

# **CIGAM Report**

**Manual de Usuário**



## Conteúdo

|  |    |
|--|----|
| Fundamentos .....                                    | 9  |
| O relatório .....                                    | 9  |
| Editor de Relatórios .....                           | 10 |
| Opções de Relatório .....                            | 10 |
| Páginas do relatório .....                           | 13 |
| Gerenciando páginas .....                            | 14 |
| Propriedades da página .....                         | 15 |
| Bandas .....   | 17 |
| Bandas no editor .....                               | 19 |
| Configurando bandas .....                            | 20 |
| Imprimindo bandas .....                              | 22 |
| Propriedades da banda .....                          | 23 |
| Objetos de Relatório .....                           | 23 |
| Propriedades comuns de objetos .....                 | 24 |
| O objeto Texto .....                                 | 26 |
| O objeto Texto Formatado .....                       | 32 |
| Editarando o objeto Texto Formatado .....            | 32 |
| O objeto Figura .....                                | 34 |
| O objeto Formas .....                                | 35 |
| O objeto Código de Barras .....                      | 36 |
| O objeto Caixa de Verificação .....                  | 38 |
| Propriedades do objeto Caixa de Verificação .....    | 38 |
| O objeto Tabela .....                                | 39 |
| O objeto Matriz .....                                | 39 |
| O objeto MS Gráfico .....                            | 40 |
| O objeto Código Postal .....                         | 41 |
| O objeto Cellular Text .....                         | 41 |
| Seu primeiro relatório com o CIGAM Report .....      | 42 |
| Exemplo 1: Criando um relatório manualmente .....    | 42 |
| Exemplo 2: Criando um relatório com o Auxiliar ..... | 45 |



|  |    |
|--|----|
| Criação de relatório:.....                     | 50 |
| Escolhendo os dados para um relatório.....     | 50 |
| Layout dinâmico.....                           | 52 |
| Propriedades CanGrow, CanShrink.....           | 52 |
| Propriedade ShiftMode.....                     | 53 |
| Propriedade GrowToBottom.....                  | 53 |
| Propriedade Anchor.....                        | 54 |
| Propriedade Dock.....                          | 55 |
| Formatando.....                                | 55 |
| Borda e preenchimento: .....                   | 56 |
| Formatação de Texto .....                      | 57 |
| Estilos .....                                  | 57 |
| Formatação de Dados .....                      | 58 |
| Destaque .....                                 | 60 |
| Escondendo valores zerados.....                | 63 |
| Esconder valores duplicados.....               | 64 |
| Marcação intercalada dos dados .....           | 65 |
| Relatório com uma banda Dados.....             | 65 |
| Conectando uma banda à uma fonte de dados..... | 65 |
| Imprimindo o texto .....                       | 66 |
| Misturando os dados.....                       | 66 |
| Filtrando os dados.....                        | 67 |
| Cabeçalho e rodapé dos dados .....             | 68 |
| Separar dados e mantê-los juntos .....         | 70 |
| Imprimindo linhas de dados vazias .....        | 73 |
| Hierarquia de Impressão.....                   | 75 |
| Relatório Mestre-Detalhe .....                 | 78 |
| Relatório Mestre-Mestre .....                  | 81 |
| Relatório Mestre-Detalhe-Detalhe.....          | 83 |
| Relatórios com Múltiplas Colunas.....          | 84 |
| Colunas das páginas .....                      | 84 |



|  |     |
|--|-----|
| Colunas da banda de dados .....              | 86  |
| Relatórios do tipo “livreto” .....           | 88  |
| Adicionando uma página em um relatório ..... | 88  |
| Configurações da página .....                | 88  |
| Impressão em páginas ímpares/pares .....     | 89  |
| Grupos e totais .....                        | 91  |
| Criando grupos .....                         | 93  |
| Disposição dos dados .....                   | 95  |
| Grupos aninhados .....                       | 96  |
| Gerenciando grupos .....                     | 97  |
| Imprimindo valores totais .....              | 98  |
| Repetindo o cabeçalho e o rodapé .....       | 100 |
| Propriedades do grupo .....                  | 102 |
| Subrelatórios .....                          | 103 |
| Modos de impressão .....                     | 103 |
| Relatório lado-a-lado .....                  | 104 |
| Subrelatórios aninhados .....                | 104 |
| Relatórios do tipo Tabela .....              | 105 |
| Configurando colunas .....                   | 105 |
| Gerenciando o tamanho da coluna .....        | 105 |
| Configurando linhas .....                    | 106 |
| Gerenciando o tamanho da linha .....         | 106 |
| Configurando células .....                   | 107 |
| Unindo e separando células .....             | 107 |
| Inserindo objetos em células .....           | 108 |
| Imprimindo uma tabela .....                  | 108 |
| Imprimindo cabeçalhos complexos .....        | 110 |
| Usando totais .....                          | 113 |
| Layout da tabela .....                       | 113 |
| Exemplos .....                               | 113 |
| Relatórios do Tipo Matriz .....              | 120 |



|  |     |
|--|-----|
| Uma pequena teoria .....                           | 121 |
| Configurando a matriz.....                         | 122 |
| Configurando cabeçalhos.....                       | 123 |
| Configurando células.....                          | 124 |
| Estilizando a matriz .....                         | 125 |
| Gerenciamento do tamanho das linhas e colunas..... | 126 |
| Exemplos .....                                     | 127 |
| Relatórios interativos: .....                      | 143 |
| Hyperlink .....                                    | 143 |
| Configuração do <i>hyperlink</i> .....             | 143 |
| Link para a URL.....                               | 144 |
| Link para uma marcação .....                       | 145 |
| Link para um relatório detalhado.....              | 145 |
| Link para uma página detalhada .....               | 146 |
| Link personalizado.....                            | 148 |
| Estrutura do relatório.....                        | 148 |
| Exemplos .....                                     | 150 |
| Herança de Relatório.....                          | 164 |
| Criando um relatório .....                         | 164 |
| Mudando o relatório base.....                      | 165 |
| Limitações .....                                   | 166 |
| Relatórios com gráficos .....                      | 166 |
| Elementos do gráfico.....                          | 166 |
| Editor de gráficos .....                           | 168 |
| Gráficos interativos .....                         | 178 |
| Dados.....   | 181 |
| A janela Dados.....                                | 181 |
| Fontes de dados .....                              | 182 |
| Criando uma fonte de dados.....                    | 183 |
| Criando uma <i>query</i> SQL.....                  | 187 |
| Parâmetros da <i>query</i> .....                   | 190 |



|  |     |
|--|-----|
| Passando um valor para o parâmetro .....   | 192 |
| Editando uma conexão.....                  | 193 |
| Editarando a fonte de dados.....           | 193 |
| Identificadores .....                      | 193 |
| Fontes hierárquicas de dados .....         | 194 |
| Relações .....                             | 194 |
| Criando uma relação .....                  | 198 |
| Editando uma relação .....                 | 199 |
| Variáveis de sistema.....                  | 199 |
| Funções .....                              | 200 |
| Matemáticas.....                           | 201 |
| Texto.....                                 | 207 |
| Data & Hora.....                           | 213 |
| Formatação .....                           | 218 |
| Conversão.....                             | 225 |
| Program Flow .....                         | 230 |
| Totais .....                               | 231 |
| Criando um Total.....                      | 232 |
| Totais condicionais.....                   | 235 |
| Totais contínuos .....                     | 236 |
| Totais de página .....                     | 237 |
| Parâmetros do relatório.....               | 237 |
| Criando um parâmetro.....                  | 237 |
| Usando parâmetros no relatório.....        | 238 |
| Expressões.....                            | 239 |
| Editor de expressão.....                   | 239 |
| Referência para objetos do relatório ..... | 240 |
| Usando funções .Net.....                   | 241 |
| Referência a elementos de dados .....      | 242 |
| Referência a fontes de dados.....          | 242 |
| Referência às variáveis de sistema.....    | 244 |



|   |     |
|---|-----|
| Referência a valores totais .....                       | 244 |
| Referência aos parâmetros do relatório .....            | 245 |
| Script .....  | 246 |
| Informações Gerais .....                                | 248 |
| Manipuladores de Eventos.....                           | 250 |
| Eventos de Relatório .....                              | 251 |
| Utilizando objetos .NET.....                            | 252 |
| Referências a objetos do relatório .....                | 253 |
| Report and Engine objects .....                         | 254 |
| Referências a fontes de dados .....                     | 258 |
| Referência a variáveis de sistema .....                 | 259 |
| Referência a valores totais .....                       | 259 |
| Reference to report parameters .....                    | 260 |
| Exemplos .....  | 260 |
| Exemplo 1. Alterando a aparência de um objeto.....      | 260 |
| Exemplo 2. Realçando linhas de número par na banda..... | 261 |
| Exemplo 3. Filtragem de dados .....                     | 262 |
| Exemplo 4. Calculando um total .....                    | 263 |
| Exemplo 5. Alterando a posição de impressão .....       | 264 |
| Forms de diálogo .....                                  | 266 |
| Controles .....   | 266 |
| Referenciando um controle no <i>script</i> .....        | 267 |
| Filtragem de dados.....                                 | 268 |
| Filtragem automática – como funciona .....              | 269 |
| Operações de Filtro .....                               | 270 |
| Adicionando um filtro ao relatório.....                 | 271 |
| Filtrando dados em intervalos .....                     | 272 |
| Filtrando em uma coluna de dados relacionada.....       | 272 |
| Controlando a filtragem a partir do <i>script</i> ..... | 272 |
| Exemplos .....  | 273 |
| Exemplo 1. Olá, CIGAM Report! .....                     | 273 |



|  |     |
|--|-----|
| Exemplo 2. Solicitar um texto para o usuário .....         | 274 |
| Exemplo 3. Manipulando controles de diálogo .....          | 275 |
| Exemplo 4. Manipulando objetos de relatório .....          | 275 |
| Exemplo 5. Filtro simples .....                            | 277 |
| Exemplo 6. Filtragem automática .....                      | 278 |
| Exemplo 7. Filtragem automática por intervalo .....        | 279 |
| Exemplo 8. Filtragem por coluna de dados relacionada ..... | 280 |
| Pré-visualização, impressão e exportação .....             | 284 |
| Editarando o relatório .....                               | 285 |
| Exportando o relatório .....                               | 288 |
| Salvando no formato FPX .....                              | 289 |
| Exportar para o Adobe Acrobat (PDF).....                   | 289 |
| Exportar para Word (RTF) .....                             | 291 |
| Exportar para HTML .....                                   | 292 |
| Exportar para o OpenOffice Calc.....                       | 294 |
| Exportar para CSV .....                                    | 295 |
| Exportar para TXT .....                                    | 296 |
| Exportar para imagem.....                                  | 297 |
| Recomendações para desenvolvimento dos relatórios .....    | 299 |
| Enviando o relatório por e-mail .....                      | 300 |



## Fundamentos

Neste capítulo, iremos aprender os princípios necessários para trabalhar com o CIGAM Report. Também veremos em detalhes os elementos de um relatório, como páginas de relatório, bandas, e objetos de relatório.

### O relatório

O processo de construção do relatório pode ser representado como na imagem a seguir:



**Template<sup>1</sup> de relatório:** isto é o que vemos no editor do CIGAM Report. Relatórios são salvos em arquivos com a extensão **.RFX**. Um relatório pode ser criado com a ajuda do editor ou através de programação.

Os dados podem ser de qualquer tipo: dados definidos no programa, ou dados de um DBMS, como por exemplo o MS SQL. O CIGAM Report também pode trabalhar com objetos de *business-logic*<sup>2</sup>.

**Relatório preparado:** isto é o que vemos na tela de pré-visualização. O relatório preparado pode ser pré-visualizado, impresso, salvo em um dos formatos suportados (.doc, .xls, .pdf e outros), ou enviado via email.

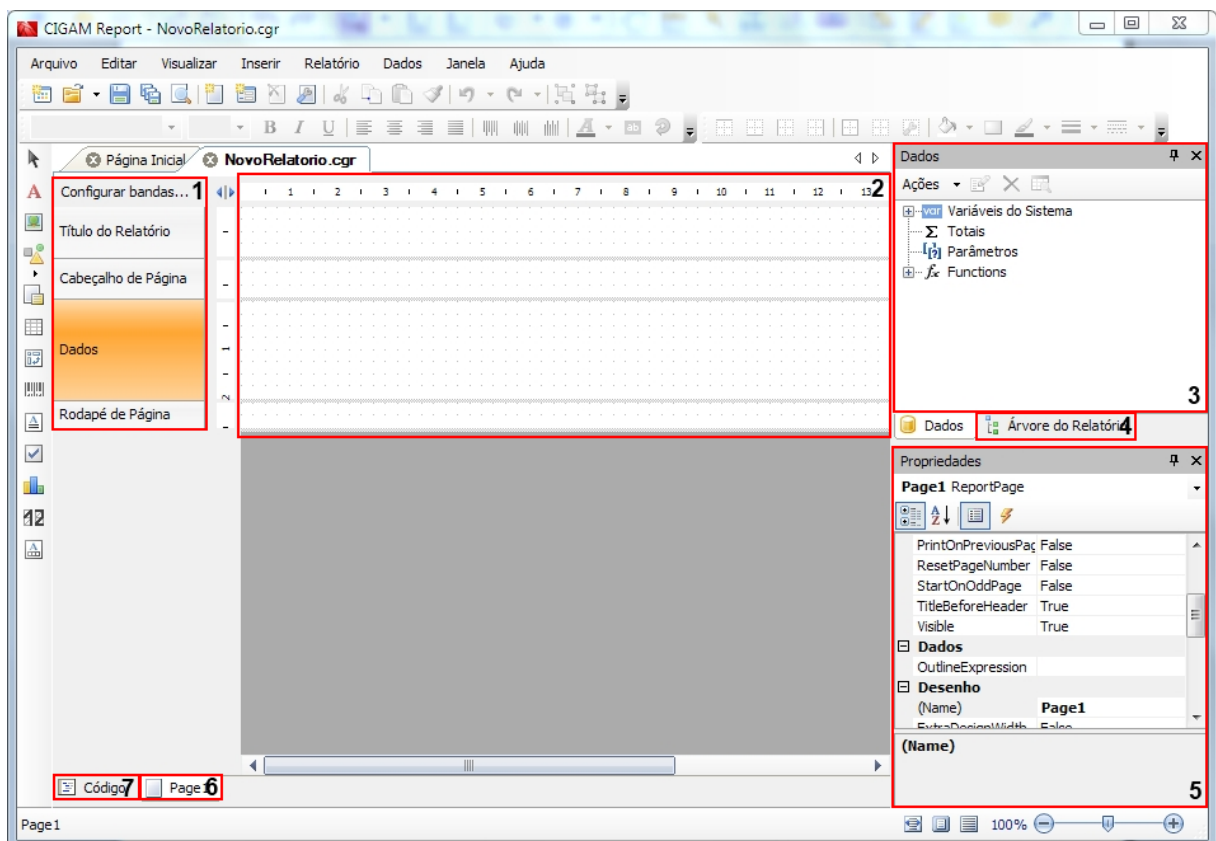
---

<sup>1</sup> **Template:** documento sem conteúdo, com apenas a apresentação visual do conteúdo. Modelo de documento para preenchimento posterior.

<sup>2</sup> **Business-logic:** termo não-técnico utilizado para descrever algoritmos que manipulam a troca de informações entre uma base de dados e uma interface de usuário.

## Editor de Relatórios

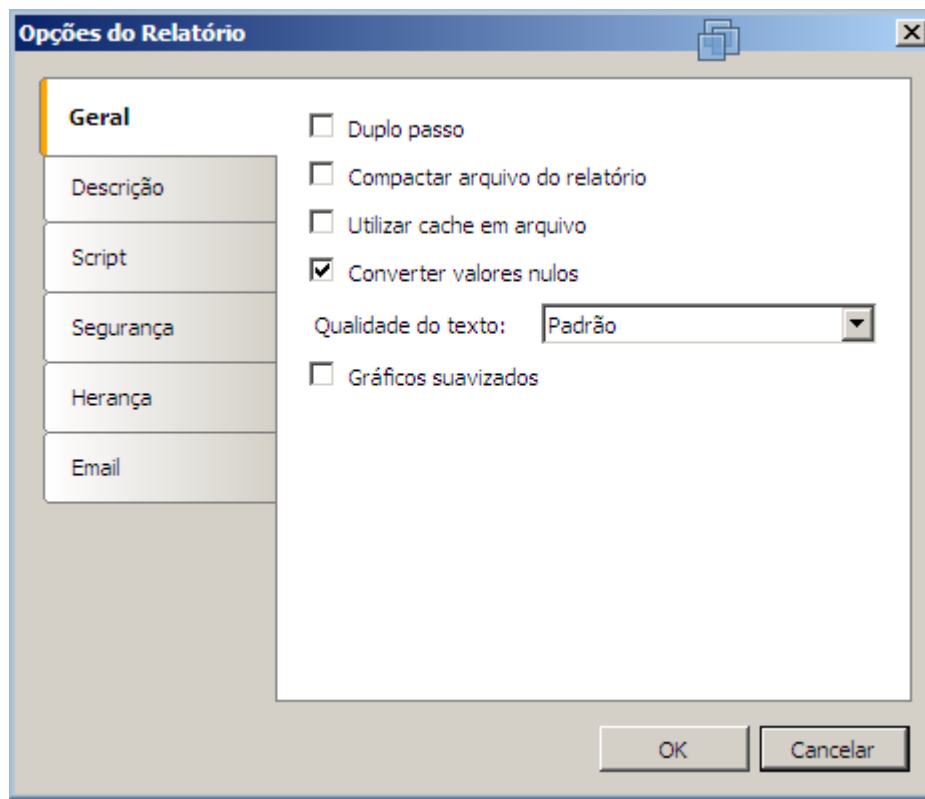
Para criar um *template* de relatório, é utilizado um editor de relatórios. Este permite ao usuário utilizar recursos simples para o desenvolvimento do relatório e, ao mesmo tempo, permite a pré-visualização do relatório.



- 1 Estrutura de bandas;
- 2 Bandas e objetos de relatórios;
- 3 Dados;
- 4 Árvore de estrutura de dados do relatório.
- 5 Propriedades;
- 6 Página do relatório;
- 7 Aba de edição do código do relatório

## Opções de Relatório

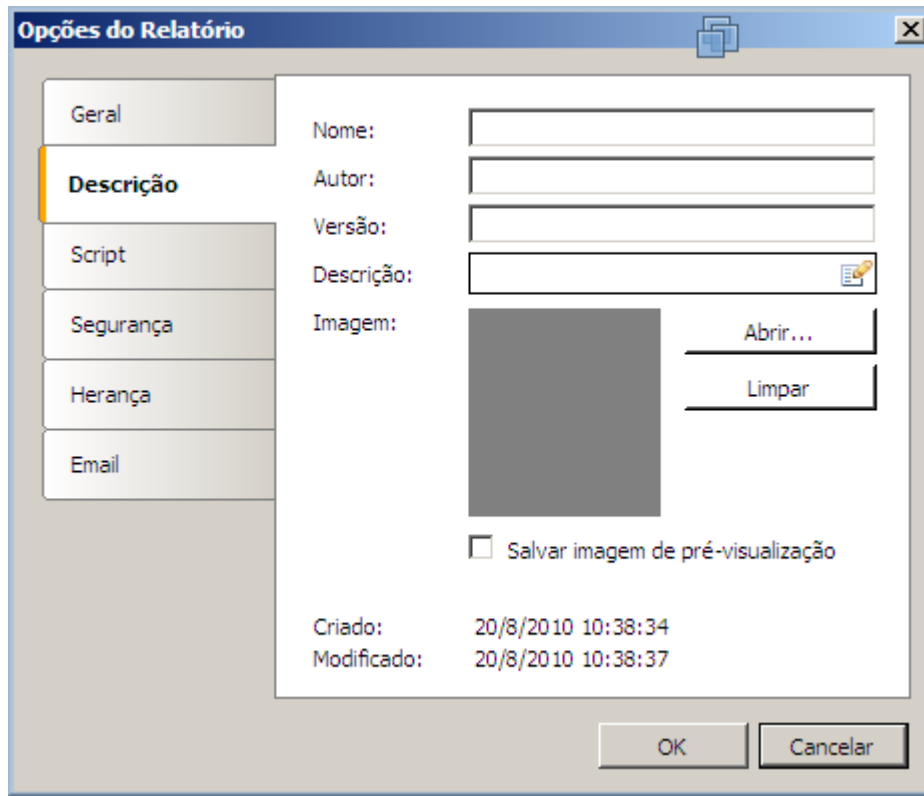
As opções do relatório podem ser acessadas através do menu **Relatório > Opções**. Você verá uma janela de diálogo com diversas abas:



Na aba **Geral**, você pode alterar os seguintes parâmetros do relatório:

- O parâmetro **Passo Duplo** permite a habilitação de duas passagens ao gerar o relatório. Isto pode ser necessário quando você utiliza a variável de sistema **número total de páginas**;
- O parâmetro **Compactar arquivo de relatório** permite que seja salvo um relatório de forma compactada. Para compressão, é utilizado o algoritmo *zip*, e por isso é possível extrair facilmente os conteúdos com a ajuda de qualquer compactador;
- O parâmetro **Usar cache de arquivo** permite o salvamento da memória quando é criado um relatório. Utilize este parâmetro se o relatório possui muitas páginas;
- O parâmetro **Converter valores nulos** controla a conversão dos valores nulos da coluna de dados aos valores padrão (0, *string vazia*, *false* – dependendo do tipo de dados de cada coluna);
- O parâmetro **Qualidade do Texto** permite a escolha do modo de exibição do texto no relatório. Este modo não afeta a impressão do relatório;
- O parâmetro **Suavizar gráficos** permite a habilitação do modo suave quando são desenhados objetos gráficos (linha, borda, gravura).

Na aba **Descrição**, você pode inserir uma descrição do relatório. Todos estes parâmetros não são obrigatórios, e servem apenas como informação:



**Opções do Relatório**

**Descrição**

Nome:

Autor:

Versão:

Descrição:

Imagem:

Salvar imagem de pré-visualização

Criado: 20/8/2010 10:38:34  
Modificado: 20/8/2010 10:38:37

Na aba **Script**, você pode escolher a linguagem de *script* para o relatório. Maiores detalhes sobre o trabalho com *scripts* pode ser encontrado no capítulo **Script**.

Na aba **Segurança**, você pode informar a senha que será requisitada quando o relatório for aberto.

---

Um relatório que possui senha é salvo de forma codificada, então não esqueça sua senha! Restaurar um relatório neste caso será praticamente impossível.

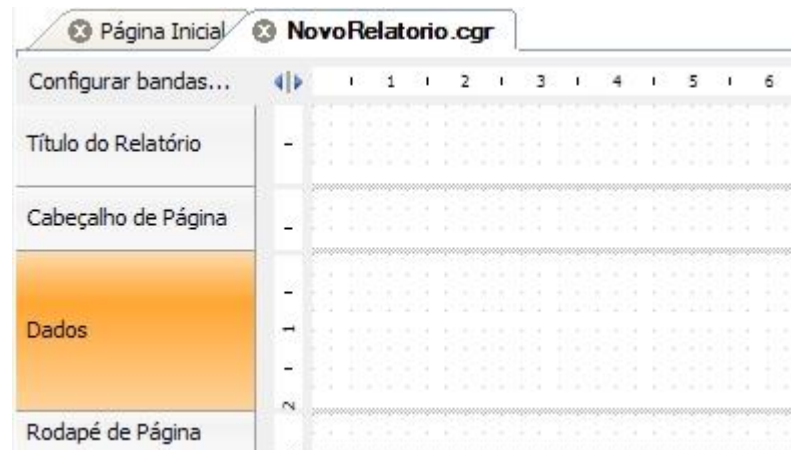
---

Na aba **Herança**, você pode controlar a herança de um relatório. Esta função será explicada em detalhes mais adiante.

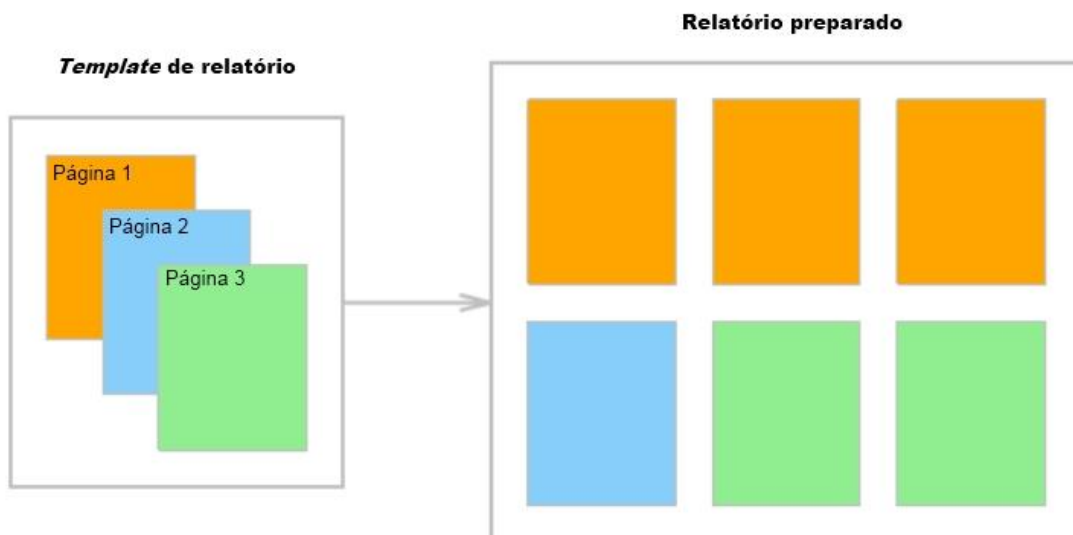
Na aba **Email**, pode ser configuradas as opções para envio do relatório por e-mail (destinatários, assunto e mensagem).

## Páginas do relatório

Um *template* consiste de uma (geralmente) ou mais páginas. Páginas de relatórios, por sua vez, contêm bandas. Objetos de relatórios como Texto, Imagem ou outros são colocados nas bandas:





Um *template* de relatório pode consistir de diversas páginas. Por exemplo, você pode criar um relatório contendo uma página-título e uma página com dados. Quando criando um relatório assim, a primeira página será impressa primeiro, e então a segunda página, e assim por diante. Cada página do *template* pode gerar uma ou mais páginas de um relatório preparado – isto dependendo diretamente dos dados contidos na página:

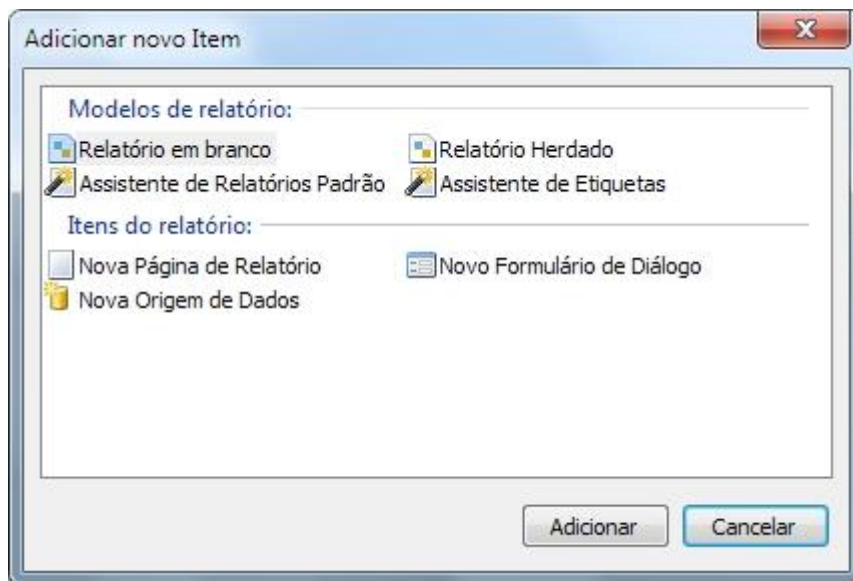



Páginas de relatório também são utilizados quando trabalhando com sub-relatórios. Ao contrário de outros geradores de relatórios, sub-relatórios no CIGAM Report são salvos em uma página de *template* separada, e não em um arquivo separado.

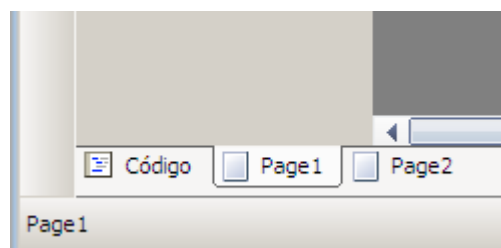
À parte das páginas de relatórios, um *template* pode conter um ou mais *forms* de diálogo. *Forms* de diálogo podem ser utilizados para solicitar alguns parâmetros antes de criar um relatório. Maiores detalhes sobre o trabalho com os *forms* de diálogo serão descritos no capítulo **Forms de diálogo**.

### Gerenciando páginas

Quando você cria um novo relatório, este já vem com uma página contendo diversas bandas. Para adicionar uma nova página, clique no botão . Uma página também pode ser adicionada clicando no botão  e escolhendo **Nova Página de Relatório** na janela.




De maneira semelhante, *forms* de diálogo também podem ser adicionados ao relatório. Para isto, utiliza o botão . Páginas de *templates* são exibidas no editor como abas:




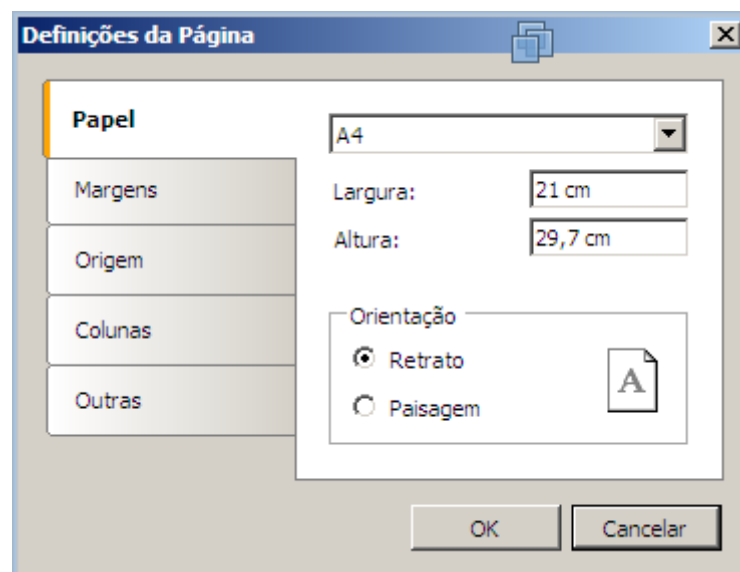
**A primeira aba é o código do relatório, e não pode ser movida nem excluída.**

Para acessar uma página, basta clicar na sua aba. As abas podem ser arrastadas para alterar a ordem destas.

Para excluir uma página, clique no botão . Este botão não estará ativo se o relatório só possui uma página.

### Propriedades da página

Cada página de relatório pode ter suas próprias configurações específicas, como tamanho do papel, orientação (paisagem ou retrato), margens, colunas, fonte de papel e outros. Um *template* de relatório pode conter diversas páginas com diferentes orientações e tamanhos de papel. A janela com a configuração da página pode ser acessada clicando no botão  ou através do menu **Arquivo > Configuração de página**.



A aba **Papel** permite definir o tamanho do papel e a orientação. É possível escolher um dos tipos suportados, utilizando a lista *drop-down*. Nesta lista estão presentes todos os tamanhos de papel que são suportados pela impressora atual.

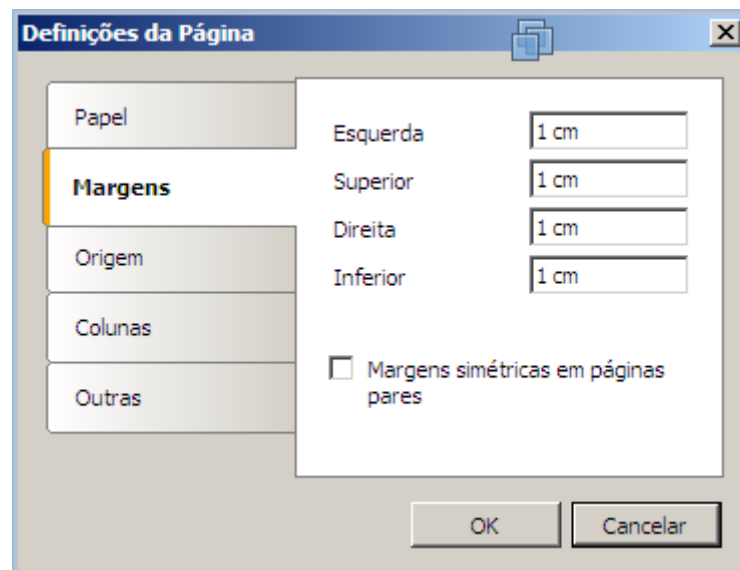
---

**A impressora atual pode ser configurada através do menu **Arquivo > Configuração de Impressora**.**

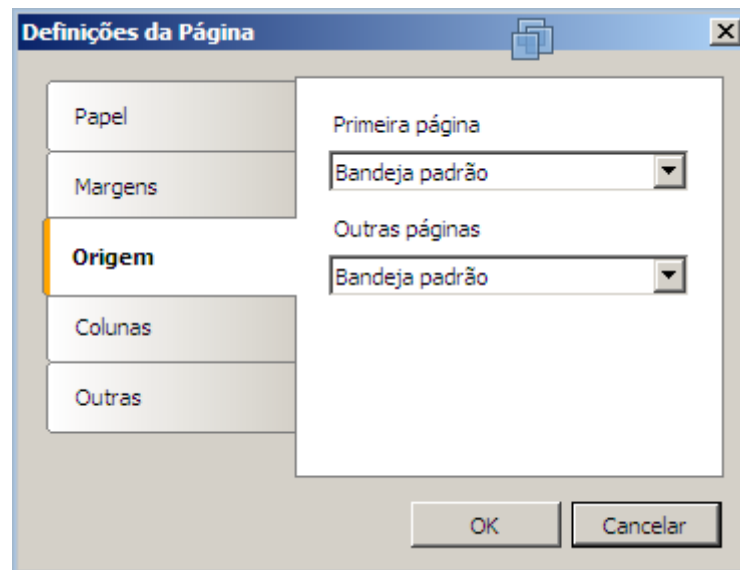
---

A aba **Margens** permite a configuração das margens da página. A opção **Espelhar margens nas páginas pares** pode ser utilizada para a impressão de livretos.





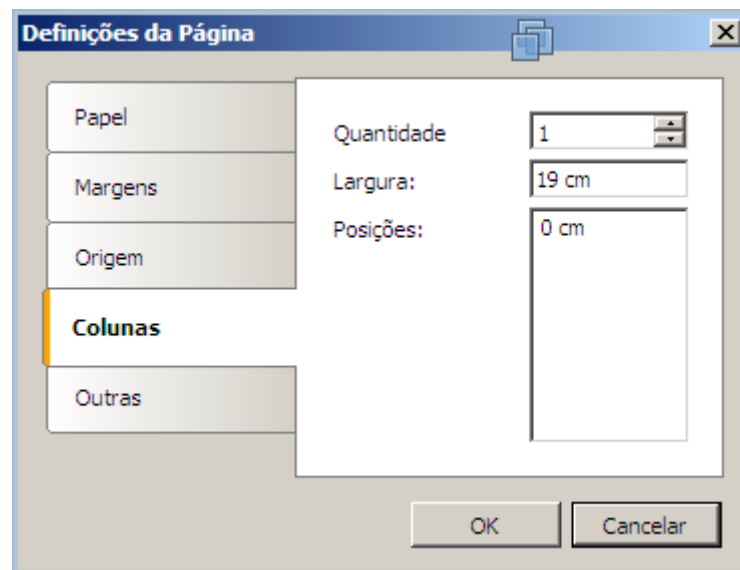
A aba **Fonte** permite a escolha da fonte de papel. Lembre-se que as fontes podem ser informadas separadamente: a fonte da primeira página do relatório preparado, e do restante das páginas.



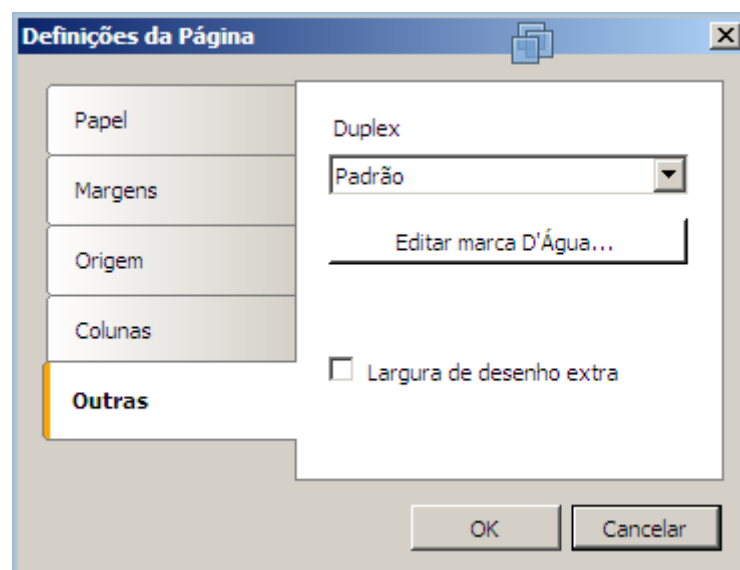
A fonte também pode ser escolhida no diálogo **Imprimir**.

A aba **Colunas** permite a configuração dos parâmetros de um relatório com múltiplas colunas. Para tanto, o número de colunas precisa ser indicado e, opcionalmente, deve ser informada a correção de largura das colunas e a posição de cada uma.





A aba **Outros** permite a configuração de certas propriedades úteis na página. É possível indicar o modo duplo (para impressão dupla), se a impressora suporta este modo de impressão. Aqui também é possível definir a marca d'água, que será impressa nas páginas preparadas do relatório.



A **checkbox Largura extra no editor** permite o aumento da largura da página no modo *design*. Esta opção pode ser útil se você trabalha com objetos como **Tabela** ou **Matriz**.

---

O modo duplo de impressão também pode ser selecionado no diálogo **Imprimir**.

---

## Bandas

Uma banda é um objeto que está localizado diretamente na página do relatório, e é um recipiente para outros objetos como **Texto**, **Picture**, e outros.



No total, no CIGAM Report existem 13 tipos de bandas. Dependendo do seu tipo, a banda é impressa em um determinado local do relatório.

| Banda                       | Como é impressa   |
|-----------------------------|---|
| <b>Título de Relatório</b>  | É impressa uma vez, no início do relatório. Você pode escolher a ordem de impressão – antes da banda <b>Cabeçalho de Página</b> ou depois- com a ajuda da propriedade de página <b>TitleBeforeHeader</b> . Esta propriedade pode ser apropriadamente modificada com a ajuda da janela de <b>Propriedades</b> . Por padrão, a propriedade é igual a <b>True</b> , ou seja, o título é impresso antes do cabeçalho. |
| <b>Sumário do Relatório</b> | É impressa uma vez ao final do relatório, depois da última linha, mas antes da banda <b>Rodapé de Página</b> .  |
| <b>Cabeçalho de Página</b>  | É impressa no topo de cada página do relatório.   |
| <b>Rodapé de Página</b>     | É impressa ao final de cada página do relatório.  |
| <b>Cabeçalho de Coluna</b>  | esta banda é utilizada quando é impresso um relatório de colunas múltiplas (quando o número de colunas indicado na configuração da página é maior do que 1). É impresso no topo de cada coluna após a banda <b>Cabeçalho de Página</b> .  |
| <b>Rodapé de Coluna</b>     | Impressa ao final de cada coluna, antes da banda <b>Rodapé de Página</b> .  |
| <b>Dados</b>                | esta banda é conectada à fonte de dados e é um número de vezes igual ao número de linhas na fonte.  |
| <b>Cabeçalho de Dados</b>   | esta banda é conectada à banda Dados e é impressa antes da primeira linha de dados.   |
| <b>Rodapé de Dados</b>      | esta banda é conectada à banda Dados e é impressa após a última linha de dados.   |
| <b>Cabeçalho de Grupo</b>   | É impressa no início de cada grupo, quando o valor da condição de grupo é alterado.   |
| <b>Rodapé de Grupo</b>      | É impressa ao final de cada grupo.  |
| <b>Filha</b>                | esta banda pode ser conectada a qualquer banda, incluindo outra banda Filha. É impressa diretamente após a <i>banda-pai</i> .   |
| <b>Deslocamento</b>         | Impressa como fundo em cada página do relatório.  |

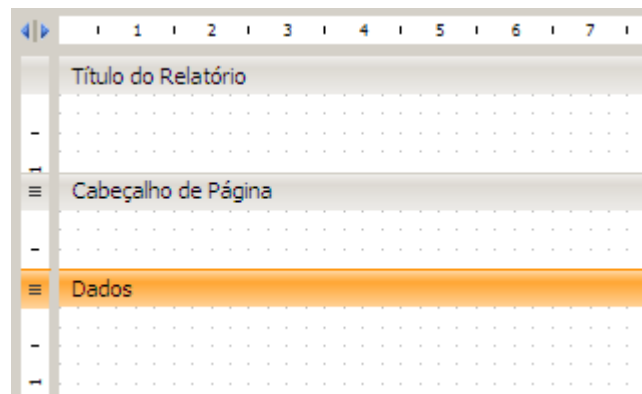
### Bandas no editor

Uma banda no editor aparece na forma de uma área retangular. Uma banda, como muitos outros objetos do relatório, pode ter uma borda e preenchimento (que por padrão estão desabilitadas). Independente disto, uma banda exibe uma grade. Para habilitar o modo de grade, vá para **Visualizar > Opções** e escolha **Página do Relatório**. Grades também podem ser habilitadas ou desabilitadas no menu **Visualizar**.

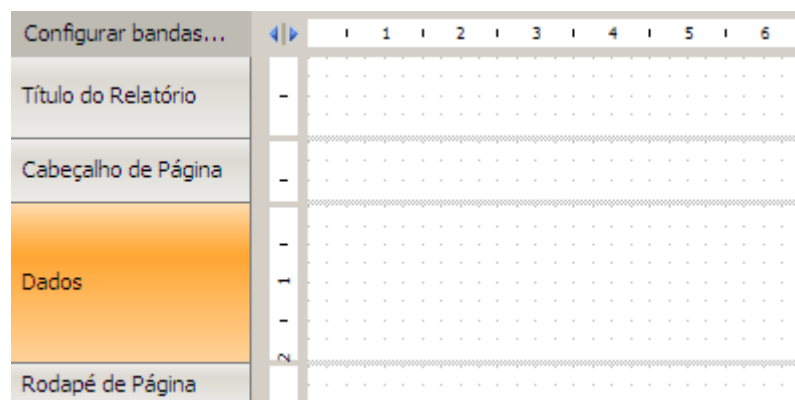
Você pode ajustar a altura da banda de três maneiras:


- Coloque o ponteiro do mouse na borda inferior da banda. O formato do cursor deve mudar para um **divisor horizontal** e você pode redimensionar a banda;
- Arraste a alça da banda na régua esquerda;
- Use a janela **Propriedades** para ajustar a propriedade **Height** da banda.

O editor tem dois modos de exibição de bandas, que podem ser alternados a qualquer momento. No primeiro modo, cada banda tem um cabeçalho, que contém o título da banda e informações úteis sobre esta (como por exemplo o nome da fonte de dados à qual está conectada).



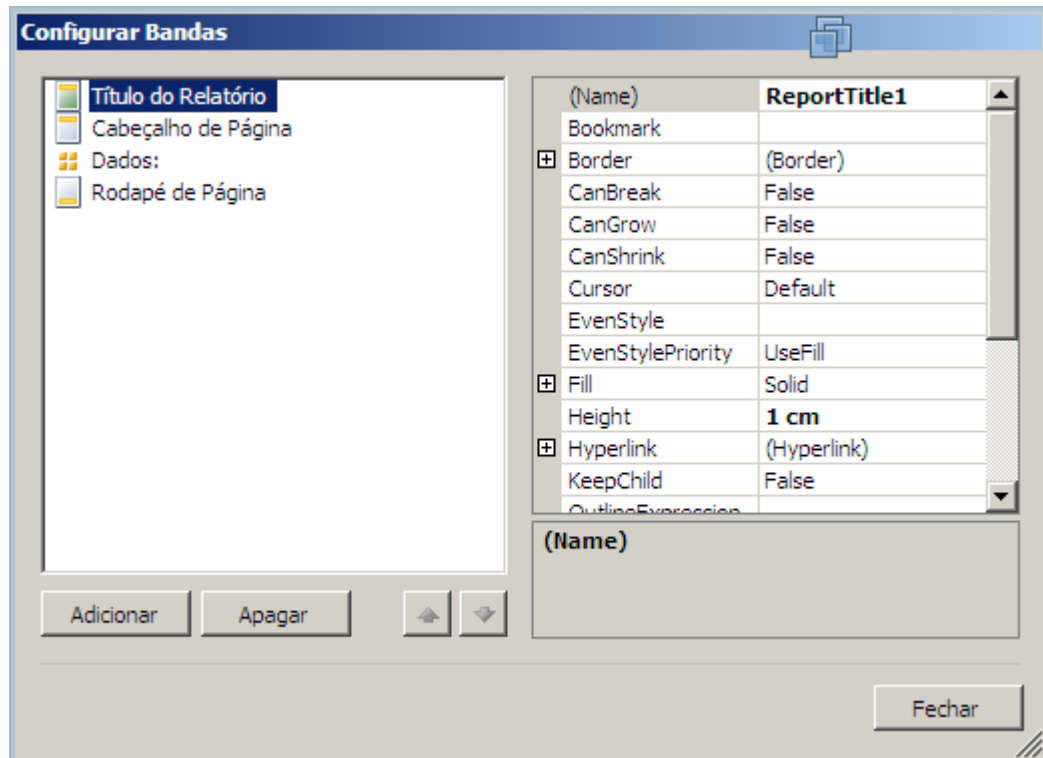
No segundo modo, a banda não possui um cabeçalho. Ao invés disso, a estrutura das bandas é exibida no lado esquerdo da janela. Este modo simplifica a compreensão do relatório, principalmente se o mesmo não foi elaborado por você.



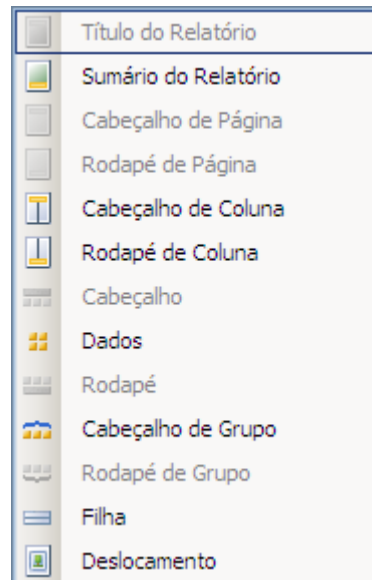
Você pode alternar entre estes modos clicando no botão .

### Configurando bandas

Você pode configurar as bandas na janela **Configurar Bandas**, acessada pelo menu **Relatório > Configurar Bandas...** ou com a ajuda do botão **Configurar bandas**, localizado acima da estrutura em árvore das bandas.

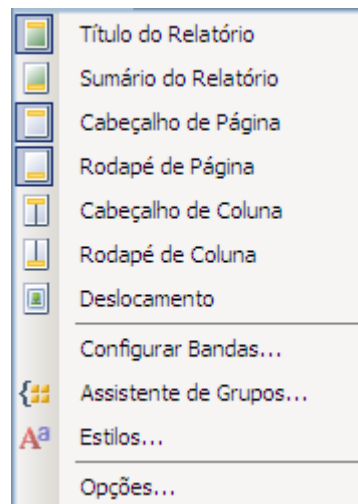


Nesta janela, é possível adicionar bandas ao relatório, excluir, ou alterar a ordem destas. Para adicionar uma banda, clique no botão **Adicionar** ou clique com o botão direito na estrutura em árvore das bandas. Um menu de contexto será exibido contendo uma lista das bandas. Uma banda que não pode ser adicionada estará esmaecida.



A operação **Adicionar** depende do tipo de operação sendo adicionada. Por exemplo, adicionar bandas do tipo **Cabeçalho de Dados** e **Rodapé de Dados** é possível somente se a banda **Dados** foi selecionada anteriormente.

Ainda há outra maneira de configurar algumas bandas. Isto pode ser feito a partir do menu **Relatório**.



Para excluir uma banda, selecione-a e pressione **Delete**.

Quando estiver configurando uma banda, o CIGAM Report não permite a execução de operações que levam à criação de um *template* de relatório que esteja errado. Por exemplo, você não pode excluir uma banda **Dados**, que está conectada a um grupo – para isto, o grupo deve ser excluído primeiro. Outro exemplo é quando você estiver excluindo uma banda **Dados**: seu cabeçalho (**Cabeçalho de Dados**) e rodapé (**Rodapé de Dados**) serão excluídos automaticamente. Também não é possível excluir uma banda se esta é a única presente na página.

### Imprimindo bandas

Temos diversas bandas já colocadas em uma página. Agora, como o CIGAM Report irá compor um relatório preparado? Vamos acompanhar o seguinte exemplo:

|                      |   |                      |
|----------------------|---|----------------------|
| Título do Relatório  | - | Título do Relatório  |
| Cabeçalho de Página  | - | Cabeçalho de Página  |
| Dados                | - | Dados                |
| Sumário do Relatório | - | Sumário do Relatório |
| Rodapé de Página     | - | Rodapé de Página     |

A banda **Título do Relatório** será impressa primeiro. A banda **Cabeçalho de Página** será impressa imediatamente após. Adiante, a banda **Dados** será impressa tantas vezes quanto houverem linhas na fonte de dados à qual a banda está conectada. Depois de todas as linhas da banda **Dados** terem sido impressas, a banda **Sumário do Relatório** é impressa e ao final da página, a banda **Rodapé de Página**. A impressão de um relatório é encerrada aqui. Um relatório preparado tem uma aparência mais ou menos assim:

|                     |                     |
|---------------------|---------------------|
| Título do Relatório | Cabeçalho de Página |
| Cabeçalho de Página | Dados               |
| Dados               | Dados               |
| Dados               |                     |
| Dados               |                     |
| Dados               |                     |
| Dados               |                     |
| Dados               |                     |
| Dados               |                     |
| Rodapé de Página    | Rodapé de Página    |

Durante o processo de impressão, o CIGAM Report checa se há espaço suficiente na página atual do relatório preparado para que a banda seja impressa. Se não houver espaço, ocorre o seguinte:

- O rodapé da página é impresso;
- Uma nova página é adicionada;
- O cabeçalho da página é impresso;
- Continua a impressão da banda que não coube na página anterior.

### Propriedades da banda

| Propriedade               | Descrição   |
|---------------------------|---|
| <b>CanGrow, CanShrink</b> | estas propriedades determinam se a banda pode crescer ou encolher dependendo do tamanho dos objetos contidos na banda. Se ambas as propriedades estiverem desabilitadas, a banda sempre terá o tamanho especificado no editor. Leia mais sobre isto no capítulo <b>Criação de relatório</b> . |
| <b>CanBreak</b>           | Se a propriedade estiver habilitada, o CIGAM Report tenta imprimir parte do conteúdo da banda no espaço disponível, ou seja, quebra a banda. Leia mais sobre isto no capítulo <b>Criação de relatório</b> .   |
| <b>StartNewPage</b>       | A impressão de uma banda com esta propriedade inicia uma nova página. Esta propriedade é geralmente utilizada quando são impressos grupos: cada grupo é impresso em uma nova página.  |
| <b>PrintOnBottom</b>      | Uma banda com esta propriedade é impressa ao final da página, antes da banda <b>Rodapé de Página</b> . Isto pode ser útil quando imprimindo certos documentos onde a soma total deve ser impressa ao fim AL da página.  |
| <b>RepeatOnEveryPage</b>  | As bandas <b>Cabeçalho de Dados, Rodapé de Dados, Cabeçalho de Grupo e Rodapé de Grupo</b> possuem esta propriedade. Este tipo de banda será impresso em cada nova página, quando a impressão de dados é feita. Leia mais sobre isto no capítulo <b>Criação de relatório</b> .                |

Cada banda possui diversas propriedades úteis, que afetam o processo de impressão. Estas podem ser configuradas utilizando o menu de contexto da banda. Para acessá-lo, clique com o botão direito em um espaço livre da banda (que não esteja ocupado por outros objetos). Também é possível acessar este menu clicando com o botão direito no cabeçalho da banda, se o modo de exibição primário, ou na estrutura da banda, no modo secundário. Outro método de fazer estas alterações é selecionando a banda e alterando suas propriedades na janela **Propriedades**.

### Objetos de Relatório

Uma variada gama de objetos pode ser utilizada no relatório.

| Ícone | Nome                          | Descrição                          |
|-------|-------------------------------|------------------------------------|
|       | <b>Texto (TextObject)</b>     | Exibe uma ou mais linhas de texto. |
|       | <b>Figura (PictureObject)</b> | Exibe uma imagem.                  |



|  |  |
|--|--|
| <b>Linha (LineObject)</b>                    | Exibe uma linha que pode ser vertical, horizontal ou diagonal.   |
| <b>Forma (ShapeObject)</b>                   | Exibe uma forma geométrica (retângulo, elipse, triângulo, e outros).   |
| <b>Texto Formatado (RichObject)</b>          | Exibe texto formatado em RTF(Rich Text Format).  |
| <b>Código de Barras (BarcodeObject)</b>      | Exibe um código de barras.   |
| <b>Caixa de Verificação (CheckBoxObject)</b> | Exibe uma <i>checkbox</i> que pode ter dois estados: <i>Enabled</i> (marcado) ou <i>Disabled</i> (desmarcado). |
| <b>Tabela (TableObject)</b>                  | Exibe uma tabela simples.  |
| <b>Matriz (MatrixObject)</b>                 | Exibe uma matriz, também conhecida como <i>Cross-tab</i> .   |
| <b>MSGrafico (MSChartObject)</b>             | Exibe um gráfico.  |
| <b>Código Postal (ZipCodeObject)</b>         | Exibe um código postal.  |
| <b>Cellular Texto (CellularTextObject)</b>   | Exibe um texto onde cada caractere está localizado em uma célula individual.                                   |

Um objeto pode ser utilizado para exibir uma informação (objeto **Texto**) ou para melhorar a aparência do relatório (objetos **Figura**, **Linha**, **Forma**). Objetos complexos como a **Tabela** e a **Matriz** podem conter outros objetos simples.

### Propriedades comuns de objetos

Todos os objetos de relatório são herdados de uma classe-base (ReportComponentBase) e possuem um conjunto de propriedades em comum. Antes de estudar cada objeto, iremos ver estas propriedades.

Os valores das propriedades podem ser alterados na janela **Propriedades**. Algumas propriedades podem ser alteradas utilizando o **menu de contexto** do objeto ou até mesmo algumas **barras de ferramentas**(como o preenchimento e a borda, por exemplo).

| Propriedade                     | Descrição  |
|---------------------------------|--|
| <b>Left, Top, Width, Height</b> | Um objeto de relatório é considerado como um retângulo, e possui coordenadas (propriedades Left e Top, respectivamente Esquerda e Topo) e tamanho (propriedades Width e Height, respectivamente largura e altura).   |
| <b>Anchor</b>                   | Determina como o objeto vai mudar sua posição e/ou tamanho quando a banda onde se encontra cresce ou encolhe. Esta propriedade pode ser utilizada para criar um layout dinâmico, fazendo com que o objeto se expanda ou mova de forma sincronizada com a banda onde se localiza. Leia mais sobre esta propriedade na seção <b>Layout Dinâmico</b> do capítulo <b>Criação de Relatórios</b> . |





|                           |   |
|---------------------------|---|
| <b>Dock</b>               | Determina em qual lado da banda o objeto será fixado. Leia mais sobre esta propriedade no seção <b>Layout Dinâmico</b> do capítulo <b>Criação de Relatórios</b> .   |
| <b>Border, Fill</b>       | Determinam, respectivamente, a borda e o preenchimento do objeto.   |
| <b>CanGrow, CanShrink</b> | Como a propriedade de banda, permite que o objeto cresça ou diminua para acomodar o conteúdo. Leia mais sobre esta propriedade no seção <b>Layout Dinâmico</b> do capítulo <b>Criação de Relatórios</b> .   |
| <b>ShiftMode</b>          | Com esta propriedade ativa, o objeto irá se mover para cima ou para baixo se o objeto acima puder crescer ou diminuir. É utilizado para a criação de layouts dinâmicos. Leia mais sobre esta propriedade no seção <b>Layout Dinâmico</b> do capítulo <b>Criação de Relatórios</b> . |
| <b>GrowToBottom</b>       | Faz com que o objeto seja estendido até o final da página. Leia mais sobre esta propriedade na seção <b>Layout Dinâmico</b> do capítulo <b>Criação de Relatórios</b> .  |
| <b>CanBreak</b>           | Como na propriedade de banda, permite que o conteúdo seja quebrado para ser enquadrado no espaço disponível do objeto.  |
| <b>PrintOn</b>            | Determina em quais páginas o objeto pode ser impresso. Leia mais sobre esta propriedade na seção <b>Relatório do tipo livreto</b> do capítulo <b>Criação de Relatórios</b> .  |
| <b>Cursor</b>             | Determina o tipo de cursor quando o objeto é focado com o mouse. Esta propriedade só é utilizada na janela de pré-visualização.   |
| <b>Visible</b>            | Determina se o objeto será exibido no relatório. Objetos invisíveis nunca são exibidos nem impressos, seja na pré-visualização ou na impressão.   |
| <b>Printable</b>          | Determina se o objeto será impresso. Com esta opção desabilitada, o objeto será visível na pré-visualização, porém não será impresso.   |
| <b>Hyperlink</b>          | Permite fazer com que o objeto de relatório seja interativo e permita utilização de um <i>hyperlink</i> . Leia mais sobre esta propriedade na seção <b>Relatórios interativos</b> do capítulo <b>Criação de Relatórios</b> .  |
| <b>Marcação</b>           | É utilizada em conjunto com a propriedade Hyperlink. Pode conter qualquer expressão. Esta expressão será calculada quando o relatório for executado, e seu valor será utilizado como nome da marcação gerada.   |
| <b>Restrictions</b>       | Restringe certas operações, como movimentação, redimensionamento e exclusão do objeto.  |

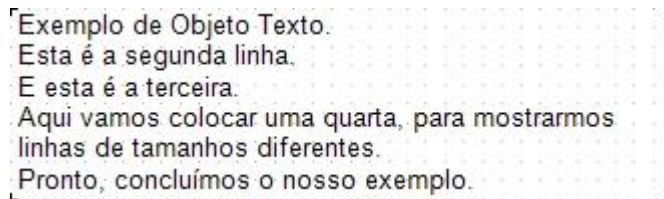
---

|              |   |
|--------------|---|
| <b>Style</b> | Nesta propriedade pode ser atribuído um nome de estilo. Quando isto for feito, o objeto será configurado conforme o estilo indicado. Se os parâmetros do estilo mudarem, a aparência também muda. |
|--------------|---|

---

### O objeto Texto

O **Texto** é o principal objeto a ser utilizado na criação de um relatório. Quando inserido na área de trabalho do editor de relatórios, ele é apresentado assim:



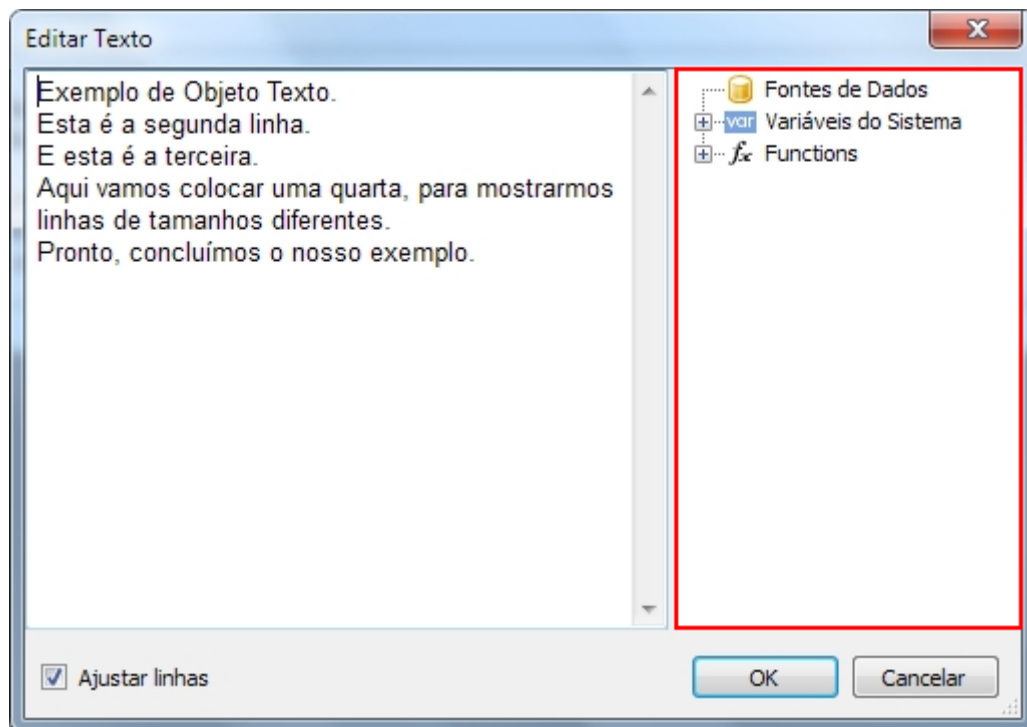
Exemplo de Objeto Texto.  
Esta é a segunda linha.  
E esta é a terceira.  
Aqui vamos colocar uma quarta, para mostrarmos  
linhas de tamanhos diferentes.  
Pronto, concluímos o nosso exemplo.

este é o **objeto mais versátil** do CIGAM Report, sendo utilizado para exibir **quaisquer tipos de dados**, tais como:

- Uma ou mais linhas de texto simples;
- Colunas de dados;
- Parâmetros de relatório;
- Valores totais;
- Expressões;
- Qualquer combinação dos itens anteriores.

### Editarando o texto

Para editar o texto de um objeto, basta dar um **duplo-clique** no mesmo. É exibida a tela de edição do objeto. Veja a imagem abaixo para saber mais:



Ao lado é exibida uma árvore de dados, com as fontes de dados às quais o relatório está conectado, as variáveis disponíveis do sistema, e as funções que podem ser utilizadas no texto.

Para adicionar uma delas, basta arrastar a mesma até o texto, ou dar um duplo-clique. Esta segunda maneira irá inserir o a opção selecionada no local do texto onde estiver o cursor.

---

Quando um objeto está sendo editado diretamente, seu tamanho pode ser alterado com o uso do mouse.

---

### Expressões

O objeto Texto pode utilizar expressões dinâmicas misturadas ao texto simples, como por exemplo:

**Hoje é [Date].**

Quando o texto é impresso, todas as expressões contidas no mesmo serão calculadas. O resultado do exemplo acima ficaria aproximadamente assim:

**Hoje é 16.07.2010.**

---

As expressões são identificadas, por padrão, através de **colchetes**. Isto pode ser alterado na **propriedade Brackets**, que por padrão contém a string [,]. Se desejado (ou necessário),

---



---

as mesmas podem ser trocadas por um conjunto diferente de símbolos, como por exemplo, <!, !>. Neste caso, o **exemplo acima ficaria assim:**

**Hoje é <!Date!>.**

---

Além disto, é possível desabilitar todas as expressões. Para tanto, basta mudar o valor da propriedade **AllowExpressions** para **false**. Neste caso o texto será mostrado diretamente como foi digitado.

Dentro dos colchetes, pode ser utilizada qualquer expressão válida. Leia mais sobre isto no capítulo **Expressões**. Por exemplo, um objeto com o seguinte texto:

**2 \* 2 = [2 \* 2]**

Será impresso assim:

**2 \* 2 = 4**

---

**Erro comum:** tentar escrever uma expressão **fora** dos colchetes. Lembrando que só é considerada expressão e executada aquela informação que é digitada **dentro dos colchetes**. Por exemplo:

**2 \* 2 = [2] \* [2]** será impresso como **2 \* 2 = 2 \* 2**

---

Em alguns casos, pode haver a necessidade de utilizar expressões dentro de outras expressões. Nestes casos, a expressão interna deve obrigatoriamente utilizar colchetes, **independentemente da configuração** da propriedade *AllowExpressions*. Veja este exemplo:

**A próxima página: [[Page]+1].**

Dentro da expressão, existe a expressão **[Page]**. Esta é cercada por seus próprios identificadores, que devem ser colchetes.

### *Exibindo colunas de dados*

Uma coluna de dados pode ser impressa da seguinte maneira:

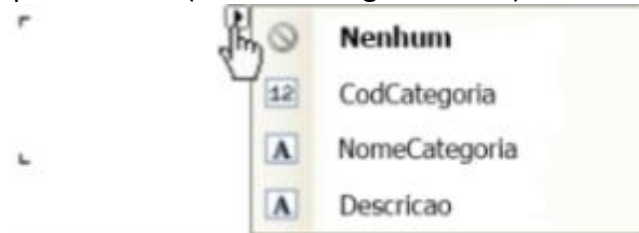
**[NomeDaFonteDados.NomeDaColuna]**

Como você pode ver, os colchetes estão sendo utilizados aqui. O nome da fonte de dados e o nome da coluna de dados são separados por um ponto. Um exemplo de coluna de dados é o seguinte:

**[Colaboradores.PrimeiroNome]**

Porém, este método de inserção de coluna é inconveniente, pois é muito fácil de ocorrerem erros no nome de uma das variáveis. Siga os passos abaixo para inserir uma coluna de dados de outra maneira:

1. No editor do objeto Texto, na área lateral com a árvore de dados, selecione a coluna desejada e arraste até a área de edição do texto.
2. Clique no botão pequeno que aparece no canto superior direito do objeto. Selecione a opção no menu que é exibido (como na imagem abaixo).

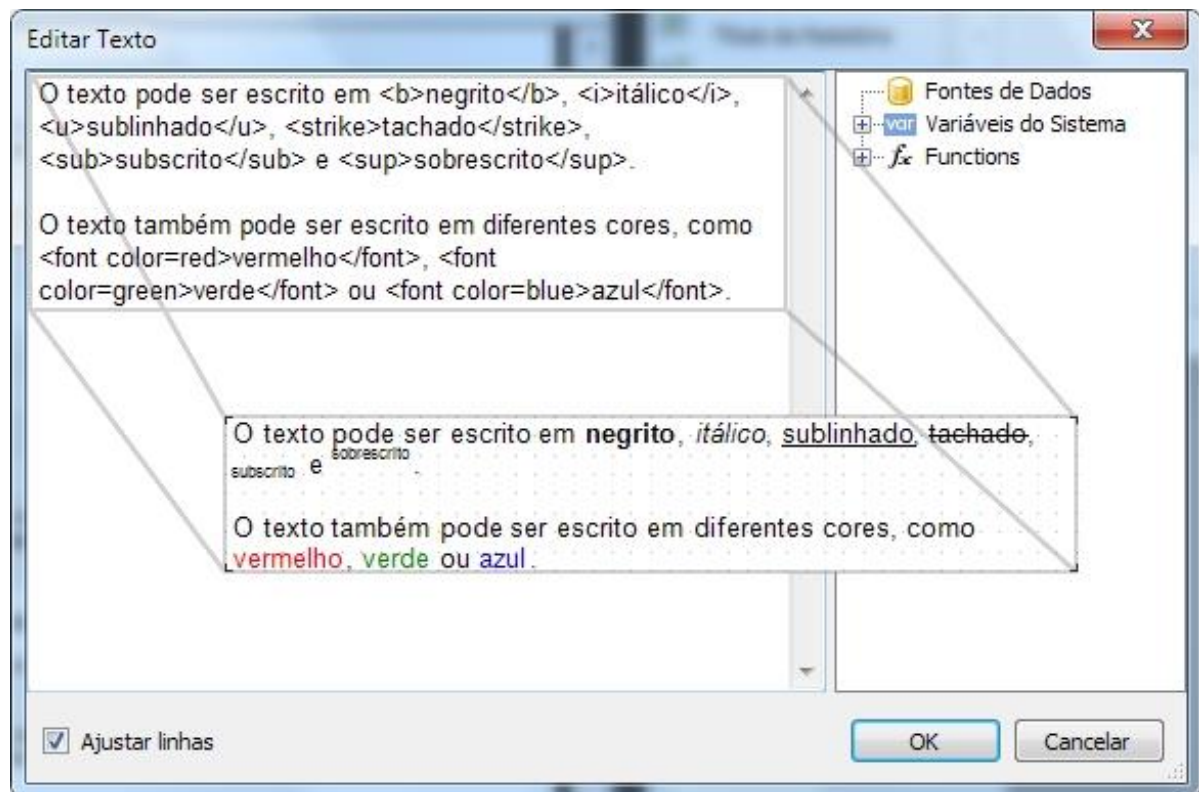


3. Clique e arraste a coluna de dados da janela aberta para a página do relatório. Neste caso é criado um objeto com um link para a coluna.

### Tags HTML

Também podem ser utilizadas *tags* HTML dentro de um objeto Texto. Por padrão, estas estão desabilitadas, mas isto pode ser alterado mudando a propriedade **HtmlTags** para **true**. As *tags* suportadas são:

- **<b> e </b>**: texto em negrito;
- **<i> e </i>**: texto em itálico;
- **<u> e </u>**: texto sublinhado;
- **<strike> e </strike>**: texto tachado;
- **<sub> e </sub>**: texto subscrito;
- **<sup> e </sup>**: texto sobrescrito;
- **<fontcolor=X> e </fontcolor>**: Cor da fonte. A cor pode tanto ser pelo nome (como DarkGray ou Red) quanto código hexadecimal #RGB (como #FF8030).



### Propriedades do objeto Texto

Além das propriedades comuns a todos os objetos, o objeto Texto possui as seguintes propriedades específicas:

| Propriedade              | Descrição  |
|--------------------------|--|
| <b>AllowExpressions</b>  | Permite ativar ou desativar a utilização de expressões.  |
| <b>Angle</b>             | Permite indicar um ângulo de rotação para o texto, em graus.   |
| <b>AutoShrink</b>        | Faz com que o tamanho e largura de fonte sejam reduzidos automaticamente para acomodar o texto no espaço disponível.   |
| <b>AutoShrinkMinSize</b> | Permite indicar um tamanho mínimo para a redução da fonte.   |
| <b>AutoWidth</b>         | Calcula automaticamente a largura do objeto.   |
| <b>Brackets</b>          | Símbolos utilizados para identificar uma expressão.  |
| <b>BreakTo</b>           | Com esta propriedade, você pode organizar o fluxo de texto de um objeto de Texto para o próximo. Por exemplo: temos um objeto A e um objeto B. O objeto A contém texto |





|                             |  |
|-----------------------------|--|
|                             | longo que não cabe no tamanho especificado para este. Com esta propriedade configurada para B, este objeto irá exibir a parte do texto que não coube em A.                 |
| <b>Clip</b>                 | Determina se é necessário cortar um texto que se estenda para fora dos limites do objeto. Vem ativada por padrão.  |
| <b>Duplicates</b>           | Determina como os valores repetidos serão impressos.   |
| <b>FirstTabOffset</b>       | Determina o primeiro recuo da tabulação, em pixels.  |
| <b>FontWidthRatio</b>       | Utilize esta propriedade para alterar a largura da fonte. Para aumentar a largura, utilize um valor maior do que 1. Para diminuir a largura, utilize um valor entre 0 e 1. |
| <b>HideValue</b>            | Propriedade do tipo String. Permite que sejam escondidos valores que são iguais ao valor especificado nesta propriedade.   |
| <b>HideZeros</b>            | Permite que sejam escondidos valores zerados.  |
| <b>Highlight</b>            | Permite a definição de marcações condicionais.   |
| <b>HorzAlign, VertAlign</b> | Determina o alinhamento do texto.  |
| <b>HtmlTags</b>             | Permite a utilização de tags HTML simples no texto, como visto na lição anterior.  |
| <b>LineHeight</b>           | esta propriedade permite definir a altura de uma linha de texto. Por padrão o valor é 0, utilizando o espaçamento padrão de linhas.  |
| <b>Padding</b>              | Permite a definição do espaçamento, em pixels.   |
| <b>RightToLeft</b>          | Indica se o texto deve ser exibido em ordem da direita para a esquerda.  |
| <b>TabWidth</b>             | Determina a largura do símbolo TAB, em pixels.   |
| <b>Texto</b>                | Contém o texto do objeto.  |
| <b>TextFill</b>             | Determina o preenchimento do texto.  |
| <b>Trimming</b>             | Determina como cortar o texto que não cabe dentro dos limites do objeto. Só é utilizado se a propriedade WordWrap estiver com o valor false.                               |

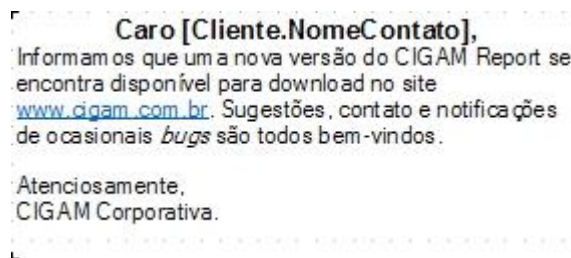
---

|                   |  |
|-------------------|--|
| <b>Underlines</b> | Faz com que seja adicionada uma linha gráfica após cada linha de texto. Só pode ser utilizada se o texto estiver alinhado com o topo.  |
| <b>WordWrap</b>   | Determina se é necessário quebrar um texto automaticamente.  |
| <b>Wysiwyg</b>    | Altera o modo de exibição do objeto Texto para ser o mesmo da tela e da impressão final. Este modo também é usado se o alinhamento justificado ou uma altura de linha que não a padrão estejam sendo utilizados. |

---

### O objeto Texto Formatado

este objeto exibe texto em formato RTF (Rich Texto Format), como na imagem abaixo:



Tente utilizar o objeto **Texto** para exibir **texto escrito**, pois quando é feita a exportação do relatório, o objeto **Texto Formatado** será exibido como **uma imagem**.

### Editarando o objeto Texto Formatado

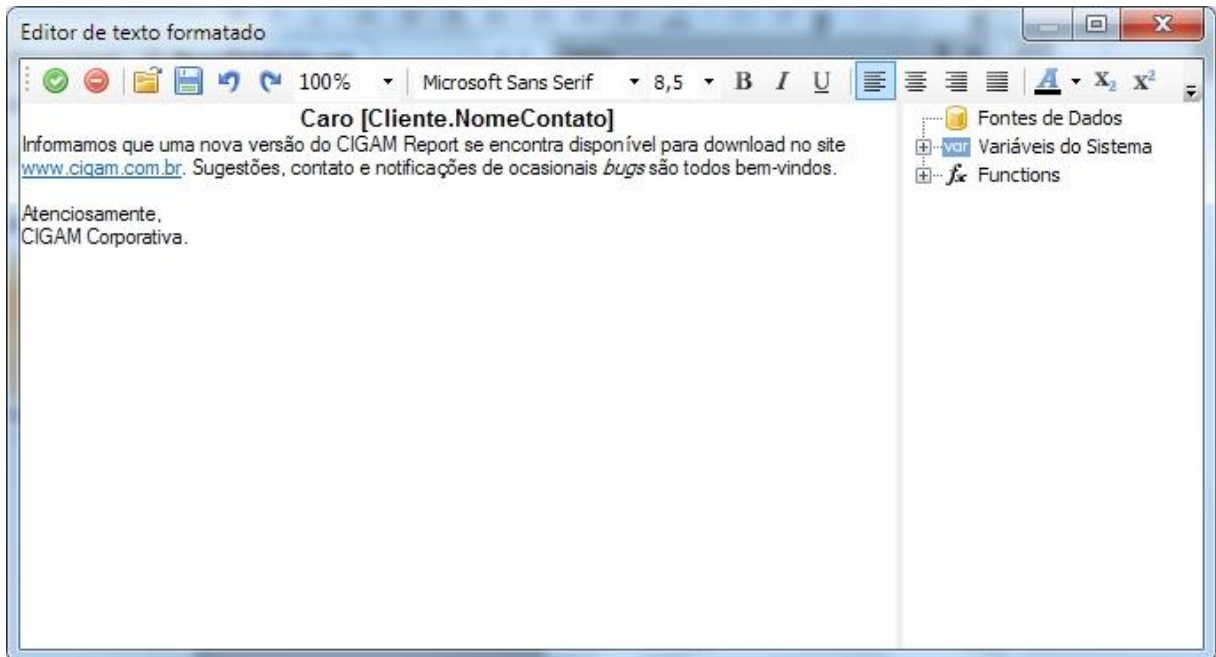
Para editar o objeto, basta dar um **duplo-clique**. A janela abaixo (extremamente semelhante à tela de edição do objeto Texto, porém com recursos WYSIWYG<sup>3</sup>) é exibida:

---


<sup>3</sup> **What You See Is What You Get:** O que você vê é o que você ganha. Editor semelhante ao padrão de editores RTF de texto, como o Microsoft Word. O que é visto na tela do editor é o mesmo que é impresso ao concluir o trabalho.

Um editor WYSIWYG possui ferramentas visuais para a edição de texto, como os botões Negrito, Itálico e Sublinhado, além de opções de tipo, cor e tamanho da fonte, subscrito e sobrescrito, alinhamento do texto, etc.





O objeto **Texto Formatado** possui algumas características em comum com o objeto **Texto**: além das notáveis semelhanças por ambos objetos lidarem com o mesmo tipo de dados (texto), o objeto Texto Formatado **também interpreta expressões**, como no exemplo visto acima. Basta inserir a coluna de dados necessária para o texto, como no objeto Texto.

O texto também pode ser criado no Microsoft Word ou qualquer outro editor que suporte o formato RTF. Crie o arquivo no editor e salve o mesmo com a extensão .RTF. Depois, dentro do editor, clique no ícone de **Abrir** , e selecione o arquivo para abertura.

É possível que não sejam suportados pelo objeto todos os recursos de formatação disponibilizados pelo editor RTF. Quando estiver criando um texto para ser utilizado neste objeto, atente aos recursos que este possui, para evitar erros na importação de um texto.

### Propriedades do objeto Texto Formatado

O objeto Texto Formatado possui as seguintes propriedades:

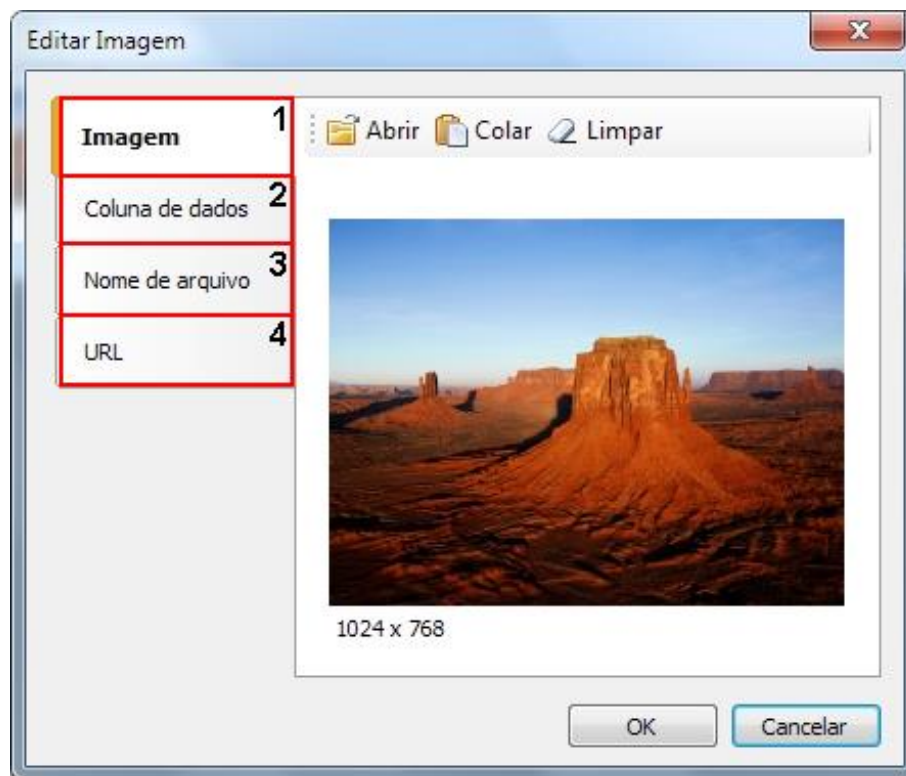
| Propriedade             | Descrição   |
|-------------------------|---|
| <b>AllowExpressions</b> | Permite utilizar (ou não) expressões no texto.  |
| <b>Brackets</b>         | Como nas propriedades do objeto Texto, aqui são definidos os indicadores da expressão.          |
| <b>DataColumn</b>       | Nesta propriedade pode ser informada uma coluna de dados à qual a expressão será vinculada.     |
| <b>Texto</b>            | Como no objeto Texto, aqui está contido o texto informado no objeto, porém aqui em formato RTF. |
| <b>Padding</b>          | O espaçamento, em pixels.   |

### O objeto Figura

este objeto pode ser utilizado para incluir uma **imagem** nos **formatos**: BMP, PNG, JPG, GIF, TIFF, ICO, EMF, WMF. Com a ajuda do objeto Figura, você pode imprimir o logotipo de sua companhia, a foto de um funcionário, ou qualquer outra informação gráfica. O objeto é exibido da seguinte maneira no editor:



Para abrir o editor do objeto, basta dar um **duplo-clique**. No editor, estão disponíveis as seguintes opções:



1. A imagem é carregada a partir de um arquivo e é salva **dentro do relatório**. A imagem é armazenada na propriedade **Image**.
2. Imagem de uma **coluna de dados**. O nome da coluna é armazenado na propriedade **DataColumn**.

3. A imagem é **carregada de um arquivo com o nome informado**. Este nome é armazenado na propriedade **ImageLocation**. A imagem não é armazenada no relatório, e **deve ser distribuída** com o relatório.
4. A imagem é **carregada de um endereço na internet** toda vez que o relatório é criado. A URL é armazenada na propriedade **ImageLocation**.

### *As propriedades do objeto Figura*

As propriedades do objeto Figura são as seguintes:

| Propriedade             | Descrição  |
|-------------------------|--|
| <b>Angle</b>            | Ângulo de rotação, em graus. Valores possíveis para esta propriedade são 0, 90, 180 e 270.   |
| <b>SizeMode</b>         | O tamanho da imagem.   |
| <b>Transparency</b>     | O grau de transparência da imagem, entre 0 e 1. O valor 0(padrão) significa que a imagem é opaca.                                      |
| <b>TransparentColor</b> | A cor que será transparente quando exibida a imagem.   |
| <b>Image</b>            | A imagem.  |
| <b>DataColumn</b>       | A coluna de dados à qual o objeto está vinculado.  |
| <b>ImageLocation</b>    | esta propriedade contém o nome (e caminho) do arquivo ou URL.  |
| <b>Padding</b>          | O espaçamento, em pixels.  |
| <b>ShowErrorMessage</b> | Exibe a imagem de erro, no caso do objeto estar vazio. Esta propriedade só é válida quando a imagem é referenciada através de uma URL. |

### *O objeto Formas*

O objeto formas possui um submenu de seleções, com formas que podem ser inseridas no relatório. As formas que podem ser inseridas são:

- Linha (horizontal e diagonal);
- Retângulo;
- Retângulo Arredondado;
- Elipse;
- Triângulo;
- Losango.

### *Propriedades do objeto Formas*

O objeto Formas pode ser separado em duas subcategorias: Linhas e Formas. As linhas possuem as seguintes propriedades:

| Propriedade     | Descrição  |
|-----------------|--|
| <b>Diagonal</b> | esta propriedade define se a linha pode ou não ser diagonal. |

**Startcap,  
EndCap**

esta propriedade define as pontas da linha. Podem ser utilizadas as formas elipse, retângulo, diamante e flecha. O tamanho da ponta pode ser definido utilizando as subpropriedades Width e Height. Podem ser configuradas ambas as pontas da linha (início e fim).

As formas restantes (retângulo, retângulo arredondado, elipse, triângulo e losango) possuem as seguintes propriedades:

| Propriedade | Descrição  |
|-------------|--|
| Shape       | Determina o tipo de forma.   |
| Curve       | É utilizada com a forma Retângulo Arredondado. Permite a definição da curvatura. |

É recomendado que, ao invés de utilizar **formas ou linhas**, sejam utilizadas as **próprias bordas** dos objetos, para simplificar o relatório e evitar possíveis problemas ao exportar o mesmo.

### O objeto Código de Barras

este objeto permite a inclusão de um código de barras válido para leitura. O resultado é semelhante à imagem abaixo:



este objeto suporta os seguintes tipos de códigos de barras:

| Código          | Caracteres | Símbolos permitidos    |
|-----------------|------------|------------------------|
| 2/5 Interleaved |            | 0 a 9                  |
| 2/5 Industrial  |            | 0 a 9                  |
| 2/5 Matriz      |            | 0 a 9                  |
| Codabar         |            | 0 a 9, -\$/+.+         |
| Code128         |            | 128 caracteres ASCII   |
| Code39          |            | 0 a 9, A a Z, -.*\$/+% |
| Code39 Extended |            | 128 caracteres ASCII   |
| Code93          |            | 0 a 9, A a Z, -.*\$/+% |
| Code93 Extended |            | 128 Caracteres ASCII   |
| EAN8            | 8          | 0 a 9                  |
| EAN13           | 13         | 0 a 9                  |
| MSI             |            | 0 a 9                  |



|              |    |          |
|--------------|----|----------|
| PostNet      |    | 0 a 9    |
| UPC-A        | 12 | 0 a 9    |
| UPC-E0       | 6  | 0 a 9    |
| UPC-E1       | 6  | 0 a 9    |
| Supplement 2 | 2  | 0 a 9    |
| Supplement 5 | 5  | 0 a 9    |
| PDF417       |    | Qualquer |
| Datamatrix   |    | Qualquer |

### *Propriedades do objeto Código de Barras*

O objeto Código de Barras possui as seguintes propriedades:

| Propriedade | Descrição  |
|-------------|--|
| Barcode     | esta propriedade contém definições específicas do código de barras.  |
| Angle       | Ângulo de rotação do código de barras, em graus. Os valores que podem ser utilizados são 0, 90, 180 e 270.   |
| Zoom        | Permite aproximar o código de barras. É utilizado em conjunto com a propriedade AutoSize.  |
| AutoSize    | Faz com que o objeto se expanda para exibir todo o código de barras. Se esta propriedade for desativada, o código de barras irá se adaptar ao tamanho dos limites do objeto. |
| ShowText    | Determina se é necessário exibir os caracteres em formato legível abaixo da barra.   |
| DataColumn  | Coluna de dados à qual o objeto é vinculado.   |
| Expression  | Expressão que retorna os dados do código de barras.  |
| Texto       | Os dados do código de barras.  |
| Padding     | O espaçamento, em pixels.  |

### *Propriedades específicas dos tipos de códigos de barras*

Os diferentes tipos de códigos de barras possuem propriedades específicas, acessíveis ao expandir a propriedade Barcode e listadas abaixo:

| Propriedade  | Descrição   |
|--------------|---|
| WideBarRatio | Todos os tipos de códigos de barra lineares possuem esta propriedade. Ela determina a escala de largura da barra. Para a maioria dos tipos, este valor deve ficar entre 2 e 3.  |
| CalcChecksum | Como a propriedade anterior, esta é comum a todos os códigos lineares. Determina se o dígito de checagem será calculado automaticamente. Se esta propriedade estiver desativada, você deve informar o dígito de checagem dos dados do código de barras. |
| AutoEncode   | esta propriedade é específica do tipo Code128. Este código possui 3 diferentes codificações: A, B e C. Ao utilizar o objeto, deve-se ou   |

|                        |   |
|------------------------|---|
|                        | ajustar a codificação explicitamente nos dados do código de barras, ou colocar o valor True nesta propriedade. Neste caso a codificação será escolhida automaticamente.   |
| <b>AspectRatio</b>     | Propriedade específica do tipo PDF417. Determina a proporção altura-para-largura e é utilizada para calcular automaticamente o tamanho do código de barras.   |
| <b>CodePage</b>        | Propriedade específica dos tipos PDF417 e Datamatrix. Determina a codificação de página utilizada para converter caracteres não-ASCII. Por exemplo, a codificação padrão do windows é 1251.   |
| <b>Colunas, Rows</b>   | estas propriedades são específicas do tipo PDF417. Determinam o número de colunas e linhas em um código de barras. Se ambas as propriedades forem 0, o tamanho do código de barras será calculado automaticamente. Neste caso a propriedade <b>AspectRatio</b> será utilizada também. |
| <b>CompactionMode</b>  | Propriedade específica do tipo PDF417, que determina o modo de compactação.   |
| <b>ErrorCorrection</b> | Propriedade específica do tipo PDF417 que determina o nível da correção de erros.   |
| <b>PixelSize</b>       | Propriedade específica do tipo PDF417 que determina o tamanho do elemento de código de barras na tela, em pixels. Como regra, a altura deve ser maior do que o elemento largura por 3 vezes ou mais.  |
| <b>Encoding</b>        | Propriedade específica do tipo Datamatrix que determina o tipo de codificação de dados.   |
| <b>PixelSize</b>       | Propriedade específica do tipo Datamatrix que determina o tamanho do elemento de código de barras, em pixels.   |
| <b>SymbolSize</b>      | Propriedade específica do tipo Datamatrix, que determina o tamanho do símbolo de código de barras.  |

### O objeto Caixa de Verificação

este objeto exibe uma *checkbox* no relatório. Este aparece no editor da seguinte maneira:



este objeto possui dois estados: marcado e desmarcado. Este status pode ser alterado de 3 maneiras:

- Altere o estado da propriedade **Checked**;
- Vincule o objeto a uma coluna de dados utilizando a propriedade **DataColumn**;
- Defina a expressão que retorna um valor **True** ou **False** na propriedade **Expression**.

### Propriedades do objeto Caixa de Verificação

A propriedade Caixa de Verificação possui as seguintes propriedades:





| Propriedade                           | Descrição  |
|---------------------------------------|--|
| <b>CheckedSymbol, UncheckedSymbol</b> | estas propriedades determinam o símbolo que é exibido no objeto dependendo do estado.                                  |
| <b>CheckColor</b>                     | Determina a cor do símbolo.  |
| <b>CheckWidthRatio</b>                | Determina a largura do símbolo. Podem ser utilizados valores entre 0.2 e 2.  |
| <b>HidelfUnchecked</b>                | Esconde o objeto se o mesmo estiver desmarcado.  |
| <b>Checked</b>                        | Controla o estado do objeto.   |
| <b>DataColumn</b>                     | A coluna de dados à qual o objeto é vinculado. A coluna deve retornar dados do tipo <i>boolean</i> ou <i>integer</i> . |
| <b>Expression</b>                     | Uma expressão que deve retornar <b>True</b> ou <b>False</b> .  |

### O objeto Tabela

O objeto Tabela, como diz o nome, insere uma tabela simples no relatório.

As propriedades do objeto são as seguintes:

| Propriedade          | Descrição  |
|----------------------|--|
| <b>ColumnCount</b>   | Define o número de colunas.  |
| <b>RowCount</b>      | Define o número de linhas.   |
| <b>FixedColunas</b>  | Determina quantas colunas na tabela são fixas. Colunas fixas formam o cabeçalho da tabela. A impressão do cabeçalho é controlado pela propriedade <b>RepeatHeaders</b> . |
| <b>FixedRows</b>     | Determina quantas linhas na tabela são fixas. Colunas fixas formam o cabeçalho da tabela. A impressão do cabeçalho é controlado pela propriedade <b>RepeatHeaders</b> .  |
| <b>RepeatHeaders</b> | Permite a impressão do cabeçalho da tabela em cada nova página. Esta propriedade é compatível apenas com tabelas que são formadas dinamicamente.                         |

### O objeto Matriz

Como o objeto **Tabela**, este é composto por linhas, colunas e células. O número de linhas e colunas que irão compor a matriz dependem dos dados presentes nesta.

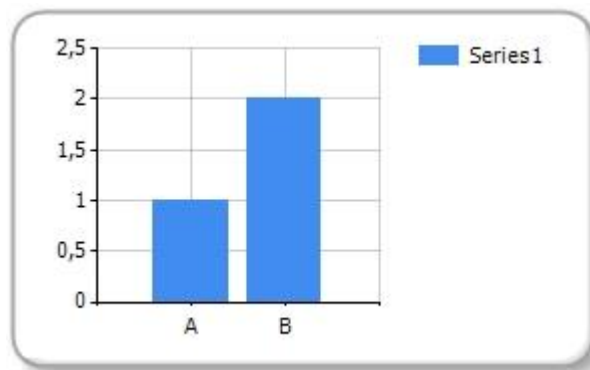
este objeto possui as seguintes propriedades:

| Propriedade | Descrição |
|-------------|-----------|
|-------------|-----------|

|                       |  |
|-----------------------|--|
| <b>RepeatHeaders</b>  | Se a matriz é dividida em várias páginas, esta propriedade permite a impressão do cabeçalho em cada página.  |
| <b>CallSideBySide</b> | Determina como as células serão localizadas se a matriz possui vários níveis de células de dados. As variantes possíveis são lado-a-lado e abaixo.   |
| <b>Style</b>          | Permite definir um estilo para toda a matriz, dentre uma lista de estilos predefinidos.  |
| <b>AutoSize</b>       | Permite que o tamanho da matriz seja calculado automaticamente.  |
| <b>DataSource</b>     | Conecta a matriz a uma fonte de dados. Esta propriedade é configurada automaticamente quando é arrastada uma coluna de dados até a matriz. Entretanto, ao utilizar expressões nas células, é importante verificar se esta propriedade está configurada corretamente. |
| <b>Filter</b>         | Contém uma expressão para a filtragem de dados que será aplicados à fonte de dados da matriz.  |

### O objeto MS Gráfico

este objeto permite a inserção de gráficos. Há mais de 30 tipos diferentes disponíveis: barras, colunas, áreas, linhas, bolhas, pizza, financeiro, pirâmide, etc. O objeto possui a seguinte forma (que pode variar de acordo com o tipo de gráfico selecionado):



As propriedades do objeto são as seguintes:

| Propriedade   | Descrição  |
|---|--|
| <b>Chart</b>  | Referência a um objeto Microsoft Chart.  |
| <b>AlignXValues</b>   | esta propriedade permite alinhar X valores em diferentes séries de gráficos. É utilizado se o gráfico possui duas ou mais séries diferentes. |
| <b>AutoSeriesColumn, AutoSeriesColor, AutoSeriesSortOrder</b> | Permite definir séries criadas automaticamente.  |
| <b>DataSource</b>   | Permite conectar o gráfico à fonte de dados.   |
| <b>Filter</b>   | Contém a expressão para filtragem de dados que será aplicada à fonte de dados do gráfico.  |



### O objeto Código Postal

este objeto permite a impressão de um código postal em envelopes. Aceita a exibição de caracteres numéricos de 0 a 9. Ao inserir, o objeto fica aproximadamente como na imagem abaixo:

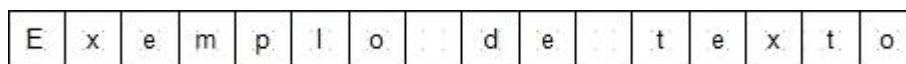


este objeto possui as seguintes propriedades:

| Propriedade                         | Descrição  |
|-------------------------------------|--|
| <b>SegmentCount</b>                 | O número de segmentos em um código.  |
| <b>Segment Width, SegmentHeight</b> | O tamanho de cada segmento do código.  |
| <b>Spacing</b>                      | O espaçamento entre dois segmentos.  |
| <b>ShowGrid</b>                     | Determina se a grade é exibida ou não.   |
| <b>ShowMarkers</b>                  | Determina se os marcadores (linhas horizontais acima do código) são exibidos ou não. |
| <b>DataColumn</b>                   | Coluna de dados à qual o objeto está vinculado.                                      |
| <b>Expression</b>                   | A expressão que retorna o código postal.   |
| <b>Texto</b>                        | Texto contendo um código postal.   |

### O objeto Cellular Text

este objeto exibe um texto com cada caractere exibido em sua célula individual. É utilizado normalmente para imprimir formulários em aplicações financeiras. O objeto tem aproximadamente a aparência abaixo:



este objeto é diretamente relacionado ao objeto Texto. O mesmo método de conexão de dados através de expressões pode ser utilizado aqui.

O objeto Cellular Texto possui as seguintes propriedades:


| Propriedade                     | Descrição  |
|---------------------------------|--|
| <b>CellWidth, CellHeight</b>    | Determinam o tamanho da célula (respectivamente, largura e altura). Se ambos os valores forem 0, o tamanho será calculado automaticamente dependendo da fonte utilizada. |
| <b>HorzSpacing, VertSpacing</b> | Determinam o espaçamento vertical e horizontal entre células adjacentes.   |

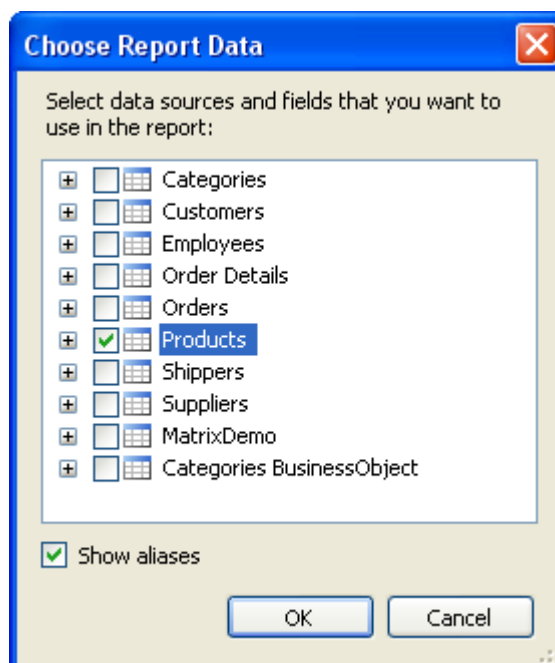
## Seu primeiro relatório com o CIGAM Report

Vamos criar um relatório simples, que imprime uma lista de produtos. Iremos utilizar a tabela **Products** que pode ser encontrada na base de dados de demonstração como nossa fonte de dados.

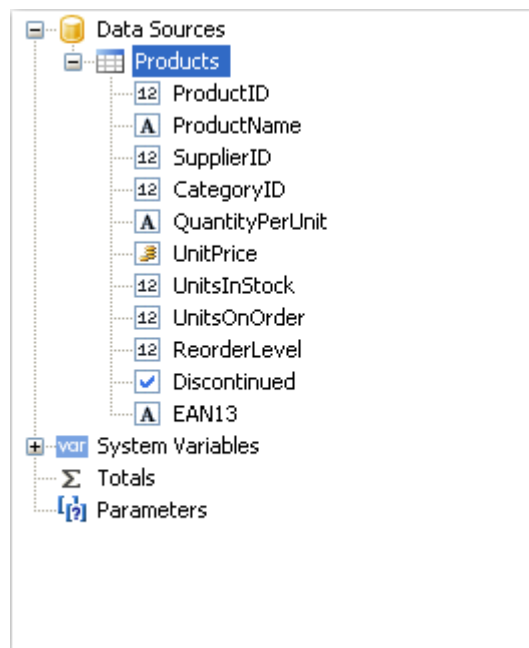
### Exemplo 1: Criando um relatório manualmente

Neste exemplo, iremos criar um relatório manualmente. Para isto, iremos fazer o seguinte:

- Pressione o botão  na barra de ferramentas, e na janela **Adicionar Novo Item**, selecione **Relatório em branco**;
- No menu **Dados**, selecione a opção **Escolher Dados do Relatório** e marque a fonte de dados **Products**:



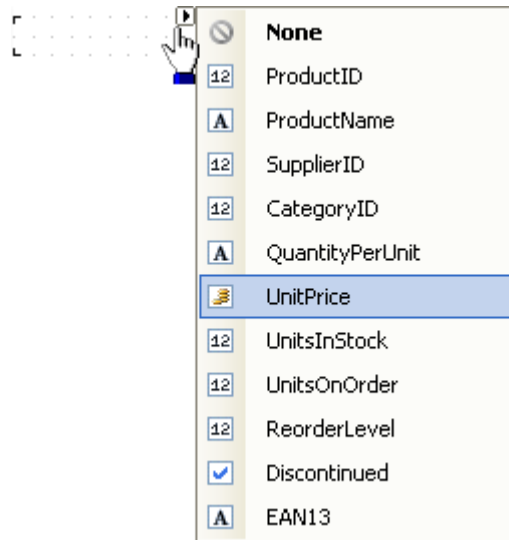
- Vá até a janela **Dados** no editor. Expanda o item **Fontes de Dados**, e então o item **Products**;




- Arraste a coluna **ProductName** para a banda **Dados**. O CIGAM Report cria um objeto **Texto**, que está conectado a esta coluna, e um cabeçalho para a mesma;
- Iremos criar a coluna de dados **UnitPrice** através de outro método. Para tanto, pressione o botão **Texto** na barra de ferramentas de objetos:




- Clique e mova o ponteiro para a banda **Dados**. Você verá que o aplicativo oferecerá a inserção de um objeto. Escolha a posição desejada e clique com o mouse para inserir o objeto;
- Clique no pequeno botão no canto direito do objeto. Você verá uma lista de colunas de dados. Selecione o item **UnitPrice** na lista:



- Crie o objeto **Texto** – o cabeçalho da coluna **UnitPrice**. Posicione o mesmo na banda **Cabeçalho de Página**. Dê um duplo-clique no objeto e escreva o texto **Unit Price**;
- Crie o objeto **Texto** – o título do relatório. Posicione o mesmo na banda **Título do Relatório** e insira o texto **Produtos**;
- Defina **Negrito** como o estilo para todos os objetos que são inseridos nas bandas **Cabeçalho de Página** e **Título do Relatório**. Para isto, selecione os objetos (segurando a tecla **Shift** para selecionar mais de um), e clique no botão . Após isto, o relatório no editor ficará assim:


|                |                        |                      |
|----------------|------------------------|----------------------|
| Report Title   | <b>PRODUCTS</b>        |                      |
| Page Header    | <b>ProductName</b>     | <b>Unit Price</b>    |
| Data: Products | [Products.ProductName] | [Products.UnitPrice] |
| Page Footer    |                        |                      |

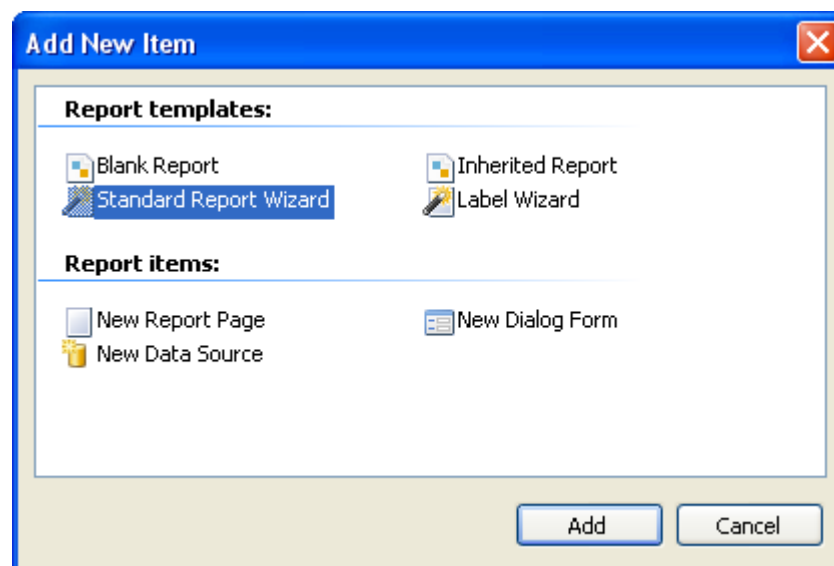
- Para gerar o relatório, clique no botão  na barra de ferramentas. O relatório será construído e exibido na janela de pré-visualização:

| PRODUCTS                     |            |
|------------------------------|------------|
| ProductName                  | Unit Price |
| Chai                         | 18         |
| Chang                        | 19         |
| Aniseed Syrup                | 10         |
| Chef Anton's Cajun Seasoning | 22         |
| Chef Anton's Gumbo Mix       | 21,35      |
| Grandma's Boysenberry Spread | 25         |
| Uncle Bob's Organic Dried    | 30         |
| Northwoods Cranberry Sauce   | 40         |
| Mishi Kobe Niku              | 97         |
| Ikura                        | 31         |

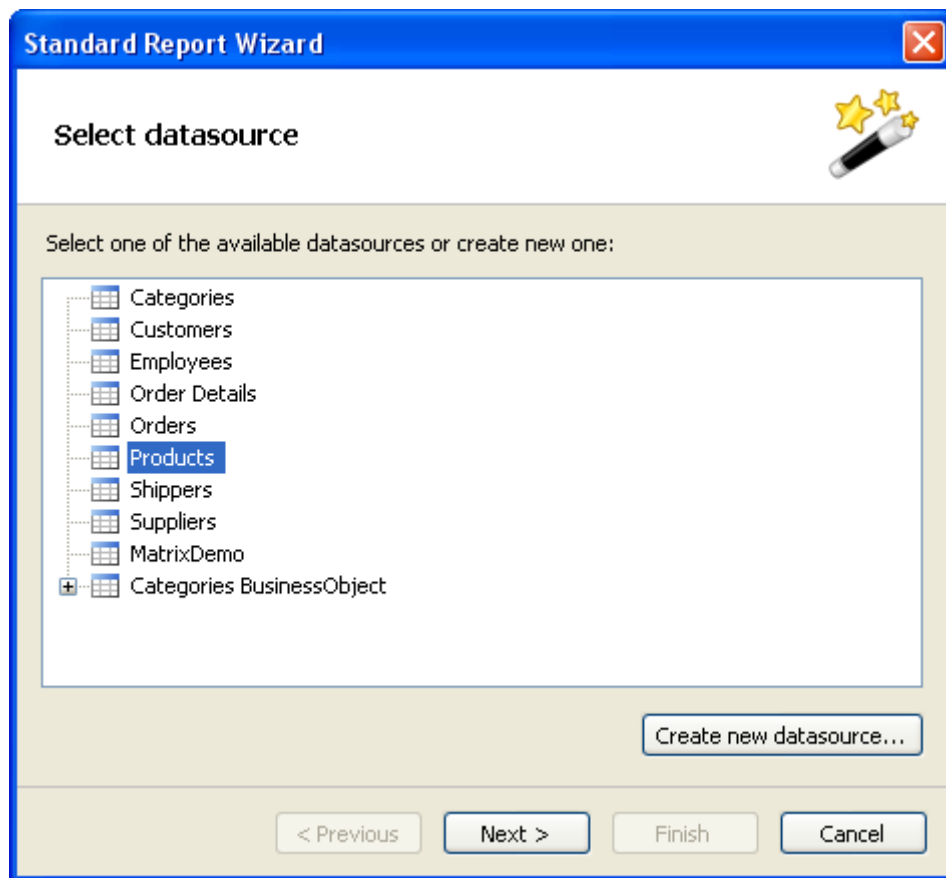
### Exemplo 2: Criando um relatório com o Auxiliar

Neste exemplo, iremos criar um relatório com o auxílio do **Assistente de Relatório Padrão**. Para tanto, faça o seguinte:

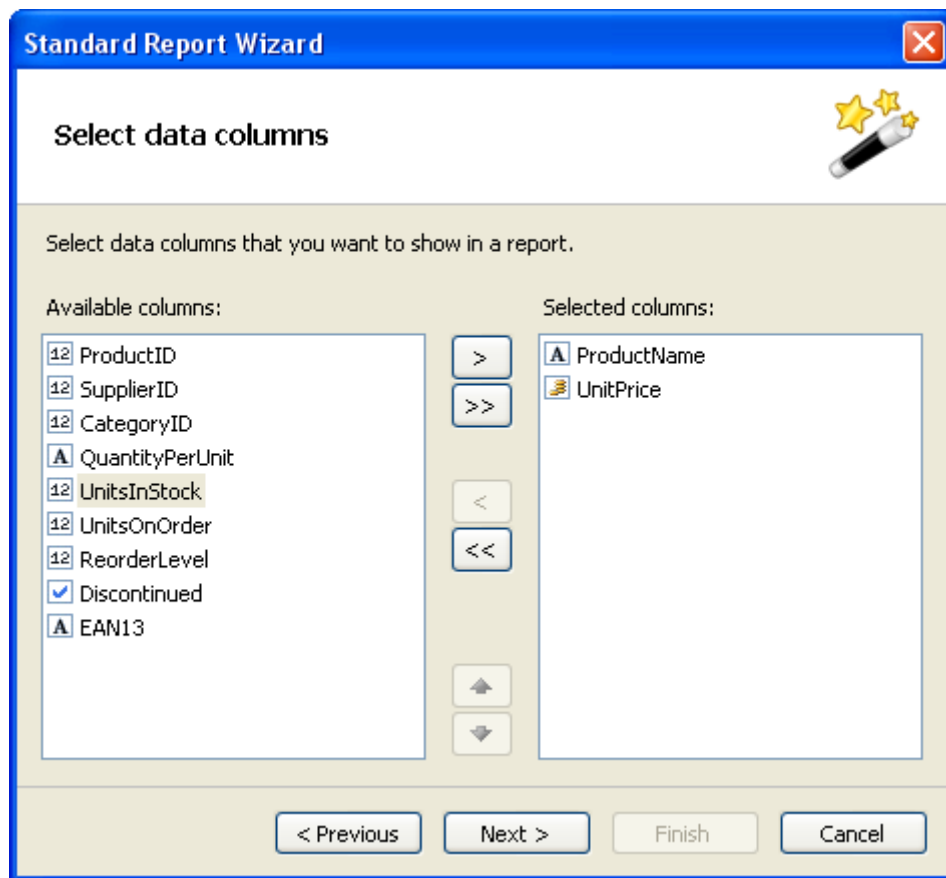
- Pressione o botão  na barra de ferramentas e na janela **Adicionar novo item**, selecione **Assistente de Relatório Padrão**:



- No primeiro passo do assistente, escolha a tabela **Products** e clique em **Próximo**:



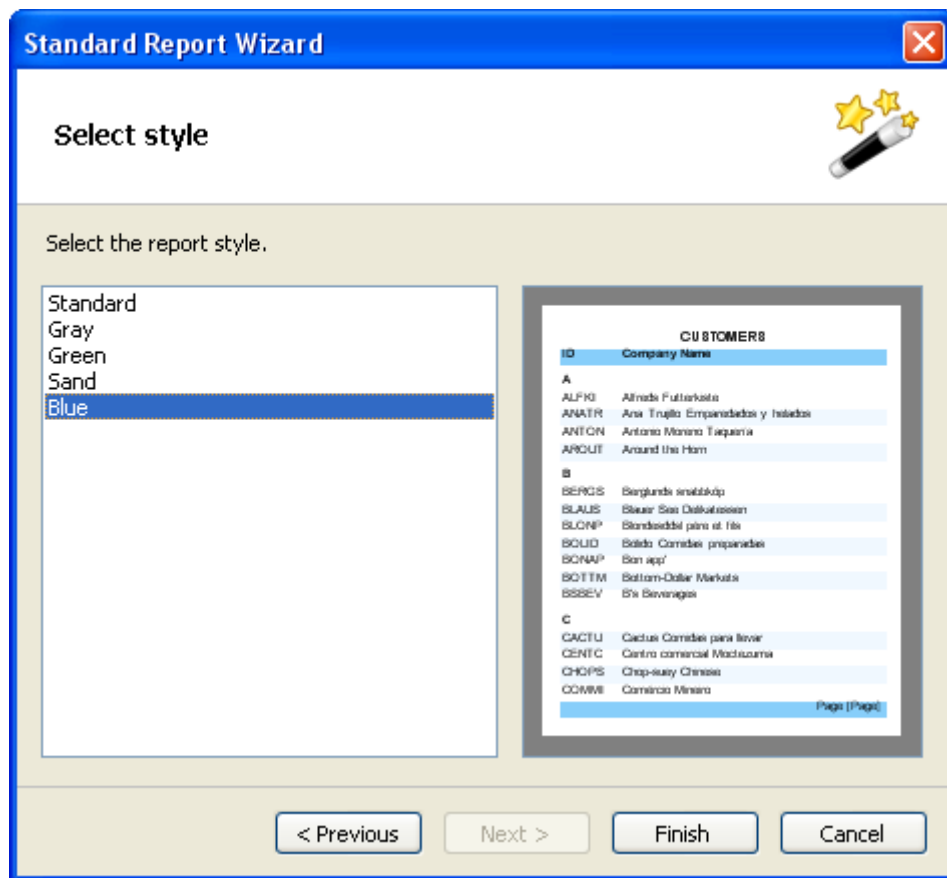
- No segundo passo, selecione as colunas de dados **Products** e **UnitPrice**:



The image shows a screenshot of the 'Standard Report Wizard' dialog box. The title bar reads 'Standard Report Wizard' with a close button. The main heading is 'Select data columns'. Below this, there is a sub-heading 'Select data columns that you want to show in a report.' The dialog is divided into two main sections: 'Available columns:' and 'Selected columns:'. The 'Available columns:' list includes ProductID, SupplierID, CategoryID, QuantityPerUnit, UnitsInStock, UnitsOnOrder, ReorderLevel, Discontinued (checked), and EAN13. The 'Selected columns:' list includes ProductName and UnitPrice. Between the two lists are navigation buttons: '>', '>>', '<', '<<', and up/down arrows. At the bottom, there are four buttons: '< Previous', 'Next >', 'Finish', and 'Cancel'. A small icon of a pen and stars is in the top right corner of the dialog area.


- Os próximos passos podem ser passados diretamente, clique no botão **Próximo**;
- No último passo do assistente de construção de relatórios, selecione o estilo **Blue** e clique no botão **Concluir**:





- O CIGAM Report cria o seguinte relatório:

| Report Title   | Products               |                      |
|----------------|------------------------|----------------------|
| Page Header    | ProductName            | UnitPrice            |
| Data: Products | [Products.ProductName] | [Products.UnitPrice] |
| Page Footer    | Page [Page]            |                      |

- Para gerar um relatório, clique no botão  na barra de ferramentas. O relatório será construído e exibido na tela de pré-visualização:

| Products                        |           |
|---------------------------------|-----------|
| ProductName                     | UnitPrice |
| Chai                            | 18        |
| Chang                           | 19        |
| Aniseed Syrup                   | 10        |
| Chef Anton's Cajun Seasoning    | 22        |
| Chef Anton's Gumbo Mix          | 21,35     |
| Grandma's Boysenberry Spread    | 25        |
| Uncle Bob's Organic Dried Pears | 30        |
| Northwoods Cranberry Sauce      | 40        |
| Mishi Kobe Niku                 | 97        |
| Ikura                           | 31        |
| Quep Pasaños                    | 21        |



Universidade Corporativa CIGAM

[www.cigam.com.br](http://www.cigam.com.br)

[treinamento@cigam.com.br](mailto:treinamento@cigam.com.br)

## Criação de relatório:

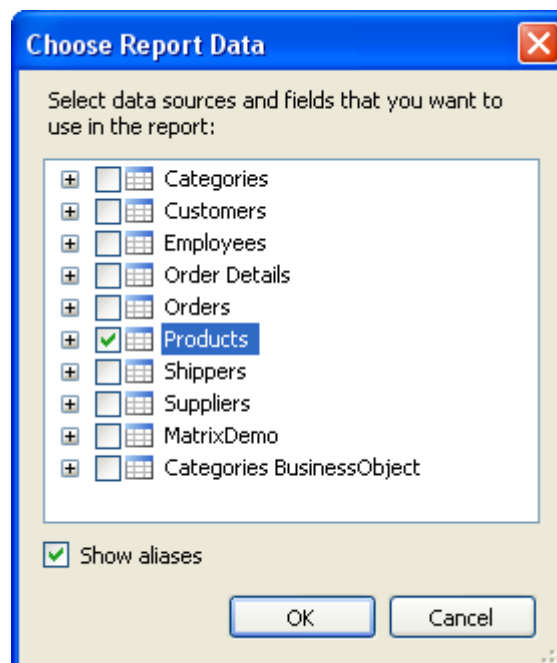
Neste capítulo nós vamos ver métodos de criar relatórios comuns. Para criar qualquer relatório, você precisa fazer o seguinte:

1. Escolher (ou criar) os dados que serão utilizados no relatório.
2. Criar a estrutura do relatório, adicionando as bandas necessárias no relatório.
3. Conectar a banda a uma fonte de dados.
4. Colocar os objetos de **Texto** nas bandas para imprimir os dados.

## Escolhendo os dados para um relatório

Antes de você começar a construir um relatório, você deve escolher quais dados irão ser impressos no relatório. Você pode fazer isso de duas maneiras:

- Você pode escolher uma das fontes de dados, as quais foram registradas no relatório por um método de programação. esta pode ser feita no menu **Dados > Escolher os Dados para o Relatório...**, marcando a fonte de dados necessária;



- Você pode criar uma nova fonte de dados no menu **Dados > Adicionar fonte de dados....**



Leia mais sobre fontes de dados no capítulo **Fontes de Dados**.

Logo após você ter escolhido a fonte de dados, a mesma aparecerá na janela **Dados**. Agora você pode usar esta fonte para seu relatório. Muitos relatórios usam somente uma fonte de dados. Para relatórios do tipo **Detalhe-Mestre**, você deve escolher duas fontes de dados, relacionadas entre si (você pode ler mais sobre relacionamentos no capítulo **Fontes de Dados**). Muitas fontes de dados podem também ser necessárias em um relatório, o qual imprime dados de fontes relacionadas.

## Layout dinâmico

Quando criamos um relatório, freqüentemente é necessário imprimir um texto que não se sabe o tamanho. Por exemplo, pode ser uma descrição de bens. Neste caso, os seguintes itens deverão ser resolvidos:

- Calcular o tamanho de um objeto, assegurando-se de envolver todo o texto;
- Calcular o tamanho da banda, assegurando-se de que envolva o objeto com uma quantidade variável de textos;
- Mova ou troque a altura de outros objetos contidos na banda, assegurando-se que estes não estejam obstruindo o *layout* geral do relatório.

Estes itens podem ser resolvidos utilizando algum objeto ou propriedades da banda:

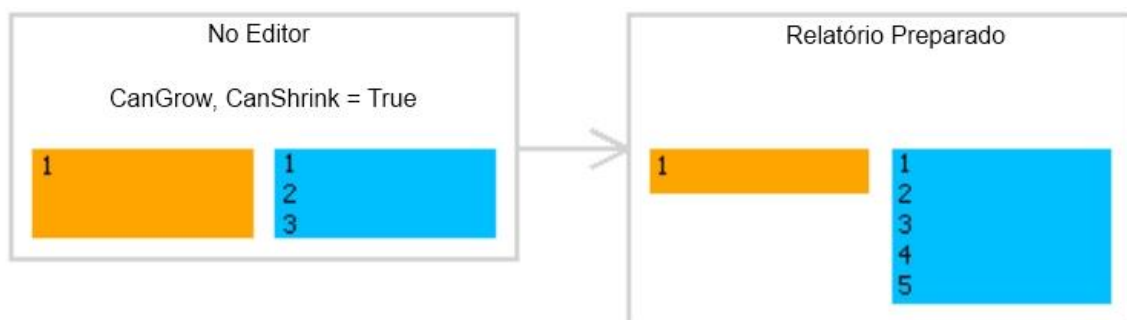
- As propriedades **CanGrow** e **CanShrink** permitem calcular a altura de um objeto automaticamente;
- A propriedade **ShiftMode** permite mover objetos que estão localizados abaixo dos objetos que expandem.
- As propriedades **Anchor** e **Dock** permitem controlar o tamanho dos objetos dependendo do tamanho da banda.

Todas estas propriedades serão mostradas abaixo.

## Propriedades CanGrow, CanShrink

Toda a banda e objetos de relatório têm estas propriedades. Eles determinam quando um objeto pode aumentar ou diminuir dependendo do tamanho e de seu conteúdo. Se ambas as propriedades estão **desativadas**, o objeto sempre tem seu **tamanho especificado no editor**.

estas propriedades são muito úteis se é preciso imprimir um texto que não se sabe o tamanho, no editor. Para um objeto acomodar um texto inteiro, ele precisa ter as propriedades **CanGrow** e **CanShrink** habilitadas.



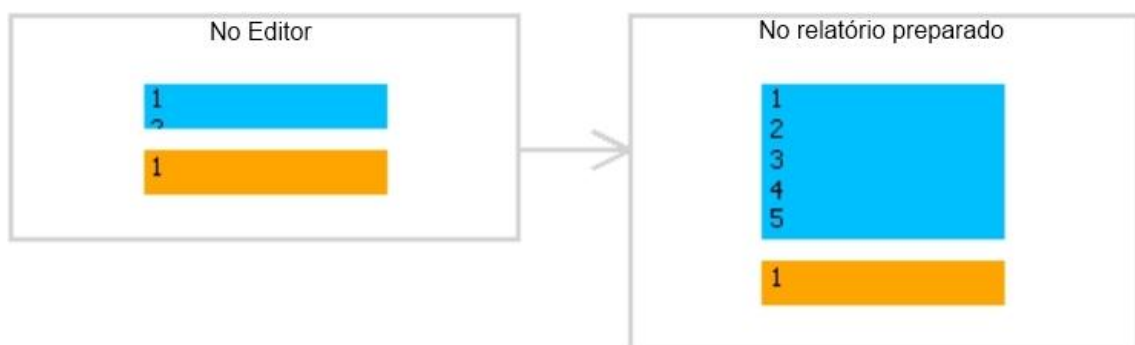
Os seguintes objetos podem afetar a altura da banda:

- Texto;

- Texto Formatado;
- Figura (com a propriedade **AutoSize** habilitado);
- Tabela.

### Propriedade ShiftMode

Todo objeto do relatório possui esta propriedade. Esta propriedade é acessível somente na janela **Propriedades**. Um objeto com a propriedade **ShiftMode** habilitada irá se mover para cima e para baixo, se o objeto acima aumentar ou diminuir.



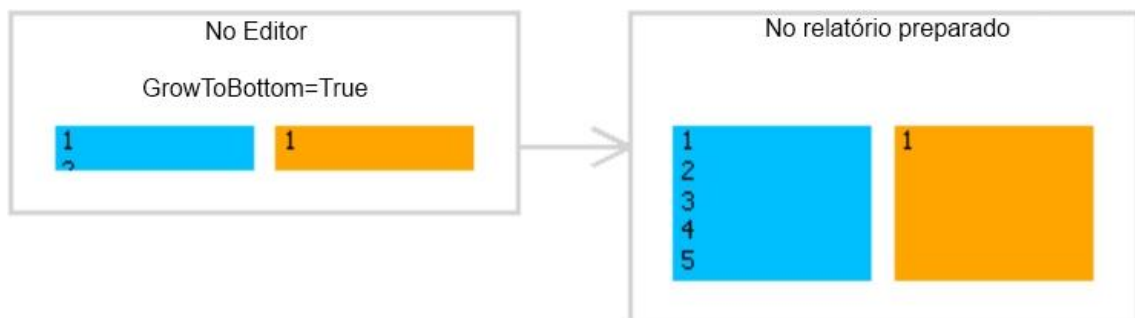
A propriedade **ShiftMode** pode ter um dos seguintes valores:

- **Always** (por padrão): Significando que o objeto deve se deslocar sempre.
- **Never**: o objeto não se desloca.
- **WhenOverlapped**: o objeto se desloca se um outro objeto se sobrepõe a ele.

É conveniente o uso desta propriedade quando se imprime informações em uma tabela de formulário, quando muitas células da tabela estão localizadas no topo umas das outras e podem ter quantidade variável de texto.

### Propriedade GrowToBottom

Todo objeto do relatório possui esta propriedade. Quando um objeto é impresso com esta propriedade, ele alonga-se até o limite inferior da banda:



Isto é necessário quando a informação é impressa em uma tabela de formulário. Em uma linha da tabela pode haver muitos objetos que podem alongar-se. Com esta propriedade é possível definir a altura de todos os objetos de acordo com a altura máxima da banda.

### Propriedade Anchor

Todo objeto do relatório possui esta propriedade. Ela determina como o objeto vai trocar sua posição e/ou seu tamanho quando a banda em que está contido muda de tamanho. Utilizando o **Anchor**, o objeto se expande ou move ao mesmo tempo em que o *container*.

O *container* em muitos casos será a banda, mas isso não é uma regra – podem ser também os objetos de uma **Tabela** ou da **Matriz**.

A propriedade **Anchor** pode ter um dos seguintes valores, e também uma combinação deles:

| Value         | Description  |
|---------------|--|
| <b>Left</b>   | Fixa o lado esquerdo do objeto. Quando o tamanho do <i>container</i> mudar, o objeto não irá se mover para esquerda/direita.   |
| <b>Top</b>    | Fixa o topo do objeto. Quando a altura do <i>container</i> mudar, o objeto não irá se mover para cima/baixo.   |
| <b>Right</b>  | Fixa o lado direito do objeto. Quando a largura do <i>container</i> mudar, a distância entre o lado direito e o <i>container</i> será constante. Se o lado esquerdo do <i>container</i> estiver fixado também, então o objeto irá aumentar e deslocar-se ao mesmo tempo que o <i>container</i> . |
| <b>Bottom</b> | Fixa a parte de baixo do objeto. Quando a altura do <i>container</i> mudar, a distância entre a parte de baixo do objeto e o <i>container</i> será constante. Se a parte de cima do objeto estiver fixado também, o objeto irá aumentar e deslocar-se ao mesmo tempo que o <i>container</i> .    |

Por padrão, o valor da propriedade é **Left, Top**. Isso significa que, quando o tamanho do *container* mudar, o objeto não irá mudar. Na tabela abaixo, combinações de alguns valores freqüentemente usando são dados:

| Value                           | Description  |
|---------------------------------|--|
| <b>Left, Top</b>                | Valor padrão. O objeto não muda quando o tamanho do <i>container</i> muda.   |
| <b>Left, Bottom</b>             | O objeto move-se para cima/baixo quando a altura do <i>container</i> muda. A posição do objeto em relação a parte inferior do <i>container</i> não muda. |
| <b>Left, Top, Bottom</b>        | Quando a altura do <i>container</i> muda, a altura do objeto de maneira assíncrona com ele.  |
| <b>Left, Top, Right, Bottom</b> | Quando a largura e a altura do <i>container</i> mudam, o objeto aumenta e desloca-se junto com ele.  |



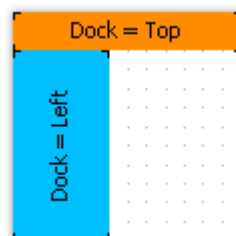
## Propriedade Dock

Todo relatório possui esta propriedade. Esta propriedade determina em qual lado do *container* o objeto será fixado.

A propriedade **Dock** pode ter um dos seguintes valores:

| Value  | Description   |
|--------|---|
| None   | Valor padrão. O objeto não é fixado.  |
| Left   | O objeto é fixado à direita do <i>container</i> . A altura do objeto será igual à altura do <i>container</i> .          |
| Top    | O objeto é fixado ao topo do <i>container</i> . A largura do objeto será igual à largura do <i>container</i> .          |
| Right  | O objeto é fixado à direita do <i>container</i> . A altura do objeto será igual à altura do <i>container</i> .          |
| Bottom | O objeto é fixado à parte inferior do <i>container</i> . A largura do objeto será igual à largura do <i>container</i> . |
| Fill   | O objeto ocupa todo o espaço livre do <i>container</i> .  |

\*No entanto, isso não é válido se muitos objetos foram fixados ao mesmo tempo. A figura abaixo mostra dois objetos, o primeiro foi fixado no topo do *container* e o segundo fixado na esquerda.



Como visto, a altura do segundo objeto é igual à altura do espaço livre, que é o que sobra depois de fixar o primeiro objeto.

O comportamento da fixação depende da ordem de criação do objeto. Você pode mudar a ordem no menu de contexto de um objeto. Para fazer isso, selecione tanto o item de menu **Trazer para a Frente** quanto **Enviar para Trás**.

## Formatando

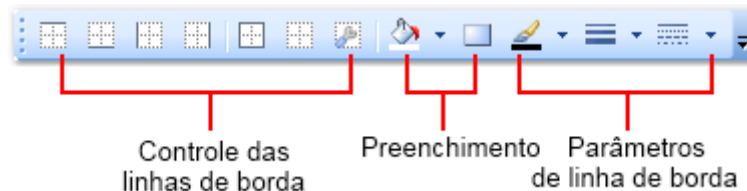
Nesta seção, nós vamos olhar as seguintes questões:

- Trocar a aparência do objeto;
- Trocar o formato de impressão dos valores;
- Trocar automaticamente a aparência de um objeto quando preencher algum tipo de condição;
- Esconder valores desnecessários;


- Destacar as linhas de dados em diferentes cores.

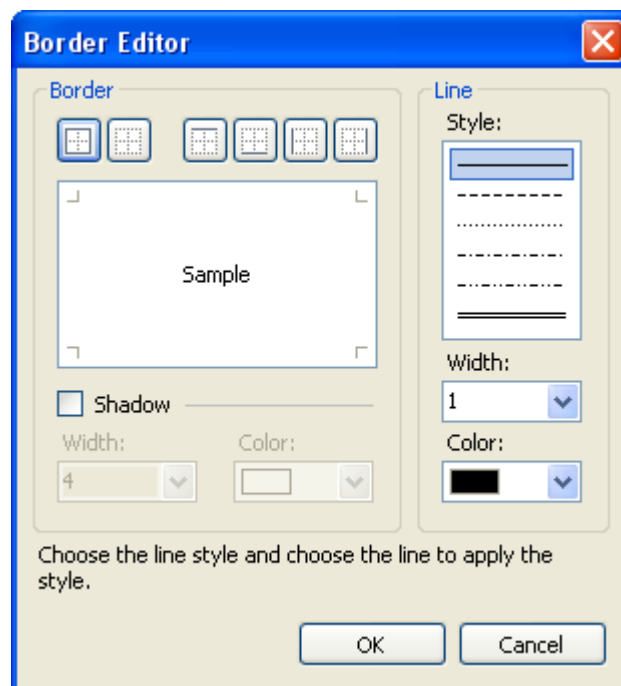
### Borda e preenchimento:



Quase todos os relatórios possuem borda e preenchimento. Para trabalhar com esta propriedade, use a barra de ferramentas **Bordas e preenchimento**.

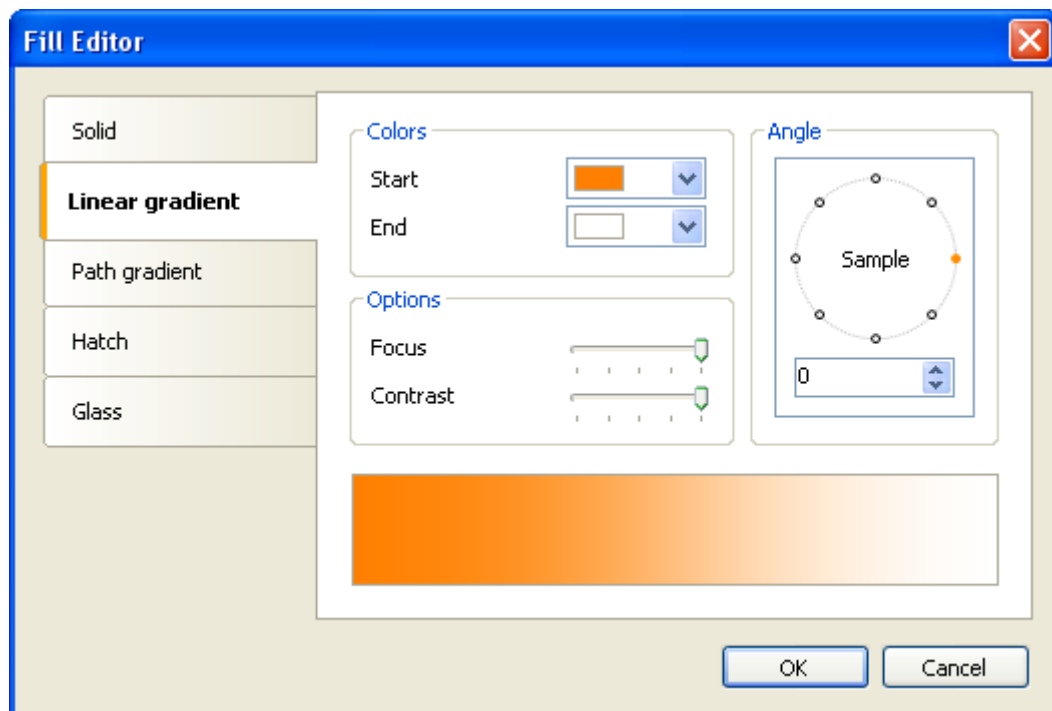


A borda do objeto consiste em quatro linhas. Cada linha pode ter largura, cor e estilo diferentes.

Os botões da barra de ferramentas afetam todas as linhas no quadro. O botão  mostra uma janela que permite editar cada linha separadamente.

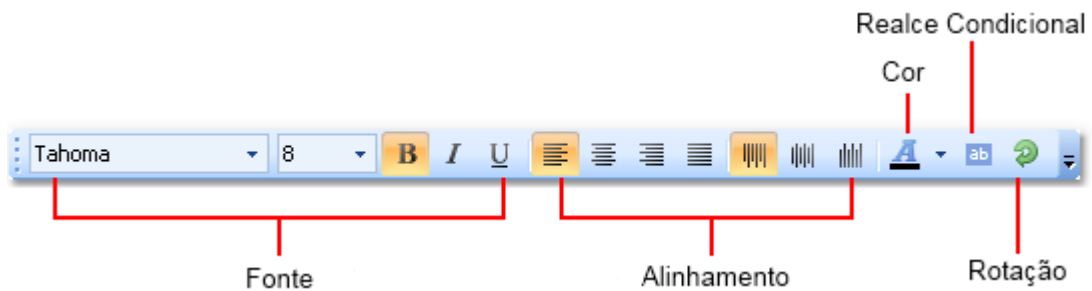


Para trabalhar com preenchimento, há dois botões. O botão  permite escolher a cor de um preenchimento sólido. O botão  mostra uma janela que permite escolher entre dois tipos diferentes de preenchimento:



### Formatação de Texto

Para mudar a aparência do objeto Texto, use a barra de ferramentas de Texto:

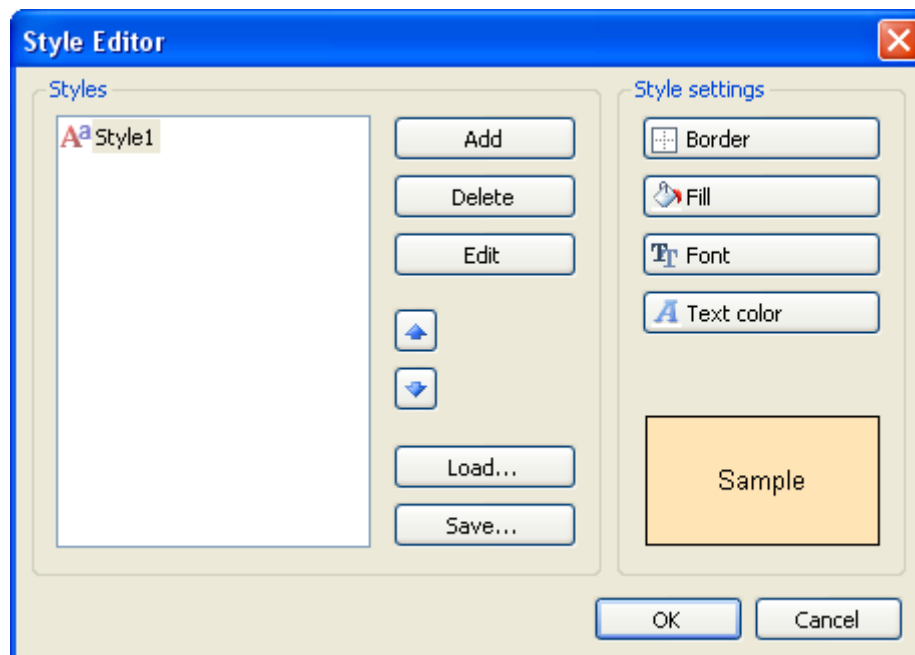


### Estilos

Para alterar a aparência de um objeto, você pode utilizar estilos. Estilo é um conjunto das seguintes propriedades:

- Borda;
- Preenchimento;
- Fonte;
- Cor do texto.

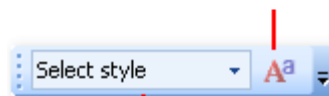
A lista de estilos é guardada em um relatório. Você pode controlar ela tanto do menu **Relatório > Estilos...** ou pelo botão **A<sup>a</sup>** na barra de ferramentas de estilo:



Você pode definir o estilo de um objeto das seguintes formas:

- Definir a propriedade **Style** na janela **Propriedades**;
- Usar a barra de ferramentas de estilos.

Chamar o editor de estilos



Estilos Disponíveis

---

Se a barra de ferramentas não está aparecendo na tela, habilite ela no menu **Visualizar > Barras de Ferramentas**.

---

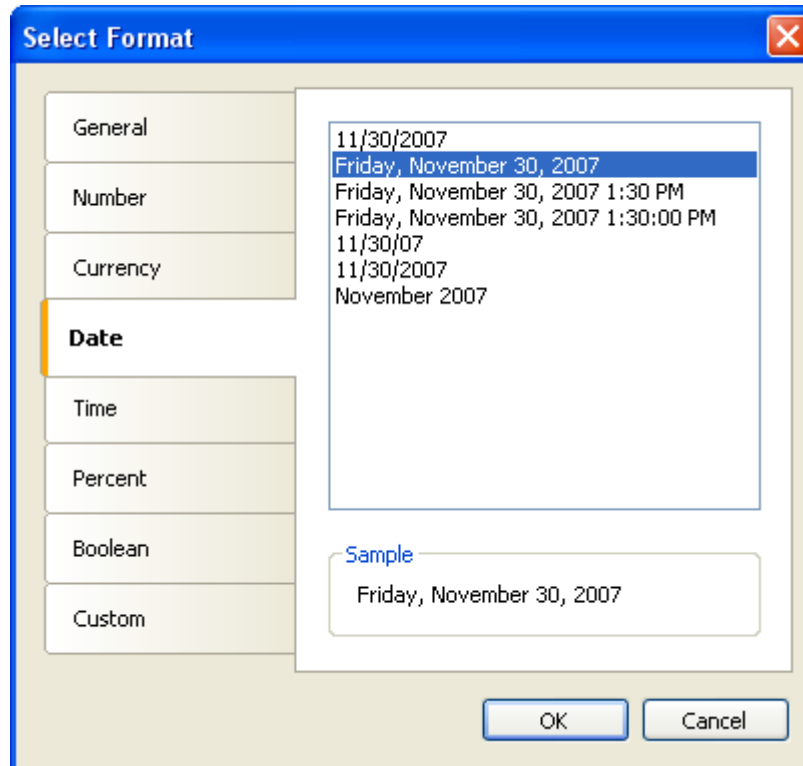
Quando você define o estilo do objeto, a aparência do objeto irá mudar de acordo com as opções do estilo. Quando você mudar as opções de estilo, o objeto com aquele estilo vai mudar automaticamente.

### Formatação de Dados

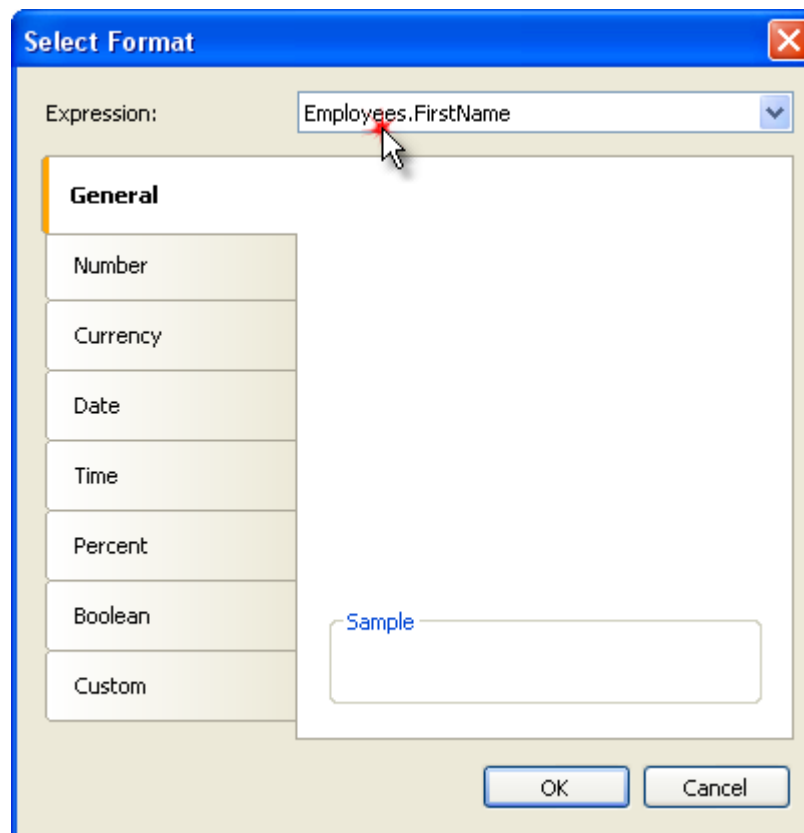
Para imprimir dados textuais no relatório, o objeto Texto é utilizado. Ele aplica a formatação padrão para todos os dados que vêm de uma fonte de dado. Por exemplo, as colunas da fonte de dados **DateTime** serão impressas no seguinte modo (isso depende das configurações regionais do seu sistema):

11.10.2008 18:04:52

Se você precisa imprimir somente a data, você deve definir a formatação da data. Para fazer isso, clique com o botão direito no objeto Texto para abrir o seu menu de contexto. No menu, escolha o item **Formatar....** Você verá a janela de edição de formato:




Você pode escolher um dos tipos de formatação disponível ou definir o seu próprio tipo. Para fazer isso, selecione a formatação **Custom**. Se o objeto Texto contém diversas colunas ou expressões de data, você deve escolher o formato apropriado para cada um deles. Para fazer isso, selecione a expressão no topo da janela, então selecione o formato:

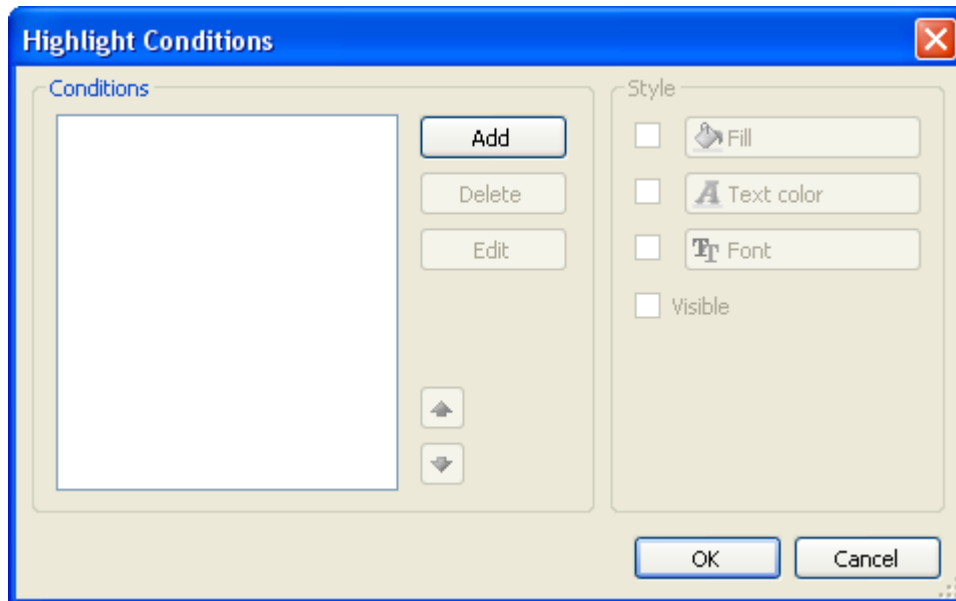


Você deve também formatar a data utilizando o método **String.Format**. Consiga ajuda sobre este método na **MSDN(Microsoft Developer Network)**.

Today is [String.Format({0;d},[Date])]

### Destaque

Há uma possibilidade de mudar a aparência do objeto **Texto** dependendo das condições dadas. Por exemplo, um objeto pode ser marcado com a cor vermelha se ele possui um valor negativo. Esta característica é chamada de **Destaque**. Para defini-la, selecione o objeto Texto e clique no botão  na barra de ferramentas de Texto. Você verá a seguinte janela:



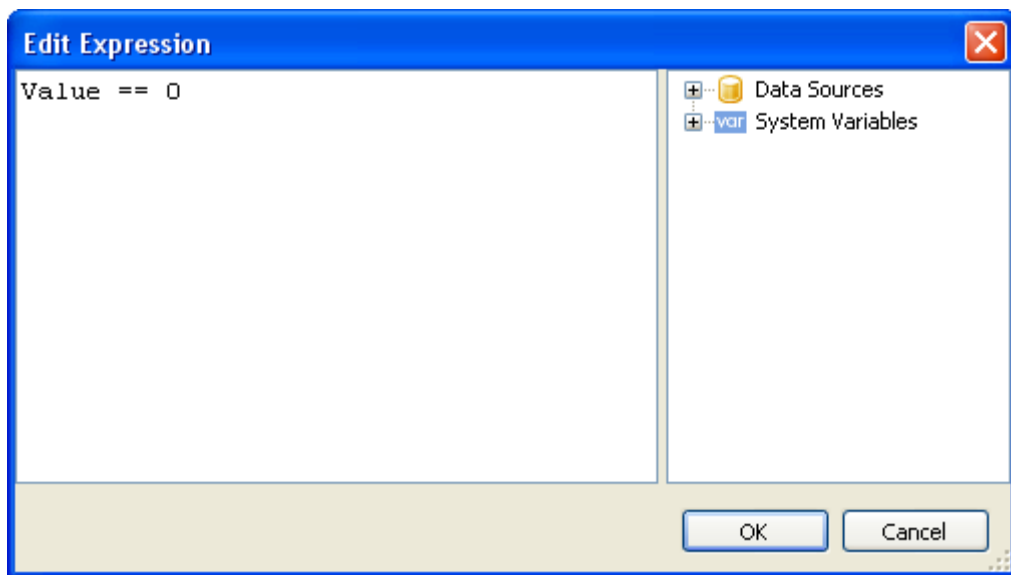
É possível definir um ou mais condições e definir o estilo de cada condição. O estilo pode conter uma ou mais configurações:

- Preenchimento;
- Cor do texto;
- Fonte;
- Visibilidade do objeto.

Você pode indicar quais configurações precisam ser mudadas quando a condição é alcançada. Para isso, marque a configuração necessária na *checkbox*. Por padrão, um novo estilo contém uma configuração – a cor do texto.

Para criar uma nova condição, clique no botão **Adicionar**. Você irá ver um editor de expressão:





Aqui é possível escrever qualquer expressão que retorne um **valor booleano**. Em muitos casos você irá usar a variável **Value**, que contém o valor da impressão presente no momento.

Vamos olhar o seguinte exemplo: nós temos um objeto Texto, no qual nós imprimiremos a quantidade de produtos no estoque:

**[Produtos.UnidadesEmEstoque]**

Nós queremos pintar o objeto de vermelho se a quantidade de produtos for igual à zero. Para isso, nós criamos a seguinte condição:

**Value == 0**

Neste caso, nós usamos a variável **Value**, a qual tem um valor impresso. Se há muitas expressões em um objeto, então esta variável terá o valor da última expressão. Ao invés de Value, você pode usar uma coluna de dados:

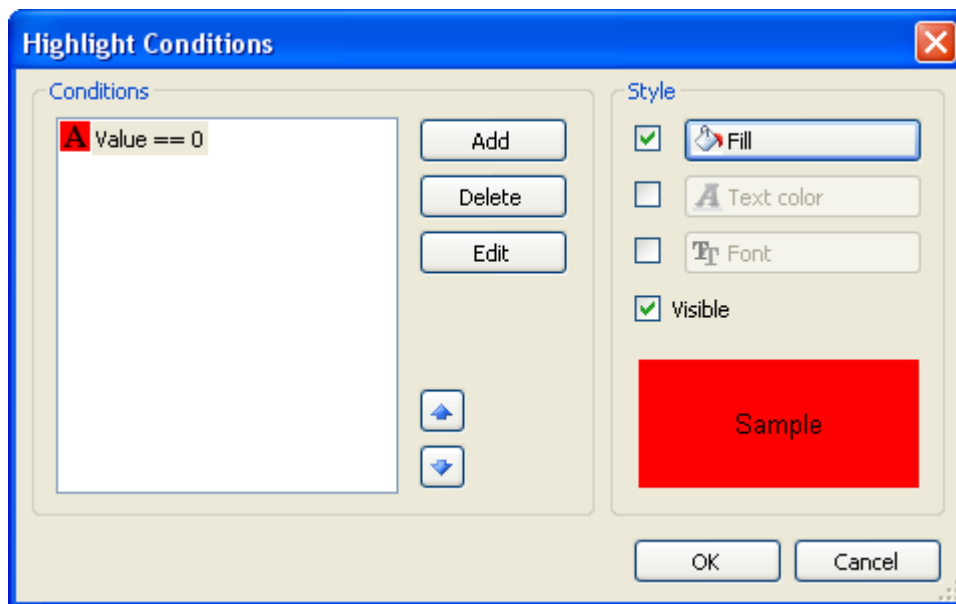
**[Produtos.UnitsInStock] == 0**

---

A expressão está escrita em padrão **C#**. Isto é válido se a linguagem escolhida for **C#**. Para VisualBasic.Net você deve utilizar somente um símbolo =. A linguagem do relatório pode ser trocada no menu **Relatório > Opções**.

---

Configure o estilo para a condição dada de uma maneira que somente preenchimento possa ser utilizado, e escolha a cor vermelha:



Quando é impresso um objeto que possui o valor zero, ele será vermelho. Vamos fazer deste exemplo um pouco mais complexo, então vamos adicionar outra condição. Se a quantidade de unidades no estoque é menor do que 10, ela deve ser impressa em amarelo. Para fazer isso, abra o editor de condição e clique no botão **Adicionar**. A segunda condição será assim:

#### **Value < 10**

No caso de muitas condições terem sido indicadas, o **CIGAM Report** checa todas as condições, começando pela primeira. Se uma condição é atingida, o **CIGAM Report** aplica seus estilos no objeto, e o processo para. É importante colocar as condições na ordem correta. A ordem que nós vimos neste exemplo está correta:

1. **Value == 0**
2. **Value < 10**

Se nós trocarmos as condições, então a marcação funcionará de forma errada.

1. **Value < 10**
2. **Value == 0**

Neste caso o **Value==0**, não será executado, porque quando o valor é zero, a primeira condição será atingida. Para mudar a ordem das condições, use os botões ▲ e ▼.

#### **Escondendo valores zerados**

O objeto Texto tem a propriedade **HideZeros** que pode ser utilizada para esconder valores zerados. Vamos dar uma olhada em um objeto com o seguinte conteúdo:

**Total elements: [CountOfElements]**

Se o valor da variável **CountOfElements** é igual a 0, e a propriedade **HideZeros** está habilitada, o objeto irá imprimir o seguinte:

#### Total elements:

O objeto Texto também tem a propriedade **HideValue** que pode ser usada para esconder o valor de uma expressão que é igual ao valor dado. Por exemplo, se o valor da propriedade é 0, então todos os campos zerados serão escondidos. Esta propriedade pode ser usada também para esconder zeros de datas. Como regra, uma data é como 1.1.0001 ou 1.1.1900. Neste caso o valor da propriedade **HideValue** deve ficar assim:

**1.1.1900 0:00:00**

Como você pode ver, você também precisa indicar o tempo. Isso é necessário porque o valor da data em .Net também contém o tempo.

---

**Importante:** este mecanismo depende das configurações regionais do seu sistema, as quais podem ser definidas no painel de controle. Isso acontece porque o CIGAM Report compara strings usando o método **ToString()**. Este método converte o valor de uma expressão em *string*. Com isso, tenha cuidado ao montar relatórios que pode ser abertos em um computador com diferentes configurações regionais.

---

#### Esconder valores duplicados

O objeto Texto tem a propriedade **Duplicates**, que possibilita controlar a maneira como valores duplicados serão impressos. Esta propriedade pode ser usada se o objeto Texto está na banda Dados. Os valores são considerados duplicados se eles são impressos próximos pelas linhas de dados.

A propriedade **Duplicates** pode ter um dos seguintes valores:

- **Show** – mostra os valores duplicados (por padrão);
- **Hide** – esconder o objeto com o valor duplicado;
- **Clear** – limpa o texto do objeto, mas mostra o objeto;
- **Merge** – concatena objetos com o mesmo valor.

A diferença entre os modos mostrados na figura abaixo:



### Marcação intercalada dos dados

Para melhorar a aparência de um relatório, você pode marcar as linhas pares de dados em diferentes cores. Isso pode ser feito usando a propriedade **EvenStyle** da banda ou dos seus objetos. A propriedade contém o nome do estilo, que será usado para marcar as linhas pares de dados.

---

É preferível usar a propriedade **EvenStyle** de um objeto ao invés de uma banda. Isso evita possíveis problemas quando exportar o relatório.

---

Para configurar a marcação, faça o seguinte:

1. Defina o estilo, o qual será usado para a marcação das linhas. Isso pode ser feito no menu **Relatório > Estilos...**
2. Indicar o nome do novo estilo na propriedade **EvenStyle** da banda ou de seus objetos.

Por padrão, os objetos usam somente a propriedade preenchimento do estilo, dada na propriedade **EvenStyle**. Este comportamento é definido em **EvenStylePriority** – por padrão é **UseFill**. Se você precisar utilizar o restante dos parâmetros, defina a propriedade como **UseAll**.

Um relatório pronto, que use esta técnica, ficará parecido com isso:

| Product name       | Unit price |
|--------------------|------------|
| Chai               | 18,00      |
| Chang              | 19,00      |
| Chartreuse verte   | 18,00      |
| Côte de Blaye      | 263,50     |
| Guaraná Fantástica | 4,50       |
| Ipoh Coffee        | 46,00      |

### Relatório com uma banda Dados

este tipo de relatório é mais frequentemente requerido. Ele permite imprimir uma lista de linhas da fonte de dados. Por exemplo, isso pode ser uma lista de clientes.

### Conectando uma banda à uma fonte de dados

Para imprimir dados de uma fonte de dados, você precisará da banda Dados, a qual deverá estar conectada a fonte de dados. A banda será impressa enquanto houver linhas na fonte de dados.

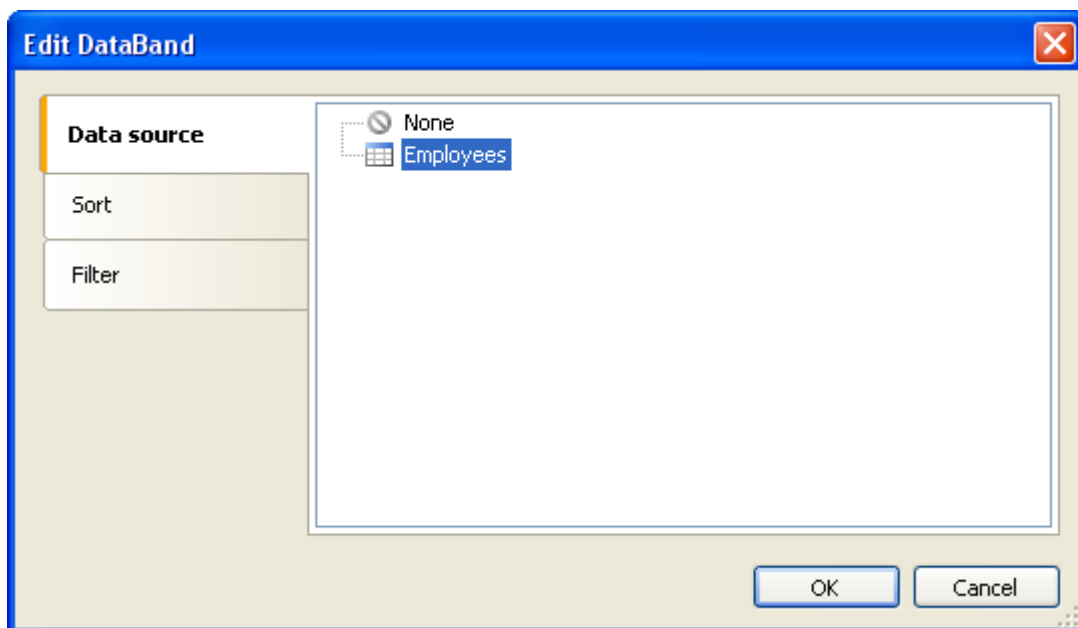
---

Se a banda Dados não está conectada à fonte, ela será impressa uma vez.

---

Quando você cria um novo relatório, ele já irá conter várias bandas vazias, incluindo a banda Dados. Esta banda pode ser também adicionada no relatório através da janela **Configurar Bandas**, escolhendo o item **Relatório > Configurar Bandas...** do menu.

Para conectar a banda aos dados, dê um duplo clique nela. Escolha a fonte de dados na janela de edição e clique **OK**.



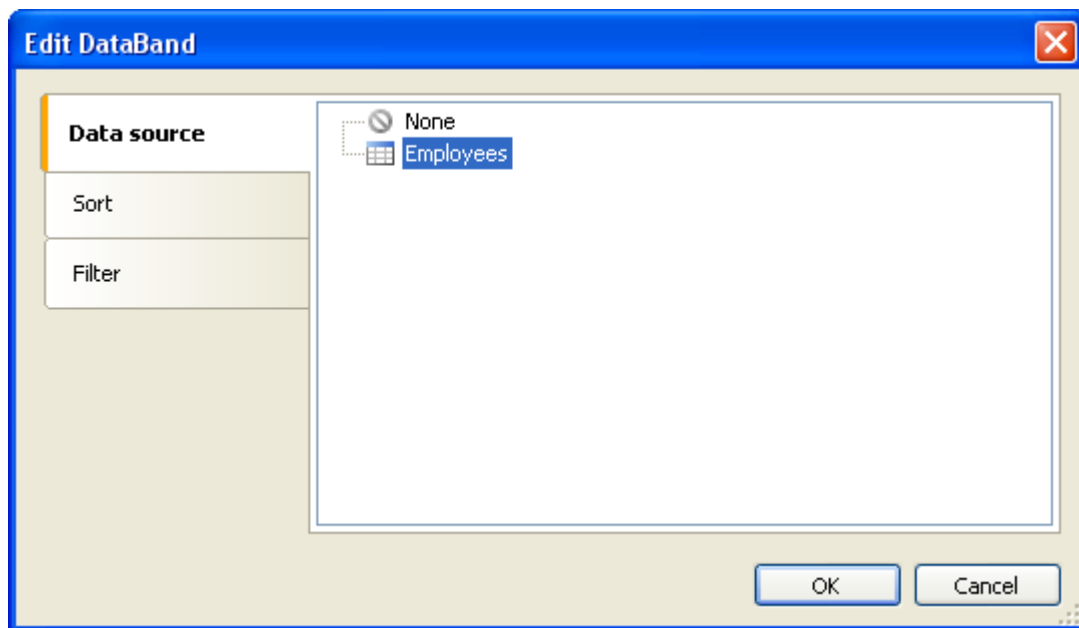
### Imprimindo o texto

Depois de você ter conectado a banda à uma fonte de dados, você pode colocar o objeto Texto na banda, a qual irá mostrar a informação de uma coluna de dados. O método mais rápido para se fazer isso – arraste a coluna da janela **Dados** e solte-a na banda. Leia mais sobre o objeto Texto no capítulo **O Objeto Texto**.

### Misturando os dados

Por padrão, a banda Dados imprime os dados em uma ordem natural. Frequentemente é necessário classificar a data antes de imprimir. Por exemplo, a lista de clientes pode ser confortavelmente apresentada em ordem alfabética.

Você pode controlar a classificação no editor da banda Dados. Para abrir o editor, dê um duplo clique em um espaço livre na banda:

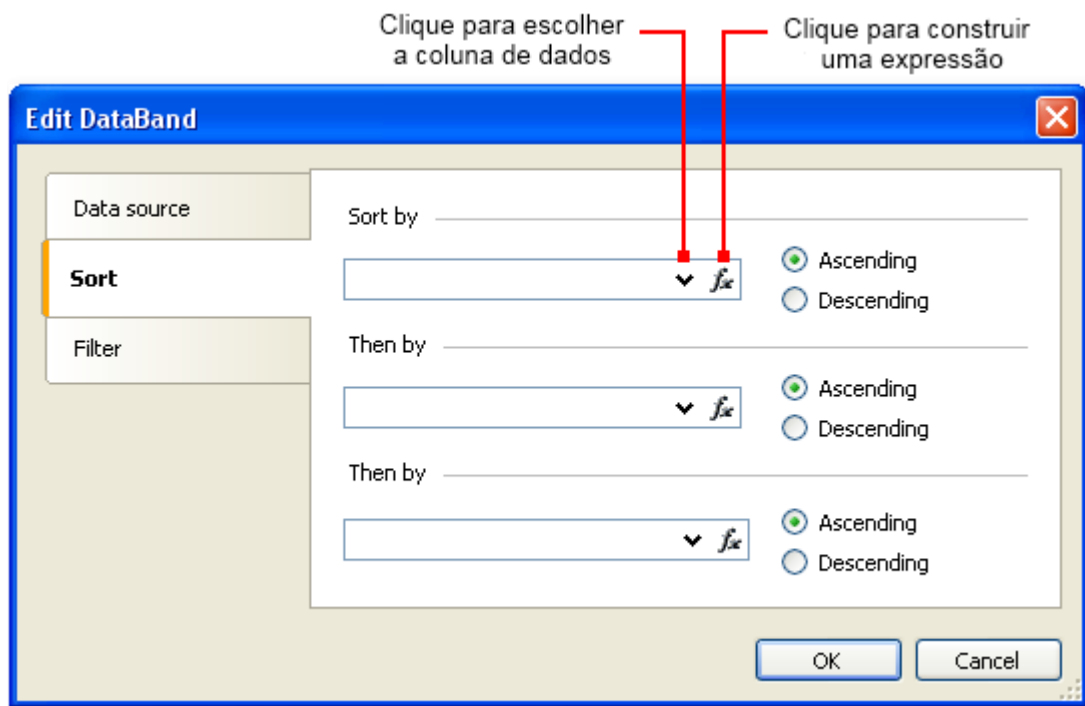


Como critério de classificação, você pode usar tanto uma coluna de dados quanto uma expressão. Você pode indicar várias (porém não mais de três) condições de classificação. Isso pode ser necessário, por exemplo, se você quiser classificar uma lista de clientes pelas cidades onde moram, e depois disso, pelo nome do cliente. Para cada condição, você pode escolher a ordem de classificação – crescente ou decrescente.

Outro método de classificar dados – usar a *query SQL* como fonte de dados. A *query* será executada no servidor de dados e retornará linhas classificadas.

### Filtrando os dados

Para filtrar uma linha, a qual é impressa na banda Dados, abra o editor e troque para a aba **Filter**:



Como um filtro, você pode indicar qualquer expressão correta. Mais detalhes sobre expressões podem ser encontradas no capítulo **Expressions**.

No exemplo acima, o seguinte filtro é usado:

**[Produtos.Discontinued] == false**

Isso significa que serão escolhidas todas as linhas de dados nas quais a *flag Disconnected* for igual a **False** e o preço menor do que 10.

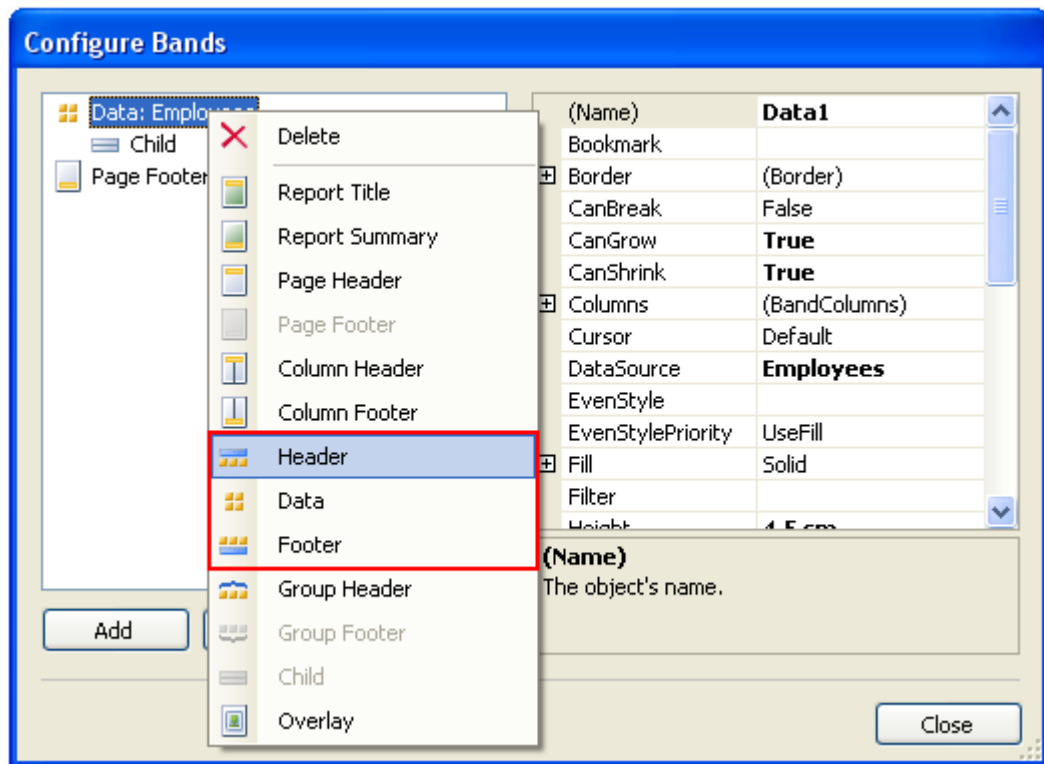
este método de filtragem supõe que a fonte de dados contém todas as linhas, parte do que será filtrado. Se a fonte de dados contém uma grande quantidade de linhas, isso pode seriamente deixar o relatório lento. Neste caso você pode usar a *query SQL* como fonte de dados, na qual você pode realizar a filtragem necessária. A *query* será executada no servidor de dados e retornará somente aquelas linhas que serão necessárias no relatório.

### Cabeçalho e rodapé dos dados

A banda **Dados** pode conter um cabeçalho e um rodapé. O cabeçalho será impresso uma vez antes dos dados, o rodapé será impresso depois de todos os dados.

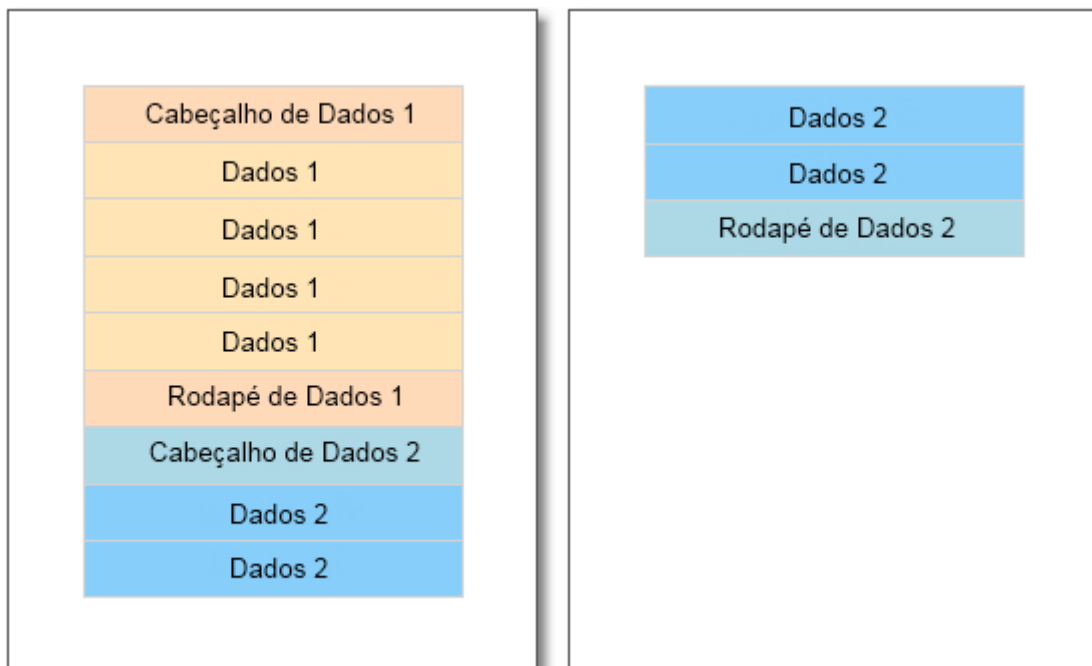
Para adicionar o cabeçalho e o rodapé à uma banda Dados, escolha o item do menu **Relatório > Configurar Bandas**. Na janela, selecione a banda Dados e clique com o botão direito. No menu de contexto, escolha os itens **Header** e/ou **Footer**:



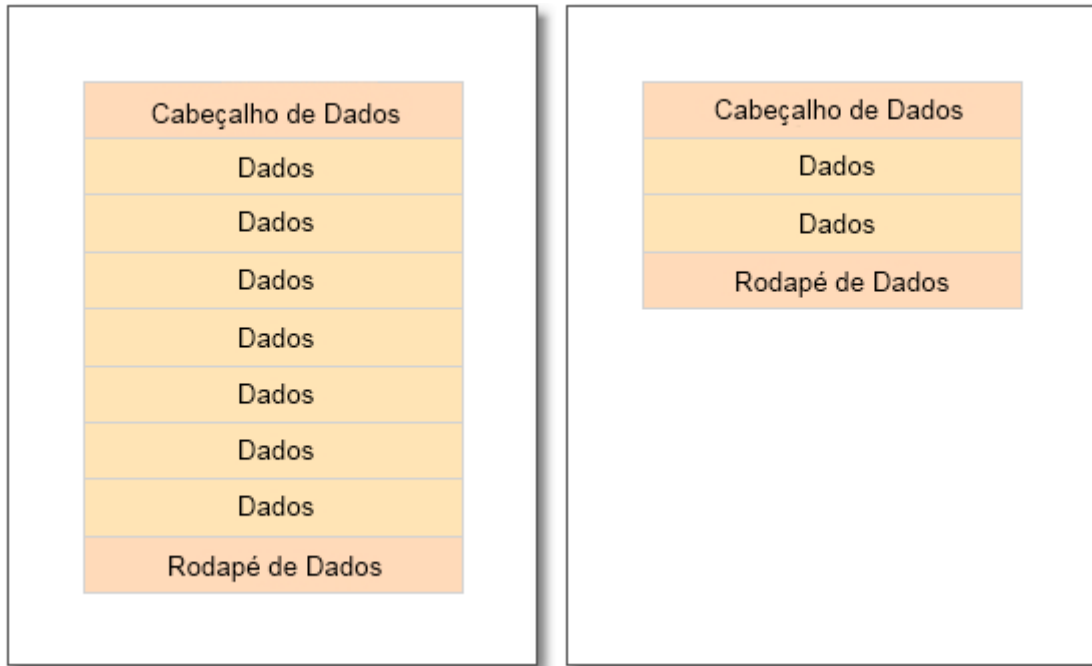


Estas bandas podem ser úteis nas seguintes situações:

- Quando imprimir várias listas em uma página (**Relatórios Mestre-Mestre**). Cada banda de dados, neste caso, pode ter seu próprio cabeçalho e rodapé:



- Quando imprimir uma lista, se a lista não cabe em uma página do relatório preparado. Usando a propriedade **Repeat On Every Page** do cabeçalho/rodapé dos dados, você pode imprimir estas bandas em cada página do relatório:



### Separar dados e mantê-los juntos

Nesta seção, nós vamos ver dois métodos de imprimir dados – **Break** e **Keep together**.

Em um modo regular de impressão de banda, o **CIGAM Report** checa se há espaço suficiente na página para imprimir a banda. Se não há, a banda é impressa inteiramente na próxima página. Se a propriedade **CanBreak** da banda estiver habilitada, o **CIGAM Report** tentará imprimir a parte da banda no espaço disponível, “quebrando” esta.

Uma tentativa para separar a banda pode ser tanto bem sucedida quanto não. Isso depende do tipo do objeto colocado na banda e suas configurações. Os seguintes objetos podem ser separados:

- **Texto**
- **Rich Texto**
- **Tabela**

Estes objetos tem a propriedade **CanBreak** também. Se estiver habilitado, então o objeto pode ser separado. Objetos não-separáveis são sempre mostrados totalmente, onde eles têm espaço suficiente.

Na figura abaixo, é mostrado como uma banda pode ser separada.



O algoritmo de separação nem sempre funciona corretamente. Os artefatos pode ocorrer quando há vários objetos na banda com diferentes tamanhos de fonte.

O objetivo de quebrar ou separar a banda é salvar espaço na folha impressa. O objetivo de manter os dados é contrário: mostrar um conjunto de bandas todos em uma folha. Neste caso haverá muito espaço inutilizado nas folhas, mas os dados são impressos de uma maneira de fácil leitura.

O mecanismo **KeepTogether** permite manter um conjunto de bandas juntas em uma página (ou coluna, se o relatório possui colunas). Se, quando imprimir, os dados chegam ao final da página, o **CIGAM Report** realoca todos os dados que já foram impressos em uma nova página.

Você pode usar o **KeepTogether** nos seguintes casos:

- Imprimir todas as linhas da banda Dados juntas;
- Imprimir todos os elementos de um grupo (cabeçalho, dados, rodapé) juntos;
- Imprimir a linha de uma fonte mestre de dados juntamente com todas as linhas de detalhe (no relatório **Mestre-Detalhe**);
- Imprimir o cabeçalho ou o cabeçalho de dados do relatório junto com ao menos uma linha de dados;
- Imprimir o rodapé ou o rodapé de dados junto com ao menos uma linha de dados;
- Imprimir as bandas pai e filha juntas.

Vamos ver o uso do mecanismo **KeepTogether**.

Para manter todas as linhas de dados ou um grupo de elementos (cabeçalho, dados, rodapé) juntos, habilite a propriedade **KeepTogether**. Esta propriedade é usada nas bandas **Dados** e **Cabeçalho de Grupo**. A figura abaixo mostra como os dados são impressos estando ou não juntos:



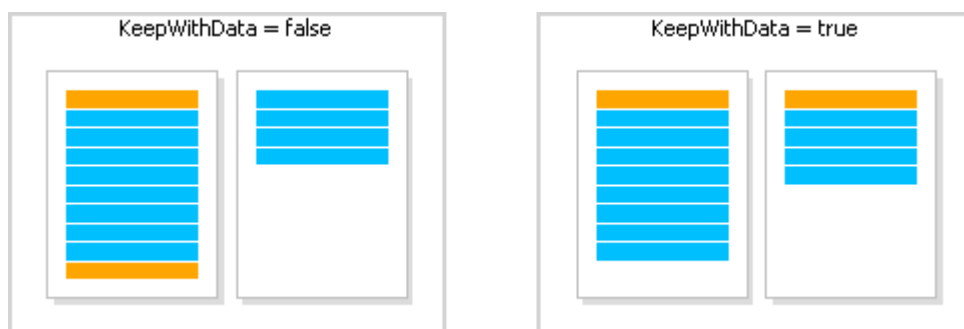
Para manter a linha mestre de dados junto com as linhas detalhes dos dados, habilite a propriedade **KeepDetail** da banda Dados. Esta propriedade é usada em um relatório do tipo **Mestre-Detalhe**:



Para prevenir o sufocamento dos cabeçalhos e rodapés, use a propriedade **KeepWithData**. As seguintes bandas têm esta propriedade:

- Cabeçalho do relatório;
- Rodapé do relatório;
- Cabeçalho dos dados;
- Rodapé dos dados;
- Grupo cabeçalho;
- Grupo rodapé.

Esta propriedade permite manter o cabeçalho/rodapé com ao menos uma linha de dados:

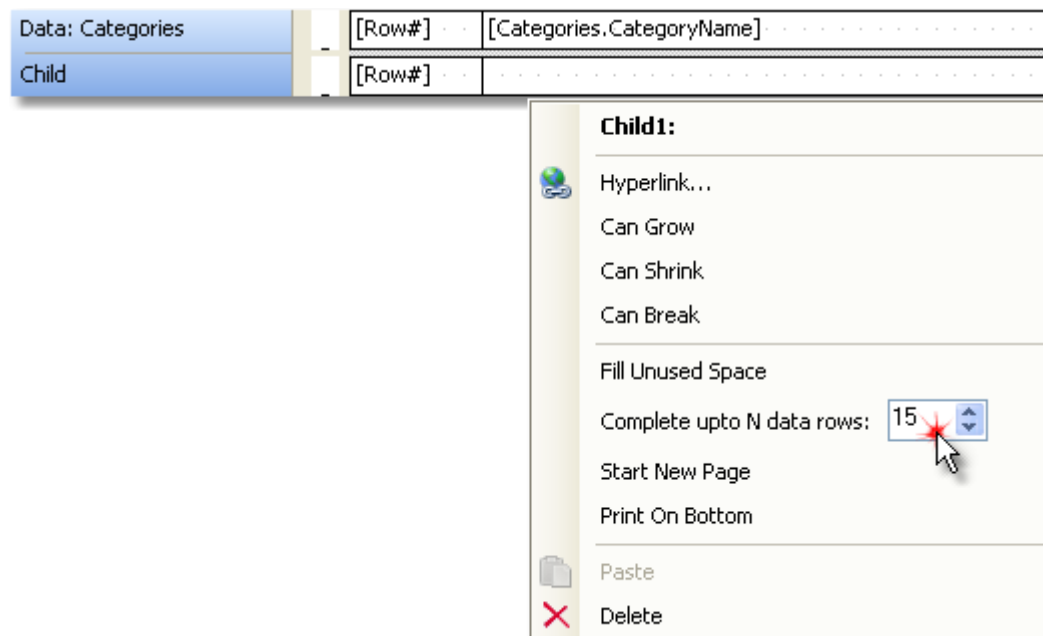


Para manter a banda e sua banda filha juntas, habilite a propriedade **KeepChild**.

### Imprimindo linhas de dados vazias

Muitas vezes, quando imprimimos em formulários pré impressos, uma certa quantidade de linhas de dados deve ser impressa. Se os dados atuais são menores do que a quantidade necessária de linhas, então as linhas vazias precisam ser impressas. Isso pode ser feito com a ajuda da banda Filha, anexando ela a banda Dados.

A banda Filha tem a propriedade **CompleteToNRows**. Se esta propriedade tiver o valor maior do que 0, a banda será utilizada para linhas de dados adicionais até a quantidade indicada. Por exemplo, nós precisamos imprimir 15 linhas, mas só há 8 linhas na fonte de dados. Neste caso, a banda Filha será impressa 7 vezes.



O relatório preparado ficará parecido com isso:

|    |                |
|----|----------------|
| 1  | Beverages      |
| 2  | Condiments     |
| 3  | Confections    |
| 4  | Dairy Products |
| 5  | Grains/Cereals |
| 6  | Meat/Poultry   |
| 7  | Produce        |
| 8  | Seafood        |
| 9  |                |
| 10 |                |
| 11 |                |
| 12 |                |
| 13 |                |
| 14 |                |
| 15 |                |

Se a fonte de dados tem mais linhas que o indicado na propriedade **CompleteToNRows**, então uma linha vazia não será impressa.

Outra maneira de imprimir uma linha vazia é preencher o espaço livre de uma página. Neste caso, a banda Filha é anexada às bandas do tipo **Rodapé de Dados** ou **Rodapé de Grupo** e preenche o espaço livre da página. A banda rodapé será impressa na parte inferior da página.

Para imprimir uma linha de dados vazia desta maneira, anexe a banda Filha na banda rodapé e habilite a sua propriedade **FillUnusedSpace**. Você verá que a banda filha é agora mostrada acima da banda a qual ela está anexada. Na figura abaixo, a banda Filha está anexada à banda **Sumário do Relatório**:

|                  |                           |
|------------------|---------------------------|
| Data: Categories | [Categories.CategoryName] |
| Child            |                           |
| Report Summary   | report summary            |
|                  |                           |

Quando nós geramos o relatório, iremos ver o seguinte:

|                |
|----------------|
| Beverages      |
| Condiments     |
| Confections    |
| Dairy Products |
| Grains/Cereals |
| Meat/Poultry   |
| Produce        |
| Seafood        |
|                |
|                |
|                |
|                |
|                |
|                |
| report summary |

### Hierarquia de Impressão

A banda Dados permite imprimir uma lista hierárquica. Para isso, uma banda e uma fonte de dados são usadas. A hierarquia deve ser definida na fonte de dados com a ajuda de duas colunas de dados:

1. Coluna chave. Esta é o identificador da linha de dados;
2. Coluna que contém a chave do pai do item.

Para imprimir tal fonte de uma forma hierárquica, você precisa definir as seguintes propriedades da banda Dados. Isso pode ser feito na janela **Propriedades**:

|                  |      |
|------------------|------|
| <b>Hierarchy</b> |      |
| IdColumn         |      |
| Indent           | 1 cm |
| ParentIdColumn   |      |

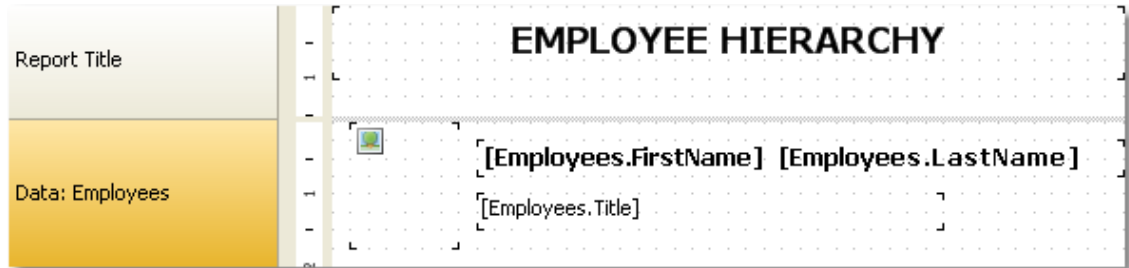
- Indicar a coluna chave na propriedade **IdColumn**;
- Indicar a coluna que contém o valor pai, na propriedade **ParentIdColumn**;
- Indicar a indentação hierárquica na propriedade **Indent**.

Vamos ver um exemplo de como imprimir uma hierarquia de empregados da tabela demonstrativa **Employees**. A tabela tem duas colunas que precisamos:

- Coluna EmployeeID é chave e contém o ID do empregado;



- Coluna ReportsTo contém o ID do empregado pai.



Define as propriedades da banda Dados, a qual é responsável pela hierarquia, do seguinte modo:

| Hierarchy      |                      |
|----------------|----------------------|
| IdColumn       | Employees.EmployeeID |
| Indent         | 1,5 cm               |
| ParentIdColumn | Employees.ReportsTo  |

Quando geramos um relatório, nós vamos ver o seguinte:

## EMPLOYEE HIERARCHY



**Andrew Fuller**

Vice President, Sales



**Nancy Davolio**

Sales Representative



**Janet Leverling**

Sales Representative



**Margaret Peacock**

Sales Representative



**Steven Buchanan**

Sales Manager



**Michael Suyama**

Sales Representative



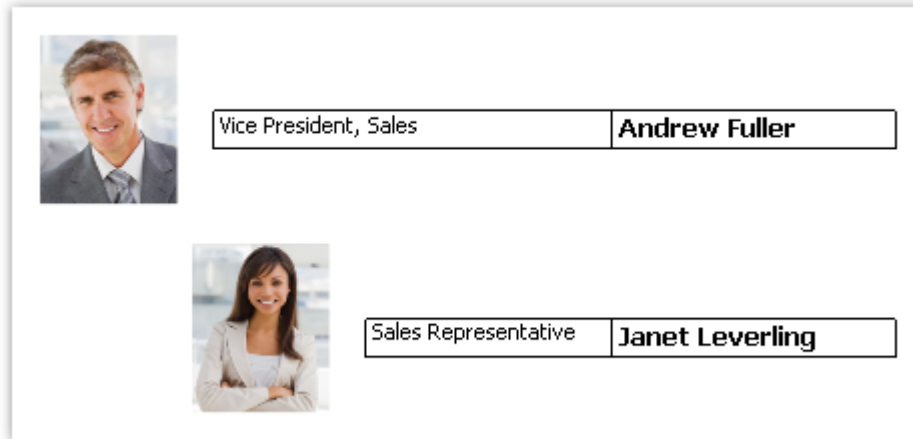
**Robert King**

Sales Representative

Quando imprimimos hierarquicamente, o **CIGAM Report** desloca a banda para a direita (pelo valor indicado na propriedade **Indent**), e também diminui a largura da banda no mesmo valor. Isso permite que você use a propriedade **Anchor** dos objetos da banda. Aqui estão possíveis valores para esta propriedade, que pode ser usados neste caso:

- **Left, Top (by default)** – o objeto é movido com a banda;
- **Right, Top** – o objeto permanece na sua posição original;
- **Left, Right, Top** – o lado direito do objeto fica na sua posição original, o lado esquerdo é movido com a banda.

Isso permite que você consiga alguns efeitos úteis:



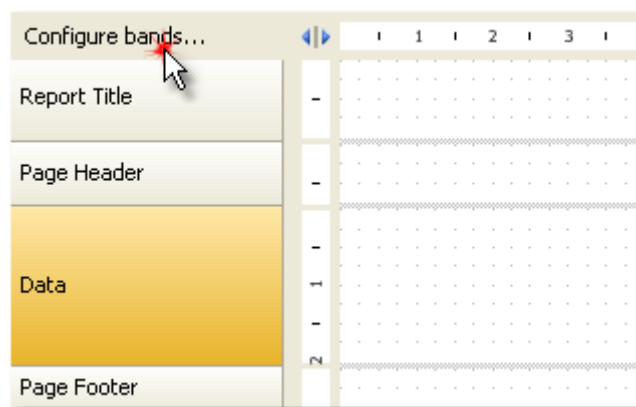
Neste exemplo, o objeto da figura tem a propriedade **Anchor** definida com **Left, Top**; o objeto com o título do cargo está fixado à **Left, Right, Top**; o objeto com o nome está fixado à **Right, Top**.

## Relatório Mestre-Detalhe

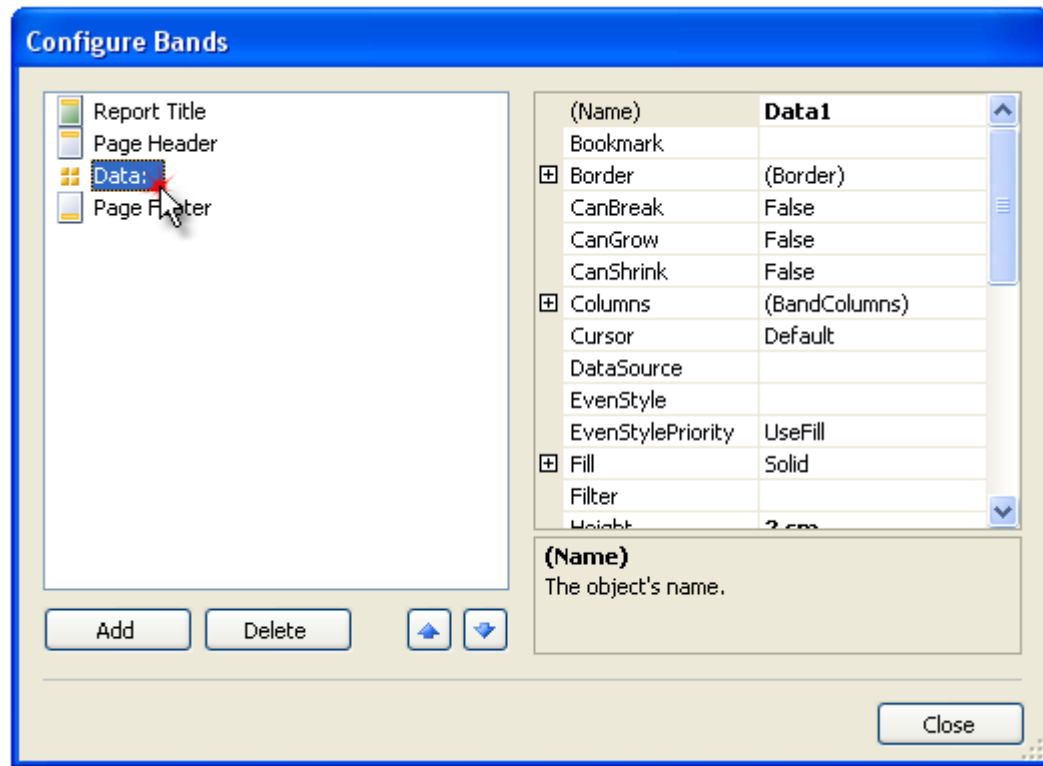
Usando duas bandas **Dados**, é fácil criar um relatório do tipo **Mestre-Detalhe**. Neste relatório, duas fontes de dados, entre as quais há uma relação, são usadas. Uma linha da fonte mestre pode corresponder com várias linhas da fonte de detalhes. Mais detalhes sobre relações pode ser encontrados no capítulo **Dados**.

É necessário colocar a banda no relatório de uma maneira que a banda-mestre contenha a banda de detalhes dentro dela. Isso pode ser feito na janela **Configurar Bandas**, a qual pode ser encontrada no menu **Relatório > Configurar Bandas....**

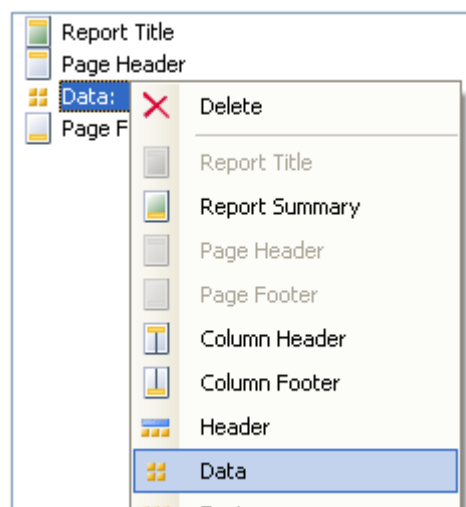
Vamos ver a criação de um relatório **Mestre-Detalhe** do zero. Para isso, nós vamos abrir o editor de relatório e criar um novo relatório vazio. Ele já possui uma banda **Dados**:



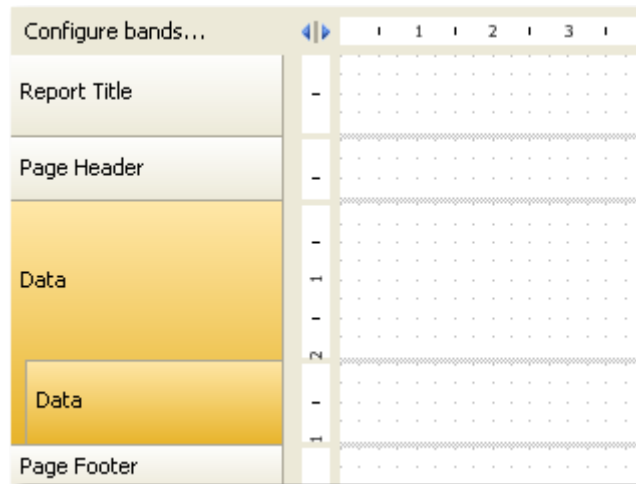
Para adicionar um detalhe a banda de dados, abra a janela **Configurar Bandas**. Isso pode ser feito pressionando o botão **Configurar Bandas...**, mostrado na figura, ou escolhendo o item de menu **Relatório > Configurar Bandas...**. Na janela de configurações, a estrutura da banda é mostrada:



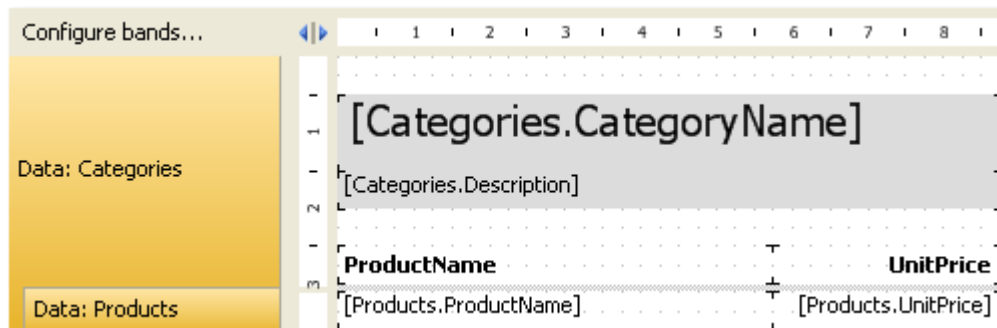
Selecione a banda Dados, como mostrado na figura, e clique com o botão direito do mouse para mostrar o menu de contexto (ou pressione o botão **Adicionar** na parte inferior da janela). Na janela que abrirá, selecione banda Dados:



Depois disso, uma **banda Dados** aninhada é adicionada à banda selecionada. Feche a janela pressionando o botão **Fechar**. Você verá que o *template* do relatório mudará da seguinte forma:



Bandas de dados aninhadas são claramente vistas em estruturas de bandas na parte esquerda da janela. Depois disso, você precisa conectar a banda à fonte de dados correspondente e colocar as colunas de dados nas bandas. Nós vamos utilizar duas fontes de dados –**Categorias** e **Produtos** – da base de dados demo que vem com o **CIGAM Report**:



Se nós gerarmos o relatório, vamos ver o seguinte:

## Beverages

Soft drinks, coffees, teas, beers, and ales

| ProductName               | UnitPrice |
|---------------------------|-----------|
| Chai                      | 18,00p.   |
| Chang                     | 19,00p.   |
| Guaraná Fantástica        | 4,50p.    |
| Sasquatch Ale             | 14,00p.   |
| Steeleye Stout            | 18,00p.   |
| Côte de Blaye             | 263,50p.  |
| Chartreuse verte          | 18,00p.   |
| Iphoh Coffee              | 46,00p.   |
| Laughing Lumberjack Lager | 14,00p.   |
| Outback Lager             | 15,00p.   |
| Rhönbräu Klosterbier      | 7,75p.    |
| Lakkalikööri              | 18,00p.   |

## Condiments

Sweet and savory sauces, relishes, spreads, and seasonings

| ProductName                  | UnitPrice |
|------------------------------|-----------|
| Aniseed Syrup                | 10,00p.   |
| Chef Anton's Cajun Seasoning | 22,00p.   |
| Chef Anton's Gumbo Mix       | 21,35p.   |

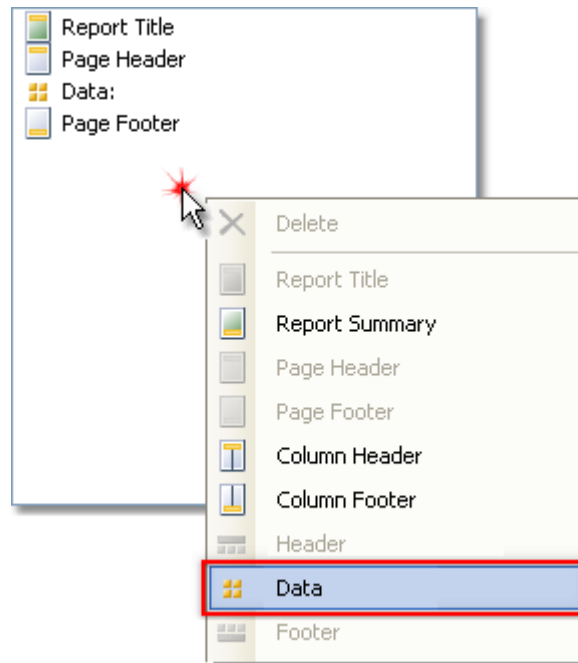
Deste modo, você pode criar um relatório do tipo **Mestre-Detalhe** com ilimitados dados aninhados, por exemplo, **Mestre-Detalhe-Subdetalhe**. Outro método, que é usado para a criação de um relatório do tipo **Mestre-Detalhe**, conectado com o uso de relatórios aninhados. Relatórios aninhados serão tratados na seção **Subrelatórios**.

## Relatório Mestre-Mestre

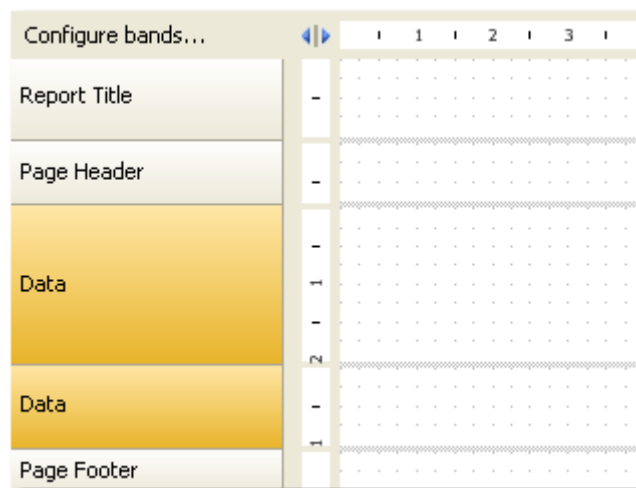
Na página do relatório, você pode imprimir várias listas simples. Isso pode ser feito colocando na página uma ou várias bandas **Dados**. Contrariamente ao relatório **Mestre-Detalhe**, onde as bandas estão aninhadas uma dentro da outra e imprimem dado de fontes relacionadas, em um relatório deste tipo ambas as bandas e fontes de dados não dependem uma da outra.

Nós vamos mostrar com um exemplo, como criar um relatório que imprime duas listas em uma página – tabela **Categorias** e tabela **Clientes**. Nós vamos criar um novo relatório e adicionar nele as

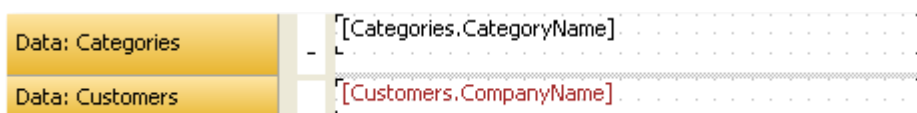
fontes de dados necessárias. Para adicionar uma segunda banda Dados, abra a janela **Configurar Bandas**.



Clique com o botão direito em um lugar vazio da lista, como mostrado na figura, e selecione a banda Dados no menu de contexto. Isso cria uma nova e independente banda Dados. O *template* do relatório vai ficar assim:



Agora, nós vamos conectar a banda à fonte de dados e colocar várias colunas de dados nela:

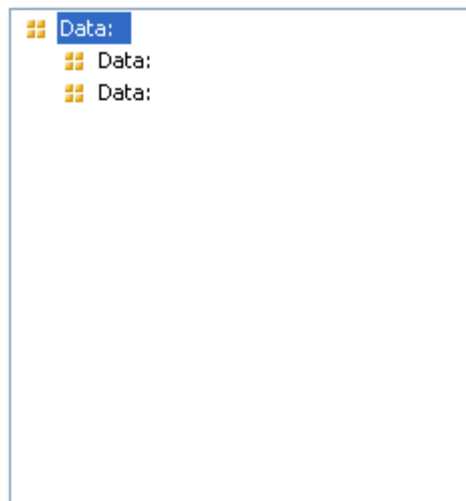


Se nós gerarmos o relatório, vamos ver o seguinte:



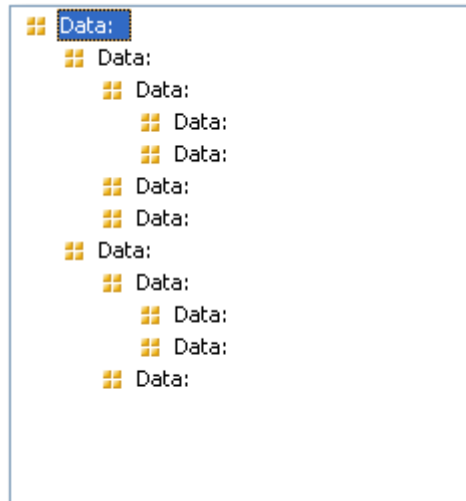
## Relatório Mestre-Detalhe-Detalhe

Uma banda Dados pode conter uma ou várias bandas **Dados** aninhadas. Isso permite construir um relatório do tipo **Mestre-Detalhe-Detalhe**. Para fazer isso, abra a janela **Configurar Bandas**, clique com o botão direito na banda-mestre **Dados** e adicione o detalhe banda Dados a ela. Repita o processo para adicionar uma segunda banda de detalhes.





Deste modo, é possível adicionar um número ilimitado de bandas de detalhe na banda-mestre **Dados**. Um exemplo desta estrutura de relatório pode ser assim (isso é somente um exemplo; isso somente demonstra as habilidade do **CIGAM Report**);



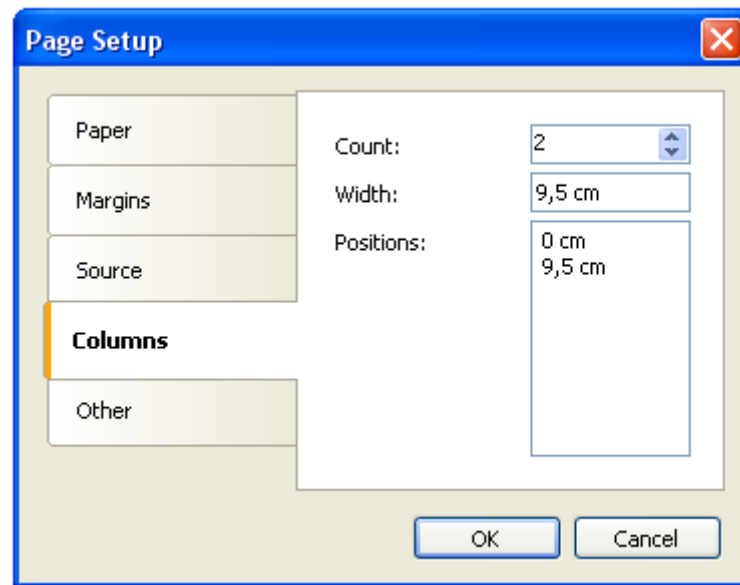
## Relatórios com Múltiplas Colunas

Um relatório comum imprime dados enquanto ele não chegar até o final da página. Depois disso, uma nova página é formada e a impressão continua nela. Um relatório com colunas imprime dados em várias colunas. Quando o fim da página é alcançado, a impressão continua em uma nova coluna na mesma página. Neste sentido, um relatório comum pode ser visto como um relatório com uma coluna.

No **CIGAM Report** há dois métodos de imprimir colunas.

## Colunas das páginas

O primeiro método é baseado em definir o número de colunas de uma página de relatório. Isso é feito na janela **Configuração de Página** na guia **Colunas**:



Como visto, você pode definir os seguintes parâmetros da coluna:

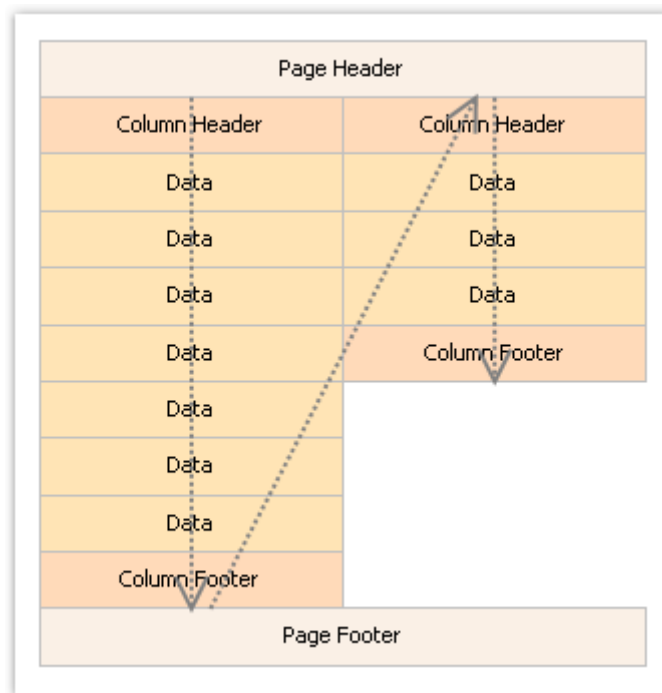
- Quantidade de colunas;
- Largura da coluna;
- A posição de cada coluna.

Para transformar um relatório comum em um relatório com colunas, você precisa definir somente o número de colunas na página. O resto dos parâmetros o **CIGAM Report** vai calcular por sua conta. Quando você habilita colunas, o modo das bandas no editor muda:



A área mostrada em cinza nunca deve ser usada para colocar objetos nela. Ela deve ser usada para imprimir objetos da coluna próximos.

Para trabalhar com colunas, as bandas **Cabeçalho de Coluna** e **Rodapé de Coluna** são usadas. Como visto nos nomes delas, elas imprimem no topo e na parte inferior de cada coluna respectivamente. A figura abaixo demonstra a ordem de impressão da banda em um relatório com colunas:



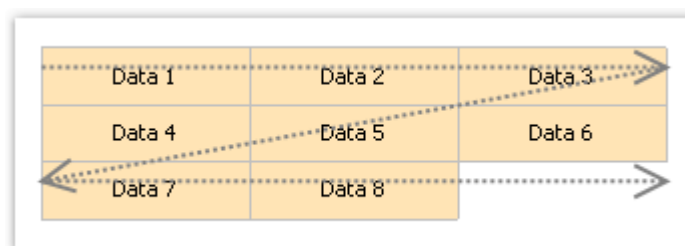
### Colunas da banda de dados

Outro método de imprimir um relatório com múltiplas colunas é baseado em usar colunas da banda Dados. O resto das bandas continua sendo impresso em uma coluna.

Os parâmetros da coluna podem ser configurados usando a propriedade **Colunas**, a qual pode ser mudada na janela Propriedades. Você pode definir os seguintes parâmetros:

- Número de colunas;
- Largura da coluna;
- Modo de impressão da coluna. Você pode escolher qualquer um dos dois modos – **AcrossThenDown** e **DownThenAcross**;
- O número mínimo de linhas em uma coluna, se o modo escolhido for **DownThenAcross**.

A coluna da banda pode ser impressa em qualquer um dos dois modos. No modo **AcrossThenDown** (o padrão), as colunas são impressas na seguinte maneira:






## Relatórios do tipo “livreto”

Quando imprimimos um relatório em forma de “livreto”, você provavelmente vai deparar-se com as seguintes exigências:

- Separar as páginas do relatório – capa, conteúdo da tabela, conteúdo do relatório, contracapa;
- Diferentes margens de página para páginas pares e ímpares;
- Diferentes cabeçalhos e rodapés em páginas pares e ímpares.

## Adicionando uma página em um relatório

Você pode adicionar qualquer número de páginas no *template* do relatório. Em cada página você pode por um relatório separado. Para adicionar uma nova página, clique no botão  na barra de ferramentas. A página pode também ser adicionada, pressionando o botão **Adicionar Novo...** e selecionando o item **Página de Relatório** na janela.

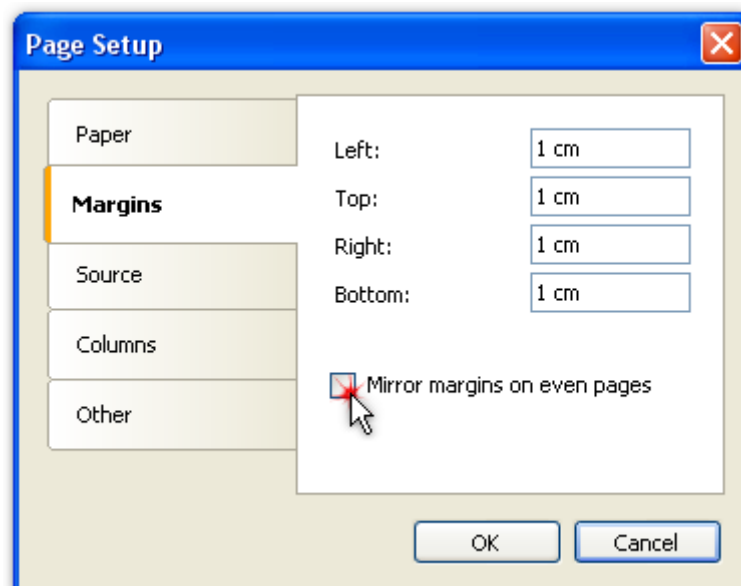
---

Para criar a seção **Sumário**, você pode usar a técnica descrita no capítulo **Relatórios Interativos**.

---

## Configurações da página

Na janela **Configuração de Página**, você pode indicar que o **CIGAM Report** deve espelhar margem esquerda e a margem direita para páginas pares:



Se você precisa começar a página em um número ímpar, defina a propriedade **StartOnOddPage** para verdadeiro. Quando necessário, o **CIGAM Report** imprime páginas em branco antes de começar a imprimir a página indicada.

## Impressão em páginas ímpares/pares

Todos os objetos do relatório tem a propriedade **PrintOn**. Usando esta propriedade, você pode imprimir diferentes objetos em páginas ímpares e pares.

*Esta propriedade pode ser definida na janela Propriedades.*

Esta propriedade determina em quais páginas o objeto pode ser impresso. Esta propriedade pode ter um dos seguintes valores ou uma combinação deles:

- FirstPage;
- LastPage;
- OddPages;
- EvenPages;
- RepeatedBand. Este valor refere-se a uma banda com a propriedade **RepeatOnEveryPage** definida como **true**.

Por padrão, o valor desta propriedade é igual à FirstPage, LastPage, OddPages, EvenPages, RepeatedBand. Isso significa que o objeto será impresso em todas as páginas do relatório. Nós vamos mostrar diversos exemplos típicos do uso desta propriedade:

| Valor da Propriedade                         | Onde o objeto será impresso   |
|--|---|
| FirstPage                                    | Somente na primeira página  |
| LastPage, OddPages, EvenPages, RepeatedBand  | Em todas as páginas, exceto a primeira.   |
| FirstPage, OddPages, EvenPages, RepeatedBand | Em todas as páginas, exceto as últimas.   |
| RepeatedBand                                 | Somente nas bandas com a propriedade <b>RepeatOnEveryPage</b> definida como verdadeira. |
| FirstPage, LastPage, OddPages, RepeatedBand  | Somente nas páginas ímpares.  |
| FirstPage, LastPage, EvenPages, RepeatedBand | Somente nas páginas pares.  |

Por exemplo, imprimir um texto diferente em páginas ímpares e pares, colocar dois objetos **Texto** na banda e definir eles nas seguintes maneiras:

- O primeiro objeto será impresso em páginas ímpares. Defina a propriedade **PrintOn** para **FirstPage, LastPage, OddPages, RepeatedBand()**
- O segundo objeto será impresso em páginas pares. Defina a propriedade **PrintOn** para **FirstPage, LastPage, EvenPages, RepeatedBand()**



Estes objetos nunca serão impressos ao mesmo tempo. Você pode colocá-los um no topo do outro.

Todas as bandas têm a mesma propriedade. Para imprimir bandas diferentes em páginas pares e ímpares, use a banda Filha. Você pode anexar isso em qualquer banda; isto pode ser feito na janela **Configurar Bandas**. Configure a banda principal e sua filha do seguinte modo:

- A banda principal será impressa em páginas ímpares. Defina a propriedade **PrintOn** para **FirstPage, LastPage, OddPages, RepeatedBand.** ( )
- A banda filha será impressa em página pares. Defina a propriedade **PrintOn** para **FirstPage, LastPage, EvenPages, RepeatedBand.**( )

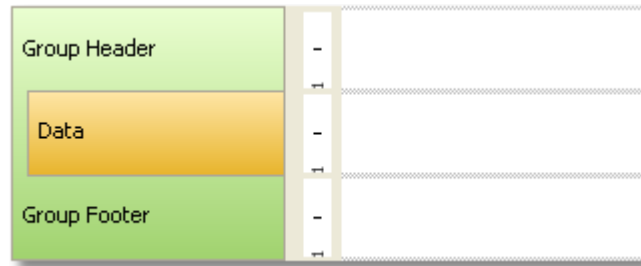
As bandas podem ter diferentes alturas, aparências e conteúdos. Veja o exemplo seguinte, o qual imprime diferentes cabeçalhos de páginas:

|             |   |  |         |
|-------------|---|--|---------|
| Page Header | - | PrintOn = FirstPage, LastPage, OddPages  | [PageN] |
| Child       | - | PrintOn = FirstPage, LastPage, EvenPages | [PageN] |

## Grupos e totais

Anteriormente vimos o relatório do tipo **Mestre-Detalhe**, o qual imprime dados de duas fontes relacionadas. **CIGAM Report** permite criar relatórios que (...), mas usam uma fonte de dados. Para isso, grupos são utilizados.

Um grupo é um conjunto de três bandas: **Cabeçalho de Grupo**, **Data** e **Rodapé de Grupo**. No editor:



---

Um grupo sempre contém um cabeçalho e dados. O rodapé do grupo é opcional, você pode deletá-lo.

---

Para usar um grupo, você deve definir a condição do grupo para cabeçalho do grupo, e conectar as fontes de dados à banda Dados. A condição pode ser qualquer expressão, mas como regra, isto é uma das colunas da fonte de dados. A impressão do grupo é feita do seguinte modo:

1. Cabeçalho do grupo é impresso
2. Linha de dados é impressa
3. Verifica se a condição de agrupamento mudou
4. Se a condição não mudou, a próxima linha de dados é impressa (p.2)
5. Se a condição mudou, o rodapé do grupo é impresso, e um novo grupo começa a ser impresso (p.1)

Supondo que temos uma tabela **Produtos** com os seguintes dados:



| CategoryName | ProductName                      |
|--------------|----------------------------------|
| Beverages    | Côte de Blaye                    |
| Beverages    | Chartreuse verte                 |
| Beverages    | Steeleye Stout                   |
| Beverages    | Guaraná Fantástica               |
| Beverages    | Sasquatch Ale                    |
| Beverages    | Rhönbräu Klosterbier             |
| Beverages    | Lakkalikööri                     |
| Beverages    | Outback Lager                    |
| Beverages    | Ipoh Coffee                      |
| Beverages    | Laughing Lumberjack Lager        |
| Beverages    | Chang                            |
| Beverages    | Chai                             |
| Condiments   | Original Frankfurter grüne Soße  |
| Condiments   | Sirop d'érable                   |
| Condiments   | Chef Anton's Gumbo Mix           |
| Condiments   | Northwoods Cranberry Sauce       |
| Condiments   | Grandma's Boysenberry Spread     |
| Condiments   | Chef Anton's Cajun Seasoning     |
| Condiments   | Aniseed Syrup                    |
| Condiments   | Louisiana Hot Spiced Okra        |
| Condiments   | Veggie-spread                    |
| Condiments   | Louisiana Fiery Hot Pepper Sauce |
| Condiments   | Gula Malacca                     |
| Condiments   | Genen Shouyu                     |

Dados podem ser agrupados na coluna **NomeCategoria**. Esta coluna será impressa no cabeçalho do grupo. Os dados são apresentados pelo campo **NomeProduto**. O relatório será como segue:

|                |   |                                      |
|----------------|---|--------------------------------------|
| Group Header:  | - | [[Products.Categories.CategoryName]] |
| Data: Products | - | [Products.ProductName]               |
| Group Footer   | - |                                      |

Se nós gerarmos o relatório, será mostrado o seguinte:

#### Beverages

|                           |
|---------------------------|
| Côte de Blaye             |
| Chartreuse verte          |
| Steeleye Stout            |
| Guaraná Fantástica        |
| Sasquatch Ale             |
| Rhönbräu Klosterbier      |
| Lakkalikööri              |
| Outback Lager             |
| Ipoh Coffee               |
| Laughing Lumberjack Lager |
| Chang                     |
| Chai                      |

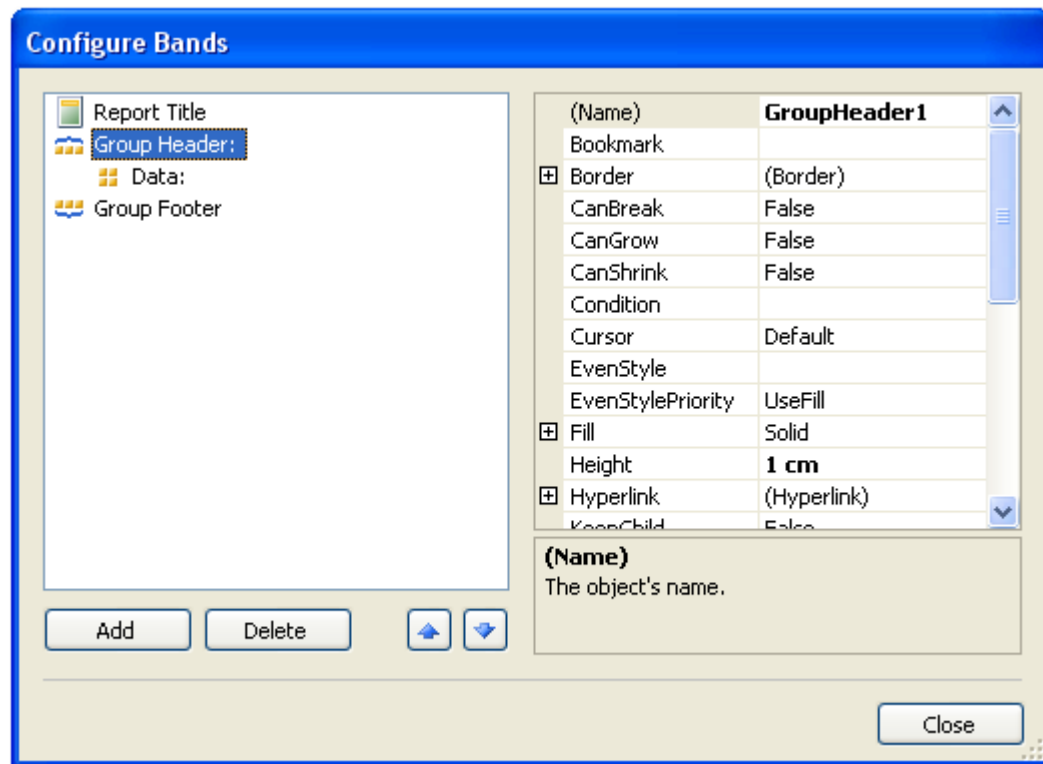
#### Condiments

|                                  |
|----------------------------------|
| Original Frankfurter grüne Soße  |
| Sirop d'érable                   |
| Chef Anton's Gumbo Mix           |
| Northwoods Cranberry Sauce       |
| Grandma's Boysenberry Spread     |
| Chef Anton's Cajun Seasoning     |
| Aniseed Syrup                    |
| Louisiana Hot Spiced Okra        |
| Vegie-spread                     |
| Louisiana Fiery Hot Pepper Sauce |
| Gula Malacca                     |
| Genen Shouyu                     |

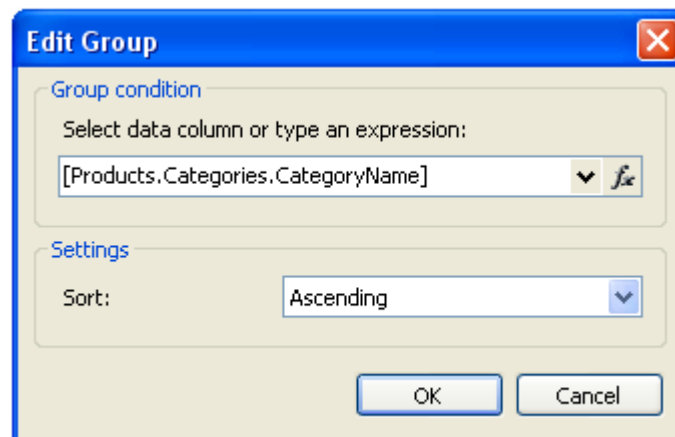
### Criando grupos

Há duas maneiras de adicionar um grupo em um relatório.

Primeiro método: você adiciona a banda **Cabeçalho de Grupo** na janela **Configurar Bandas**. Para fazer isso, pressione o botão **Adicionar** e selecione a banda **Cabeçalho de Grupo**. O **CIGAM Report** adiciona o grupo à banda Dados disponível ou cria todo o grupo, se a banda não está no relatório:

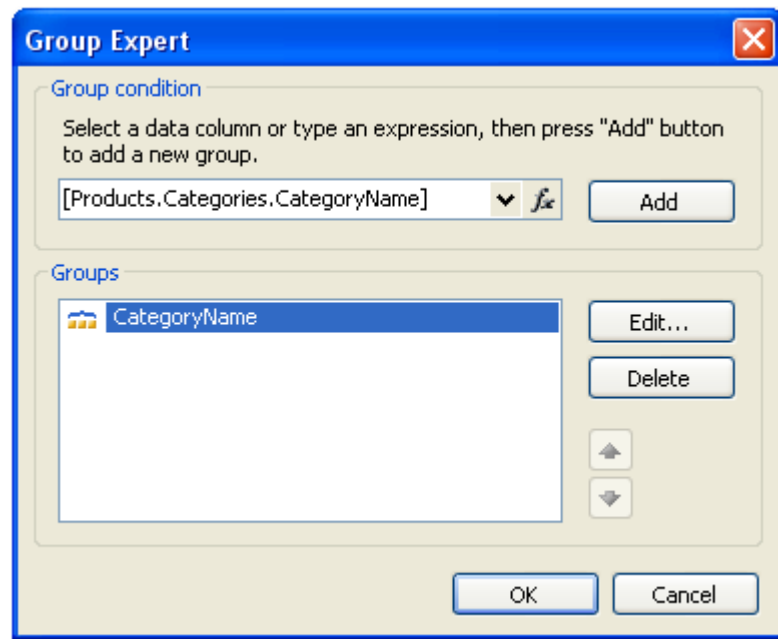


Para configurar um grupo, dê um duplo clique na banda **Cabeçalho de Grupo**. Você verá a janela de edição do cabeçalho do grupo:

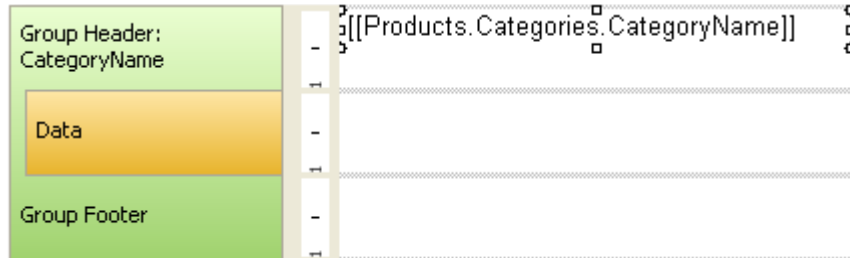


Você precisa definir a condição do grupo. Isto pode ser qualquer expressão ou coluna da fonte de dados. Também escolha a disposição. Por padrão, os dados são dispostos na ordem crescente.

Segundo método: você usa o **Assistente**, o qual pode ser aberto pelo menu **Relatório > Assistente de Grupos....** Para criar um grupo, coloque a condição do grupo e pressione o botão **Adicionar** :



O **Assistente** adicionará todos os elementos do grupo no relatório. Também cria o objeto Texto no cabeçalho do grupo, no qual a condição é impressa:



### Disposição dos dados

Para o funcionamento correto do grupo, é necessário obedecer à seguinte condição: a fonte de dados deve ser disposta na coluna que é usada na condição do grupo. Se esta condição não for obedecida, você verá vários grupos contendo 1-2 linhas de dados:

| Beverages      |
|----------------|
| Sasquatch Ale  |
| Steeleye Stout |

| Seafood     |
|-------------|
| Inlagd Sill |
| Gravad lax  |

| Beverages        |
|------------------|
| Côte de Blaye    |
| Chartreuse verte |

| Seafood                         |
|---------------------------------|
| Boston Crab Meat                |
| Jack's New England Clam Chowder |

Felizmente, há possibilidade de organizar a fonte de dados de duas maneiras.

- Você pode definir a organização dos dados no editor de grupo. A fonte de dados será automaticamente organizada pela condição do grupo;
- Você pode definir a organização (disposição) no editor da banda Dados.

Ambos os métodos são equivalentes, no entanto é mais fácil utilizar o primeiro método. Quando você cria grupos, você define os grupos e a disposição dos dados em uma janela.

Em certas situações, o primeiro método não deve ser usado. Com isso, nós vamos definir o agrupamento pela primeira letra do nome do produto. Neste caso, o produto será organizado somente pela primeira letra, o que não é aceitável. Você deve usar o segundo método e indicar a organização pelo nome inteiro do produto.

### Grupos aninhados

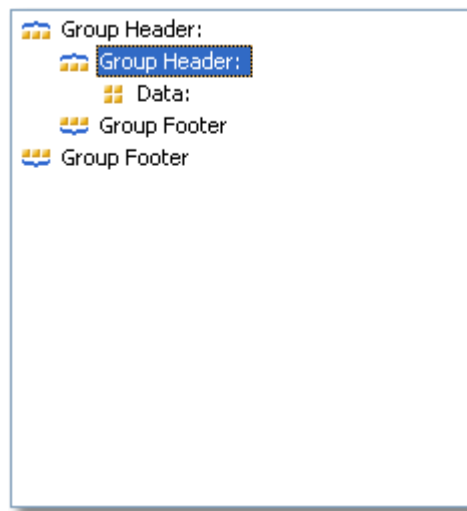
Um grupo aninhado tem várias bandas **Cabeçalho de Grupo**. A última banda contém a banda Dados:

|              |   |  |
|--------------|---|--|
| Group Header | - |  |
| Group Header | - |  |
| Data         | - |  |
| Group Footer | - |  |
| Group Footer | - |  |

Cada cabeçalho do grupo tem sua própria condição de grupo.

Criar um grupo aninhado pode ser feito do mesmo modo que um grupo comum. No primeiro caso, você cria um grupo simples e adiciona um grupo aninhado na janela **Configurar Bandas**.

Para isso, selecione a banda **Cabeçalho de Grupo**, pressione o botão **Adicionar** e adicione outra banda **Cabeçalho de Grupo**:



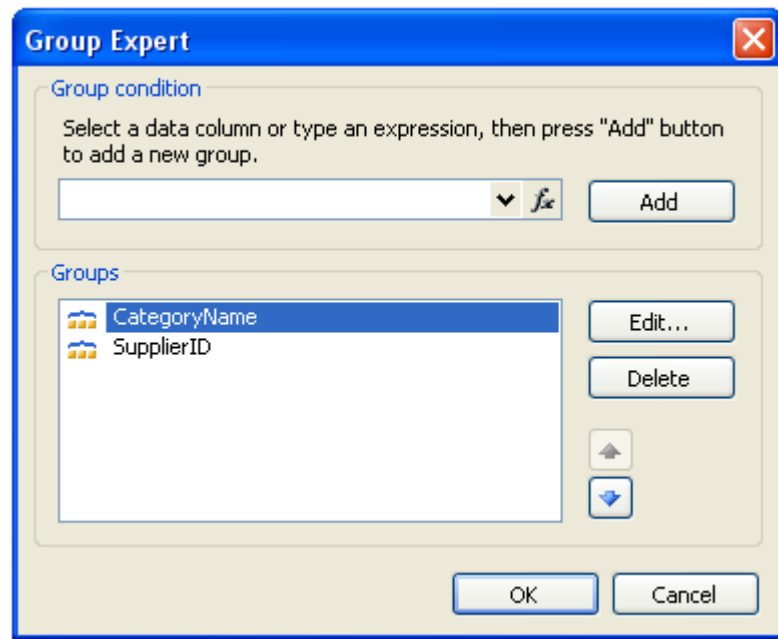
Depois disso, abra o editor do grupo adicionado e defina a condição do grupo.

No segundo caso, você usa o **Assistente de Grupo**. Defina a condição de grupo e clique no botão **Adicionar**. O programa irá adicionar o novo grupo ao existente.

A impressão do grupo aninhado não muda muito em relação à impressão de um grupo comum. Quando imprime dados, o **CIGAM Report** checará todas as condições de grupos de todos os grupos. Se a condição muda, o grupo correspondente termina e um novo grupo começa a imprimir.

### Gerenciando grupos

Para gerenciar grupos, o **group expert** pode ser usado. Ele pode ser aberto através do menu **Relatório > Assistente de Grupos...**:



Com a ajuda do **Assistente** você pode tanto adicionar quanto excluir um grupo, e mudar a ordem de agrupamento também. Para mudar a ordem de agrupamento, os botões (cima) e (baixo) são usados. Com a ajuda do botão **Editar...**, você pode mudar a condição do grupo do grupo escolhido..

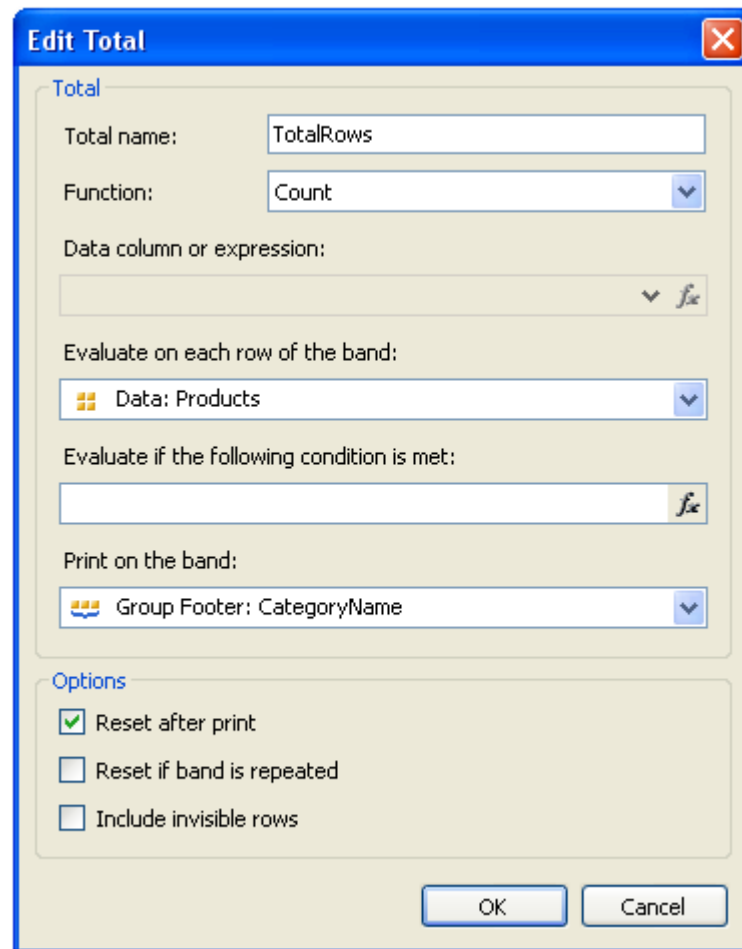
## Imprimindo valores totais

Agrupamento é freqüentemente usado para imprimir alguns valores totais em cada grupo. Por exemplo, isso pode ser o número de linhas no grupo ou a soma de uma das colunas de dados. Para imprimir tais valores, o total é usado. O uso dos totais é descrito no capítulo **Dados**.

Para imprimir o total do valor no grupo, você precisa fazer o seguinte:

- Crie um total, selecionando o item **Ações > Novo total...** na janela **Dados**;
- Escolha a banda de dados do grupo no *combobox* **Calcular em cada linha da banda**;
- Escolha o rodapé do grupo no *combobox* **Imprimir na banda**.
- Coloque o objeto Texto, que imprima um valor total, no rodapé do grupo.

Por exemplo, para imprimir o número de linhas em cada grupo, configure o total como segue:



Para mostrar o valor do total, arraste-o para o rodapé do grupo:

|                |   |                                    |
|----------------|---|------------------------------------|
| Group Header:  | - | [Products.Categories.CategoryName] |
| CategoryName   | - |                                    |
| Data: Products | - | [Products.ProductName]             |
|                | - |                                    |
| Group Footer   | - | [TotalRows]                        |
|                | - |                                    |

O relatório preparado ficará parecido com o seguinte:



| Meat/Poultry            |
|-------------------------|
| Alice Mutton            |
| Perth Pasties           |
| Thüringer Rostbratwurst |
| Pâté chinois            |
| Tourtière               |
| Mishi Kobe Niku         |

6

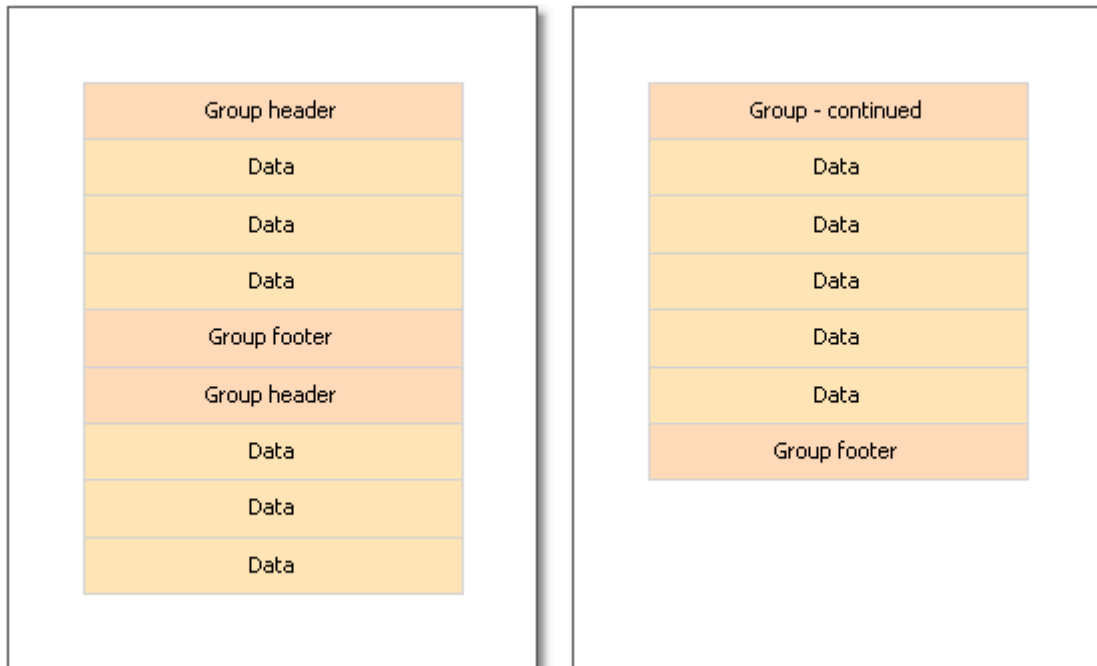
| Produce                         |
|---------------------------------|
| Rössle Sauerkraut               |
| Uncle Bob's Organic Dried Pears |
| Manjimup Dried Apples           |
| Longlife Tofu                   |
| Tofu                            |

5

### Repetindo o cabeçalho e o rodapé

O cabeçalho e o rodapé do grupo tem a propriedade **RepeatOnEveryPage**. Isto pode ser útil se o grupo não cabe em uma página do relatório feito. Usando esta propriedade, você pode imprimir o cabeçalho/rodapé do grupo em cada página, onde o grupo é impresso. Quando imprimimos tal cabeçalho/rodapé, o **CIGAM Report** define sua **flag Repeated**. Isto pode ser usado para imprimir diferentes objetos em um cabeçalho de um grupo comum e em seguida, por exemplo, imprimir o texto **continue...** em uma nova página. Para isso, use a propriedade **PrintOn** do objeto Texto (veja mais detalhes na seção **Relatório do tipo "livreto"**).

Para imprimir diferentes textos, coloque dois objetos no cabeçalho do grupo, um em cima do outro:



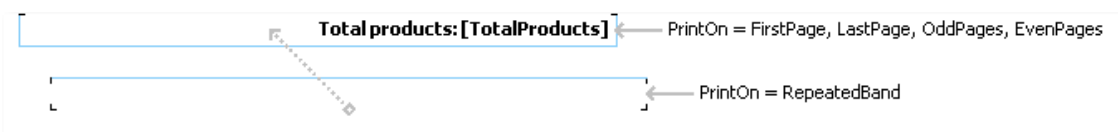
O rodapé do grupo pode também ser repetido em cada página:

| R                        |       |
|--------------------------|-------|
| Raclette Courdavault     | 55,00 |
| Ravioli Angelo           | 19,50 |
| Rhönbräu Klosterbier     | 7,75  |
| Röd Kaviar               | 15,00 |
| Røgede sild              | 9,50  |
| <b>Total products: 6</b> |       |

| R                        |       |
|--------------------------|-------|
| Rössle Sauerkraut        | 45,60 |
| <b>Total products: 6</b> |       |

Neste relatório, o rodapé do grupo tem dois objetos, colocados um acima do outro:





### Propriedades do grupo

A banda **Cabeçalho de Grupo** tem algumas propriedades úteis.

A propriedade **StartNewPage** permite formar uma nova página antes de imprimir o grupo. Como resultado, cada grupo será colocado em uma nova página.

---

A nova página não será adicionada antes do primeiro grupo. Isso é feito para evitar uma primeira página vazia.

---

A propriedade **ResetPageNumber** permite resetar o número das páginas quando o grupo é impresso. Normalmente isto é utilizado em conjunto com a propriedade **StartNewPage**. Como resultado, se ambas as propriedades estiverem habilitadas, cada grupo será impresso em uma nova página, e terá sua própria numeração de página.

## Subrelatórios

Às vezes em certo local do relatório principal, dados a mais precisam ser mostrados. Isto pode ser um relatório separado com uma estrutura muito complicada. Você pode tentar resolver esta tarefa usando a rica coleção de bandas do **CIGAM Report**. No entanto, em certos casos, é preferível utilizar o objeto **Subreport**.

O objeto **Subreport** é um objeto comum de relatório, o qual pode ser colocado em uma das bandas. Quando você faz isso, o **CIGAM Report** adiciona uma página extra no relatório e conecta com ela com o subrelatório. Nesta página é possível criar um relatório extra, contendo qualquer estrutura.

Quando imprimimos um relatório, no qual exista um objeto **Subreport**, o seguinte será feito:

1. O relatório principal será impresso, enquanto o objeto **Subreport** não for alcançado.
2. As bandas de subrelatórios serão impressas.
3. A impressão do relatório principal continuará.

---

Desde que o subrelatório é formado na folha do relatório principal, ele não pode contém as seguintes bandas: **Cabeçalho/Rodapé de Relatório, Cabeçalho/Rodapé de Página, Cabeçalho/Rodapé de Coluna, Deslocamento**.

---

## Modos de impressão

O subrelatório pode ser impresso de dois modos.

No primeiro método de impressão, bandas e objetos do subrelatório são impressos na página do relatório principal. Objetos que podem ser colocados nas bandas e tem limites:

- O objeto **Subreport** deve estar localizado parte mais inferior da banda;
- Nunca coloque objetos abaixo do objeto **Subreport**. Quando o relatório estiver em funcionamento, tais objetos serão sobrepostos com os objetos do subrelatório.



Para colocar outros objetos abaixo do subrelatório, use a banda Filha. Coloque os objetos no seguinte modo:



O segundo método de impressão difere naqueles objetos de subrelatórios impressos na banda, os quais contém o objeto **Subreport**. Você pode habilitar este modo no menu de contexto do objeto **Subreport**. Para fazer isso, selecione o item **Imprimir no pai**. Este modo não põe um limite na



colocação de objetos. Fora à isso, neste modo, a banda pai pode tanto aumentar como diminuir dependendo de quantos dados foram impressos no subrelatório.

O único problema com o segundo modo é que deve haver um monte de dados no subrelatório. Quando imprimi-lo, a banda pai terá uma grande altura. Para imprimir tal banda corretamente, é necessário quebrar o seu conteúdo (propriedade **CanBreak**). O algoritmo que quebra não garante 100% de qualidade e em alguns casos ele pode causar o desordenamento dos objetos.

### Relatório lado-a-lado

Colocando dois objetos **Subreports** lado-a-lado na mesma banda, você pode imprimir listas de dados independentes. Quando imprimimos tal relatório ,o **CIGAM Report** age da seguinte maneira:

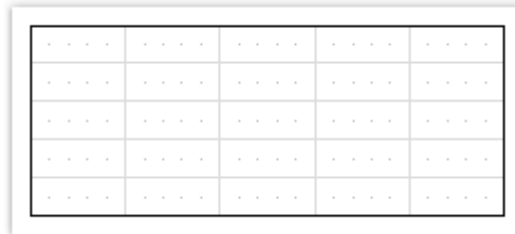
- Imprime o relatório principal até que o objeto **Subreport** seja alcançado;
- Imprime o primeiro subrelatório;
- Volta para a página onde a impressão do relatório começou, e imprime o próximo subrelatório;
- Depois de todos os subrelatórios terem sido impressos, o relatório principal continua sendo impresso de onde o subrelatório mais longo parou.

### Subrelatórios aninhados

Na página do subrelatório, você pode por outro objeto **Subreport**, então o subrelatório torna-se aninhado. O número de níveis de aninhamento não é restrito; no entanto não se preocupe com isso. Múltiplos aninhamentos são muito difíceis de entender. Se você tiver a possibilidade, use bandas de dados para imprimir dados aninhados. Uma banda Dados pode ter um ou várias bandas **Dados** aninhadas. Se você precisar imprimir um relatório do tipo **Mestre-Detalhe** ou **Mestre-Detalhe-Subdetalhe**, não há necessidade de usar o subrelatório.

## Relatórios do tipo Tabela

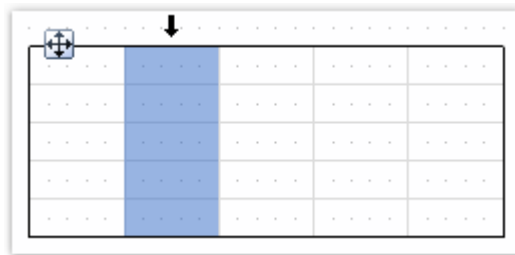
O objeto Tabela é feito de linhas, colunas e células. Ele é uma simplificação análoga à tabela do Microsoft Excel. Ela se parece com a seguinte figura:



### Configurando colunas

Você pode adicionar ou excluir colunas com a ajuda do menu de contexto. Para isso:

- Selecione a tabela ou qualquer um de seus elementos e coloque o cursor na coluna escolhida. A forma do cursor muda para uma pequena flecha preta.



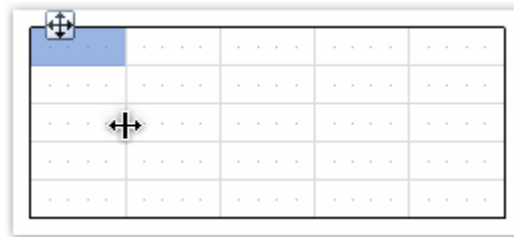
- Clique com o botão esquerdo, para selecionar a coluna;
- Clique com o botão direito para mostrar a menu de contexto da coluna;
- Se você precisar selecionar várias colunas adjacentes, clique com o botão direito e, sem soltá-lo, mova o mouse para direita ou para a esquerda, para selecionar colunas adjacentes.

O menu de contexto das colunas pode também ser aberto na janela **Árvore do Relatório**. Abra a janela, selecione a coluna necessária e clique com o botão direito do mouse.

### Gerenciando o tamanho da coluna

Você pode definir a largura da coluna usando um dos seguintes métodos:

- Selecione a tabela ou qualquer um de seus elementos e posicione o cursor na borda entre as duas colunas. A forma do cursor muda para um separador horizontal:



- Selecione a coluna e indique a largura necessária na propriedade **Width**. A propriedade é acessível na janela Propriedades.

Você pode também habilitar a propriedade **Autosize** da coluna. Quando gerarmos o relatório, a largura da coluna será calculada automaticamente. Para limitar a largura da coluna, você pode indicar as propriedades **MinWidth** e **MaxWidth**.

---

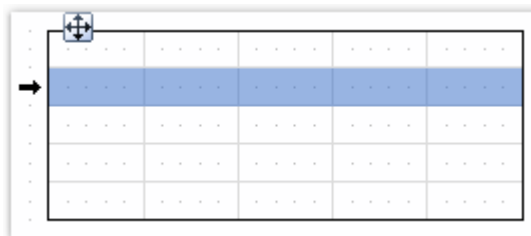
A largura de uma coluna nunca deve ser maior do que a largura da página.

---

### Configurando linhas

Linhas são configuradas da mesma maneira. Para selecionar uma linha, faça o seguinte:

- Selecione a tabela ou qualquer um de seus elementos e posicione o cursor para a esquerda da linha necessária. A forma do cursor mudará para uma pequena seta preta:



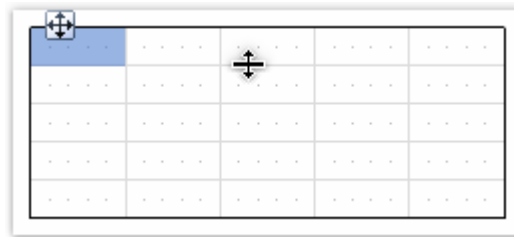
- Clique com o botão esquerdo do mouse para selecionar a linha;
- Clique com o botão direito do mouse para mostrar o menu de contexto da linha.

Se você precisar selecionar várias linhas adjacentes, então clique com o botão esquerdo do mouse e, sem soltá-lo, mova o mouse para direita ou para esquerda.

### Gerenciando o tamanho da linha

Você define a altura da linha usando um dos seguintes métodos:

- Selecione a tabela e qualquer um dos seus elementos e posicione o cursor na borda entre as duas linhas. A forma do cursor muda para um separador vertical:



Clique com o botão esquerdo e mova o mouse para mudar o tamanho da linha.

- Selecione a linha e indique a altura necessária na propriedade **Height**. Esta propriedade está disponível na janela Propriedades.

Você pode também habilitar a propriedade **AutoSize** da linha. Quando o relatório é gerado, a altura da linha será calculada automaticamente. Para limitar a altura da linha, você pode usar as propriedades altura mínima (**MinHeight**) e altura máxima(**MaxHeight**):

---

A altura da linha nunca deve ser maior que a altura da página.

---

### Configurando células

Células são objetos de texto. Na essência, as classes das células são herdadas do objeto Texto. Tudo que foi dito acima sobre o objeto Texto se aplica as células das tabelas também.

A edição do texto das células pode ser feita da mesma maneira que o objeto Texto. Além disso, você pode arrastar e soltar um elemento da janela Dados nas células.

A borda e o preenchimento da célula podem ser configurados com a ajuda da barra de tarefas **Filtro e Bordas**.

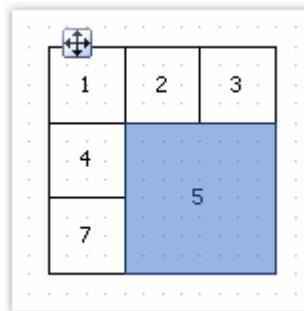
Para abrir o menu de contexto da célula, clique com o botão direito sobre a célula.

### Unindo e separando células

Você pode juntar duas células adjacentes da tabela. Como resultado, haverá uma grande célula. Para fazer isso:

- Selecione a primeira célula com a ajuda do mouse;
- Clique com o botão esquerdo, sem soltá-lo, e mova o mouse para selecionar um grupo de células;
- Na região selecionada, clique com o botão direito do mouse para mostrar o menu de contexto das células;
- No menu de contexto, escolha o item **Juntar células**.





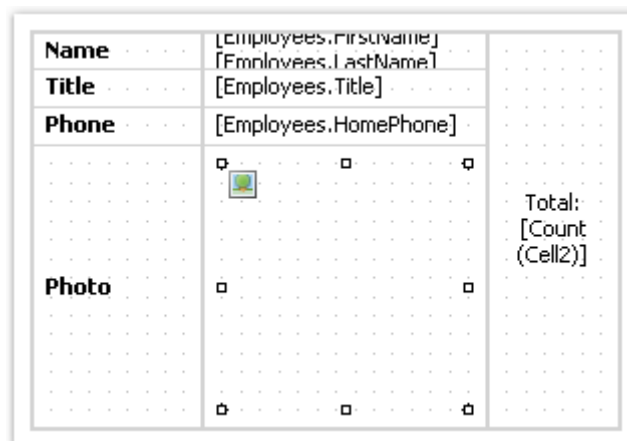
Para separar uma célula, abra o menu de contexto dela e escolha o item **Dividir células**.

### Inserindo objetos em células

Nas células, você pode inserir outros objetos, por exemplo, uma figura. Os seguintes objetos nunca poderão ser adicionados nas células:

- **Tabela;**
- **Matriz;**
- **Subrelatório.**

Para adicionar um objeto em uma célula, simplesmente arraste-o para dentro da célula. Você pode mover livremente um objeto entre células, e também levá-lo além do limite das tabelas.



Uma célula serve como um *container* para objetos serem colocados dentro. Isso significa que você pode usar as propriedades **Dock** e **Anchor** de um objeto dentro da célula. Isso permite mudar o tamanho de um objeto quando o tamanho da célula mudar.

### Imprimindo uma tabela

Uma tabela pode ser impressa de duas maneiras:

No primeiro modo, a tabela é impressa dentro da banda a qual pertence, e fica igual como no **editor**. Neste modo, a tabela não se separa durante as páginas se a sua largura é maior que a largura da página do relatório. Este é o modo de impressão padrão.

O segundo modo é dinâmico. Neste modo, a tabela é montada com a ajuda de **script**. Durante isso, a tabela resultante pode ser diferente da tabela inicial, assim como o relatório preparado do **CIGAM Report** difere do **template** do relatório. No modo dinâmico, a tabela pode se dividir nas páginas se ela não couber na página do relatório.

No modo dinâmico, a tabela não fica impressa na banda na qual foi colocada. Ao invés disso, a tabela por si mesma gera um conjunto de bandas, as quais contêm partes da tabela resultante. Este modo de trabalho impõe os seguintes limites:

- Nunca coloque outros objetos abaixo ou perto da tabela. Ao invés disso, use a banda Filha;
- Nunca coloque dois objetos Tabela nos detalhes.

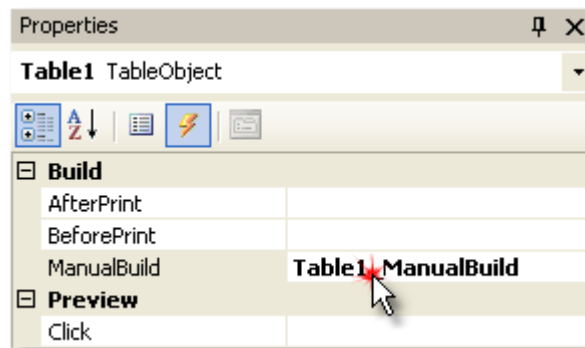
Vamos dar uma olhada no modo dinâmico em detalhes.

---

este modo é conectado com programação e requer altas qualificações do desenvolvedor do relatório.

---

A formação de uma tabela é feita com a ajuda de **script**. Para criar um **script**, selecione o objeto Tabela, na janela Propriedades clique no botão **Eventos** e dê um duplo clique no evento **ManualBuild**:



Quando isto é feito, um manipulador de eventos vazio é adicionado no código do relatório:

```
private void Table1_ManualBuild(object sender, EventArgs e)
{
}
```

Neste modo, a tabela fonte é usada como **template**. No código do evento, você pode imprimir linhas e colunas de uma tabela fonte quantas vezes for preciso. Durante isso, a tabela resultante será formada, a qual pode conter um número ilimitado de linhas e colunas. Tal tabela pode se separar através das páginas se ela não couber na página do relatório.

Para imprimir a tabela, os seguintes métodos do objeto Tabela são usados:

| Método              | Parâmetros    | Descrição   |
|---------------------|---------------|---|
| <b>PrintRow</b>     | Int index     | Imprime a linha com o índice especificado. A numeração da linha inicia com 0. |
| <b>PrintColumn</b>  | Int index     | Imprime a coluna com o índice especificado.                                   |
| <b>PrintRows</b>    | Int[] índices | Imprime várias linhas da tabela.  |
| <b>PrintRows</b>    | -             | Imprime todas as linhas da tabela.  |
| <b>PrintColunas</b> | Int[] índices | Imprime varias colunas da tabela.   |
| <b>PrintColunas</b> | -             | Imprime todas as colunas da tabela.   |
| <b>PageBreak</b>    | -             | Insere uma quebra de página antes de imprimir a próxima coluna ou linha.      |

A impressão de uma tabela pode ser feita usando um dos seguintes métodos:

O primeiro método – imprimir do topo até o fim, então da esquerda para direita. Este método se encaixa melhor em uma tabela com um tamanho variável de linhas. Você deve chamar os métodos na seguinte ordem:

- **PrintRow**(índice da linha);
- Uma ou mais chamadas dos métodos **PrintColumn**(índice da coluna) ou **PrintColunas**(índices das colunas) para imprimir as colunas indicadas;
- Ou uma chamada do método **PrintColunas()** para imprimir todas as colunas;
- Repita esta seqüência para imprimir todas as linhas necessárias da tabela.

---

Cada coluna da tabela deve conter o mesmo número de linhas. Tenha isso em mente quando usar os métodos **PrintRow(int índice)** e **PrintRows(int [] índices)**.

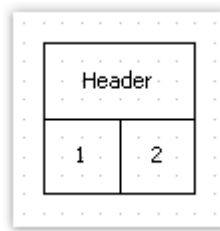
---

A violação da ordem de chamada dos métodos de impressão ocasiona erros quando geramos o relatório. Um dos erros – tentando imprimir uma tabela com a ajuda do seguinte código:

esta seqüência de métodos não está correta. Você deve começar imprimindo a tabela com ambos os métodos (**PrintRow** ou **PrintColumn**).

### Imprimindo cabeçalhos complexos

Aqui estamos nós, falando sobre cabeçalhos que contém células estendidas. Quando imprimimos linhas ou colunas da tabela, a qual tem células estendidas, as células automaticamente irão aumentar o tamanho. Nós vamos mostrar isto no próximo exemplo:



| Header |   |
|--------|---|
| 1      | 2 |
|        |   |

Nós vamos criar um manipulador de eventos **ManualBuild**, o qual imprimirá a primeira coluna três vezes e a segunda coluna uma vez:

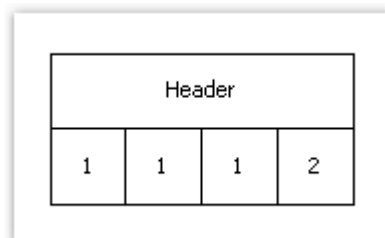
```
private void Table1_ManualBuild(object sender, EventArgs e)
{
    // Imprimindo linha 1 e colunas 0, 0, 0, 1
    Table1.PrintRow(0);
    Table1.PrintColumn(0);
    Table1.PrintColumn(0);
    Table1.PrintColumn(0);
    Table1.PrintColumn(1);
    // Imprimindo linha 1 e colunas 0, 0, 0, 1
    Table1.PrintRow(1);
    Table1.PrintColumn(0);
    Table1.PrintColumn(0);
    Table1.PrintColumn(0);
    Table1.PrintColumn(1);
}
```

---

Veja que nós imprimimos o mesmo número de colunas em cada linha. Se esta regra é violada, então nós vamos obter um resultado inesperado.

---

Como resultado da execução deste código, nós vamos obter o seguinte:



| Header |   |   |   |
|--------|---|---|---|
| 1      | 1 | 1 | 2 |
|        |   |   |   |

Como visto, a célula do cabeçalho é estendida automaticamente. Nós vamos tornar este código um pouco mais complexo, então dois grupos de colunas serão impressas:

```
private void Table1_ManualBuild(object sender, EventArgs e)
{
    // imprimir linha 0 e dois grupos de colunas 0,0,0,1
```

```
Table1.PrintRow(0);
// grupo1
Table1.PrintColumn(0);
Table1.PrintColumn(0);
Table1.PrintColumn(0);
Table1.PrintColumn(1);
// grupo 2
Table1.PrintColumn(0);
Table1.PrintColumn(0);
Table1.PrintColumn(0);
Table1.PrintColumn(1);
// imprimir linha 1 e dois grupos de colunas 0,0,0,1
Table1.PrintRow(1);
// grupo 1
Table1.PrintColumn(0);
Table1.PrintColumn(0);
Table1.PrintColumn(0);
Table1.PrintColumn(1);
// grupo 2
Table1.PrintColumn(0);
Table1.PrintColumn(0);
Table1.PrintColumn(0);
Table1.PrintColumn(1);
}
```

Quando nós gerarmos o relatório, vamos ver o seguinte resultado:

| Header |   |   |   | Header |   |   |   |
|--------|---|---|---|--------|---|---|---|
| 1      | 1 | 1 | 2 | 1      | 1 | 1 | 2 |

Quando imprimirmos a segunda coluna com o seguinte código:

```
Table1.PrintColumn (1);
```

O cabeçalho é finalizado e uma impressão mais longa da primeira coluna iniciará um novo cabeçalho:

```
//grupo 2
Table1. PrintColumn (0);
```

## Usando totais

No modo dinâmico do objeto Tabela, as seguintes funções de totais são suportadas:

| Função       | Parâmetros              | Descrição   |
|--------------|-------------------------|---|
| <b>Sum</b>   | Célula <b>TableCell</b> | Retorna a soma dos valores contidos na célula.                |
| <b>Min</b>   | Célula <b>TableCell</b> | Retorna o mínimo dos valores contidos na célula.              |
| <b>Max</b>   | Célula <b>TableCell</b> | Retorna o máximo dos valores contidos na célula.              |
| <b>Avg</b>   | Célula <b>TableCell</b> | Retorna a média dos valores contidos na célula.               |
| <b>Count</b> | Célula <b>TableCell</b> | Retorna o número de linhas, contidas na células especificada. |

Em um modo de impressão comum (não dinâmico) estas funções não irão funcionar.

Para usar a função total, coloque-a na célula da tabela. Por exemplo, a seguinte função calcula a soma dos valores contidos na célula chamada **Cell1**;

**[Sum(Cell1)]**

Durante este processo, todas as células localizadas acima e na esquerda da célula atual (na qual nós estamos calculando a soma) são analisadas.

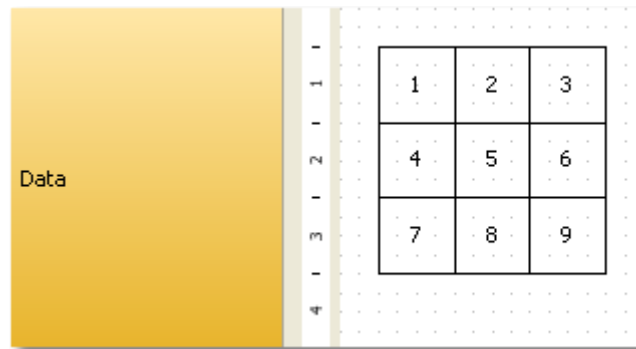
## Layout da tabela

A tabela que é montada dinamicamente, pode ser automaticamente separada durante as páginas. Este comportamento é controlado pela propriedade **Layout** da tabela. Você pode escolher um dos seguintes valores:

| Valor                 | Descrição  |
|-----------------------|--|
| <b>AcrossThenDown</b> | A tabela é processada horizontalmente e depois verticalmente.                        |
| <b>DownThenAcross</b> | A tabela é processada verticalmente e depois horizontalmente.                        |
| <b>Wrapped</b>        | A largura da página é condicionada a largura da página e processada na mesma página. |

## Exemplos

Vamos ver a impressão de tabelas de exemplo. Como **template**, nós vamos usar o seguinte relatório:



|   |   |   |
|---|---|---|
| 1 | 2 | 3 |
| 4 | 5 | 6 |
| 7 | 8 | 9 |

Para começar, selecione a tabela e crie um manipulador de eventos para um evento **ManualBuild**.

**Exemplo 1: Imprimindo toda a tabela do início ao fim**

```
private void Table1_ManualBuild(object sender, EventArgs e)
{
    // imprimir linha 0 e todas suas colunas
    Table1.PrintRow(0);
    Table1.PrintColumns();
    // imprimir linha 1 e todas suas colunas
    Table1.PrintRow(1);
    Table1.PrintColumns();
    // imprimir linha 2 e todas suas colunas
    Table1.PrintRow(2);
    Table1.PrintColumns();
}
```

Como resultado, a seguinte tabela será impressa, a qual não difere do **template**:

|   |   |   |
|---|---|---|
| 1 | 2 | 3 |
| 4 | 5 | 6 |
| 7 | 8 | 9 |

**Exemplo 2: Imprimindo a tabela do início ao fim com linhas repetidas**

```
private void Table1_ManualBuild(object sender, EventArgs e)
{
    // imprimir linha 0 e todas suas colunas
    Table1.PrintRow(0);
    Table1.PrintColumns();
    // imprimir 3 cópias da linha 1 e todas suas colunas
    for (int i = 0; i < 3; i++)
    {
```

```
    Table1.PrintRow(1);  
    Table1.PrintColumns();  
}  
// imprimir linha 2 e todas suas colunas  
Table1.PrintRow(2);  
Table1.PrintColumns();  
}
```

Neste exemplo, a linha do meio é impressa três vezes. Como resultado vamos obter o seguinte:

|   |   |   |
|---|---|---|
| 1 | 2 | 3 |
| 4 | 5 | 6 |
| 4 | 5 | 6 |
| 4 | 5 | 6 |
| 7 | 8 | 9 |

**Exemplo 3: Imprimindo toda a tabela da esquerda para direita**

```
private void Table1_ManualBuild(object sender, EventArgs e)  
{  
    // Imprime a coluna 0 e todas as suas linhas  
    Table1.PrintColumn(0);  
    Table1.PrintRows();  
    // Imprime a coluna 1 e todas as suas linhas  
    Table1.PrintColumn(1);  
    Table1.PrintRows();  
    // Imprime a coluna 2 e todas as suas linhas  
    Table1.PrintColumn(2);  
    Table1.PrintRows();  
}
```

Como resultado, a seguinte tabela será impressa, a qual não difere do **template**:

|   |   |   |
|---|---|---|
| 1 | 2 | 3 |
| 4 | 5 | 6 |
| 7 | 8 | 9 |



**Exemplo 4: Imprimindo a tabela da esquerda para direito com as colunas repetidas**

```
private void Table1_ManualBuild(object sender, EventArgs e)
{
    // Imprime coluna 0 e todas suas linhas
    Table1.PrintColumn(0);
    Table1.PrintRows();
    // Imprime 3 cópias da coluna 1 e todas suas linhas
    for (int i = 0; i < 3; i++)
    {
        Table1.PrintColumn(1);
        Table1.PrintRows();
    }
    // Imprime coluna 2 e todas suas linhas
    Table1.PrintColumn(2);
    Table1.PrintRows();
}
```

Neste exemplo, a coluna do meio da tabela é impressa três vezes. Como resultado, vamos obter o seguinte:

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 2 | 2 | 3 |
| 4 | 5 | 5 | 5 | 6 |
| 7 | 8 | 8 | 8 | 9 |

**Exemplo 5: Imprimindo a tabela com linhas e colunas repetidas**

```
private void Table1_ManualBuild(object sender, EventArgs e)
{
    // imprime linha 0
    Table1.PrintRow(0);
    // imprime coluna 0
    Table1.PrintColumn(0);
    // imprime 3 cópias da coluna 1
    for (int i = 0; i < 3; i++)
    Table1.PrintColumn(1);
    // imprime coluna 2
    Table1.PrintColumn(2);
    // imprime 3 cópias da linha 1
    for (int j = 0; j < 3; j++)
```

```
{
    Table1.PrintRow(1);
    // imprime coluna 0
    Table1.PrintColumn(0);
    // imprime 3 cópias da coluna 1
    for (int i = 0; i < 3; i++)
        Table1.PrintColumn(1);
    // imprime coluna 2
    Table1.PrintColumn(2);
}
// imprime linha 2
Table1.PrintRow(2);
// imprime coluna 0
Table1.PrintColumn(0);
// imprime 3 cópias da coluna 1
    for (int i = 0; i < 3; i++)
        Table1.PrintColumn(1);
// imprime coluna 2
Table1.PrintColumn(2);
}
```

---

Veja que nós imprimimos o mesmo número de colunas em cada linha. Se esta regra é violada, então nós vamos obter um resultado inesperado.

---

Neste exemplo, a linha do meio e a coluna do meio da tabela foram impressas 3 vezes. E como resultados, nós obtemos o seguinte:

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 2 | 2 | 3 |
| 4 | 5 | 5 | 5 | 6 |
| 4 | 5 | 5 | 5 | 6 |
| 4 | 5 | 5 | 5 | 6 |
| 7 | 8 | 8 | 8 | 9 |

### Exemplo 6: Usando a fonte de dados

Em todos os exemplos considerados, nós imprimimos a tabela, a qual contém um texto comum. Neste exemplo nós mostraremos como formar uma tabela usando uma fonte de dados. Para isso, criaremos uma tabela que tem o seguinte formato:

| Product Name           | Unit Price           | Units In Stock          |
|------------------------|----------------------|-------------------------|
| [Products.ProductName] | [Products.UnitPrice] | [Products.UnitsInStock] |
|                        |                      |                         |

Nós vamos criar o manipulador de eventos **ManualBuild**, o qual estará fazendo o seguinte:

- Obter a fonte de dados, definida no relatório;
- Inicializá-lo (preenche-lo com dados);
- Imprimir as linhas da tabela enquanto houver linhas na fonte de dados.

Aqui está o código do manipulador:

```
private void Table1_ManualBuild(object sender, EventArgs e)
{
    // pegar a fonte de dados pelo nome
    DataSourceBase rowData = Report.GetDataSource("Products");
    // inicializá-la
    rowData.Init();
    // imprimir o cabeçalho da tabela
    Table1.PrintRow(0);
    Table1.PrintColumns();
    // processa as linhas da fonte de dados em Loop
    while (rowData.HasMoreRows)
    {
        // imprime uma linha na tabela
        Table1.PrintRow(1);
        Table1.PrintColumns();
        // selecionar a próxima linha
        rowData.Next();
    }
    // imprimir o rodapé da tabela
    Table1.PrintRow(2);
    Table1.PrintColumns();
}
```

Se nós gerarmos o relatório, nós vamos obter o seguinte:

| Product Name                    | Unit Price | Units In Stock |
|---------------------------------|------------|----------------|
| Chai                            | 18,00      | 39             |
| Chang                           | 19,00      | 17             |
| Aniseed Syrup                   | 10,00      | 13             |
| Chef Anton's Cajun Seasoning    | 22,00      | 53             |
| Chef Anton's Gumbo Mix          | 21,35      | 0              |
| Grandma's Boysenberry Spread    | 25,00      | 120            |
| Uncle Bob's Organic Dried Pears | 30,00      | 15             |
| Northwoods Cranberry Sauce      | 40,00      | 6              |

### Exemplo 7: Inserindo quebras de página

Usando o método **PageBreak** do objeto Tabela, você pode inserir uma quebra de página quando imprimir a tabela. Abra o método antes de imprimir uma linha/coluna.

Nós vamos usar o exemplo 1 para demonstrar como o método **PageBreak** trabalha. Vamos imprimir a terceira linha em uma nova página.

```
private void Table1_ManualBuild(object sender, EventArgs e)
{
    // imprime a linha 0 e suas colunas
    Table1.PrintRow(0);
    Table1.PrintColumns();
    // imprime a linha 1 e suas colunas
    Table1.PrintRow(1);
    Table1.PrintColumns();
    // insere uma quebra de página antes da linha 2
    Table1.PageBreak();
    // imprime a linha 2 e suas colunas
    Table1.PrintRow(2);
    Table1.PrintColumns();
}
```

Como resultado, obteremos o seguinte:

|   |   |   |
|---|---|---|
| 1 | 2 | 3 |
| 4 | 5 | 6 |

|   |   |   |
|---|---|---|
| 7 | 8 | 9 |
|---|---|---|

### Exemplo 8: Imprimindo totais

Nós veremos o uso da função total no exemplo 6. Nós a modificaremos da seguinte maneira:

| Product Name           | Unit Price           | Units In Stock          |
|------------------------|----------------------|-------------------------|
| [Products.ProductName] | [Products.UnitPrice] | [Products.UnitsInStock] |
|                        | <b>Total:</b>        | <b>[Sum(Cell8)]</b>     |

A célula Cell8, que iremos calcular

O valor da soma

Se nós gerarmos o relatório, obteremos o seguinte:

|                                 |               |             |
|---------------------------------|---------------|-------------|
| Outback Lager                   | 15,00         | 15          |
| Fløtemysost                     | 21,50         | 26          |
| Mozzarella di Giovanni          | 34,80         | 14          |
| Rød Kaviar                      | 15,00         | 101         |
| Longlife Tofu                   | 10,00         | 4           |
| Rhönbräu Klosterbier            | 7,75          | 125         |
| Lakkalikööri                    | 18,00         | 57          |
| Original Frankfurter grüne Soße | 13,00         | 32          |
|                                 | <b>Total:</b> | <b>3119</b> |

### Relatórios do Tipo Matriz

O objeto **Matriz** é uma tabela diversificada e é parecido com o objeto Tabela, é feito de linhas, colunas e células. Ao mesmo tempo, não se sabe antes quantas linhas e colunas a matriz irá ter – isto depende dos dados aos quais ela está conectada.

O objeto se parecerá com isso:

| Employee | [Year]    | Total |
|----------|-----------|-------|
| [Name]   | [Revenue] |       |
| Total    |           |       |

Quando imprimimos, a matriz preenche os valores e cresce para cima e para baixo. O resultado pode ser como o seguinte:

| Employee        | 1999        | 2000        | 2001        | 2002       | Total              |
|-----------------|-------------|-------------|-------------|------------|--------------------|
| Andrew Fuller   | \$3,900.00  | \$2,100.00  |             | \$1,800.00 | \$7,800.00         |
| Janet Leverling | \$6,100.00  | \$3,200.00  |             |            | \$9,300.00         |
| Nancy Davolio   | \$3,300.00  | \$2,700.00  | \$3,100.00  | \$1,700.00 | \$10,800.00        |
| Steven Buchanan |             | \$3,999.00  | \$8,100.00  |            | \$12,099.00        |
| <b>Total</b>    | \$13,300.00 | \$11,999.00 | \$11,200.00 | \$3,500.00 | <b>\$39,999.00</b> |

### Uma pequena teoria

Vamos ver os elementos da matriz:

|   | 1  | 2  | 3  | 4  |
|---|----|----|----|----|
| a | a1 | a2 | a3 | a4 |
| b | b1 | b2 | b3 | b4 |

Na figura, nós vemos a matriz com 2 linhas e 4 colunas. Aqui, a, b – cabeçalho da linha, 1,2,3,4 – cabeçalho da coluna, a1...a4, b1...b4 – células. Para montar tal relatório, somente uma fonte de dados será necessária, a qual tem 3 colunas e contém os seguintes dados:

|   |   |    |
|---|---|----|
| a | 1 | a1 |
| a | 2 | a2 |
| a | 3 | a3 |
| a | 4 | a4 |
| b | 1 | b1 |
| b | 2 | b2 |
| b | 3 | b3 |
| b | 4 | b4 |

Como visto, a primeira coluna representa as linhas da matriz, a segunda – colunas da matriz, e a terceira – conteúdo das células na inserção de linhas e colunas com o número indicado. Quando cria um relatório, o **CIGAM Report** cria uma matriz na memória e preenche com dados. Durante este processo, a matriz cresce dinamicamente, se a linha ou coluna com o número dado não existe ainda.

Um cabeçalho pode conter mais que um nível. Vamos ver o seguinte exemplo:

|   | 10    |       | 20    |       |
|---|-------|-------|-------|-------|
|   | 1     | 2     | 1     | 2     |
| a | a10.1 | a10.2 | a20.1 | a20.2 |
| b | b10.1 | b10.2 | b20.1 | b20.2 |

Neste exemplo, a coluna é composta, isto é, ela tem dois valores. Este relatório requer os seguintes dados:

|   |    |   |       |
|---|----|---|-------|
| a | 10 | 1 | a10.1 |
| a | 10 | 2 | a10.2 |
| a | 20 | 1 | a20.1 |
| a | 20 | 2 | a20.2 |
| b | 10 | 1 | b10.1 |
| b | 10 | 2 | b10.2 |
| b | 20 | 1 | b20.1 |
| b | 20 | 2 | b20.2 |

Aqui, a primeira coluna representa as linhas, a segunda e a terceira representam a coluna da matriz. A última coluna de dados contém o valor da célula.

Os próximos elementos da matriz – subtotal e total geral, a próxima figura os demonstra:

|       | 10          |             |                         | 20          |             |                         | Total         |
|-------|-------------|-------------|-------------------------|-------------|-------------|-------------------------|---------------|
|       | 1           | 2           | Total                   | 1           | 2           | Total                   |               |
| a     | a10.1       | a10.2       | a10.1+a10.2             | a20.1       | a20.2       | a20.1+a20.2             | sum(a)        |
| b     | b10.1       | b10.2       | b10.1+b10.2             | b20.1       | b20.2       | b20.1+b20.2             | sum(b)        |
| Total | a10.1+b10.1 | a10.2+b10.2 | a10.1+b10.1+a10.2+b10.2 | a20.1+b20.1 | a20.2+b20.2 | a20.1+b20.1+a20.2+b20.2 | sum(a)+sum(b) |

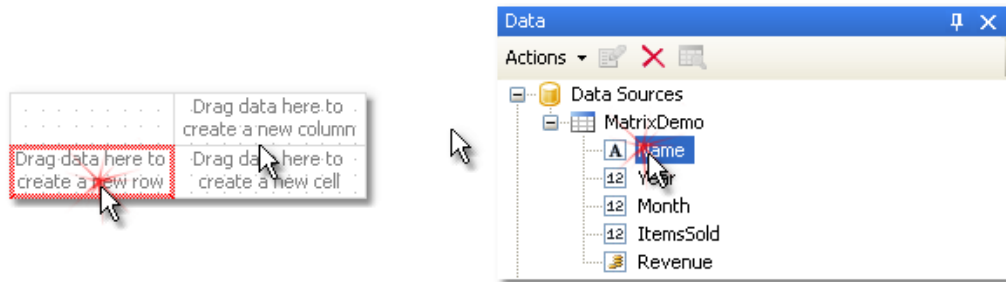
este relatório é construído com os mesmos dados usados no exemplo anterior. As linhas mostradas em cinza na figura são calculadas automaticamente.

### Configurando a matriz

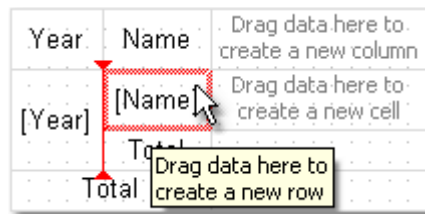
Depois de você colocar um novo objeto **Matriz** em uma folha, ela irá se parecer com o seguinte:



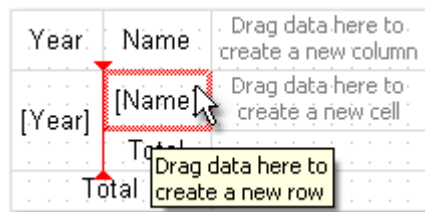
A matriz pode ser configurada com a ajuda do mouse. Para fazer isso, arraste e solte as colunas da fonte de dados da janela **Dados** na matriz, para criar linhas, colunas e células. A matriz destaca um quadro vermelho para colocar onde a nova data será colocada:



Se a matriz já contém alguns elementos, então quando colocar um novo elemento, um indicador será mostrado. Neste caso, os novos dados serão colocados entre os elementos **Year** e **Name**:



Você pode também trocar a ordem dos elementos existentes. Para fazer isso, clique na borda do elemento (mostrado com preto) e arraste-o para o lugar desejado:

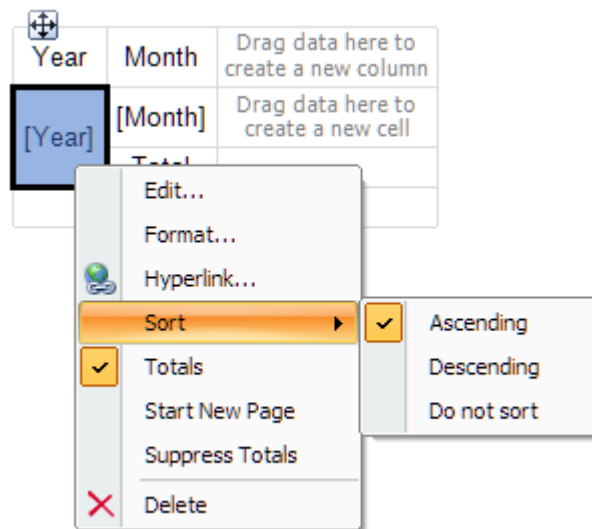


Para excluir um elemento, selecione-o com o mouse e aperte a tecla Delete.

### Configurando cabeçalhos

Para configurar o cabeçalho de um elemento, selecione-o e clique com o botão direito para mostrar o seu menu de contexto:





Por padrão, os dados no cabeçalho da matriz são dispostos na ordem crescente. Você pode mudar a ordem de disposição selecionando o item **Sort**.

Comumente todo item no cabeçalho da matriz tem um total (isto é a célula com o texto **Total**). Você pode excluir o total selecionando-o e apertando a tecla **Delete**. Para habilitá-lo novamente selecione um elemento ao qual ele pertence, e escolha o item **Total** no seu menu de contexto.

O item de menu **Iniciar Nova Página** diz para a matriz inserir quebras de página depois de imprimir cada valor do cabeçalho. Por exemplo, se você habilitar as quebras de página para o item **Year** (como mostrado na figura acima), cada valor do ano será impresso na sua própria página.

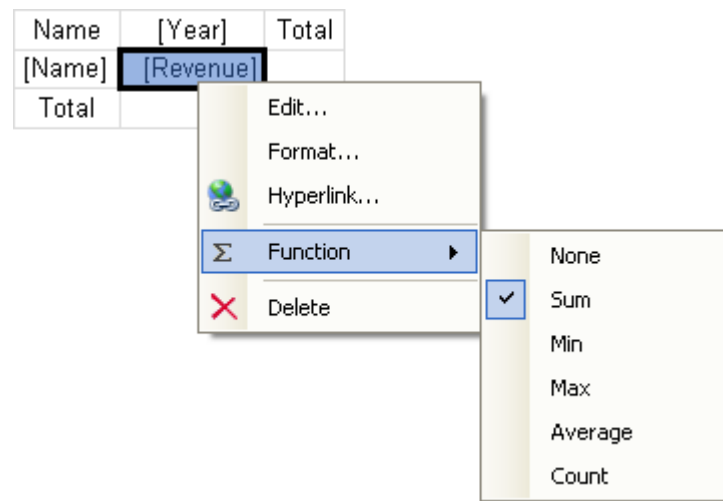
O item **Suprimir Totais** permite suprimir os totais no caso de um grupo (no qual o valor total é calculado) conter somente um valor.

### Configurando células

Para uma célula de matriz, você pode escolher uma função que será usada quando calcular o total. A lista de funções que podem ser usadas é a seguinte:

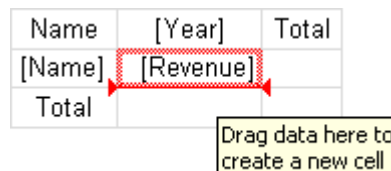
| Função  | Descrição                                       |
|---------|---|
| None    | O valor da celular não é processado.            |
| Sum     | Retorna a soma dos valores na célula da matriz. |
| Min     | Retorna o menor valor.                          |
| Max     | Retorna o maior valor.                          |
| Average | Retorna o valor médio.                          |
| Count   | Retorna o número de valores não vazios.         |

Por padrão, a função **Sum** é usada. Você pode mudar isso selecionando a célula e escolhendo o item **Function** no menu de contexto:



Escolha a função **None**, se você não pretende imprimir o total na célula dada.

Na matriz, pode haver uma ou mais células de dados. No caso da matriz ter várias células, elas podem ser organizadas lado-a-lado ou no topo uma da outra. A propriedade **CellsSideBySide** da matriz controla como as células são organizadas. Isso pode ser trocado através do menu de contexto do objeto **Matriz**. Você também pode escolher a ordem de organização (disposição) quando você adiciona uma segunda célula na matriz. Quando fizer isso, olhe para o indicador vermelho que mostra onde o segundo valor será colocado:



Depois de você ter adicionado o segundo valor, o resto dos valores serão adicionados na ordem escolhida.

### Estilizando a matriz

Para escolher a aparência da célula da matriz, clique na célula desejada. Com a ajuda da barra de ferramentas você pode definir a fonte, a borda e o preenchimento. Para trocar a aparência de várias células ao mesmo tempo, selecione o grupo de células. Para fazer isso, selecione a célula esquerda do topo, e, sem soltar o mouse, mova o mouse para selecionar o grupo:



Você pode usar estilos para mudar a aparência de toda a matriz. Para fazer isso, abra o menu de contexto do objeto **Matriz** e escolha o estilo:



### Gerenciamento do tamanho das linhas e colunas

O objeto **Matriz** é um tipo de objeto Tabela, portanto ele permite a definição do tamanho das linhas e colunas do mesmo modo.

Por padrão, a matriz tem o modo **AutoSize** habilitado. Neste modo, a matriz calcula o tamanho da coluna/linha automaticamente. Você pode também gerenciar o tamanho do objeto manualmente. Para fazer isso, desabilite a propriedade **AutoSize** da matriz. Linhas e colunas têm a mesma propriedade, você pode usá-la se o tamanho automático da matriz estiver habilitado.

Para limitar a largura máxima e a mínima de uma linha, selecione a linha e defina suas propriedades **MinWidth** e **MaxWidth**.

Para limitar a altura máxima e a mínima de uma linha, selecione a linha e defina suas propriedades **MinHeight** e **MaxHeight**.

## Exemplos

Vamos ver exemplos de uso do objeto **Matriz**. De início, crie um novo relatório e coloque o objeto **Matriz** na banda **Report Title**. Você pode também usar a banda **Dados** – no caso de não haver necessidade de conectar a banda à uma fonte de dados. Neste caso, não importa em qual das duas bandas você colocará a matriz, visto que ambas as bandas serão impressas uma vez quando o relatório é iniciado. O visual do relatório ficará assim:



Não ponha o objeto **Matriz** em bandas que serão impressas em cada página – **Cabeçalho de Página, Rodapé de Página, etc.** A matriz neste caso será criada toda vez que a banda for impressa.

A maioria dos exemplos usarão a tabela **MatrixDemo**, a qual vem junto com o pacote do **CIGAM Report**. esta tabela contém os seguintes dados:

| Nome            | Ano  | Mês | Itens Vendidos | Receita |
|-----------------|------|-----|----------------|---------|
| Nancy Davolio   | 1999 | 2   | 1              | 1000    |
| Nancy Davolio   | 1999 | 11  | 1              | 1100    |
| Nancy Davolio   | 1999 | 12  | 1              | 1200    |
| Nancy Davolio   | 2000 | 1   | 1              | 1300    |
| Nancy Davolio   | 2000 | 2   | 2              | 1400    |
| Nancy Davolio   | 2001 | 2   | 2              | 1500    |
| Nancy Davolio   | 2001 | 3   | 2              | 1600    |
| Nancy Davolio   | 2002 | 1   | 2              | 1800    |
| Andrew Fuller   | 2002 | 1   | 2              | 1800    |
| Andrew Fuller   | 1999 | 10  | 2              | 1900    |
| Andrew Fuller   | 1999 | 11  | 2              | 2000    |
| Andrew Fuller   | 2000 | 2   | 2              | 2100    |
| Janet Leverling | 1999 | 10  | 3              | 3000    |
| Janet Leverling | 1999 | 11  | 3              | 3100    |
| Janet Leverling | 2000 | 3   | 3              | 3200    |
| Steven Buchanan | 2001 | 1   | 3              | 4000    |
| Steven Buchanan | 2001 | 2   | 4              | 4100    |
| Steven Buchanan | 2000 | 1   | 4              | 3999    |

### Exemplo 1: Matriz simples

A matriz irá conter um valor em uma linha e uma coluna assim como uma célula de dados. Para construir uma matriz você precisa adicionar colunas de dados **MatrixDemo** da seguinte maneira:

- Adicione a coluna de dados **Year** ao cabeçalho da linha;
- Adicione a coluna de dados **Name** ao cabeçalho da coluna;
- Adicione a coluna de dados **Revenue** à célula da matriz.

Depois disso, a matriz se parecerá com o seguinte:

|        |           |       |
|--------|-----------|-------|
| Year   | [Name]    | Total |
| [Year] | [Revenue] |       |
| Total  |           |       |

Vamos melhorar a aparência da matriz:

- Escolha o estilo **Orange** para a matriz;
- Escolha a fonte **Tahoma,8** para todas as células da matriz;
- Selecione a palavra **Total** com o tipo negrito;
- Escolha o preenchimento do tipo **Glass** para as células da linha acima;
- Desabilite o auto-redimensionamento da matriz e aumente o tamanho das linhas e colunas.

Depois disso, a matriz terá o seguinte visual:

|              |           |              |
|--------------|-----------|--------------|
| <b>Year</b>  | [Name]    | <b>Total</b> |
| [Year]       | [Revenue] |              |
| <b>Total</b> |           |              |

Gere o relatório e você verá o seguinte resultado:

| Year         | Andrew Fuller | Janet Leverling | Nancy Davolio | Steven Buchanan | Total |
|--------------|---------------|-----------------|---------------|-----------------|-------|
| 1999         | 3900          | 6100            | 3300          |                 | 13300 |
| 2000         | 2100          | 3200            | 2700          | 3999            | 11999 |
| 2001         |               |                 | 3100          | 8100            | 11200 |
| 2002         | 1800          |                 | 1700          |                 | 3500  |
| <b>Total</b> | 7800          | 9300            | 10800         | 12099           | 39999 |

Há uma falta dos seguintes itens na matriz:

- Não há título para a coluna de dados **Name**;
- As somas não são impressas no formato monetário;

Você pode adicionar um título para a coluna de dados **Name** da seguinte maneira:

- O texto **Year/Employee** pode ser colocado no canto superior esquerdo da matriz;
- A linha diagonal e o segundo objeto Texto pode ser colocados lá, como mostrado abaixo:

|              | Employee  |              |
|--------------|-----------|--------------|
| <b>Year</b>  | [Name]    | <b>Total</b> |
| [Year]       | [Revenue] |              |
| <b>Total</b> |           |              |

- Habilite o título da matriz. Para fazer isso, escolha o item **Show Title** no menu de contexto do objeto **Matriz**. Qualquer texto pode ser incluso no título:

| Employee     |           |              |
|--------------|-----------|--------------|
| Year         | [Name]    | <b>Total</b> |
| [Year]       | [Revenue] |              |
| <b>Total</b> |           |              |

Para definir a formatação dos dados, selecione toda a área da célula, como mostrado na figura abaixo, e defina o formato selecionando o item **Format...** no menu de contexto:

| Employee     |           |              |
|--------------|-----------|--------------|
| Year         | [Name]    | <b>Total</b> |
| [Year]       | [Revenue] |              |
| <b>Total</b> |           |              |

Depois disso, o relatório preparado será assim:

|              | Employee      |                 |               |                 |              |
|--------------|---------------|-----------------|---------------|-----------------|--------------|
| Year         | Andrew Fuller | Janet Leverling | Nancy Davolio | Steven Buchanan | <b>Total</b> |
| 1999         | \$3,900.00    | \$6,100.00      | \$3,300.00    |                 | \$13,300.00  |
| 2000         | \$2,100.00    | \$3,200.00      | \$2,700.00    | \$3,999.00      | \$11,999.00  |
| 2001         |               |                 | \$3,100.00    | \$8,100.00      | \$11,200.00  |
| 2002         | \$1,800.00    |                 | \$1,700.00    |                 | \$3,500.00   |
| <b>Total</b> | \$7,800.00    | \$9,300.00      | \$10,800.00   | \$12,099.00     | \$39,999.00  |

### Exemplo 2: Cabeçalhos multinível

A matriz terá um valor em uma linha, dois valores em uma coluna e uma célula de dados. Nós vamos utilizar o exemplo anterior como base e adicionar um novo item nele:

- Nós vamos adicionar a coluna de dados **Month** ao cabeçalho da linha à direita do item **Year**.

Depois de adicionar um novo item, melhore a aparência da matriz. Isto também é necessário para definir a formatação das células. Depois disso, a matriz se parecerá com o seguinte:

|              |         | Employee  |       |
|--------------|---------|-----------|-------|
| Year         | Month   | [Name]    | Total |
| [Year]       | [Month] | [Revenue] |       |
|              | Total   |           |       |
| <b>Total</b> |         |           |       |

Gere o relatório e você terá o seguinte resultado:

|              |       | Employee      |                 |               |                 |             |
|--------------|-------|---------------|-----------------|---------------|-----------------|-------------|
| Year         | Month | Andrew Fuller | Janet Leverling | Nancy Davolio | Steven Buchanan | Total       |
| 1999         | 2     |               |                 | \$1,000.00    |                 | \$1,000.00  |
|              | 10    | \$1,900.00    | \$3,000.00      |               |                 | \$4,900.00  |
|              | 11    | \$2,000.00    | \$3,100.00      | \$1,100.00    |                 | \$6,200.00  |
|              | 12    |               |                 | \$1,200.00    |                 | \$1,200.00  |
|              | Total | \$3,900.00    | \$6,100.00      | \$3,300.00    | \$0.00          | \$13,300.00 |
| 2000         | 1     |               |                 | \$1,300.00    | \$3,999.00      | \$5,299.00  |
|              | 2     | \$2,100.00    |                 | \$1,400.00    |                 | \$3,500.00  |
|              | 3     |               | \$3,200.00      |               |                 | \$3,200.00  |
|              | Total | \$2,100.00    | \$3,200.00      | \$2,700.00    | \$3,999.00      | \$11,999.00 |
| 2001         | 1     |               |                 |               | \$4,000.00      | \$4,000.00  |
|              | 2     |               |                 | \$1,500.00    | \$4,100.00      | \$5,600.00  |
|              | 3     |               |                 | \$1,600.00    |                 | \$1,600.00  |
|              | Total | \$0.00        | \$0.00          | \$3,100.00    | \$8,100.00      | \$11,200.00 |
| 2002         | 1     | \$1,800.00    |                 | \$1,700.00    |                 | \$3,500.00  |
|              | Total | \$1,800.00    | \$0.00          | \$1,700.00    | \$0.00          | \$3,500.00  |
| <b>Total</b> |       | \$7,800.00    | \$9,300.00      | \$10,800.00   | \$12,099.00     | \$39,999.00 |

### Exemplo 3: Imprimindo o nome do mês

No exemplo anterior, havia números de meses impressos na matriz. Isso ocorreu porque a coluna de dados **Month** contém o número do mês, não o seu nome. Para imprimir o nome do mês, faça o seguinte:

- Escolha a célula onde o número do mês será impresso. No nosso caso é uma célula com o nome **Cell8**;
- Na janela Propriedades pressione o botão [raio] e dê um duplo clique no evento **BeforePrint**;
- O **CIGAM Report** adicionará um manipulador de eventos vazio no **script** do relatório. Escreva o seguinte código:

```
private void Cell8_BeforePrint(object sender, EventArgs e)
{
```

```

string[] month Names = new string[] {
    "January", "February", "March", "April",
    "May", "June", "July", "August",
    "September", "October", "November", "December"};
// Cell8 é uma célula que imprime o número do mês.
// Cell8.Value é um valor impresso na célula(o número do mês).
// Este valor é do tipo System.Object, portanto necessitamos convertê-lo
para int.
Cell8.Text = monthNames[(int)Cell8.Value - 1];
}


```

Quando você gerar o relatório, você terá o seguinte resultado:

| Year | Month        | Andrew Fuller | Janet Leverli |
|------|--------------|---------------|---------------|
| 1999 | February     |               |               |
|      | October      | \$1,900.00    | \$2,000.00    |
|      | November     | \$2,000.00    |               |
|      | December     |               |               |
|      | <b>Total</b> | \$3,900.00    |               |
| 2000 | January      |               |               |
|      | February     |               |               |
|      | March        |               |               |

#### Exemplo 4: Marcação condicional

Você pode definir um destaque (marcação) condicional para células da matriz, assim como o objeto Texto. Mais detalhes sobre isso podem ser encontrados na seção **Realce Condicional**.

Vamos ver o Exemplo 2 e veja como destacar a quantidade em vermelho se for maior que 3000. Para isso, selecione a célula com o texto **Revenue** e pressione o botão  na barra de ferramentas de Texto. No editor de condições, adicione a seguinte condição:

**Value > 3000**

Escolha a cor de texto vermelha para a condição. O relatório pronto se parecerá com o seguinte:



|              |              | Employee      |                 |               |                 |             |
|--------------|--------------|---------------|-----------------|---------------|-----------------|-------------|
| Year         | Month        | Andrew Fuller | Janet Leverling | Nancy Davolio | Steven Buchanan | Total       |
| 1999         | 2            |               |                 | \$1,000.00    |                 | \$1,000.00  |
|              | 10           | \$1,900.00    | \$3,000.00      |               |                 | \$4,900.00  |
|              | 11           | \$2,000.00    | \$3,100.00      | \$1,100.00    |                 | \$6,200.00  |
|              | 12           |               |                 | \$1,200.00    |                 | \$1,200.00  |
|              | <b>Total</b> | \$3,900.00    | \$6,100.00      | \$3,300.00    | \$0.00          | \$13,300.00 |
| 2000         | 1            |               |                 | \$1,300.00    | \$3,999.00      | \$5,299.00  |
|              | 2            | \$2,100.00    |                 | \$1,400.00    |                 | \$3,500.00  |
|              | 3            |               | \$3,200.00      |               |                 | \$3,200.00  |
|              | <b>Total</b> | \$2,100.00    | \$3,200.00      | \$2,700.00    | \$3,999.00      | \$11,999.00 |
| 2001         | 1            |               |                 |               | \$4,000.00      | \$4,000.00  |
|              | 2            |               |                 | \$1,500.00    | \$4,100.00      | \$5,600.00  |
|              | 3            |               |                 | \$1,600.00    |                 | \$1,600.00  |
|              | <b>Total</b> | \$0.00        | \$0.00          | \$3,100.00    | \$8,100.00      | \$11,200.00 |
| 2002         | 1            | \$1,800.00    |                 | \$1,700.00    |                 | \$3,500.00  |
|              | <b>Total</b> | \$1,800.00    | \$0.00          | \$1,700.00    | \$0.00          | \$3,500.00  |
| <b>Total</b> |              | \$7,800.00    | \$9,300.00      | \$10,800.00   | \$12,099.00     | \$39,999.00 |

Como visto, os valores totais não estão destacados. Isso ocorreu porque nós escolhemos a condição de marcação (destaque) para somente uma célula. Para destacar o resto dos valores, é necessário definir o destaque para todas as células da matriz.

Neste exemplo nós usamos a marcação condicional, a qual depende do valor da célula. Além disso, você pode destacar a célula dependendo dos valores dos cabeçalhos da matriz. Nós vamos mostrar com o seguinte exemplo, como destacar células, as quais pertençam ao ano 2000, em vermelho. Para isso, selecione as células da matriz como é mostrado na figura abaixo:

|              |              | Employee  |       |
|--------------|--------------|-----------|-------|
| Year         | Month        | [Name]    | Total |
| [Year]       | [Month]      | [Revenue] |       |
|              | <b>Total</b> |           |       |
| <b>Total</b> |              |           |       |

Defina a seguinte condição de destaque:

**(int)Matrix1.RowValues[0] == 2000**

Neste caso **Matrix1** é o nome da nossa matriz. A propriedade **RowValues** da matriz possui um tipo **object[]** e contém um vetor de valores do cabeçalho da linha, da linha atual impressa. O número de valores do vetor é igual ao número de níveis no cabeçalho. Há dois valores no nosso exemplo, o primeiro é **Year** e o segundo é **Month**.

Não destaque a última linha. A propriedade **RowValues** tem um valor indeterminado e causará erro quando construir o relatório.

Quando nós gerarmos o relatório, vamos ter o seguinte resultado:

|              |              | Employee      |                 |               |                 |             |
|--------------|--------------|---------------|-----------------|---------------|-----------------|-------------|
| Year         | Month        | Andrew Fuller | Janet Leverling | Nancy Davolio | Steven Buchanan | Total       |
| 1999         | 2            |               |                 | \$1,000.00    |                 | \$1,000.00  |
|              | 10           | \$1,900.00    | \$3,000.00      |               |                 | \$4,900.00  |
|              | 11           | \$2,000.00    | \$3,100.00      | \$1,100.00    |                 | \$6,200.00  |
|              | 12           |               |                 | \$1,200.00    |                 | \$1,200.00  |
|              | <b>Total</b> | \$3,900.00    | \$6,100.00      | \$3,300.00    | \$0.00          | \$13,300.00 |
| 2000         | 1            |               |                 | \$1,300.00    | \$3,999.00      | \$5,299.00  |
|              | 2            | \$2,100.00    |                 | \$1,400.00    |                 | \$3,500.00  |
|              | 3            |               | \$3,200.00      |               |                 | \$3,200.00  |
|              | <b>Total</b> | \$2,100.00    | \$3,200.00      | \$2,700.00    | \$3,999.00      | \$11,999.00 |
| 2001         | 1            |               |                 |               | \$4,000.00      | \$4,000.00  |
|              | 2            |               |                 | \$1,500.00    | \$4,100.00      | \$5,600.00  |
|              | 3            |               |                 | \$1,600.00    |                 | \$1,600.00  |
|              | <b>Total</b> | \$0.00        | \$0.00          | \$3,100.00    | \$8,100.00      | \$11,200.00 |
| 2002         | 1            | \$1,800.00    |                 | \$1,700.00    |                 | \$3,500.00  |
|              | <b>Total</b> | \$1,800.00    | \$0.00          | \$1,700.00    | \$0.00          | \$3,500.00  |
| <b>Total</b> |              | \$7,800.00    | \$9,300.00      | \$10,800.00   | \$12,099.00     | \$39,999.00 |

Você pode também usar a propriedade **ColumnValues** da matriz para referência de valores da coluna.

### Exemplo 5: Destacando linhas pares

Para melhorar a aparência de uma matriz, você pode destacar linhas pares ou colunas com outra cor. Nós vamos usar o Exemplo 2 para mostrar como fazer isso.

Selecione toda a área de dados da matriz como mostrado na figura:

|              |              | Employee  |       |
|--------------|--------------|-----------|-------|
| Year         | Month        | [Name]    | Total |
| [Year]       | [Month]      | [Revenue] |       |
|              | <b>Total</b> |           |       |
| <b>Total</b> |              |           |       |

Abra o editor de marcação condicional. Adicione a seguinte condição:

**Matrix1.RowIndex % 2 != 0**

E escolha uma cor de fundo mais escura do que a anterior. Neste exemplo, o nome da nossa matriz é **Matrix1**. A propriedade **RowIndex** da matriz retorna o número da linha atual impressa.

Para destacar uma coluna, use a propriedade **ColumnIndex** da matriz do mesmo modo.

Quando nós gerarmos o relatório, nós veremos o seguinte:

|              |              | Employee      |                 |               |                 |             |
|--------------|--------------|---------------|-----------------|---------------|-----------------|-------------|
| Year         | Month        | Andrew Fuller | Janet Leverling | Nancy Davolio | Steven Buchanan | Total       |
| 1999         | 2            |               |                 | \$1,000.00    |                 | \$1,000.00  |
|              | 10           | \$1,900.00    | \$3,000.00      |               |                 | \$4,900.00  |
|              | 11           | \$2,000.00    | \$3,100.00      | \$1,100.00    |                 | \$6,200.00  |
|              | 12           |               |                 | \$1,200.00    |                 | \$1,200.00  |
|              | <b>Total</b> | \$3,900.00    | \$6,100.00      | \$3,300.00    | \$0.00          | \$13,300.00 |
| 2000         | 1            |               |                 | \$1,300.00    | \$3,999.00      | \$5,299.00  |
|              | 2            | \$2,100.00    |                 | \$1,400.00    |                 | \$3,500.00  |
|              | 3            |               | \$3,200.00      |               |                 | \$3,200.00  |
|              | <b>Total</b> | \$2,100.00    | \$3,200.00      | \$2,700.00    | \$3,999.00      | \$11,999.00 |
| 2001         | 1            |               |                 |               | \$4,000.00      | \$4,000.00  |
|              | 2            |               |                 | \$1,500.00    | \$4,100.00      | \$5,600.00  |
|              | 3            |               |                 | \$1,600.00    |                 | \$1,600.00  |
|              | <b>Total</b> | \$0.00        | \$0.00          | \$3,100.00    | \$8,100.00      | \$11,200.00 |
| 2002         | 1            | \$1,800.00    |                 | \$1,700.00    |                 | \$3,500.00  |
|              | <b>Total</b> | \$1,800.00    | \$0.00          | \$1,700.00    | \$0.00          | \$3,500.00  |
| <b>Total</b> |              | \$7,800.00    | \$9,300.00      | \$10,800.00   | \$12,099.00     | \$39,999.00 |

### Exemplo 6: Usando Expressões

Nos exemplos anteriores nós criamos uma matriz arrastando colunas da janela Dados. Você pode também usar expressões para este propósito. Para inserir uma expressão na matriz, faça o seguinte:

- Adicione qualquer elemento da janela **Dados** na matriz. Isso pode ser qualquer elemento, por exemplo, uma variável de sistema **Date** – nós somente a usamos para criar um elemento da matriz;
- Dê um duplo clique no elemento e selecione a expressão necessária na janela de edição de expressões.

Se a sua matriz tem expressões ao invés de campos de dados, você deve checar se a propriedade **DataSource** da matriz foi definida corretamente. Quando trabalhamos com colunas de dados, a propriedade é preenchida automaticamente quando voce arrasta a coluna para a matriz.

Vamos considerar um exemplo de como usar expressões. Para isso, nós vamos usar a tabela **OrderDetails** como fonte de dados, a qual contém uma lista de produtos vendidos, agrupado por empregados. Há vários relacionamentos nesta tabela, os quais dão acesso ao nome do empregado, o nome do produto e a sua categoria.

Nossa matriz mostrará as vendas de cada empregado, organizadas por produtos. Para construir nossa matriz, faça o seguinte:

- Adicione a coluna de dados **OrderDetails.Produtos.Categorias.NomeCategoria** ao cabeçalho da coluna;
- Adicione qualquer item ao cabeçalho da linha para criar um elemento da matriz. Depois defina a seguinte expressão para a célula:

**[OrderDetails.UnitPrice] \* [OrderDetails.Quantity] \* (decimal)(1 – [OrderDetails.Discount])**

---

Por que nós indicamos uma coluna tão longa como o nome do empregado se nós poderíamos pegar um nome do **Employees.FirstName**? Nós fizemos isto porque a matriz está conectada à fonte de dados **OrderDetails**. Usando relacionamentos entre esta fonte de dados e as outras tabelas, é fácil de se referir as colunas dela (mais detalhes sobre relacionamentos podem ser lidos no capítulo Data). Se nós nos referirmos diretamente à coluna de dados **Employees.FirstName**, nós teremos o primeiro nome do empregado na tabela.

---

Defina a aparência da matriz. Depois disso, ela se parecerá com o seguinte:

**[imagem\_tabela\_amarela]**

Quando nós gerarmos o relatório, veremos uma grande matriz que ocupará duas folhas:

### *Exemplo 7: Imagens nas células*

As células da matriz são herdadas do objeto Texto e podem mostrar dados textuais. Se isto não é o suficiente, você pode colocar qualquer objeto dentro da célula. Vamos ver como mostrar a figura na matriz.

Para isso nós vamos usar o Exemplo 6 como base. Vamos adicionar a foto de um empregado (coluna de dados **Employees.Photo**) e a categoria imagem (coluna de dados **Categorias.Picture**). Faça o seguinte:

- Selecione a célula que contém o nome do empregado e aumente o seu tamanho;
- Adicione o objeto **Picture** a esta célula;
- Para mostrar a foto de um empregado, vincule o objeto **Picture** à seguinte coluna de dados (isto pode ser feito no editor de objeto):

OrderDetail.Orders.Employees.Photo

- Selecione a célula que contém o nome da categoria e aumente o seu tamanho;
- Adicione o objeto **Picture** a esta coluna;
- Para mostrar uma figura da categoria, vincule o objeto **Picture** à seguinte coluna de dados (isto pode ser feito no editor do objeto):

OrderDetails.Produtos.Categorias.Picture

Depois disso, a matriz se parecerá com o seguinte:

| Employee  | [CategoryName]  | Total |
|---|---|-------|
| [Order<br>Details.Orders.Empl<br>oyees.FirstName] + " "<br>+ [Order<br>Details.Orders.Empl<br>oyees.LastName] | [[Order Details.UnitPrice] *<br>[Order Details.Quantity] *<br>(decimal)(1 - [Order<br>Details.Discount])] |       |
| <b>Total</b>  |   |       |

Se nós gerarmos o relatório, vamos ter o seguinte:

| Employee         | Beverages | Condiments | Confections | Dairy Products | Grains/Cereals |
|------------------|-----------|------------|-------------|----------------|----------------|
| Andrew Fuller    | 40 248,25 | 14 850,67  | 21 455,69   | 23 812,55      | 11 111,11      |
| Anne Dodsworth   | 19 642,56 | 10 125,55  | 8 053,16    | 21 101,13      | 1 234,56       |
| Janet Leverling  | 44 757,41 | 13 381,64  | 33 622,40   | 32 320,84      | 21 098,76      |
| Laura Callahan   | 17 897,85 | 14 637,66  | 21 699,91   | 21 269,47      | 10 987,65      |
| Margaret Peacock | 50 308,21 | 23 314,87  | 27 768,73   | 35 987,65      | 15 432,10      |
| Michael Suyama   | 9 450,20  | 4 648,47   | 6 859,63    | 5 432,10       | 3 210,98       |
| Nancy Davolio    | 46 599,36 | 13 561,56  | 28 568,21   | 25 432,10      | 12 345,67      |
| Robert King      | 27 963,83 | 8 851,38   | 15 432,10   | 10 987,65      | 5 432,10       |
| Steven Buchanan  | 11 000,53 | 5 432,10   | 3 210,98    | 2 109,87       | 1 098,76       |

### Exemplo 8: Objetos nas células

Usando objetos inseridos nas células da matriz você pode ter vários efeitos visuais. Nós vamos mostrar no próximo exemplo, como fazer uma simples escala indicando o nível de vendas do empregado.

A matriz usará a fonte de dados **MatrixDemo**. Para construir a matriz, adicione colunas de dados da seguinte maneira:

- Adicione a coluna de dados **Year** ao cabeçalho da linha;
- Adicione a coluna de dados **Name** ao cabeçalho da coluna;
- Adicione a coluna de dados **Revenue** à célula da matriz.

Defina a aparência da matriz da seguinte maneira:

| Employee | [Year]    | Total |
|----------|-----------|-------|
| [Name]   | [Revenue] |       |
| Total    |           |       |

Agora, vamos adicionar três objetos **Shape** na célula com o valor **Revenue**. Estes objetos servirão como indicadores da seguinte maneira:

- Se o valor em uma célula for menor que 100, somente um objeto da cor vermelha será mostrado;
- Se o valor em uma célula for menor que 3000, dois objetos da cor amarela serão mostrados;
- Se o valor em uma célula for maior ou igual a 3000, três objetos da cor verde serão mostrados.

Agora a matriz ficará da seguinte maneira:

| Employee | [Year]        | Total |
|----------|---------------|-------|
| [Name]   | ■■■ [Revenue] |       |
| Total    |               |       |

Para controlar objetos, usaremos um manipulador de eventos para uma célula da matriz. Para isso, selecione a célula **Revenue**, e crie o manipulador de eventos usando a janela Propriedades. Escreva o seguinte código no manipulador:

```
private void Cell4_BeforePrint(object sender, EventArgs e)
{
    // Em nosso exemplo, a célula possui o nome Cell4.
    // Pegue o valor que está na propriedade Cell4.Value.
    // Algumas células de nossa matriz estarão vazias. Levaremos isto em
    // conta (null check).
    // O valor deve ser do tipo decimal, pois a coluna de dados
    // [MatriDemo.Revenue] é do tipo System.Decimal.
    decimal value = Cell4.Value == null ? 0 : (decimal)Cell4.Value;
    // Altere a visibilidade dos objetos dependendo do valor:
    // Valor < 100 - um objeto visível;
    // Valor < 3000 - dois objetos visíveis;
    // Valor >= 3000 - três objetos visíveis.
    Shape1.Visible = true;
    Shape2.Visible = value >= 100;
}
```

```

Shape3.Visible = value >= 3000;
// Escolha a cor dos objetos:
// Valor < 100 - vermelho;
// Valor < 3000 - amarelo;
// Valor >= 3000 - verde.
Color color = Color.Red;
if (value >= 100)
color = Color.Yellow;
if (value >= 3000)
color = Color.GreenYellow;
// Defina a cor do objeto
Shape1.Fill = new SolidFill(color);
Shape2.Fill = new SolidFill(color);
Shape3.Fill = new SolidFill(color);
}

```

Quando nós gerarmos o relatório, vamos ver o seguinte:

| Employee        | 1999             | 2000             | 2001             | 2002            | Total            |
|-----------------|------------------|------------------|------------------|-----------------|------------------|
| Andrew Fuller   | 3 900,00         | 2 100,00         |                  | 1 800,00        | 7 800,00         |
| Janet Leverling | 6 100,00         | 3 200,00         |                  |                 | 9 300,00         |
| Nancy Davolio   | 3 300,00         | 2 700,00         | 3 100,00         | 1 700,00        | 10 800,00        |
| Steven Buchanan |                  | 3 999,00         | 8 100,00         |                 | 12 099,00        |
| <b>Total</b>    | <b>13 300,00</b> | <b>11 999,00</b> | <b>11 200,00</b> | <b>3 500,00</b> | <b>39 999,00</b> |

### Exemplo 9: Preenchendo a matriz manualmente

Em todos os exemplos que nós vimos, a matriz era preenchida automaticamente porque era conectada a uma fonte de dados. A fonte de dados para a matriz é indicada na propriedade **DataSource**. Embora nós não tenhamos definido manualmente o valor desta propriedade, isso ocorreu implicitamente quando adicionamos as colunas de dados à matriz.

Usando **script** é possível preencher a matriz manualmente. Para isso, é necessário criar um manipulador de eventos **ManualBuild** da matriz. Chame o método **Adicionar Value** no código do manipulador para adicionar um valor. Vamos mostrar como criar uma matriz que imprimirá uma tabela 10x10 do seguinte tipo:



|     |   |   |   |     |
|-----|---|---|---|-----|
|     | 1 | 2 | 3 | ... |
| 1   | 1 |   |   |     |
| 2   |   | 2 |   |     |
| 3   |   |   | 3 |     |
| ... |   |   |   | ... |

Faça o seguinte:

- Adicione uma matriz vazia no relatório;
- Coloque quaisquer elementos da janela Dados na linha, coluna e célula da matriz. Então abra o editor de expressão clicando duas vezes no elemento da matriz e limpando a expressão;
- Limpe a propriedade **DataSource** da fonte de dados.

Estes passos são necessários para criar uma matriz burra que tenha uma linha, coluna e célula. Como resultado, a matriz será a seguinte:

|       |   |       |
|-------|---|-------|
|       | ☐ | Total |
| ☐     | ☐ |       |
| Total |   |       |

Agora crie um manipulador de eventos **ManualBuild**. Para isso, selecione a matriz, vá na janela propriedades e pressione o botão ⚡. Dê um duplo clique no evento **ManualBuilt** e o **CIGAM Report** criará um manipulador de eventos vazio. Escreva o seguinte código nele:

```
private void Matrix1_ManualBuild(object sender, EventArgs e)
{
    // Nossa matriz possui um nível em linha, coluna e célula.
    // Crie 3 arrays do tipo object[], cada um com um elemento
    // (por número de níveis).
    object[] columnValues = new object[1];
    object[] rowValues = new object[1];
    object[] cellValues = new object[1];
    for (int i = 1; i <= 10; i++)
    {
        // Preenchendo os arrays
        columnValues[0] = i;
```



```
rowValues[0] = i;  
cellValues[0] = i;  
// Adicionando dados à matriz  
Matrix1.AddValue(columnValues, rowValues, cellValues);  
}  
}
```

Em um manipulador, você deve usar o método **Adicionar Value** do objeto **Matriz** para preenchê-la com dados. Este método tem três parâmetros cada dos quais estão em um vetor do tipo **System.Object**. O primeiro parâmetro é o valor da coluna, o segundo é o valor da linha, e o terceiro é o valor da célula. Perceba que o número de valores em cada vetor deve obedecer às configurações dos objetos! No nosso caso, um objeto tem um nível na coluna, linha e célula, correspondentemente nós suprimos um valor para colunas, um para linhas e um para as células.

Quando nós gerarmos o relatório, nós veremos o seguinte:

|       | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Total |
|-------|---|---|---|---|---|---|---|---|---|----|-------|
| 1     | 1 |   |   |   |   |   |   |   |   |    | 1     |
| 2     |   | 2 |   |   |   |   |   |   |   |    | 2     |
| 3     |   |   | 3 |   |   |   |   |   |   |    | 3     |
| 4     |   |   |   | 4 |   |   |   |   |   |    | 4     |
| 5     |   |   |   |   | 5 |   |   |   |   |    | 5     |
| 6     |   |   |   |   |   | 6 |   |   |   |    | 6     |
| 7     |   |   |   |   |   |   | 7 |   |   |    | 7     |
| 8     |   |   |   |   |   |   |   | 8 |   |    | 8     |
| 9     |   |   |   |   |   |   |   |   | 9 |    | 9     |
| 10    |   |   |   |   |   |   |   |   |   | 10 | 10    |
| Total | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 55    |

Vamos demonstrar como adicionar um valor 21 à matriz, na intersecção da linha 3 com a coluna 7. Para isso, mude o código da seguinte maneira:

```
private void Matrix1_ManualBuild(object sender, EventArgs e)  
{
```

```

object[] columnValues = new object[1];
object[] rowValues = new object[1];
object[] cellValues = new object[1];
for (int i = 1; i <= 10; i++)
{
    columnValues[0] = i;
    rowValues[0] = i;
    cellValues[0] = i;
    Matrix1.AddValue(columnValues, rowValues, cellValues);
}
columnValues[0] = 7;
rowValues[0] = 3;
cellValues[0] = 21;
Matrix1.AddValue(columnValues, rowValues, cellValues);
}
    
```

Como resultado, teremos o seguinte:

|       | 1 | 2 | 3 | 4 | 5 | 6 | 7  | 8 | 9 | 10 | Total |
|-------|---|---|---|---|---|---|----|---|---|----|-------|
| 1     | 1 |   |   |   |   |   |    |   |   |    | 1     |
| 2     |   | 2 |   |   |   |   |    |   |   |    | 2     |
| 3     |   |   | 3 |   |   |   | 21 |   |   |    | 24    |
| 4     |   |   |   | 4 |   |   |    |   |   |    | 4     |
| 5     |   |   |   |   | 5 |   |    |   |   |    | 5     |
| 6     |   |   |   |   |   | 6 |    |   |   |    | 6     |
| 7     |   |   |   |   |   |   | 7  |   |   |    | 7     |
| 8     |   |   |   |   |   |   |    | 8 |   |    | 8     |
| 9     |   |   |   |   |   |   |    |   | 9 |    | 9     |
| 10    |   |   |   |   |   |   |    |   |   | 10 | 10    |
| Total | 1 | 2 | 3 | 4 | 5 | 6 | 28 | 8 | 9 | 10 | 76    |

Como visto, a matriz calcula automaticamente os totais.



Universidade Corporativa CIGAM

[www.cigam.com.br](http://www.cigam.com.br)

[treinamento@cigam.com.br](mailto:treinamento@cigam.com.br)

Você pode usar o manipulador de eventos **ManualBuild** para a matriz que esteja conectada aos dados. Neste caso, o manipulador de eventos é chamado antes e então a matriz é preenchida com dados da fonte de dados.



## Relatórios interativos:

Um relatório do **CIGAM Report** preparado pode se tornar interativo. Isso significa que, isso irá reagir às ações do usuário na janela de **pré-visualização**. Você pode usar a seguinte interação:

- Quando clicamos em um objeto do relatório, algum tipo de operação é realizada. Por exemplo, você pode gerar um relatório detalhado e mostrá-lo em uma janela separada;
- A janela **pré-visualização** pode mostrar um contorno do relatório, o qual pode ser usado para navegação no relatório.

## Hyperlink

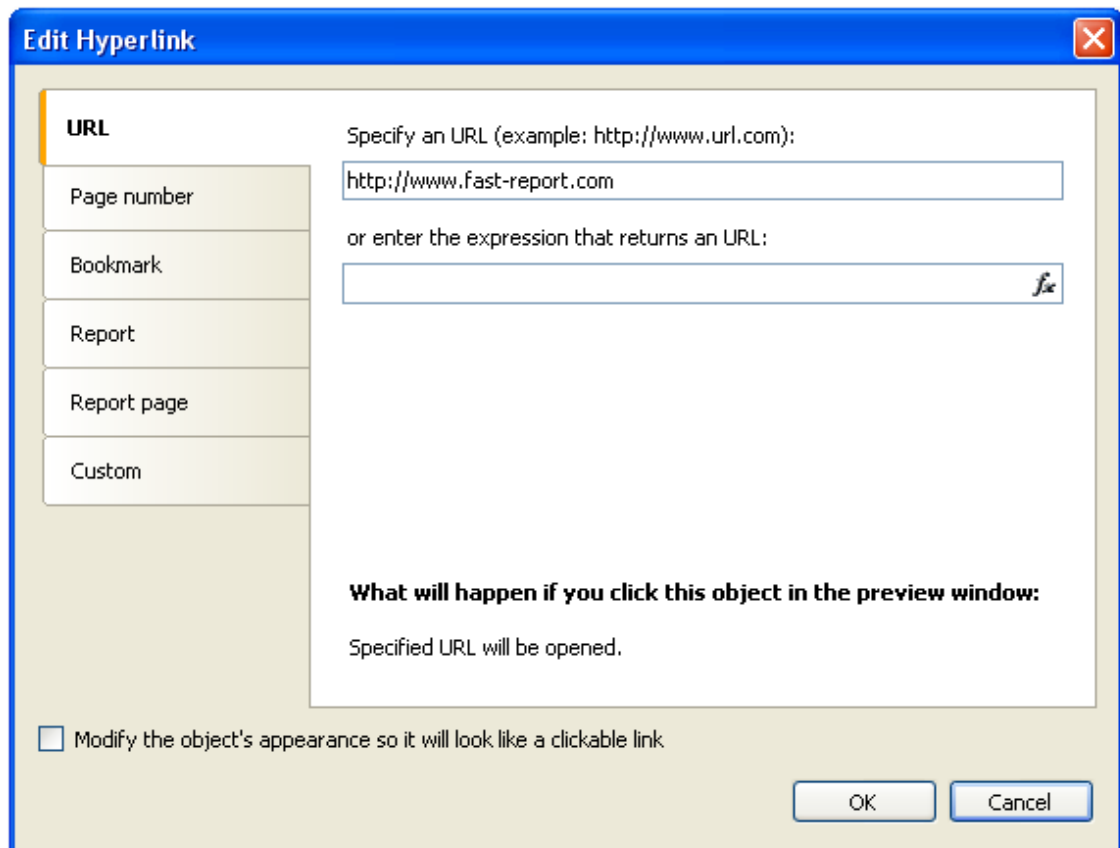
Quase todos os objetos do relatório têm a propriedade **Hyperlink**. Usando esta propriedade, você pode definir a reação de um objeto ao clique do mouse na janela **pré-visualização**.

Quando clicar em tal objeto, um dos seguintes pode ocorrer:

- Navegar até o endereço URL;
- Mandar e-mail;
- Executar qualquer tipo de comando do sistema;
- Navegar até a página do relatório com o número indicado;
- Navegar até o , definido em outro objeto do relatório;
- Gerar um relatório detalhado em uma janela **pré-visualização** separada;
- Ação customizada, definida no **script**.

## Configuração do *hyperlink*

Para configurar um *hyperlink*, selecione o objeto que você quer tornar interativo, e clique com o botão direito nele. No menu de contexto, selecione o item **Hyperlink....** A janela de edição do *hyperlink* abrirá:



Escolha o tipo de *hyperlink* selecionando a guia na parte esquerda da janela. Depois de você ter feito isso, você deve clicar na checkbox **Modificar a aparência do objeto...** na parte inferior da janela. A aparência do objeto mudará do seguinte modo:

- A cor azul será definida para o texto e ele será sublinhado;
- Um cursor em forma de mão será definido.

Em alguns casos o *hyperlink* precisa ser mostrado na janela **pré-visualização**, mas não há necessidade de imprimi-lo. Isso é fácil de fazer, se você desabilitar a propriedade **Printable** do objeto. Isso pode ser feito na janela Propriedades.

### Link para a URL

Usando este tipo de link, você pode:

- Navegar ao endereço de internet dado;
- Executar algum tipo e comandos de sistema, por exemplo, **mailto:** para enviar um e-mail;

---

Quando clicamos em um link deste tipo, o método **System.Diagnostics.Process.Start** é executado com o valor do link como parâmetro.

---

Você pode indicar o valor do link usando dois métodos:

- Indicar o número diretamente, por exemplo, 1;
- Indicar a expressão, a qual retorna o número página. Esta expressão será calculada quando você gerar o relatório.

## Link para uma marcação

Usando este tipo de link, você pode navegar até uma marcação, definido em outro objeto do relatório.

Para aqueles que conhecem HTML, é suficiente dizer que o marcador funciona como uma **âncora**. O marcação tem um nome e uma posição definida no relatório (número da página e posição na página). Quando você se direciona à uma marcação pelo seu nome, você navega até a posição indicada.

Para usar este tipo de link, você precisa primeiro definir o marcador. Para fazer isso, selecione o objeto, onde você quer mover quando clica no link. Quase todos os objetos do relatório têm a propriedade **Marcação**. A mudança desta propriedade pode ser feita com a ajuda da janela Propriedades.

A Marcação contém uma expressão, que pode ser usada da seguinte maneira:

- Indicar o nome do marcador como uma *string*:

*MyBookmark*

- Indicar uma expressão, que retorne o nome do marcador. Por exemplo, você pode usar a coluna de dados como uma expressão. O valor da expressão será calculado quando o relatório for gerado.

Depois que o marcador tiver sido definido, você pode indicar o nome dele na janela de configurações do *hyperlink*. Isso pode ser feito utilizando dois métodos:

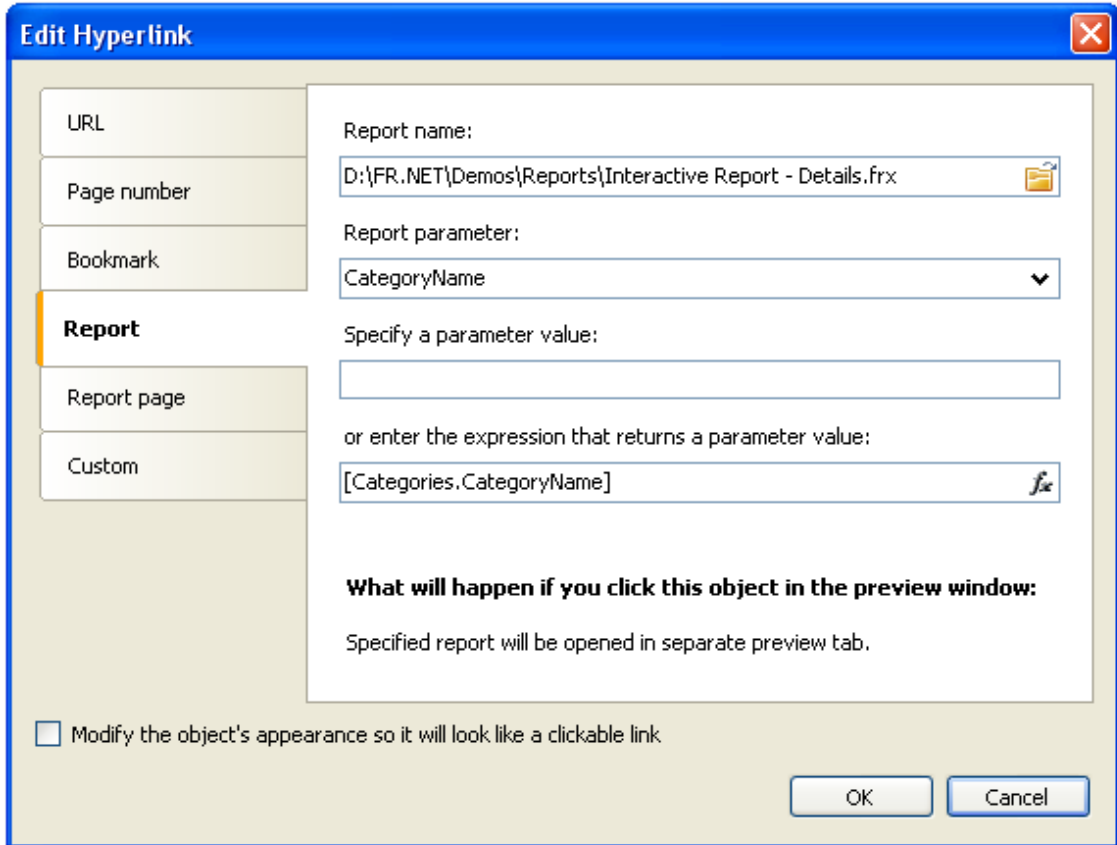
- Indicar o nome do marcador diretamente;
- Indicar uma expressão que retorne o nome do marcador. Por exemplo, isto pode ser uma coluna de dados. Esta expressão será calculada quando o relatório for gerado.

## Link para um relatório detalhado

Usando este tipo de link, você pode executar outro relatório e mostrá-lo em uma janela de **pré-visualização** separada.

Você deve definir os seguintes parâmetros para este tipo de *hyperlink*:

- Nome do relatório detalhado;
- Nome do parâmetro do relatório, que pegará o valor do *hyperlink*;
- O valor do *hyperlink*.



Quando o link é clicado, o seguinte acontecerá:

- O relatório indicado será carregado;
- O parâmetro do relatório será definido para o valor do *hyperlink*;
- O relatório irá ser montado e gerado em uma janela **pré-visualização** separada.

O valor do parâmetro do relatório pode ser indicado usando dois métodos:

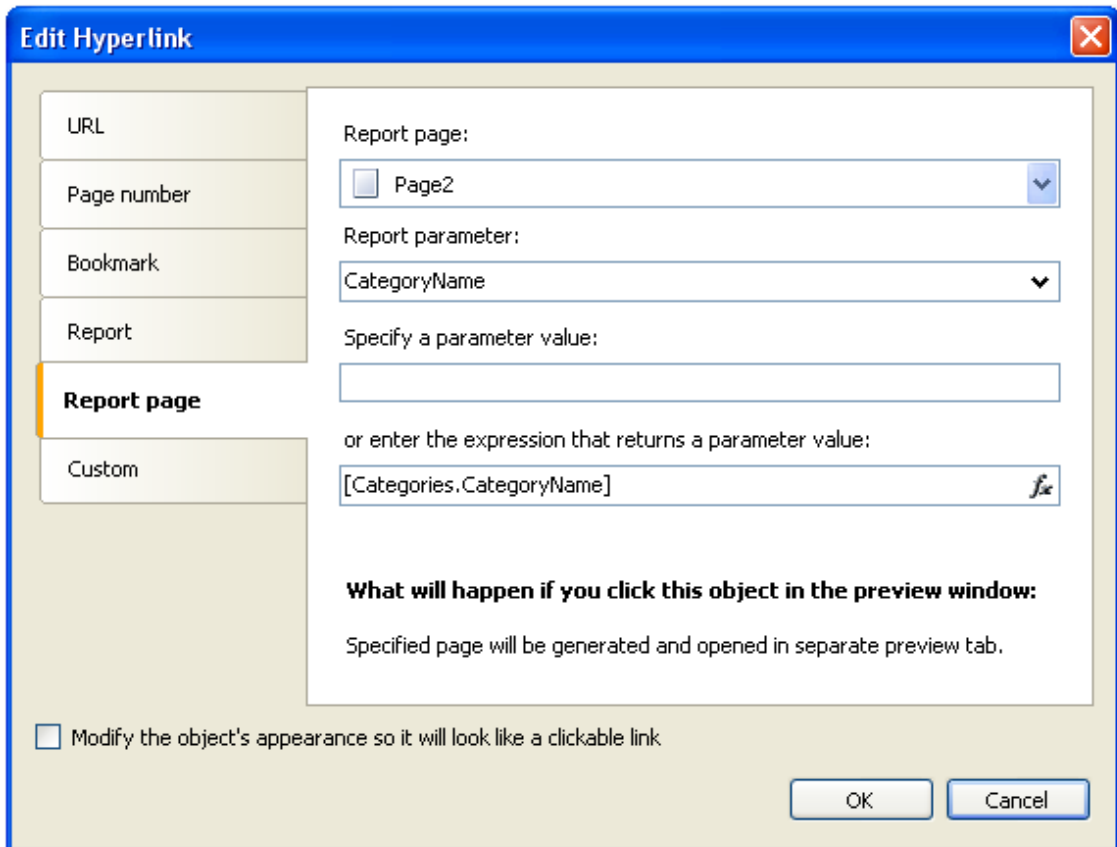
- Indicar o valor diretamente;
- Indicar uma expressão, que retorne o valor. Esta expressão será calculada quando o relatório for gerado.

### Link para uma página detalhada

Este tipo de link funciona da mesma maneira, exceto quando outra página do relatório atual é usada como um relatório detalhado. Para isso, seu relatório deve conter ao menos duas páginas: uma com o relatório principal e outra com o detalhado.

Você deve definir para este tipo de *hyperlink* os seguintes parâmetros:

- O nome da página naquele relatório;
- Nome do parâmetro do relatório, que terá o valor do *hyperlink*;
- O valor do *hyperlink*.



Quando o link é clicado, o seguinte acontecerá:

- O parâmetro do relatório será definido para o valor do *hyperlink*;
- A página indicada do relatório será montada e mostrada em uma página **pré-visualização** separada.

O valor do parâmetro do relatório pode ser indicado usando dois métodos:

- Indicar o valor diretamente;
- Indicar uma expressão, que retorne o valor. Esta expressão será calculada quando o relatório for gerado.

---

Quando você escolhe uma página do relatório, a propriedade **Visible** dele fica **false**. Isto significa que, quando o relatório principal for montado, esta página será pulada.

---





## Link personalizado

Usando este tipo de link você pode definir e personalizar a ação dele quando for clicado. Para isso, use o manipulador de eventos **Click** do objeto. Para fazer isso:

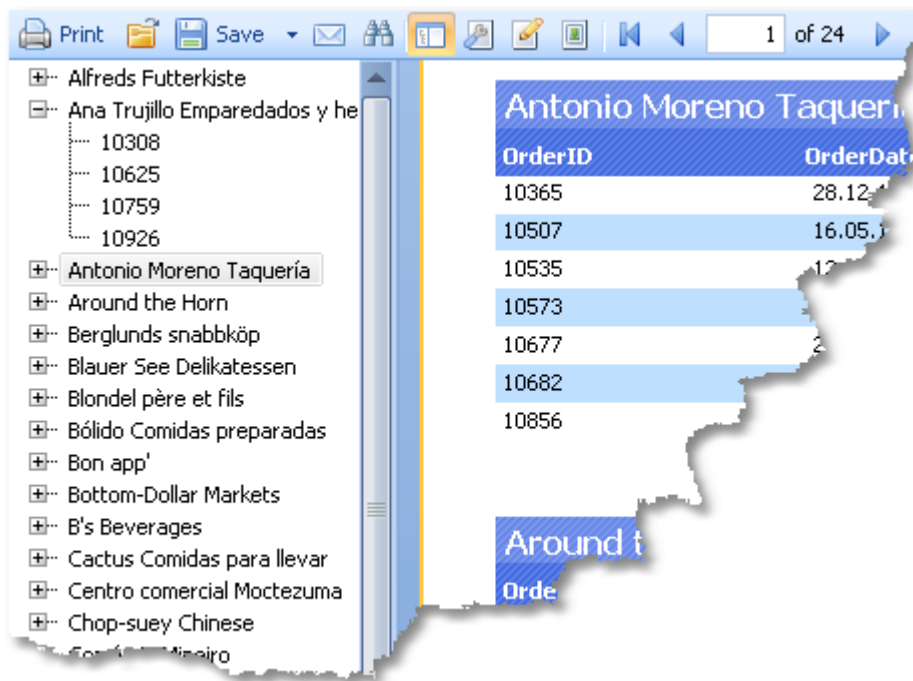
- Selecione o objeto e abra a janela Propriedades;
- Clique no botão ⚡ para mostrar os eventos do objeto;
- Dê um duplo no evento **Click**. O **CIGAM Report** troca a janela **Codee** cria um manipulador de eventos vazio.

No código do manipulador, faça tudo que você precisar. Você, geralmente, precisará criar um link para o objeto, o qual o manipulador chama, e um valor do *hyperlink*. Use o parâmetro **sender** do manipulador:


```
private void Text2_Click(object sender, EventArgs e)
{
    // sender - Este é o objeto que foi clicado.
    // Para receber o valor do hyperlink, você deve
    // converter o "sender" para o tipo ReportComponentBase.
    object hyperlinkValue = (sender as ReportComponentBase).Hyperlink.Value;
    MessageBox.Show("Hyperlink value = " + hyperlinkValue.ToString());
}
```

## Estrutura do relatório

A estrutura do relatório (também conhecida como mapa do documento) é um controle **TreeView** mostrado na janela **pré-visualização**:



este controle mostra uma estrutura em árvore, que foi formada durante a construção do relatório. Se você clicar no elemento da árvore, você navegará ao elemento correspondente do relatório.

Se o relatório tem uma estrutura ela irá ser mostrada automaticamente. Você pode mostrar ou esconder esta estrutura clicando no botão  na barra de ferramentas. O relatório não cria uma estrutura automaticamente- você deve cuidar com isso.

A página do relatório e todas as suas bandas tem a propriedade **OutlineExpression**. Para preencher a estrutura, indique uma expressão que retorne um elemento texto nesta propriedade. Esta expressão será calculada quando a banda for impressa, e este valor será adicionado à estrutura. Se o relatório é do tipo **Mestre-Detalhe** ou grupo, a estrutura será similar à estrutura do relatório.

A propriedade **OutlineExpression** pode ser definida na janela Propriedades.

Aqui estão as recomendações de como configurar a estrutura para diferentes tipos de relatório:

- Se você quer mostrar as folhas do relatório preparado na estrutura, defina a propriedade **OutlineExpression** da página do relatório. A expressão retornará o número da página:

[PageN]

- No relatório do tipo **Lista Simples** com uma banda Dados, defina a propriedade **OutlineExpression** da banda. Como uma expressão, use qualquer coluna de dados que é impressa na banda;
- No relatório do tipo **Mestre-Detalhe** com duas bandas Dados, defina a propriedade **OutlineExpression** das bandas correspondentes. Por exemplo, no relatório do tipo

**Category/Product**, o **OutlineExpression** para a primeira banda conterà o nome da categoria, para a segunda – o nome do produto;

- No relatório de grupos, configure a propriedade **OutlineExpression** do cabeçalho do grupo e a banda Dados. Como uma expressão para o cabeçalho do grupo, use a condição de agrupamento. Para a banda Dados, use qualquer coluna de dados que seja impressa na banda.

## Exemplos

### Exemplo 1: Construindo um relatório detalhado

Neste exemplo nós vamos construir um relatório que mostre uma lista de categorias. Quando clicamos no nome da categoria, um relatório detalhado que contém a lista de produtos da categoria escolhida será mostrado.

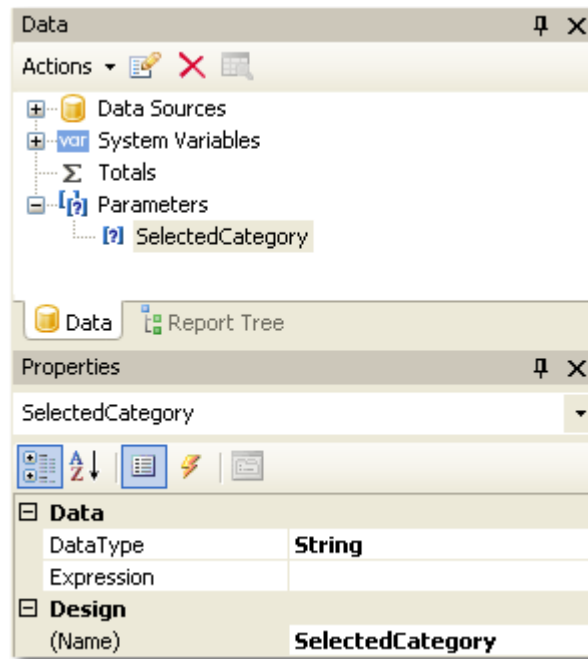
Você precisa fazer o seguinte:

- Primeiro crie um relatório detalhado;
- Defina o parâmetro do relatório que identifica uma categoria;
- Defina a filtragem de dados neste parâmetro;
- Crie o relatório principal;
- No relatório principal, configure o *hyperlink* de um modo que, o relatório detalhado possa ser gerado com o parâmetro definido para a categoria escolhida.

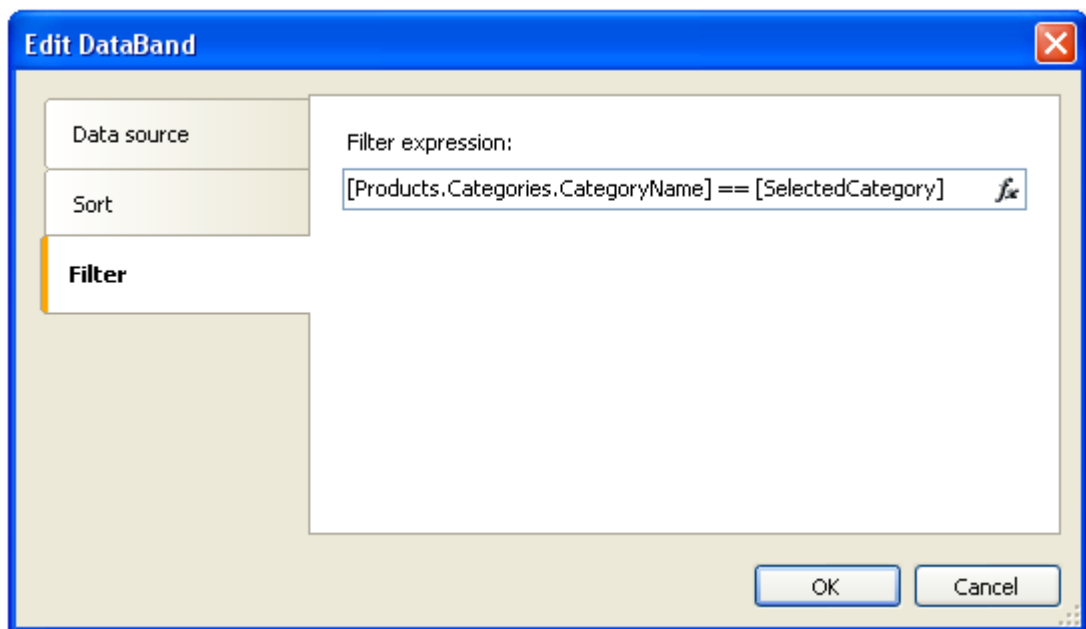
Primeiramente, criaremos um relatório detalhado que imprime uma lista de produtos. Para isso, crie um novo relatório e escolha a tabela **Produtos** como fonte de dados. Coloque os objetos da seguinte maneira:

| Report Title   | -   |
|----------------|---|
| Page Header    | <b>Product Name</b> <b>Quantity Per Unit</b> <b>UnitPrice</b>                           |
| Data: Products | [Products.ProductName]      [Products.QuantityPe<br>Unit]      [Products.UnitPric<br>e] |

Crie o parâmetro que será usado para passar a categoria selecionada do relatório principal para o detalhado. Para identificação da categoria, nós vamos usar a coluna **CategoryID** que está contida em ambas tabelas **Categorias** e **Produtos**. Configure o parâmetro da seguinte forma:



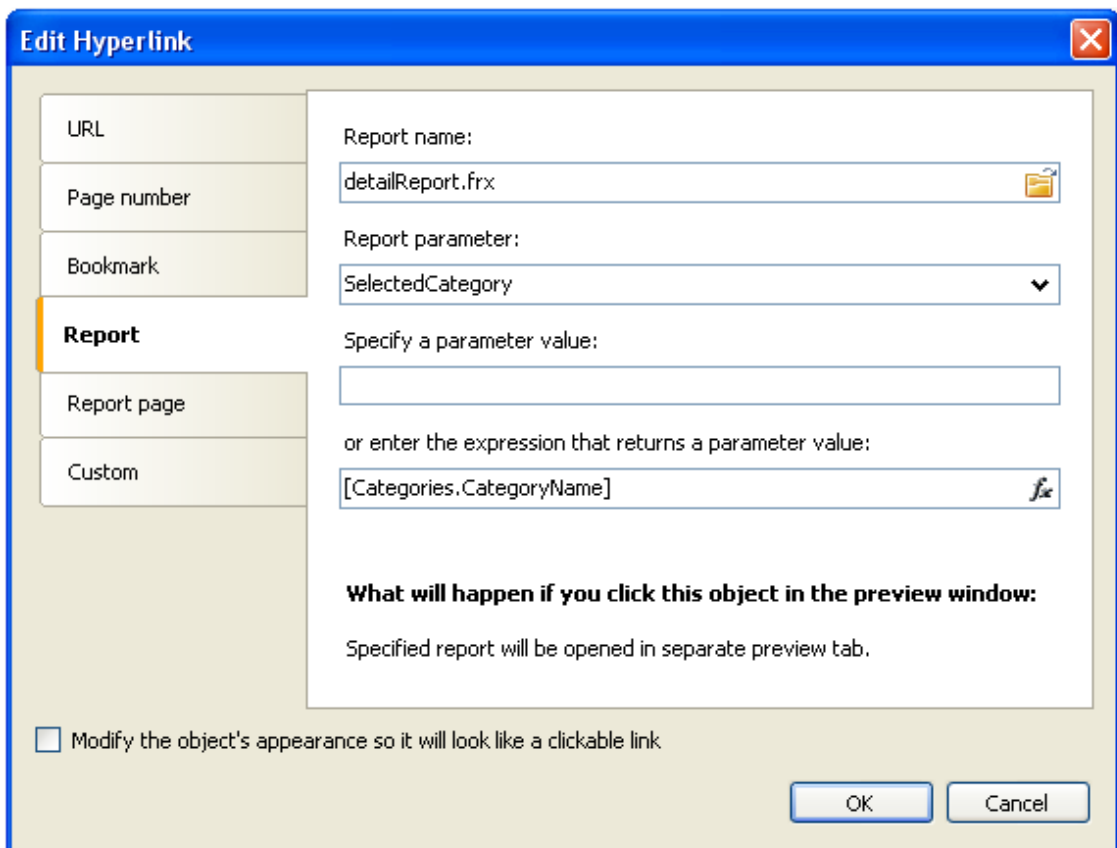
Agora nós precisamos definir o filtro de dados para filtrar todos os produtos que pertencem a uma categoria específica. Para fazer isso, dê um duplo clique na banda Dados. Vá para a aba **Filter** e indique a seguinte condição:



Agora criar o relatório principal. Crie um novo relatório e escolha a tabela **Categorias** como fonte de dados. Coloque os objetos da seguinte maneira:

|                  |   |                           |
|------------------|---|---------------------------|
| Report Title     | - |                           |
| Data: Categories | - | [Categories.CategoryName] |

Clique com o botão direito no objeto Texto e selecione o item **Hyperlink..** do menu. Defina o link da seguinte maneira:



**Edit Hyperlink**

URL

Page number

Bookmark

**Report**

Report page

Custom

Report name: detailReport.frx

Report parameter: SelectedCategory

Specify a parameter value:

or enter the expression that returns a parameter value: [Categories.CategoryName]

**What will happen if you click this object in the preview window:**

Specified report will be opened in separate preview tab.

Modify the object's appearance so it will look like a clickable link

OK Cancel

Como nome do relatório escolha o nome do arquivos do relatório detalhado. O parâmetro do relatório pode ser escolhido do menu **drop-down**, pressionando o botão no lado direito da lista. Como valor do parâmetro, indique a expressão **[Categorias.CategoryID]**.

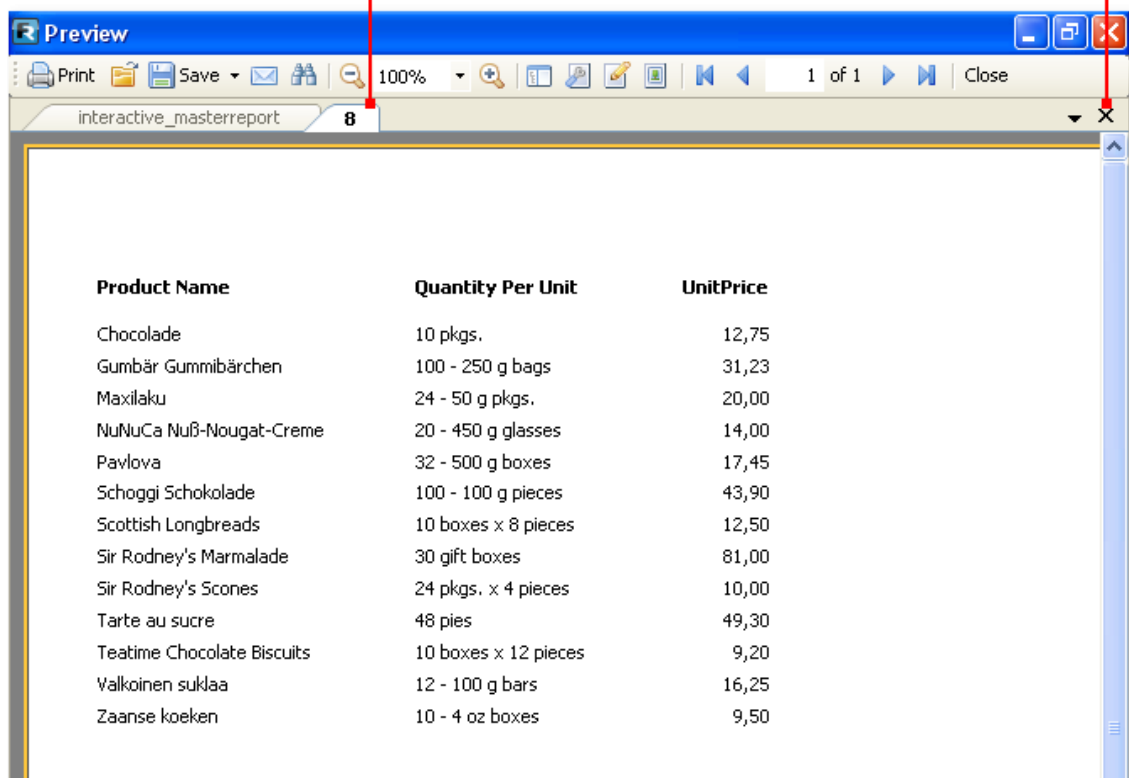
Gere o relatório e você verá a lista de categorias:

[Beverages](#)[Condiments](#)[Confections](#)[Dairy Products](#)[Grains/Cereals](#)[Meat/Poultry](#)[Produce](#)[Seafood](#)

Se você clicar em uma destas categorias, um relatório detalhado será construído. Ele será mostrado em uma guia/aba separada da janela **pré-visualização**.

Aba do relatório detalhado

Fechar relatório detalhado



| Product Name               | Quantity Per Unit    | UnitPrice |
|----------------------------|----------------------|-----------|
| Chocolade                  | 10 pkgs.             | 12,75     |
| Gumbär Gummibärchen        | 100 - 250 g bags     | 31,23     |
| Maxilaku                   | 24 - 50 g pkgs.      | 20,00     |
| NuNuCa Nuß-Nougat-Creme    | 20 - 450 g glasses   | 14,00     |
| Pavlova                    | 32 - 500 g boxes     | 17,45     |
| Schoggi Schokolade         | 100 - 100 g pieces   | 43,90     |
| Scottish Longbreads        | 10 boxes x 8 pieces  | 12,50     |
| Sir Rodney's Marmalade     | 30 gift boxes        | 81,00     |
| Sir Rodney's Scones        | 24 pkgs. x 4 pieces  | 10,00     |
| Tarte au sucre             | 48 pies              | 49,30     |
| Teatime Chocolate Biscuits | 10 boxes x 12 pieces | 9,20      |
| Valkoinen suklaa           | 12 - 100 g bars      | 16,25     |
| Zaanse koeken              | 10 - 4 oz boxes      | 9,50      |



Como visto na imagem, o título da tabela é definido pelo valor do *hyperlink*. No nosso caso, ele é um valor número contido na coluna de dados **CategoryID**. Isso não aparenta ser informativo nem bonito. Vamos mudar para que o nosso relatório use o nome da categoria ao invés do seu número. Para isso, faça o seguinte:

No relatório detalhado:

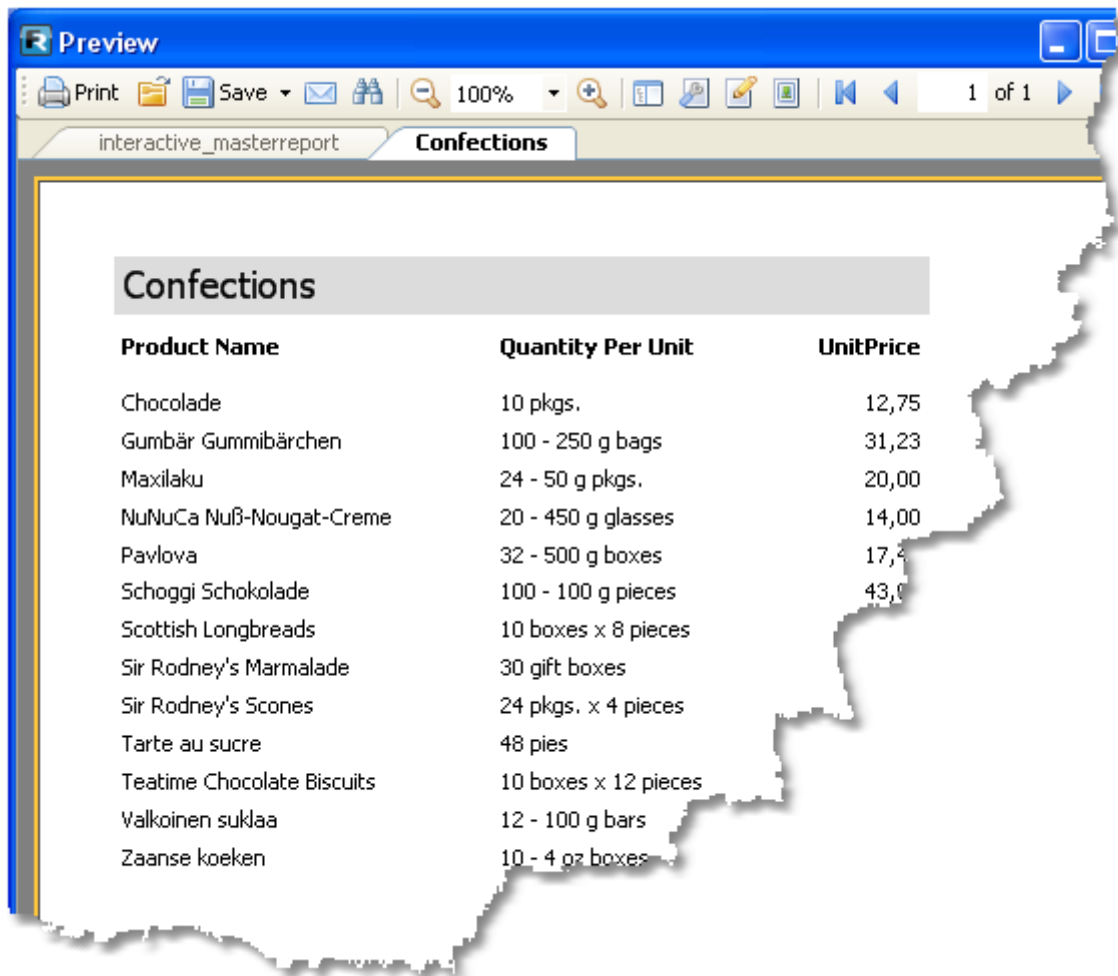
- Mude a propriedade do parâmetro **DataType** para **String**;
- Adicione a fonte de dados **Categorias** ao relatório. Ela será usada para se referir à coluna **NomeCategoria** quando filtrarmos os dados;
- Mude a expressão de filtragem da banda Dados:

**[Produtos.Categorias.NomeCategoria] == [SelectedCategory]**

No relatório principal:

- Mude as configurações do *hyperlink*. Agora nós vamos passar o valor **[Categorias.NomeCategoria]** dentro dos parâmetros do relatório.

Se nós gerarmos o relatório agora, veremos que o título daquela guia é o nome da categoria. Nós podemos melhorar um pouco o relatório detalhado. Adicione o objeto Texto, que imprimirá no título do relatório o nome da categoria escolhida:

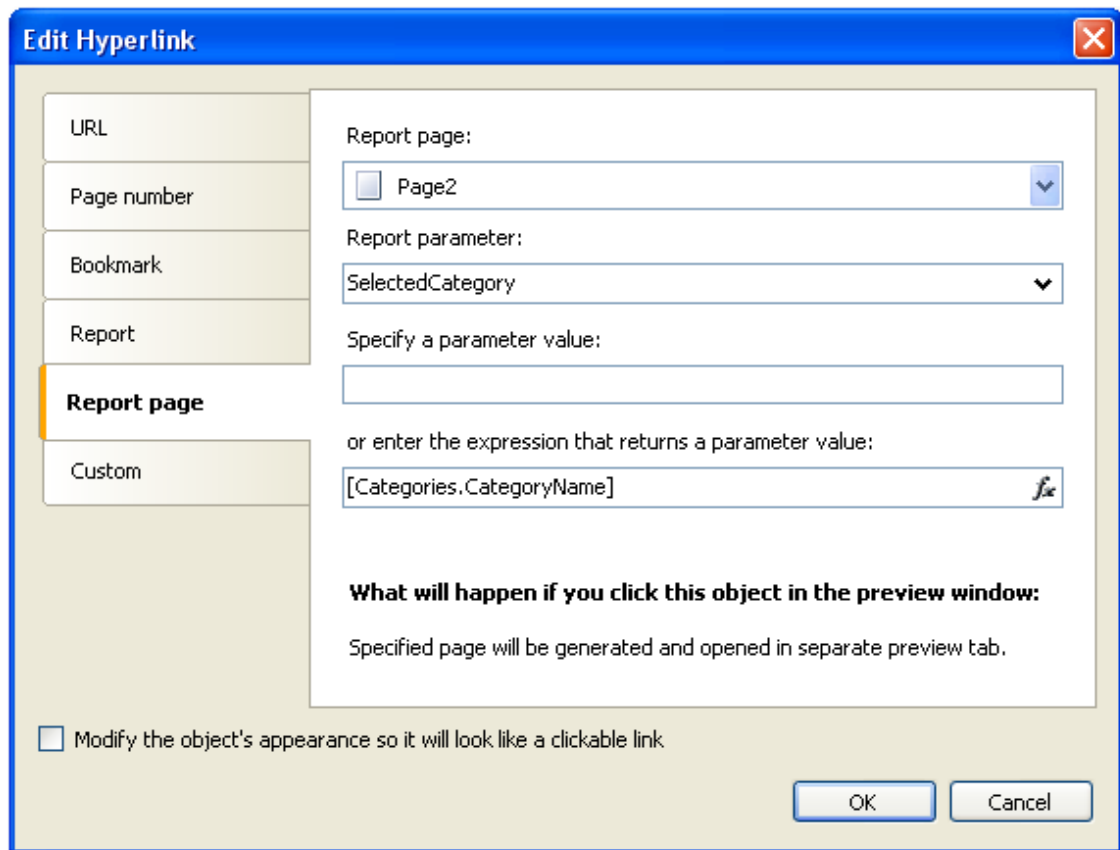


The screenshot shows a software interface with a blue title bar labeled 'Preview'. Below the title bar is a toolbar with icons for Print, Save, and other functions. The main content area displays a report titled 'Confections' with a table of product data.

| Product Name               | Quantity Per Unit    | UnitPrice |
|----------------------------|----------------------|-----------|
| Chocolade                  | 10 pkgs.             | 12,75     |
| Gumbär Gummibärchen        | 100 - 250 g bags     | 31,23     |
| Maxilaku                   | 24 - 50 g pkgs.      | 20,00     |
| NuNuCa Nuß-Nougat-Creme    | 20 - 450 g glasses   | 14,00     |
| Pavlova                    | 32 - 500 g boxes     | 17,4      |
| Schoggi Schokolade         | 100 - 100 g pieces   | 43,0      |
| Scottish Longbreads        | 10 boxes x 8 pieces  |           |
| Sir Rodney's Marmalade     | 30 gift boxes        |           |
| Sir Rodney's Scones        | 24 pkgs. x 4 pieces  |           |
| Tarte au sucre             | 48 pies              |           |
| Teatime Chocolate Biscuits | 10 boxes x 12 pieces |           |
| Valkoinen suklaa           | 12 - 100 g bars      |           |
| Zaanse koeken              | 10 - 4 oz boxes      |           |

Enquanto nós trabalhamos com este exemplo, nós criamos dois relatórios e trocamos entre eles várias vezes. Isso não é muito confortável. Para tornar esta tarefa mais fácil, dois relatórios podem ser colocados em um: o relatório principal ficará na primeira página e o detalhado na segunda página. Neste caso o *hyperlink* precisa ser definido da seguinte forma:





**Edit Hyperlink**

URL

Page number

Bookmark

Report

**Report page**

Custom

Report page:

Page2

Report parameter:

SelectedCategory

Specify a parameter value:

or enter the expression that returns a parameter value:

[Categories.CategoryName]

**What will happen if you click this object in the preview window:**

Specified page will be generated and opened in separate preview tab.

Modify the object's appearance so it will look like a clickable link

OK Cancel

Neste caso, nós precisamos escolher **Page2** como uma página do relatório detalhado.

### *Exemplo 2: Objeto Matriz interativo*

Neste exemplo nós vamos ver como criar um relatório detalhado se nós clicarmos na célula do objeto **Matriz**. Como exemplo, nós utilizaremos a matriz que mostra as vendas dos empregados, agrupadas por ano.

Como fonte de dados da matriz, a tabela **MatrixDemo** é usada. Ela contém as vendas dos empregados, agrupadas por ano e mês:



| Name            | Year | Month | ItemsSold | Revenue |
|-----------------|------|-------|-----------|---------|
| Nancy Davolio   | 1999 | 2     | 1         | 1000    |
| Nancy Davolio   | 1999 | 11    | 1         | 1100    |
| Nancy Davolio   | 1999 | 12    | 1         | 1200    |
| Nancy Davolio   | 2000 | 1     | 1         | 1300    |
| Nancy Davolio   | 2000 | 2     | 2         | 1400    |
| Nancy Davolio   | 2001 | 2     | 2         | 1500    |
| Nancy Davolio   | 2001 | 3     | 2         | 1600    |
| Nancy Davolio   | 2002 | 1     | 2         | 1700    |
| Andrew Fuller   | 2002 | 1     | 2         | 1800    |
| Andrew Fuller   | 1999 | 10    | 2         | 1900    |
| Andrew Fuller   | 1999 | 11    | 2         | 2000    |
| Andrew Fuller   | 2000 | 2     | 2         | 2100    |
| Janet Leverling | 1999 | 10    | 3         | 3000    |
| Janet Leverling | 1999 | 11    | 3         | 3100    |
| Janet Leverling | 2000 | 3     | 3         | 3200    |
| Steven Buchanan | 2001 | 1     | 3         | 4000    |
| Steven Buchanan | 2001 | 2     | 4         | 4100    |
| Steven Buchanan | 2000 | 1     | 4         | 3999    |

Configure a matriz da seguinte maneira:

- Coloque a coluna de dados **MatrixDemo.Year** no cabeçalho da coluna;
- Coloque a coluna de dados **MatrixDemo.Name** no cabeçalho da linha;
- Coloque a coluna de dados **MatrixDemo.Revenue** na célula.

A matriz pronta será como segue:

| Employee        | Year      |           |           |          | Total     |
|-----------------|-----------|-----------|-----------|----------|-----------|
|                 | 1999      | 2000      | 2001      | 2002     |           |
| Andrew Fuller   | 3 900,00  | 2 100,00  |           | 1 800,00 | 7 800,00  |
| Janet Leverling | 6 100,00  | 3 200,00  |           |          | 9 300,00  |
| Nancy Davolio   | 3 300,00  | 2 700,00  | 3 100,00  | 1 700,00 | 10 800,00 |
| Steven Buchanan |           | 3 999,00  | 8 100,00  |          | 12 099,00 |
| <b>Total</b>    | 13 300,00 | 11 999,00 | 11 200,00 | 3 500,00 | 39 999,00 |

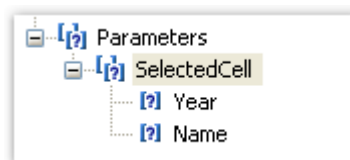
Como visto, o valor da célula é a soma das vendas do empregado do ano inteiro. Vamos criar um relatório detalhado que será mostrado quando nós clicarmos na célula. No nosso caso o relatório detalhado contém as vendas do empregado selecionado para cada mês do ano selecionado.

Como conectar a célula com dados, com base no que foi impresso? Cada célula da matriz tem seu próprio endereço. Isso é uma combinação de valores dos cabeçalhos das linhas e colunas. No nosso exemplo, o endereço da célula é uma combinação do ano e do nome do empregado. Exatamente estes dados podem ser passados para o relatório detalhado. Como isso pode ser feito? Muito simples: defina que o *hyperlink* deve mostrar somente o nome do relatório e o nome do parâmetro. Os valores do parâmetro não precisam ser indicados: para uma célula da matriz, o **CIGAM Report** forma o valor e o coloca dentro do parâmetro sozinho.

Depois disso, nós clicamos na parte superior esquerda da célula, contendo o número 3900. Isso é a soma das vendas do empregado chamado **Andrew Fuller** para o ano de 1999. Que forma é usada para passar este valor dentro do parâmetro? O **CIGAM Report** combina os valores das colunas e linhas, usando o separador:

### 1999;Andrew Fuller

Isso significa que nós devemos pegar o valor do ano e o nome do empregado desta *string*, converter o ano para inteiro e usar Estes valores para filtrar os dados? Não, é muito mais simples. Tudo que nós precisamos fazer é criar um parâmetro que tenha **nested parameters**. Você pode aprender sobre isso no capítulo **Data**. Neste caso, o parâmetro pai pode ser assim:



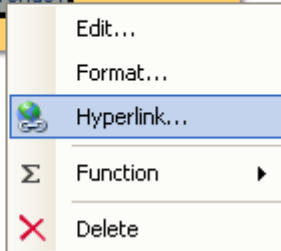
Quando criar o parâmetro, considere os seguintes momentos:

- Você não precisa definir o parâmetro pai. Só dê o nome dele;
- Um parâmetro pai deve ter parâmetros aninhados enquanto houver valores vindos da matriz. Neste caso, há dois valores;

- A ordem dos parâmetros aninhados deve corresponder à ordem dos valores passados da matriz. No nosso caso, o ano será passado no primeiro parâmetro, e o nome do empregado será passado no segundo;
- Parâmetros aninhados podem ser nomeados como você quiser, mas é melhor dar nomes que correspondam com os nomes dos elementos da matriz;
- É muito importante definir o tipo de dados corretamente para cada parâmetro aninhado. O tipo de dados deve corresponder com o valor, o qual é passado dentro do parâmetro. No nosso caso, o primeiro parâmetro (Ano) deve ser do tipo inteiro, e o segundo (nome do empregado) – *string*.

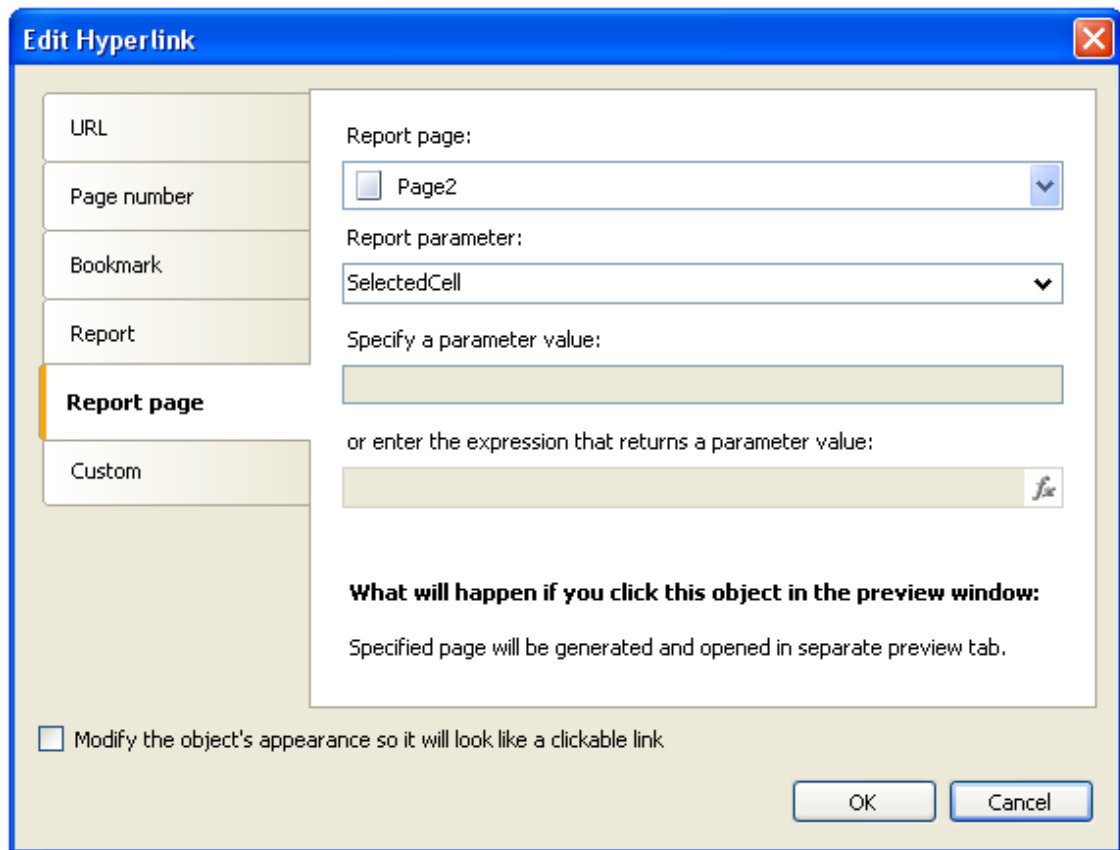
Depois de nós termos esclarecido todas as coisas necessárias, nós vamos criar o relatório. Selecione a célula da matriz e abra o editor de *hyperlink*:

|          | Year      |       |
|----------|-----------|-------|
| Employee | [Year]    | Total |
| [Name]   | [Revenue] |       |
| Total    |           |       |



The image shows a context menu for the selected cell in the matrix. The menu options are: Edit..., Format..., Hyperlink..., Function, and Delete. The 'Hyperlink...' option is highlighted.

Na configuração do *hyperlink*, indique o parâmetro pai como um parâmetro de relatório (no nosso exemplo **SelectedCell**):



O **CIGAM Report** passa os valores dentro dos parâmetros aninhados **SelectedCell.Year** e **SelectedCell.Name**. Estes valores serão convertidos para os tipos de dados, indicados na configuração do parâmetro – é por isso que é importante configurar corretamente os tipos de dados.

O relatório detalhado é colocado em uma página separada do relatório e usa a mesma fonte de dados:

|                  |   |                 |
|------------------|---|-----------------|
| Report Title     | [SelectedCell.Name] sales in [SelectedCell.Year] year |                 |
| Page Header      | Month   | Revenue         |
| Data: MatrixDemo | [MatrixDemo.Mon                                       | [MatrixDemo.Rev |

Para mostrar as vendas do empregado escolhido para um ano escolhido, defina o filtro. Para isso, abra o editor da banda Dados e indique a seguinte condição de filtragem:

**[MatrixDemo.Year] == [SelectedCcell.Year]&&[MatrixDemo.Name] == [SelectedCell.Name]**

O relatório está pronto. Execute-o e clique na parte superior esquerda da célula. Um relatório detalhado será aberto, tendo os seguinte dados:

### Andrew Fuller sales in 1999 year

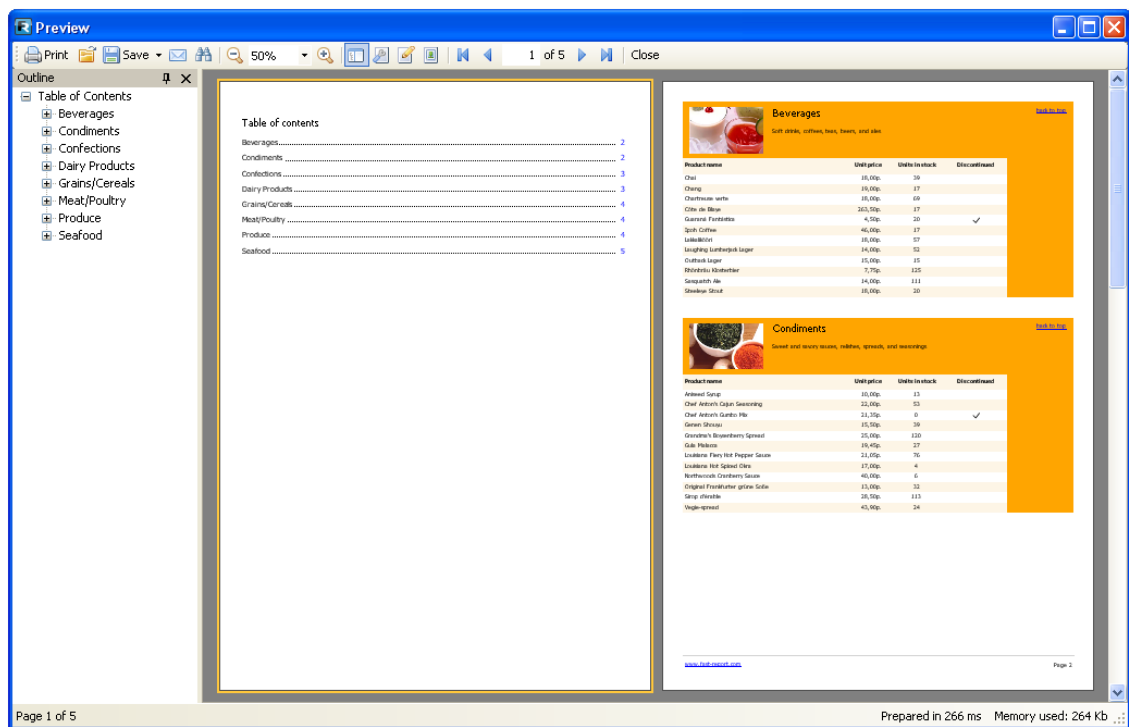
| Month | Revenue  |
|-------|----------|
| 10    | 1 900,00 |
| 11    | 2 000,00 |

Como visto, a soma dos valores (1900+2000) corresponde com a célula da matriz, na qual nós clicamos.

#### Exemplo 3: Relatório com tabelas de conteúdo, navegação e estrutura.

Neste exemplo, iremos olhar a criação de um relatório, que tem as seguintes características:

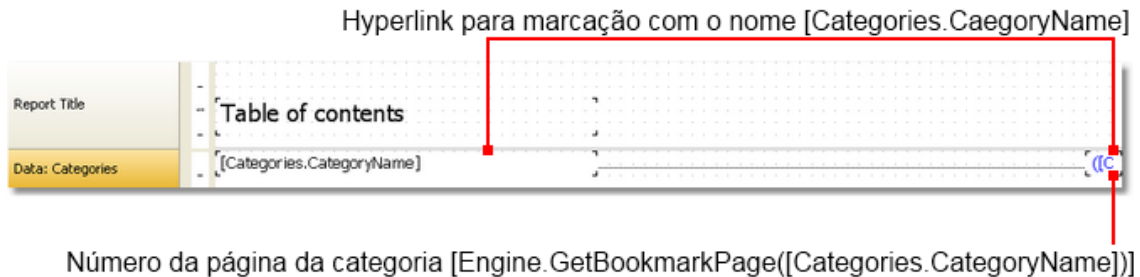
- Na primeira página iremos imprimir o **Sumário**, que será interativo (você pode clicar nos seus elementos para navegar para as páginas correspondentes);
- Na janela **pré-visualização**, ele irá mostrar a estrutura, a qual será interativa também.



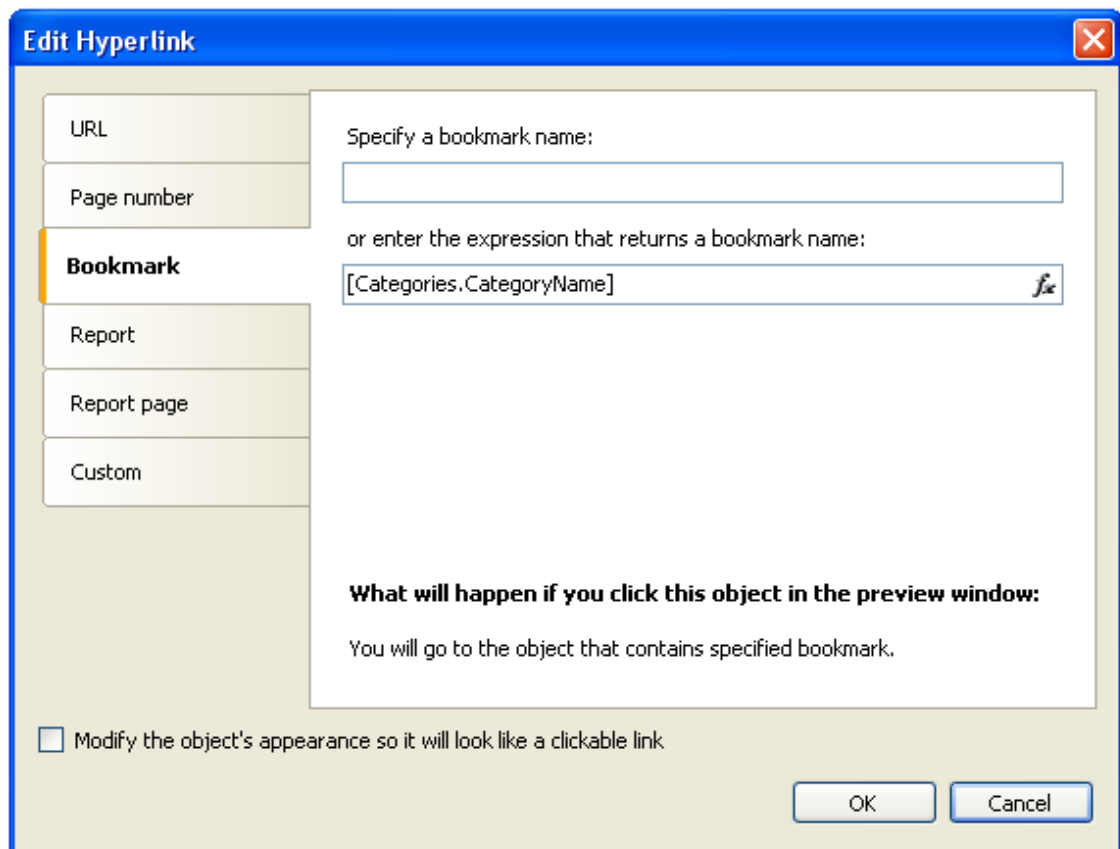
The screenshot shows a software interface with a 'Preview' window. On the left, there is an 'Outline' pane with a 'Table of Contents' section listing categories like Beverages, Condiments, Confections, Dairy Products, Grains/Cereals, Meat/Poultry, Produce, and Seafood. The main area is divided into two sections: 'Beverages' and 'Condiments'. Each section contains a table with columns for 'Product name', 'Unit price', 'Units in stock', and 'Discontinued'. The 'Beverages' table lists items like Cola, Orange, and Coffee. The 'Condiments' table lists items like Ancho Reyes, Chutney, and Sauces. The interface includes a toolbar at the top with options like Print, Save, and a page indicator '1 of 5'.

O relatório usará as tabelas **Categorias e Produtos**. No **TOC**, nós iremos imprimir a lista de categorias. O restante do relatório imprimirá a lista de produtos categorizada. O **template** do relatório será formado por duas páginas: a primeira página imprimirá o **TOC**; e a segunda será a parte principal do relatório.

Nós vamos falar sobre o **TOC** primeiro. Crie um novo relatório e adicione as fontes de dados **Categorias** e **Produtos** nele. Conecte a banda Dados à tabela **Categorias** e coloque os objetos da seguinte maneira:



Para tornar os objetos do **TOC** interativos, configure a propriedade *hyperlink* dele:



Indique o nome da categoria como uma marcação. Nós iremos definir a marcação do objeto mais tarde.

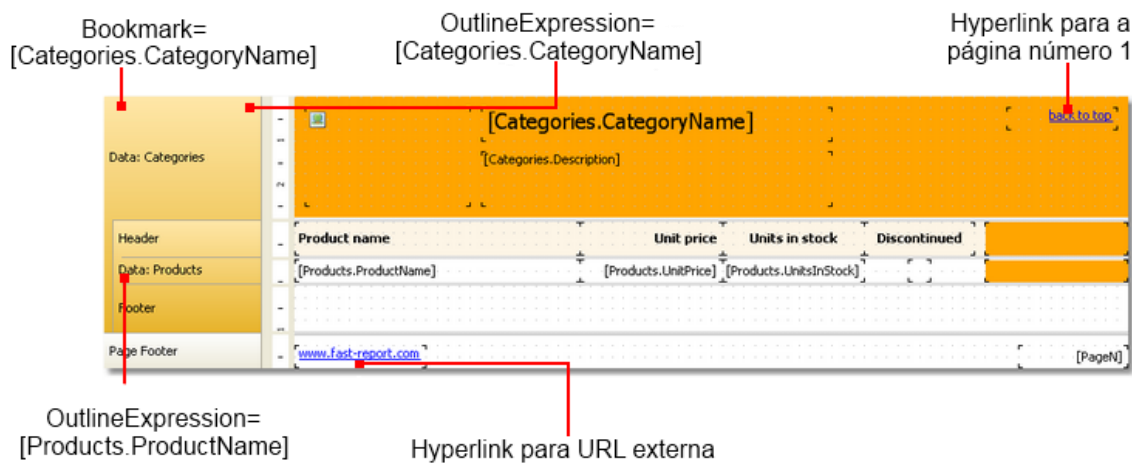
Para imprimir o número da página no **TOC**, você precisa fazer o seguinte:



- Habilitar a configuração **Passo Duplo** do relatório. Isto pode ser feito no menu **Relatório > Opções....** É necessário fazer isso porque a página do **TOC** é impressa antes das outras páginas. Neste momento o **CIGAM Report** não sabe quando as categorias serão impressas;
- Use a função **Engine.GetBookmarkPage**, que retorna o número da página para o marcação especificado. No nosso caso, use a expressão **[Categorias.NomeCategoria]** como um nome de marcação, então a chamada da função será o seguinte:

**[Engine.GetBookmarkPage([Categorias.NomeCategoria])]**

Na segunda página do relatório, nós criaremos um relatório **Mestre-Detalhe** como mostrado na figura abaixo:



Defina o marcação para o qual navegaremos quando clicarmos em um elemento do **TOC**. Para isso, selecione a primeira banda Dados e na propriedade **Marcação** indique a seguinte expressão :

**[Categorias.NomeCategoria]**

Para definir a estrutura de um relatório, faça o seguinte:

- Selecione a primeira página do relatório. Isto pode ser feito indo para a página;
- Na janela Propriedades, defina o seguinte valor para a propriedade **OutlineExpression**:

**Contents**

- Vá para a segunda página do relatório;
- Selecione a primeira banda Dados e defina a sua propriedade **OutlineExpression**

**[Categorias.NomeCategoria]**

- Selecione a segunda **Data** e defina a sua propriedade **OutlineExpression**:

**[Produtos.NomeProduto]**



## Herança de Relatório

Freqüentemente nós tivemos muitos relatórios com os mesmos dados nele – por exemplo, mesmo cabeçalho/rodapé com o logo da companhia e alguns dados – e-mail, endereço, etc. Agora imagine a situação, que você precise mudar dados da companhia – por exemplo, e-mail. Você terá que fazer isso em cada relatório! Para evitar isso, você pode usar a herança de relatório. Mas, o que é isso:

Por exemplo, se você tiver alguns elementos comuns em cada relatório (logo, nome da companhia, e-mail, etc). Estes elementos são tipicamente colocados no título do relatório e/ou no cabeçalho da página. Você pode criar um relatório base que contém somente elementos comuns. Todos os outros relatórios usarão o relatório base e então irão conter tais elementos comuns mais os seus próprios elementos, definidos em um relatório.

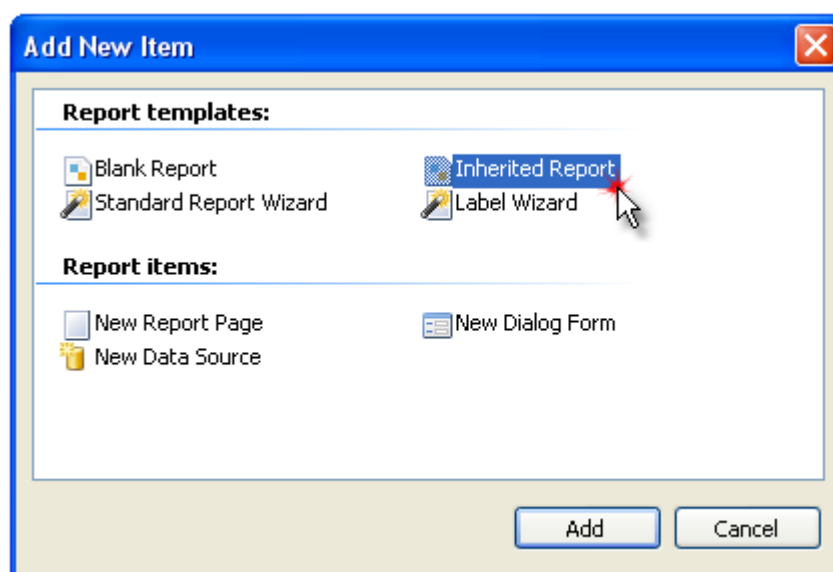
Caso você precisa mudar algo (logo, e-mail) você somente abre o relatório base e faz as alterações necessárias. Todos os outros relatórios que herdam de um base, mudarão automaticamente. No entanto, quando você abre um relatório que é o herdado, o relatório base é aberto primeiro e depois o herdado.

## Criando um relatório

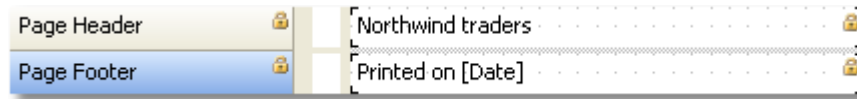
Para usar a herança, você precisa fazer o seguinte:

- Crie um relatório base e salve-o em um arquivo;
- Crie um novo relatório que herda do base;

Para criar um relatório herdado, escolhe o item do menu **Arquivo > Novo....**, então escolha o item **Relatório herdado** na janela:



Você deverá escolher um arquivo de relatório base. Este arquivo deve ser criado neste momento. Depois disso, o relatório base será carregado no **editor**. Você pode mudá-lo como preferir. Você verá que os objetos do relatório base estarão marcados com o sinal **Lock**:



Isto significa que você não pode excluir, renomear ou mover tais objetos para outra banda.

Você pode adicionar novos objetos ou bandas, mudar a aparência, o tamanho e a localização dos objetos. Quando você estiver pronto, salve o relatório.

### Mudando o relatório base

Vamos ver o que acontece se nós mudarmos o relatório base. Com isso, nós poderemos:

- Excluir o objeto do relatório base. Este objeto também será deletado do relatório herdado;
- Adicionar o objeto no relatório base. Este objeto aparecerá automaticamente no relatório herdado;
- Mudar o tamanho, a localização, o texto e a aparência do objeto. Todas as mudanças refletirão no relatório herdado, no caso deste objeto não ter sido mudado no relatório herdado.

O último item requer algumas explicações. Vamos ver dois exemplos de uso da herança. No primeiro exemplo, vamos fazer o seguinte:

- Crie um relatório base que contém o objeto **Text1**;
- Crie um relatório herdado;
- No relatório herdado, mova o objeto **Text1** para uma nova posição e salve o relatório;
- Abra o relatório base e mova o objeto **Text1** para uma posição diferente;
- Abra o relatório herdado e nós veremos que o objeto **Text1 não se moveu**;

Isso acontece porque nós mudamos o objeto no relatório herdado. Esta mudança foi salva no arquivo do relatório herdado. Agora, se nós mudarmos o objeto original no relatório base, ele será ignorado no relatório herdado. Neste caso, a localização do novo objeto será ignorada. Todas as outras alterações (como cor do texto, por exemplo) continuarão refletindo no relatório herdado.

Este comportamento ficará mais claro se nós olharmos para o conteúdo do arquivo do relatório herdado. Por exemplo, é assim que o objeto original é salvo em um relatório herdado, no caso deste objeto não ter mudado:

```
<inherited Name=Texto/>
```

Se nós mudarmos a localização do objeto no relatório herdado, ele será salvo assim:

```
<inherited Name=Text1 Left=... Right=.../>
```



Quando abrirmos o relatório herdado, o **CIGAM Report** carregará todas as propriedades dos objetos, definidas no relatório base, mais propriedades salvas no relatório herdado.

### Limitações

A herança do relatório foi designada para atingir o seguinte o objeto: salvar os elementos comuns do relatório como cabeçalhos e rodapés em arquivos separados, e reutilizar eles em relatórios herdados. Não tente usar a herança para fazer tarefas mais complexas. Evite de fazer o seguinte:

- Não herde o relatório do herdado (por exemplo, não herde duas vezes);
- Não use objetos complexos como **Tabela** e **Matriz** no relatório base;
- Não use *scripts* no relatório base;
- Não use parâmetros no relatório base.

### Relatórios com gráficos

O **CIGAM Report** usa a biblioteca **Microsoft Charts** para mostrar gráficos. Esta biblioteca será incluída no **Framework 4.0**. Por enquanto, ele está disponível como um **download** separado:

[HTTP://www.microsoft.com/downloads/Detalhes.aspx?FamilyID=130f7986-bf49-4fe5-9ca8-910ae6ea442c](http://www.microsoft.com/downloads/Detalhes.aspx?FamilyID=130f7986-bf49-4fe5-9ca8-910ae6ea442c)

Esta biblioteca requer o **.Net Framework 3.5 SP1**. Ele é um **freeware**, e você pode aprender mais sobre as suas funcionalidades aqui:

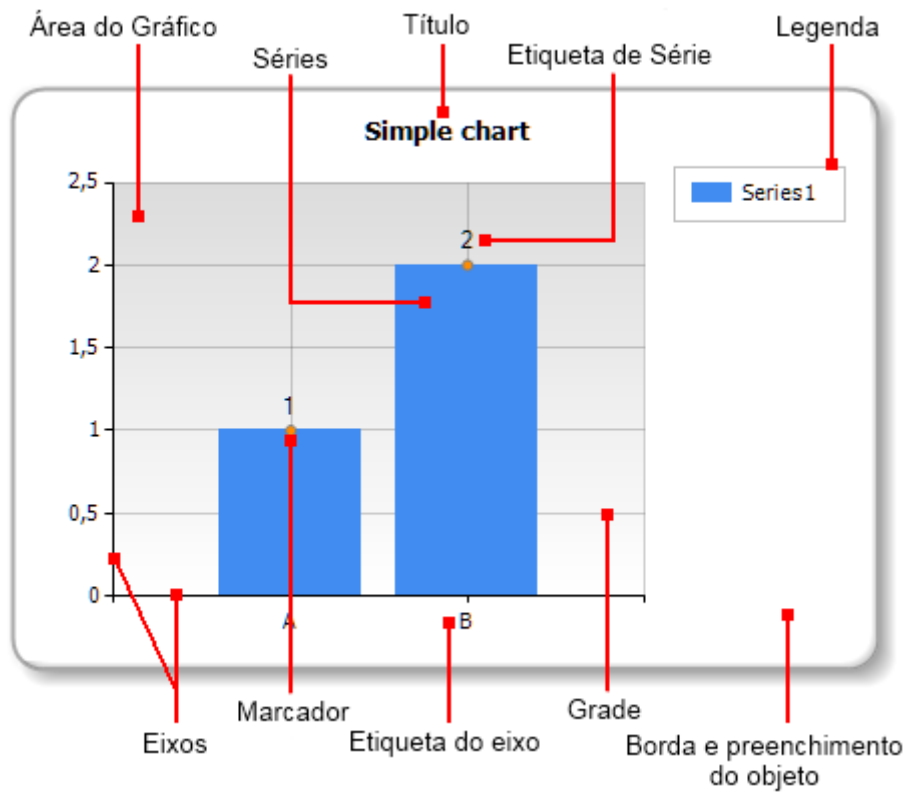
[HTTP://code.msdn.microsoft.com/mschart](http://code.msdn.microsoft.com/mschart)

Vamos destacar algumas características do **Microsoft Chart**:

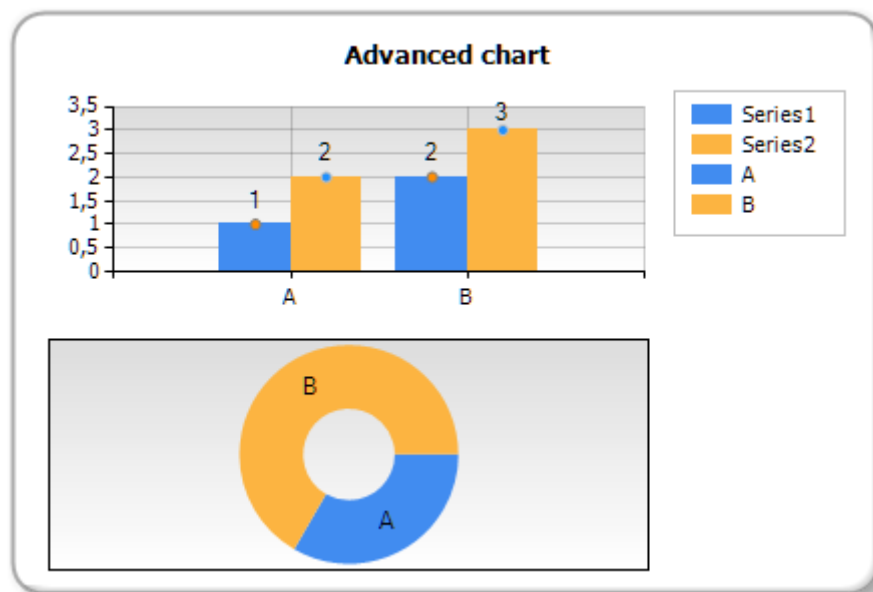
- Mais de 30 tipos de séries (barras, colunas, áreas, linhas, bolhas, torta, circular, financeiro, piramidal, séries);
- Suporte 3D;
- Suporta várias séries de diferentes tipos em um gráfico;
- Controle total sobre aparência e comportamento de cada elemento do gráfico.

### Elementos do gráfico

O **Microsoft Chart** consiste dos seguintes elementos:



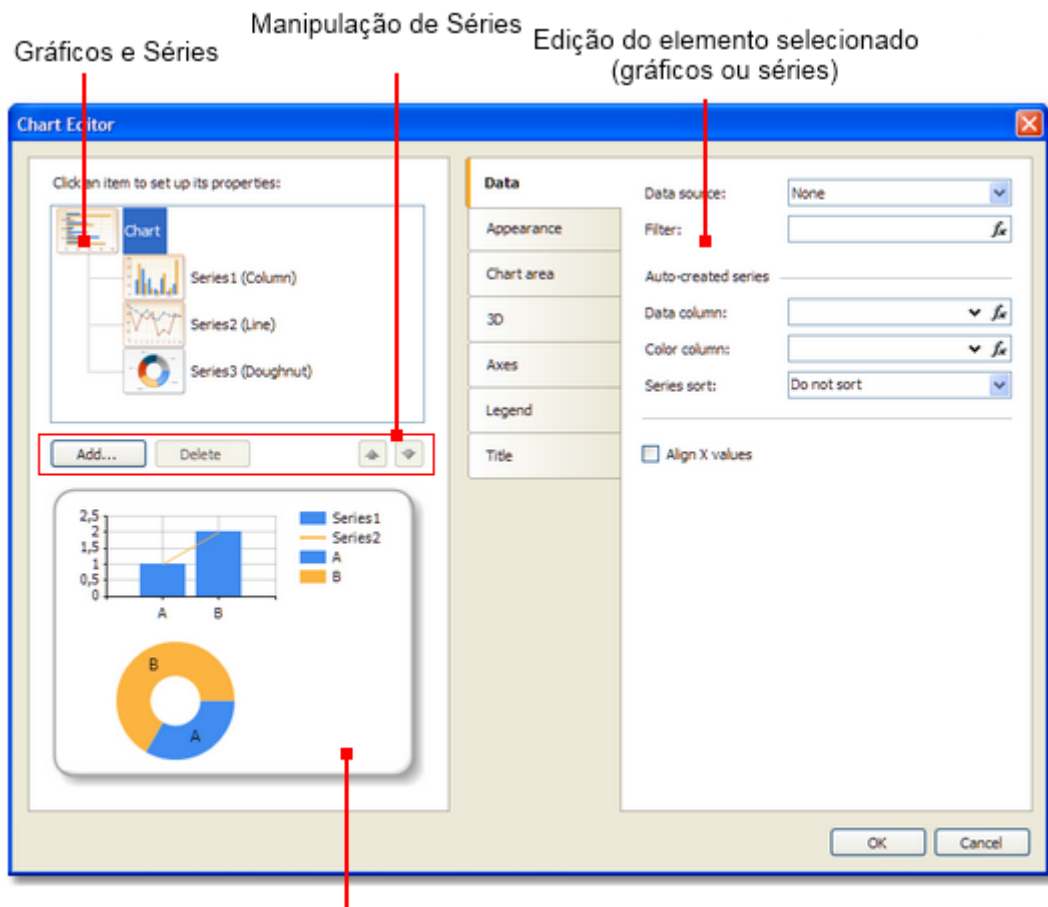
Um gráfico pode ter uma ou várias seções. Uma seção pode conter uma ou várias séries. Abaixo você pode ver o gráfico que contém duas seções (a primeira contém duas séries e a segunda contém uma série):



Algumas séries (por exemplo, séries de torta) requerem uma seção exclusiva do gráfico.

## Editor de gráficos

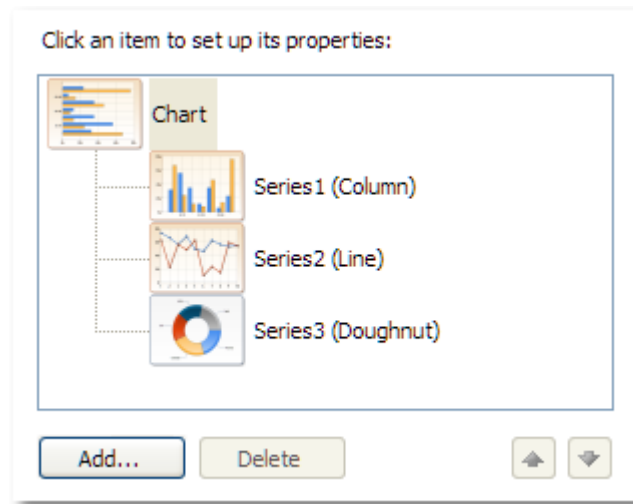
O objeto MS Gráfico contém várias configurações que podem ser alteradas no editor de gráficos. Para abrir o editor, dê um duplo clique no objeto MS Gráfico:



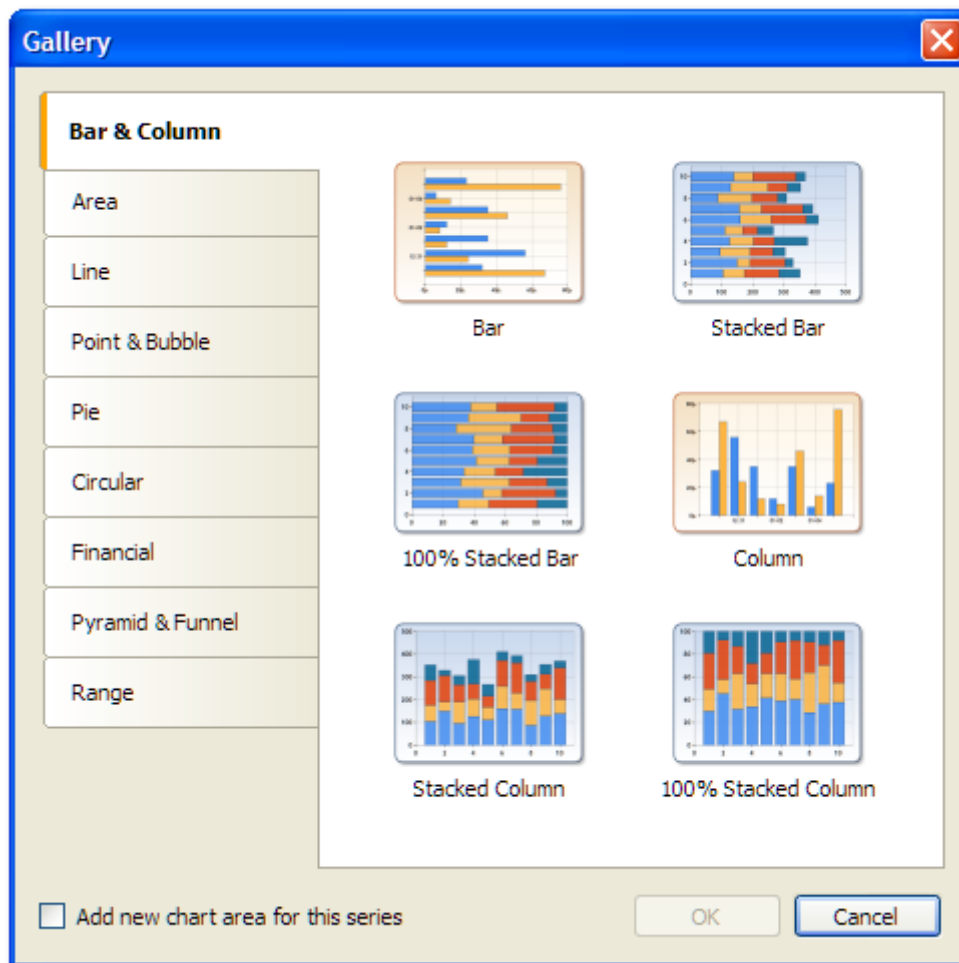
Exemplo. Os elementos são clicáveis.

## Manipulando séries

O objeto MS Gráfico pode conter uma ou mais séries. A lista de séries é mostrada no editor:



Para adicionar novas séries, pressione o botão **Adicionar....** Você verá a janela **Gallery**:



Selecione a categoria necessária e então os tipos de série necessários. Se você quiser colocar as séries na sua área do gráfico, marque o *checkbox* **Adicionar nova área de gráficos para esta série.**

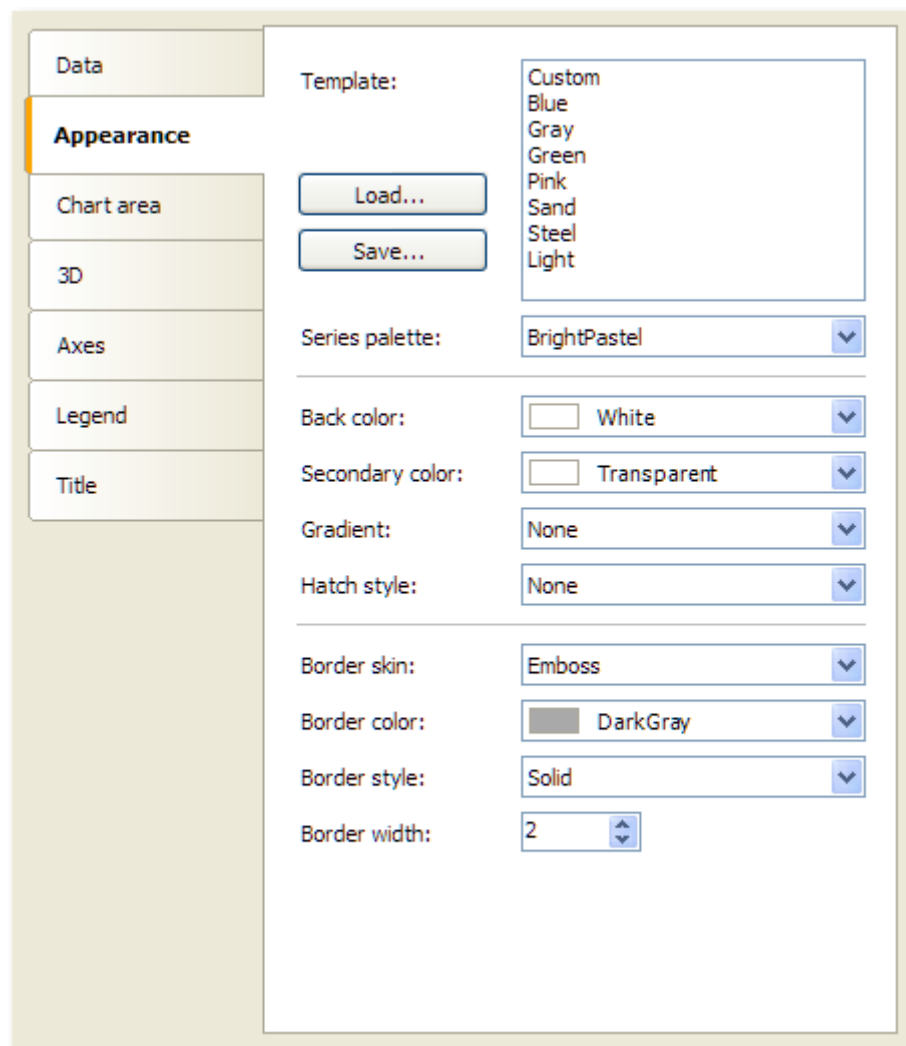
Para alguns tipos de séries (como torta, financeiro, piramidal), a seção do gráfico é adicionada automaticamente independentemente do estado do *checkbox*.

Para excluir as séries, pressione o botão Delete. Para mudar as ordens das séries, use os botões **Up** e **Down**.

### Definindo a aparência

Usando o editor de gráficos, você pode definir a aparência de cada elemento do gráfico. Todas as propriedades (mais de 100) são divididas em diversas categorias. Algumas delas são específicas do objeto MS Gráfico, enquanto outras são das séries.

Se você escolher o objeto MS Gráfico na lista de séries, você verá as seguintes propriedades da página:



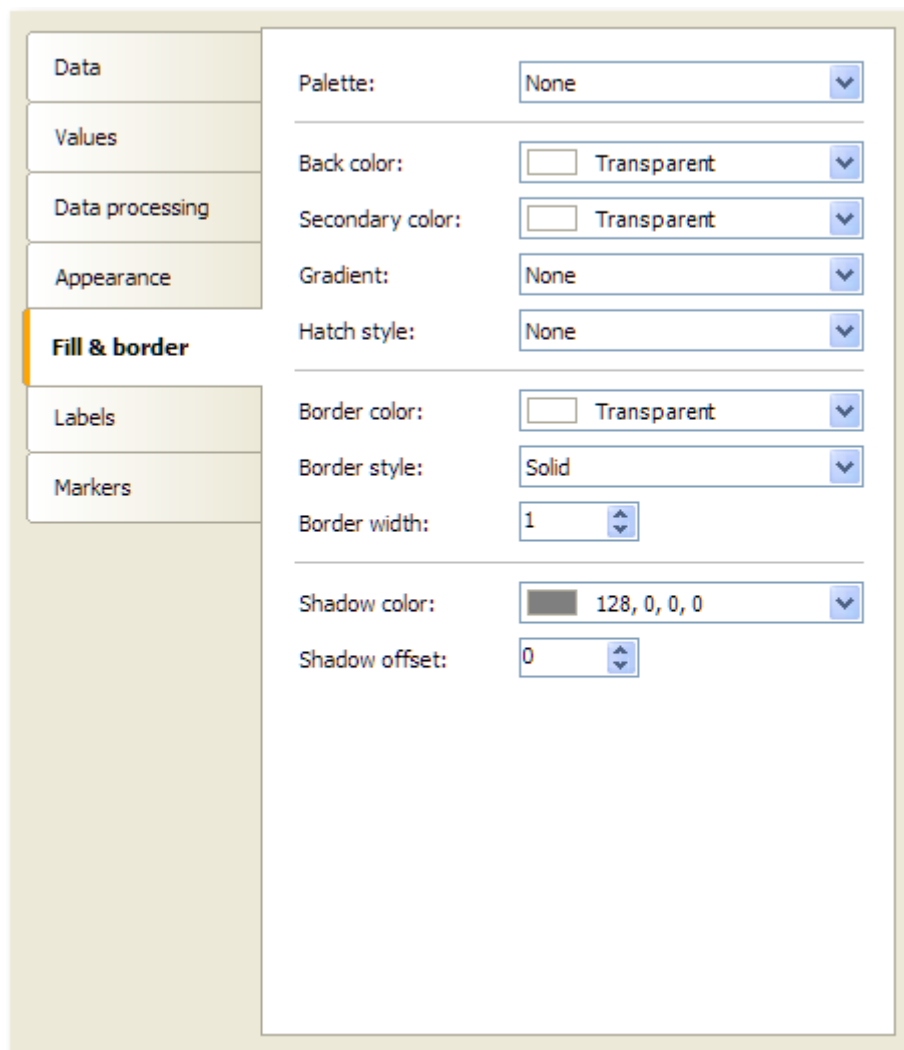
The screenshot shows the 'Appearance' tab of the MS Chart properties dialog. The left sidebar contains tabs for 'Data', 'Appearance', 'Chart area', '3D', 'Axes', 'Legend', and 'Title'. The 'Appearance' tab is active. The main area contains the following settings:

- Template:** A list box with options: Custom, Blue, Gray, Green, Pink, Sand, Steel, Light. Below it are 'Load...' and 'Save...' buttons.
- Series palette:** A dropdown menu set to 'BrightPastel'.
- Back color:** A dropdown menu set to 'White'.
- Secondary color:** A dropdown menu set to 'Transparent'.
- Gradient:** A dropdown menu set to 'None'.
- Hatch style:** A dropdown menu set to 'None'.
- Border skin:** A dropdown menu set to 'Emboss'.
- Border color:** A dropdown menu set to 'DarkGray'.
- Border style:** A dropdown menu set to 'Solid'.
- Border width:** A spinner box set to '2'.

- **Appearance** – borda e preenchimento do gráfico;

- **Chart Area** – borda, preenchimento e sombra;
- **3D** – configurações 3D;
- **Axes** – defina a aparência do eixo, o seu título, rótulos, **grid**, marcadores, rótulos personalizados e listras;
- **Legend** – estilo da legenda, ancoramento, borda, preenchimento, sombra e fonte;
- **Title** – estilo do título, ancoramento, borda, preenchimento, sombra, fonte.

Se você escolher os objetos da série da lista de séries, você verá as seguintes páginas de propriedades:



- **Appearance** – algumas configurações específicas para o tipo de série selecionado;
- **Fill & border** – preenchimento e borda dos valores das séries;
- **Labels** – rótulos das séries. Você pode escolher o tipo de rótulo, fonte, cor e preenchimento;
- **Markers** – marcadores das séries. Você pode escolher o tipo do marcador, sua cor e borda.



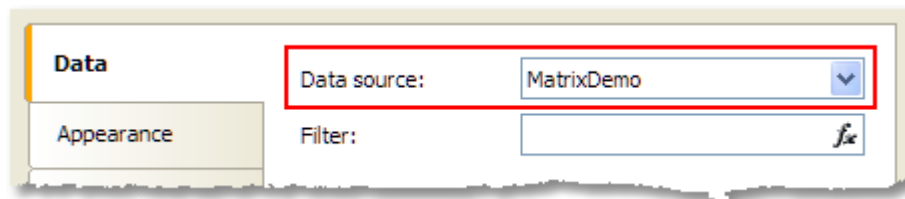
### Conectando o gráfico aos dados

Você pode preencher o gráfico com dados de várias maneiras:

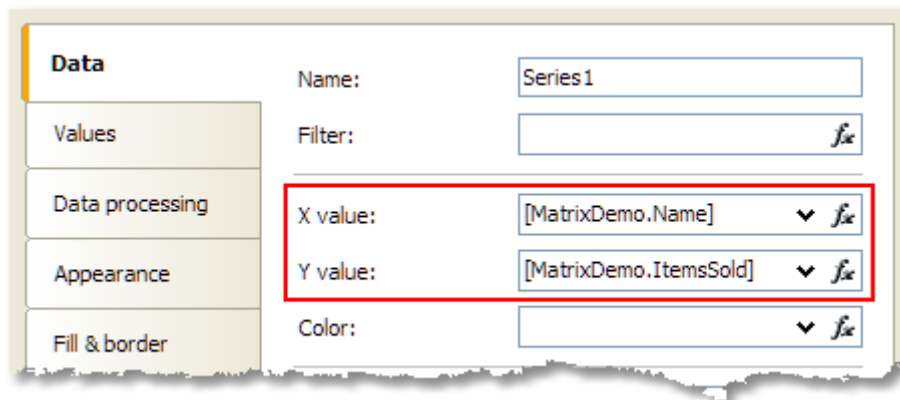
1. Use a fonte de dados. Para fazer isso, você precisa indicar a fonte de dados para o objeto MS Gráfico e conectar cada série às colunas de dados.
2. Usar valores fixos para cada série.
3. Preencher o objeto com dados através de *script*.

Para conectar o gráfico a uma fonte de dados, siga Estes passos:

- Selecione o objeto MS Gráfico na lista das séries;
- Troque para a aba **Data**;
- Escolha a fonte de dados;



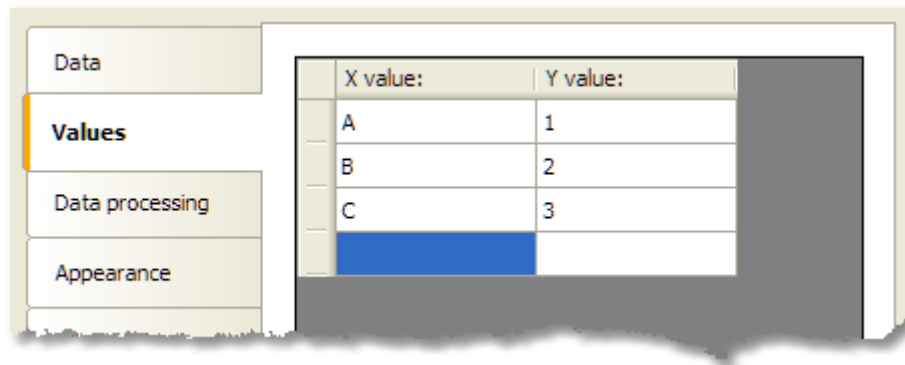
- Se necessário, defina a expressão do filtro de dados. Este filtro será aplicado para todas as séries do gráfico;
- Selecione as séries na lista de séries;
- Alterne para a guia **Dados**;
- Escolha as colunas de dados para cada valor de série. Dependendo do tipo de série, podem haver dois ou mais valores. A maioria dos tipos de séries possuem dois valores: X e Y:



- Se necessário, defina o filtro de dados da expressão. Este filtro será aplicado apenas à série atual;
- No controle **Cor**, você pode indicar a coluna de dados que retorna um valor de cor.

Você também deve fornecer uma lista de valores para as séries. Neste caso, a conexão com os dados não é necessária. Para fazer isso, selecione uma série na lista de séries e vá para a guia **Values**.

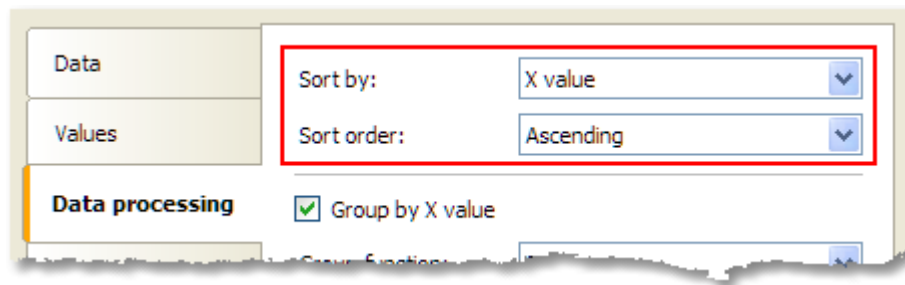
Preencha a tabela com valores:



| X value: | Y value: |
|----------|----------|
| A        | 1        |
| B        | 2        |
| C        | 3        |

### Disposição dos dados

Por padrão, o objeto gráfico mostra dados em uma ordem natural. Você pode mudar a ordem de disposição; para fazer isso, selecione as séries da lista de séries e troque para a guia **Data processing**:



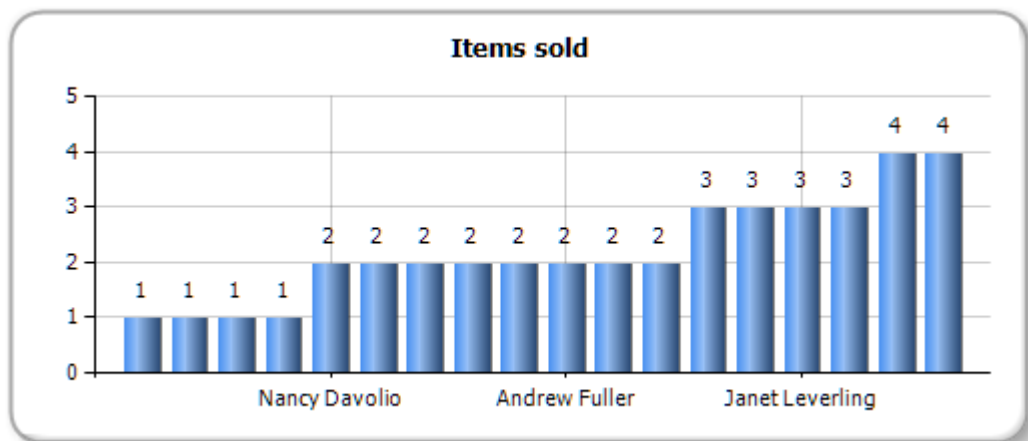
Você pode escolher um dos modos de disposição – não ordenar, ordenar pelo valor X e ordenar pelo valor Y.

### Agrupando os dados

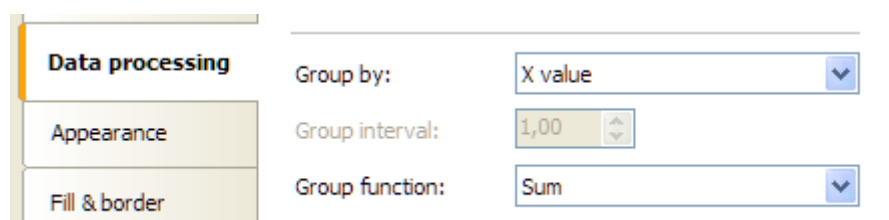
Algumas vezes nós enfrentamos um problema quando as séries são preenchidas com dados com vários valores X idênticos. Por exemplo, a tabela **MatrixDemo**, a qual é usada para demonstrar gráficos, tem os seguintes dados:

| Name            | Year | Month | ItemsSold | Revenue |
|-----------------|------|-------|-----------|---------|
| Andrew Fuller   | 2002 | 1     | 2         | 1800    |
| Andrew Fuller   | 1999 | 10    | 2         | 1900    |
| Andrew Fuller   | 1999 | 11    | 2         | 2000    |
| Andrew Fuller   | 2000 | 2     | 2         | 2100    |
| Janet Leverling | 1999 | 10    | 3         | 3000    |
| Janet Leverling | 1999 | 11    | 3         | 3100    |
| Janet Leverling | 2000 | 3     | 3         | 3200    |
| ...             |      |       |           |         |

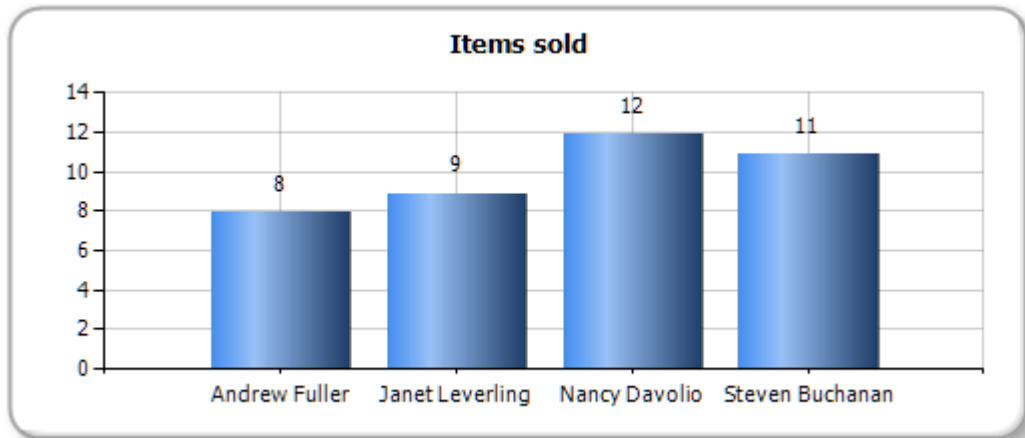
Se nós tentarmos construir um gráfico baseado nestes dados (por exemplo, vendas dos empregados – defina o valor X para a coluna dos Nomes, o valor Y para a coluna Itens Vendidos), nós vamos obter o seguinte resultado errôneo:



Nesta situação, nós precisamos agrupar os mesmos empregados em um valor. Para fazer isso, selecione as séries na lista de séries, e troque para a guia **Processamento de Dados**. Selecione o tipo do grupo – valor X e selecione Soma como uma função de grupo:



Como resultado, todos os empregados com o mesmo nome serão agrupados em uma valor, as suas vendas serão somadas. Você verá o seguinte resultado:



### Coletando os dados

Este instrumento para processamento de dados permite coletar vários valores de séries e transformá-los em um valor. Você pode escolher um dos seguintes algoritmos:

| Algoritmo                   | Descrição  |
|-----------------------------|--|
| <b>TopN</b>                 | Somente os N maiores valores serão exibidos. Todos os outros valores serão somados e apresentados como um valor <b>Outros</b> (você pode escolher a etiqueta deste valor). |
| <b>BottomN</b>              | Somente os N menores valores serão exibidos. Se o texto para o valor coletado não foi definido, este valor não é exibido.  |
| <b>Less than value</b>      |  |
| <b>Less than percent</b>    |  |
| <b>Greater than value</b>   |  |
| <b>Greater than percent</b> |  |

Por exemplo, para mostrar os cinco primeiros valores, defina as séries da seguinte forma:

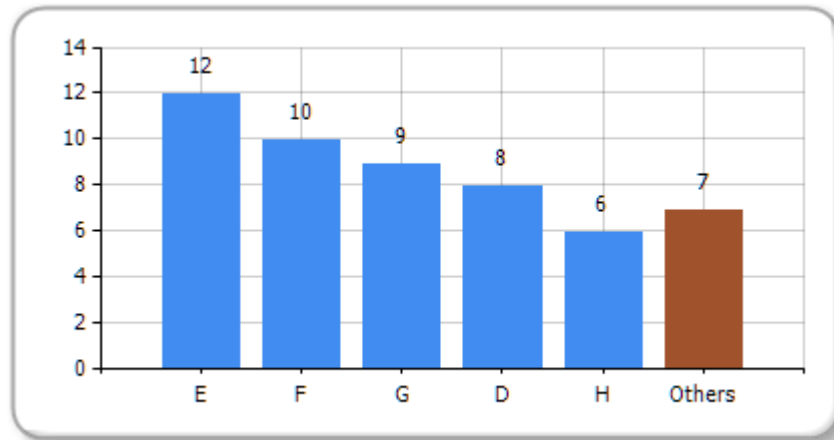
Collect data:

Value:

Collected text:


Collected color:


Você irá obter o seguinte resultado:



### Explodindo os valores

Para séries do tipo torta, você pode explodir alguns valores. Para fazer isso, selecione as séries na lista de séries e troque para a guia **Processamento de Dados**:

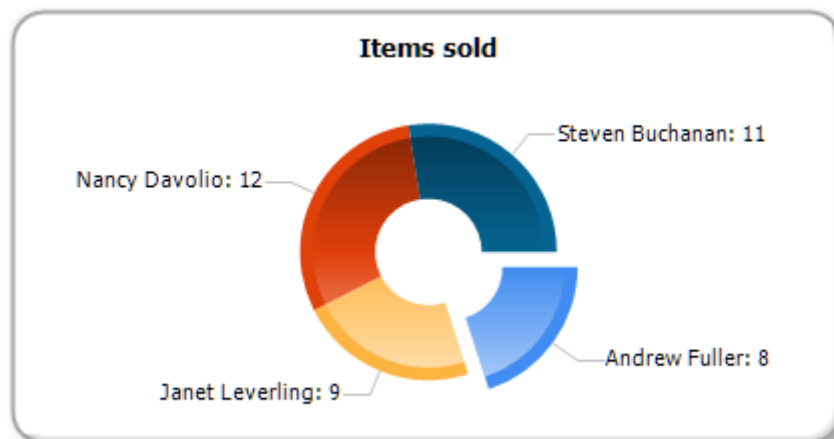
Explode:  

Value:  

Você pode usar um dos seguintes tipos de explosão: maior valor, menor valor e valor específico. Se você escolher a última opção, você tem que especificar o valor que você quer explodir. Isso pode ser qualquer expressão (veja o capítulo **Expressões** para mais detalhes). Por exemplo, se você precisar explodir o valor Andrew Fuller, use a seguinte expressão:

**Andrew Fuller**

Você obterá o seguinte resultado:



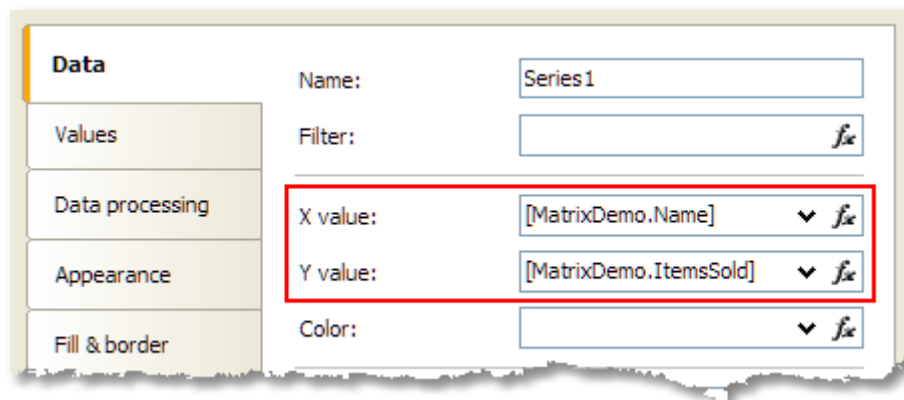
### Definindo séries automáticas

Você pode definir o gráfico criar automaticamente novas séries, dependendo dos dados na fonte de dados. Para definir séries automáticas, faça o seguinte:

- Crie uma série e defina as suas propriedades. Esta série será usada como padrão para todas as novas séries;
- Selecione o objeto MS Gráfico e defina a coluna de dados séries automáticas. O valor desta coluna será o nome de novas séries. Se ainda não houver séries com tal nome, uma nova série será adicionada.

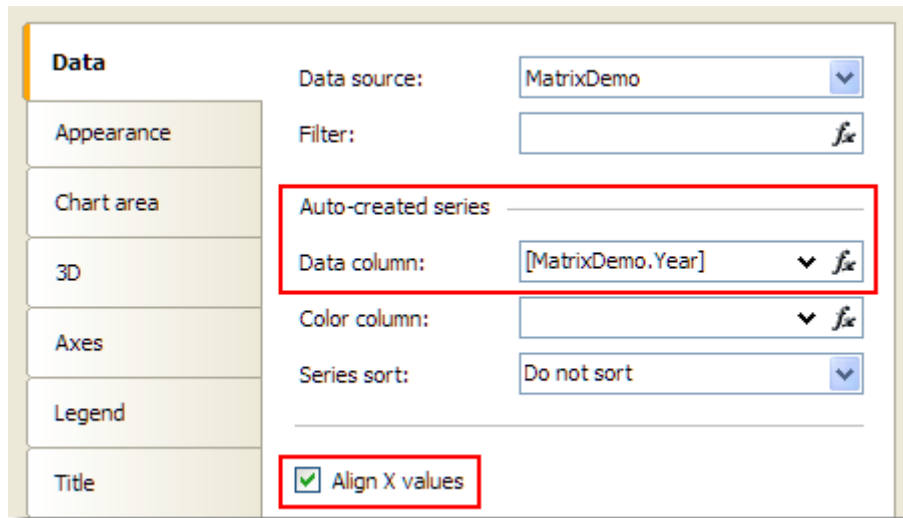
Vamos demonstrar como criar séries automaticamente. Nós vamos usar a tabela de dados **MatrixDemo** para obter o gráfico de vendas dos empregados por ano. Uma série representará o ano. Para fazer isto:

- Conecte o gráfico à fonte de dados **MatrixDemo**;
- Crie uma série e defina seus dados:



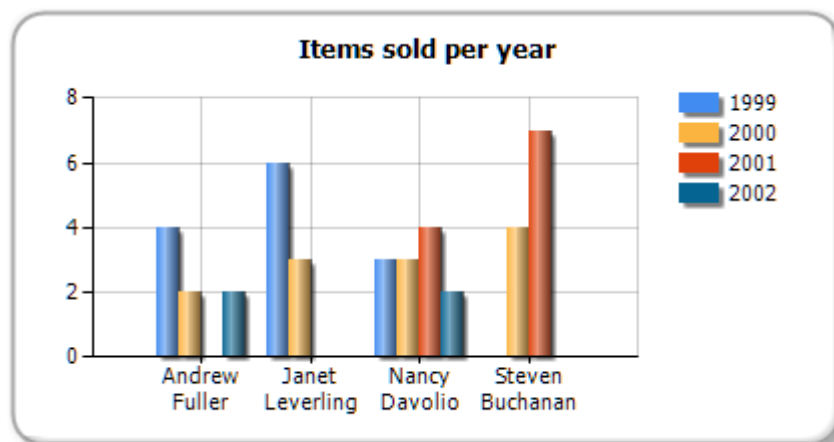
|                 |          |                             |
|-----------------|----------|-----------------------------|
| <b>Data</b>     | Name:    | Series1                     |
| Values          | Filter:  | <input type="text"/> fx     |
| Data processing | X value: | [MatrixDemo.Name] v fx      |
| Appearance      | Y value: | [MatrixDemo.ItemsSold] v fx |
| Fill & border   | Color:   | <input type="text"/> v fx   |

- Na guia **Processamento de Dados**, marque o *checkbox* **Agrupar por valor X**. Isto é necessário porque nossa fonte de dados tem vários empregados com o mesmo nome;
- Selecione o gráfico na lista de séries e defina sua coluna de auto-série na aba **Dados**:



- Nossas séries podem ter diferentes números de valores (porque alguns empregados não tem vendas em determinado ano). Para alinhar os valores das séries, marque o *checkbox* **Alinhar valores X**.

Nós obteremos o seguinte resultado:



### Gráficos interativos

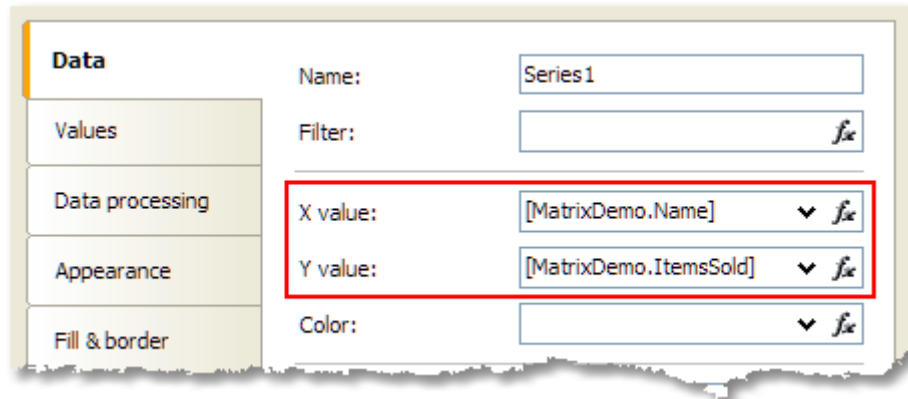
O gráfico, assim como todos os outros objetos do relatório, pode ser interativo. Você pode definir que quando você clicar em determinado valor do gráfico, outro relatório (detalhado) será exibido. Para fazer isso, você deve configurar a propriedade **Hyperlink**, como descrito no capítulo **Relatórios Interativos**. O gráfico irá transformar o valor em um *hyperlink* por si mesmo, quando você clicar em cima do seu elemento.

Vamos observar o relatório Charts/Interactive Chart da versão demo do **CIGAM Report**.

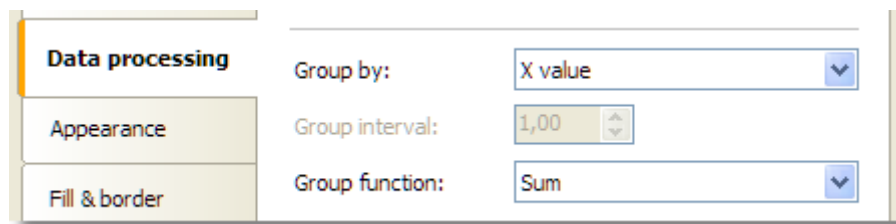
Crie um relatório com duas páginas. A primeira página irá conter um gráfico, enquanto a segunda irá conter um relatório detalhado que será mostrado quando você clicar em um valor do gráfico.

Coloque o objeto MS Gráfico na primeira página do relatório e defina as suas propriedades no editor de gráficos:

- Selecione o elemento **Chart** da lista de séries e escolha a fonte de dados **MatrixDemo**;
- Selecione a série da lista de séries e defina os valores X e Y: X = [MatrixDemo.Name], Y = [MatrixDemo.ItemsSold]>:



- Troque para a aba **Processamento de Dados** e selecione o tipo de grupo – X value:



Na segunda página do relatório, crie um relatório do tipo lista:

- Na janela Dados, crie um novo parâmetro de relatório chamado **SelectedEmployee**;
- Crie o seguinte *layout* de relatório:

| ReportTitle      | [SelectedEmployee] orders |                   |                    |                        |
|------------------|---------------------------|-------------------|--------------------|------------------------|
| Page Header      | Name                      | Year              | Month              | ItemsSold              |
| Data: MatrixDemo | [MatrixDemo.Name]         | [MatrixDemo.Year] | [MatrixDemo.Month] | [MatrixDemo.ItemsSold] |
| Report Summary   | Total:                    |                   |                    | [TotalItems]           |

- Abra o editor da banda de dados e indique a seguinte condição de filtragem:  
**[MatrixDemo.Name] == [SelectedEmployee]**

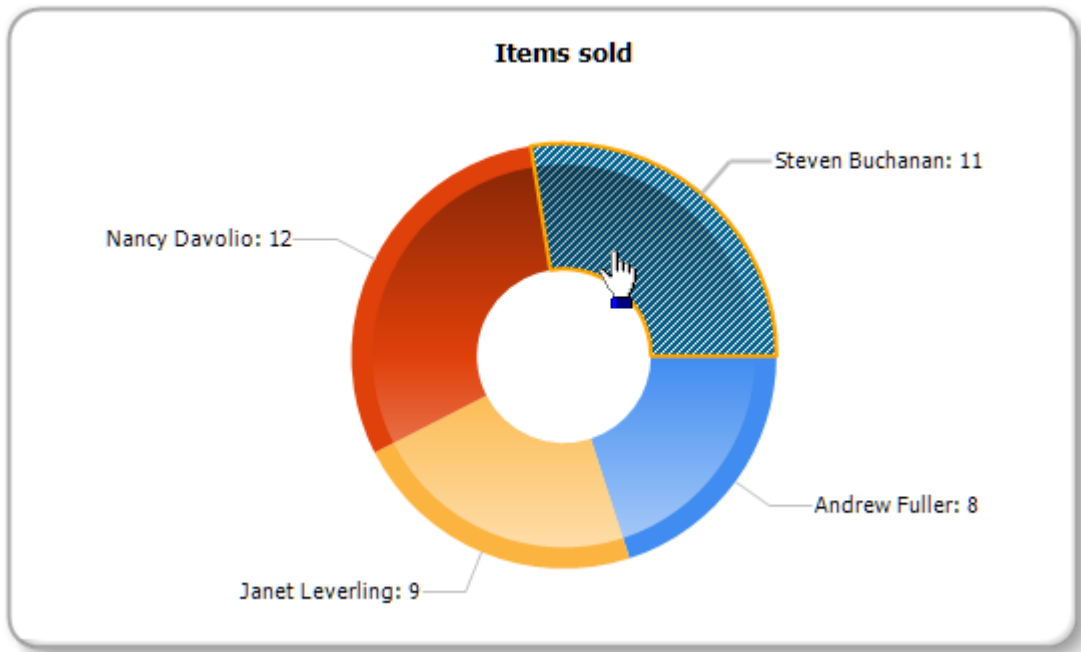
Agora defina o *hyperlink* do objeto MS Gráfico:

- No menu de contexto do objeto MS Gráfico, selecione **Hyperlink...**;
- Escolha o tipo de *hyperlink* – Página de Relatório;



- Escolha a segunda página do relatório e indique o nome do parâmetro – **SelectedEmployee**.

O relatório está pronto. Gere-o e mova o mouse para qualquer valor do gráfico. Este valor será selecionado visualmente, e o cursor do mouse mudará o seu formato:



Se você clicar no valor, você verá o seguinte relatório detalhado:

| Name            | Year | Month | ItemsSold |
|-----------------|------|-------|-----------|
| Steven Buchanan | 2001 | 1     | 3         |
| Steven Buchanan | 2001 | 2     | 4         |
| Steven Buchanan | 2000 | 1     | 4         |
| <b>Total:</b>   |      |       | 11        |

## Dados

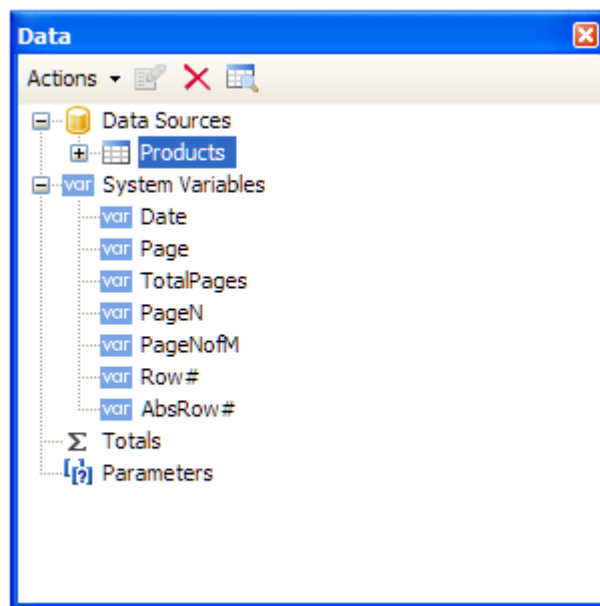
Qualquer relatório imprime alguns dados. No **CIGAM Report**, você pode operar com os seguintes dados:

- Fontes de dados;
- Variáveis de sistema;
- Valores totais;
- Parâmetros do relatório;
- Expressões, contendo qualquer um dos dados mencionados acima.

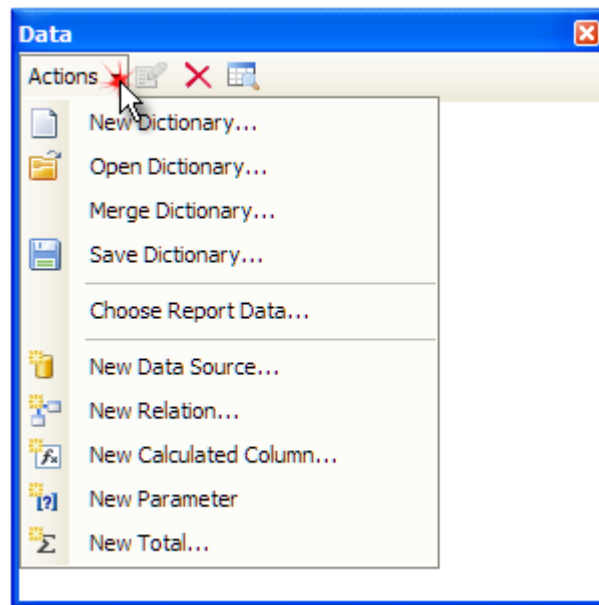
Neste capítulo, nós vamos ver como trabalhar com estes dados.

### A janela Dados

Todos os dados estão acessíveis na janela Dados. Esta janela pode ser exibida selecionando o menu **Dados > Exibir Janela de Dados**.



A janela Dados permite operar com todos os elementos dos dados e também arrastá-los para a página do relatório. Todas as operações pode ser feitas com a ajuda da barra de ferramentas e do menu **Action**:



Uma parte destas operações pode ser encontrada no menu de contexto da janela Dados. Por exemplo, se você selecionar a fonte de dados, você pode usar o menu de contexto dela para criar uma coluna calculada, excluir uma fonte de dados ou visualizar seus dados.

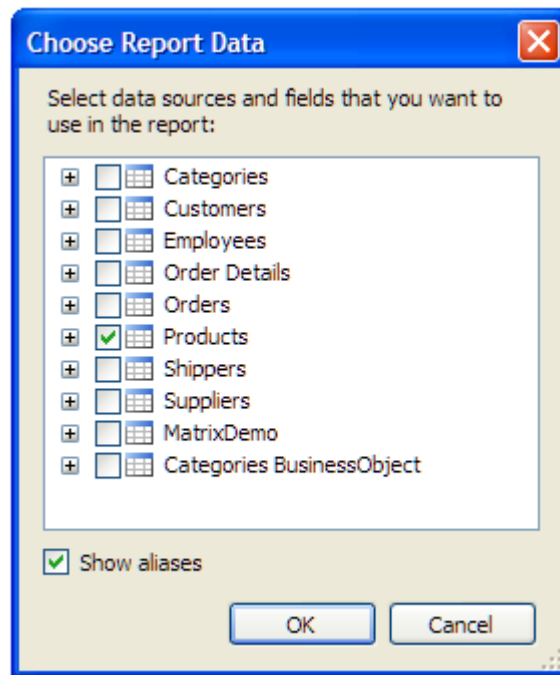
## Fontes de dados

Normalmente, a fonte de dados representa a tabela do BD ou a *query* SQL. Pode haver várias fontes de dados em um relatório. Para a maioria dos relatórios, somente uma fonte de dados é necessária. Um relatório como o **Mestre-Detalhe** precisa de duas fontes de dados as quais estão conectadas uma com a outra, usando o relacionamento (nós aprenderemos sobre isso mais tarde neste capítulo).

A fonte de dados tem uma ou várias colunas. Cada coluna tem um tipo de dados definido. Para ver o tipo de coluna, selecione-a e abra a janela Propriedades. O tipo da coluna é indicado na propriedade **DataType**. O ícone próximo ao nome da coluna também ajuda a determinar o seu tipo.

Há duas maneiras de definir a fonte de dados para o relatório.

O primeiro método – a fonte de dados é definida na aplicação e registrada no relatório. Esta tarefa é do programador que criou esta aplicação (veja os detalhes no **Programmers manual**). Um usuário deve escolher somente a fonte de dados necessária para usar no relatório. Isto pode ser feito no menu **Dados > Choose Report Data**.



Todos os dados registrados no relatório são listados nesta janela. Somente pegue os dados que serão necessários no seu relatório. Isto pode ser feito a qualquer momento enquanto estivermos trabalhando com um relatório.

O segundo método – você cria uma nova fonte de dados sozinho. Isto pode ser uma tabela do BD ou uma *query* SQL. Em tais casos, a definição da fonte de dados é salva em um arquivo do relatório.

O **CIGAM Report** permite conectar a muitos DBMS (data base management systems) populares, tais como MS SQL, Oracle, Interbase, Access. Além disso, você pode usar arquivos de dados que são salvos no formato xml/xsd.

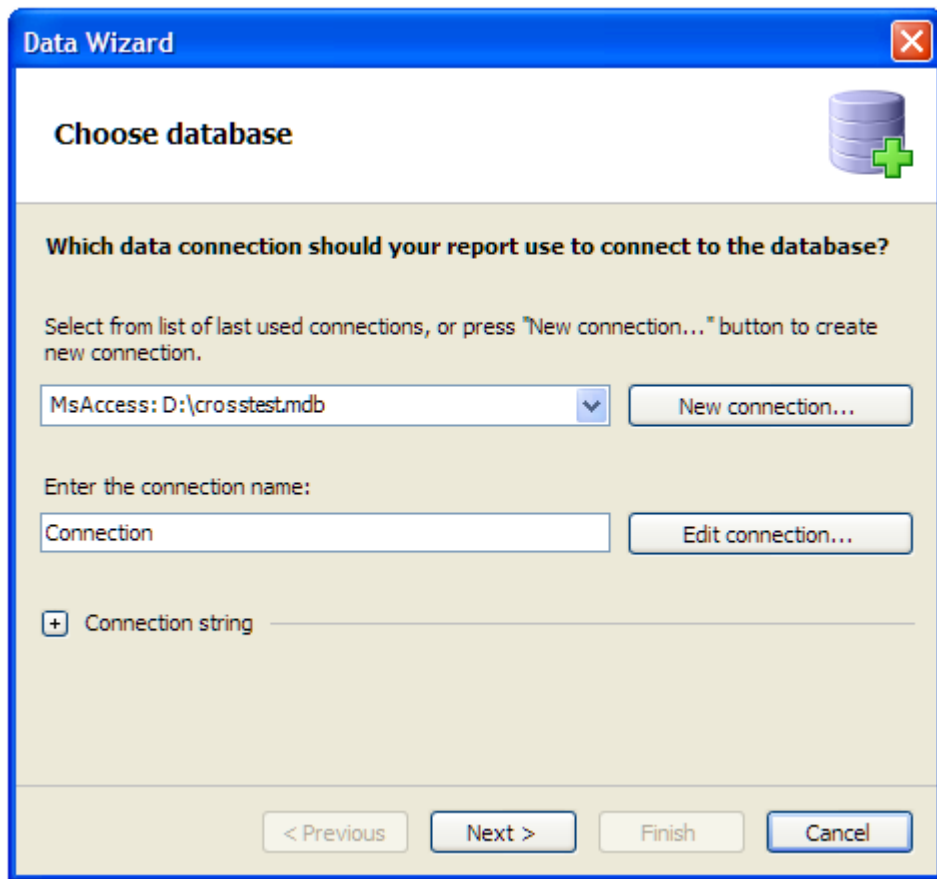
---

O conteúdo da tabela do BD não é salvo em um arquivo do relatório. Ao invés disso, a *string* de conexão e o esquema da fonte de dados são armazenados. A *string* de conexão pode conter dados como **login** e **senha**, e isto explica porque eles são mantidos criptografados em um arquivo do relatório. Quando necessário você pode aumentar a segurança usando sua própria chave para encriptação de dados. Neste caso o arquivo do relatório pode ser aberto corretamente somente no seu programa.

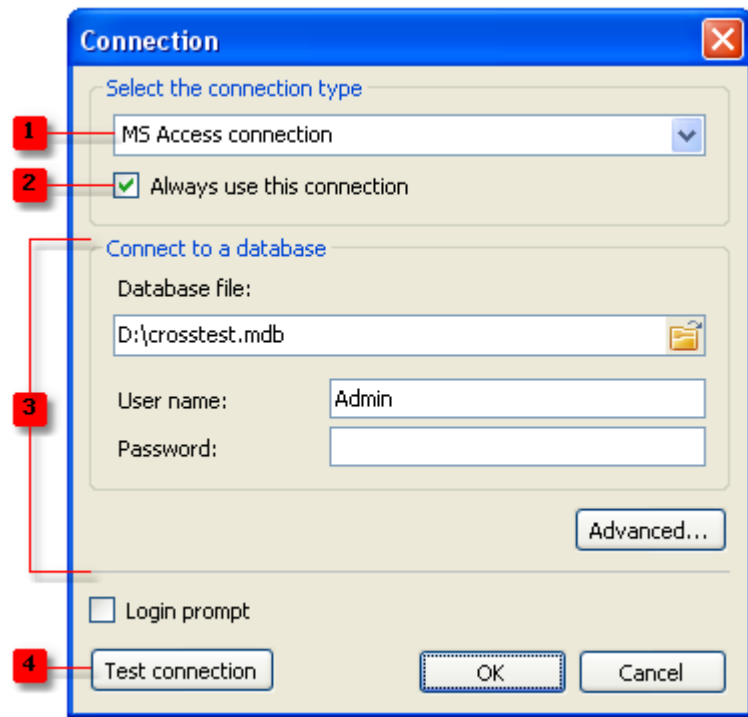
---

### Criando uma fonte de dados

Para criar uma nova fonte de dados, escolha o item de menu **Dados > Adicionar Fonte de Dados...** ou pressione o botão **Ações** na janela Dados e escolha o item **Nova fonte de Dados....** você verá a janela **Assistente de Dados**:



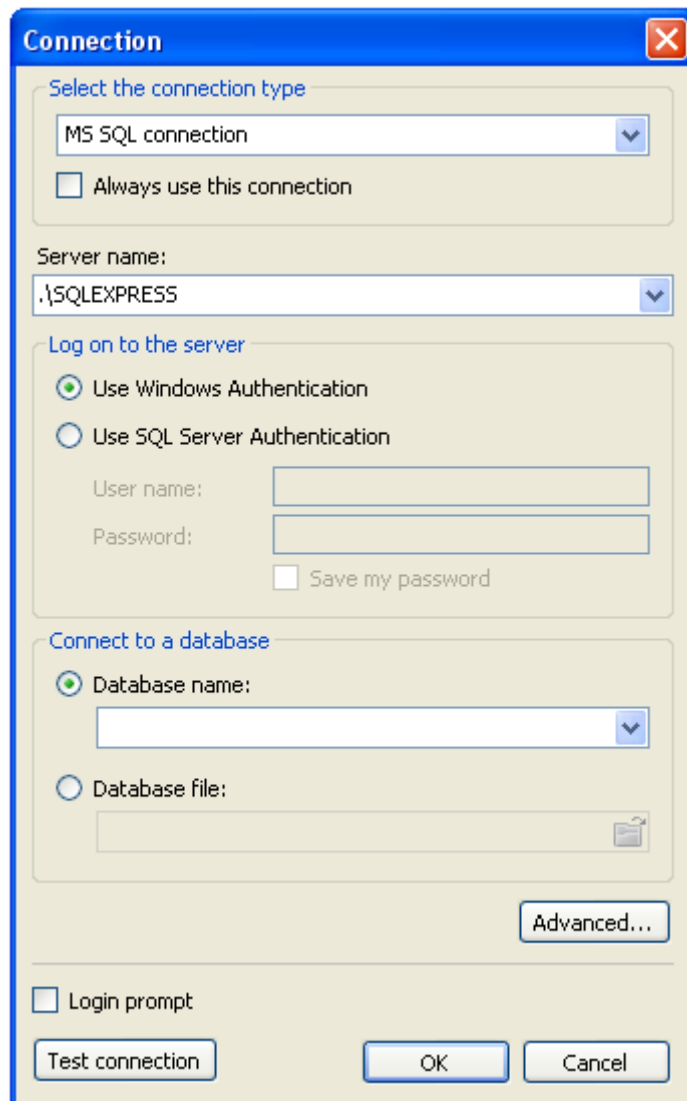
Antes de tudo, você deve criar uma conexão. Para isso, pressione o botão **Nova Conexão**. Você verá uma janela com as configurações da conexão:



Os seguintes elementos são mostrados na figura:

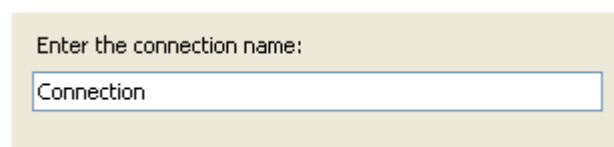
1. Tipo de conexão;
2. Se habilitado, o tipo de conexão escolhida será usado como padrão;
3. Configurações da conexão;
4. Botão para testar a conexão

A conexão com o banco de dados do MS Access é mostrada na figura. Se outro tipo de conexão for escolhido, então a área de configurações de conexão (3) será mudada. Por exemplo, a conexão para o banco de dados MS SQL tem as seguintes configurações:

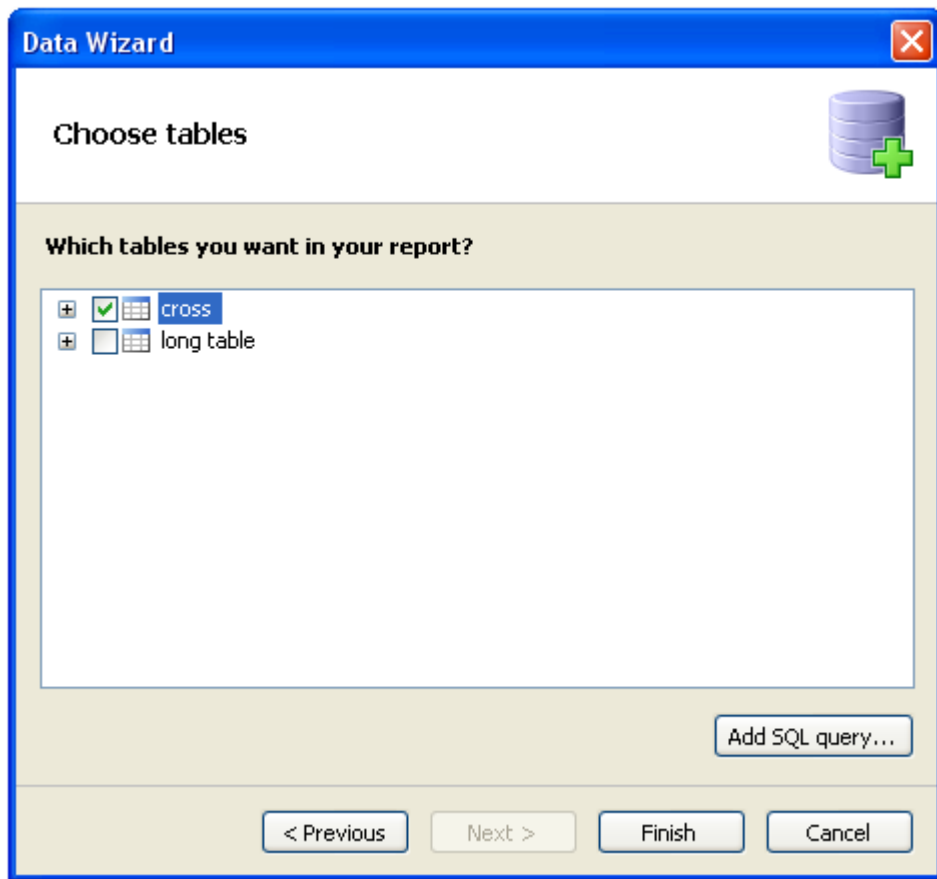


Escolha o tipo conexão necessário e defina os seus parâmetros. Depois de pressionar o botão OK, a janela será fechada e retornará para a janela do assistente de dados.

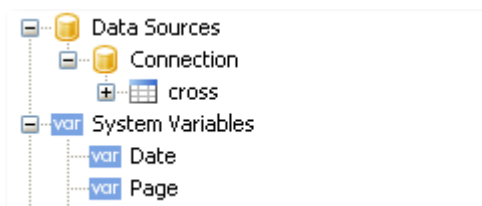
Depois, você precisa definir o nome da conexão. Este nome aparecerá na janela Dados.



Pressione o **botão Próximo** para continuar. Aqui você deverá escolher tabelas que estão acessíveis no banco de dados:



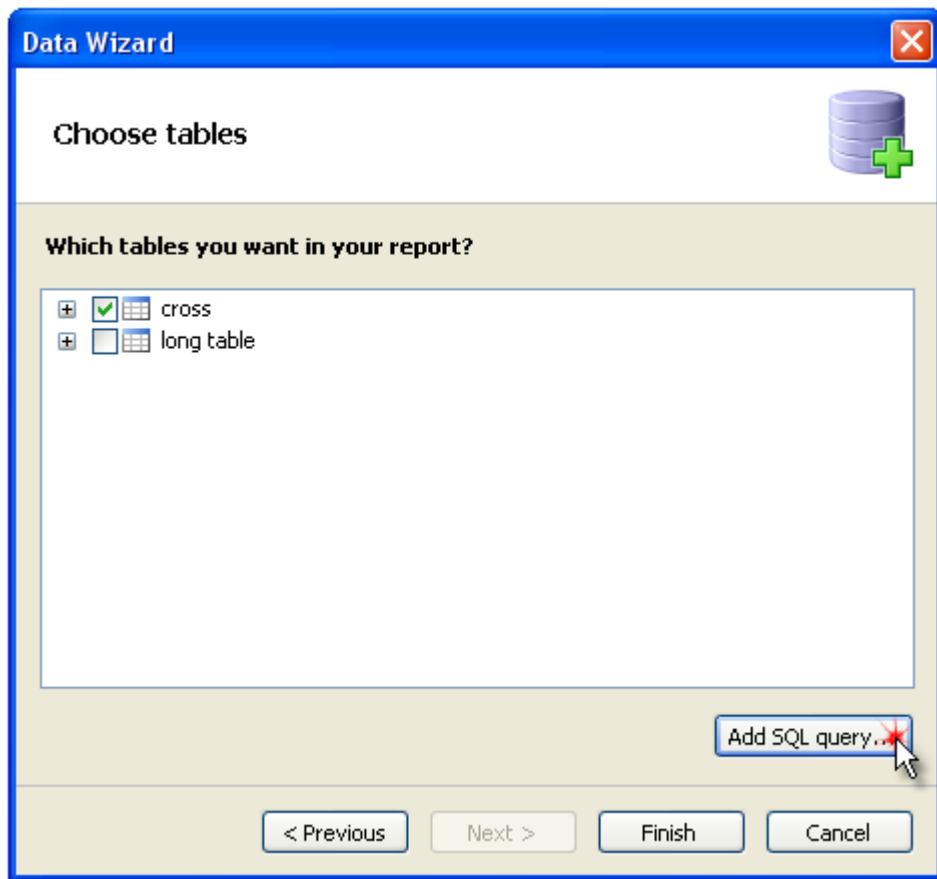
Marque as tabelas necessárias e feche o assistente, pressionando o botão **Concluir**. Agora você pode ver na janela Dados a conexão criada por você, a qual contém as fontes de dados escolhidas:



### Criando uma *query* SQL

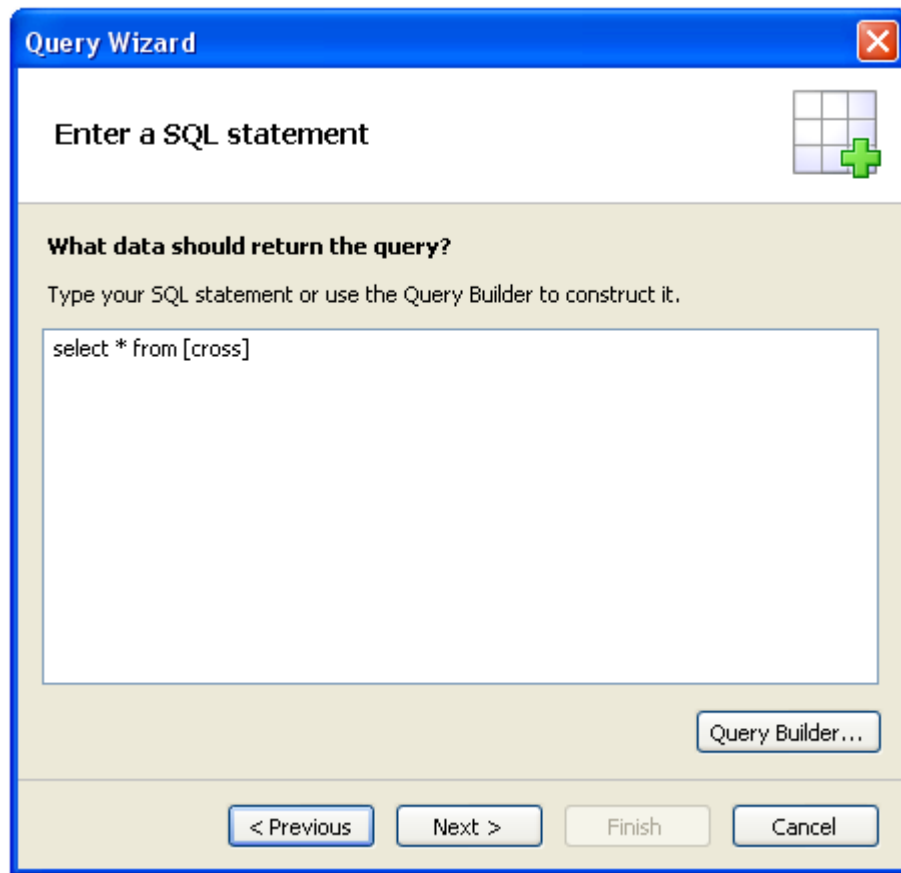
O assistente de dados permite escolher as tabelas rapidamente, contidas em uma fonte de dados. A criação da *query* SQL requer trabalho extra. Para isso, pressione o botão **Adicionar *query* SQL...** no segundo passo do assistente.





Você verá a janela do assistente de *query*. O assistente de *query* tem quatro páginas. Use os botões **Próximo** e **Voltar** para mudar de página.

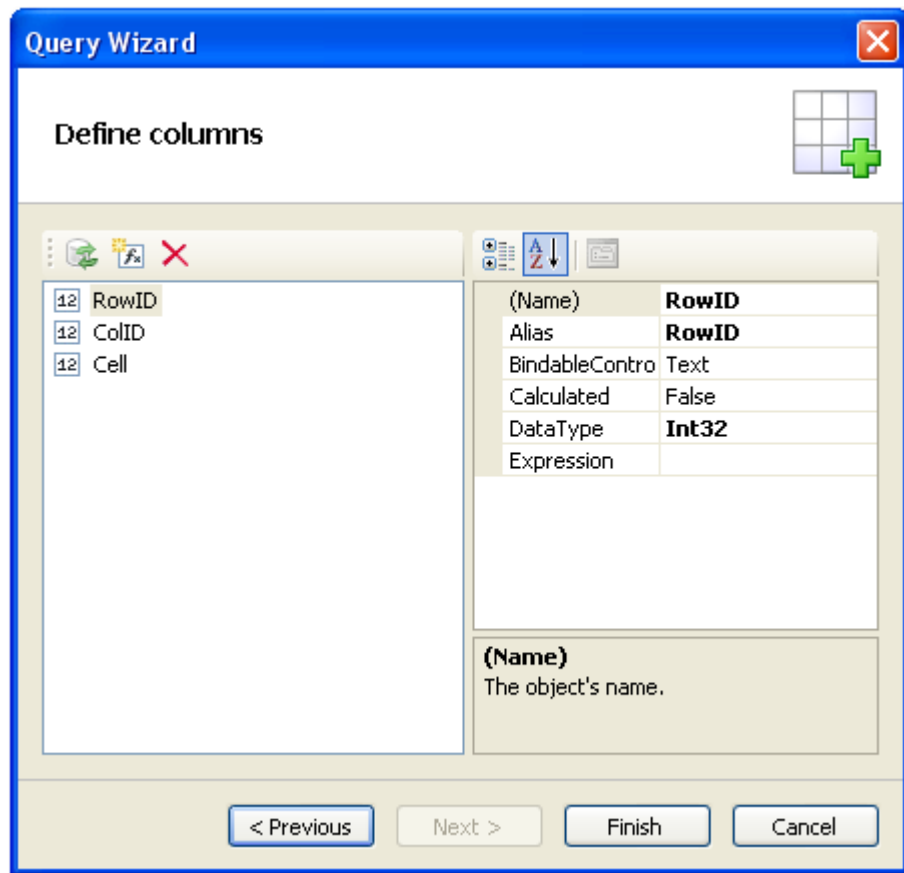
No primeiro passo, você precisa definir o nome de uma *query*. Este nome irá aparecer na janela Dados. Coloque qualquer nome único e pressione o **botão Próximo**.



No segundo passo, você deve colocar uma *query* na linguagem SQL. Use o dialeto da linguagem o qual é suportado pelo seu MSDB. Você pode usar o construtor de *query* para uma criação visual da *query*. Para fazer isso, pressione o botão **Query Builder**. O construtor de *query* será explicado com detalhes mais a frente.

Depois de você ter colocado o texto da *query*, pressione o **botão Próximo**. No terceiro passo, você pode definir os parâmetros da *query*. Isso é requerido se a sua *query* tiver parâmetros. Nós vamos falar sobre eles mais tarde, neste capítulo.

No último passo do assistente, você pode definir as colunas que serão retornadas pela *query*:



---

Se você tiver cometido um erro no texto da *query* ou na definição de parâmetros, você verá a mensagem de erro quando ir para a última página do assistente.

---

Como regra, é suficiente assegurar que a *query* retornou todas as colunas necessárias. Neste passo, você pode fazer o seguinte:

- Excluir colunas desnecessárias usando o botão **Deletar**;
- Redefinir as colunas, pressionando o botão **Atualizar**;
- Adicionar uma coluna calculada pressionando o botão **Adicionar uma coluna calculada**. Para uma nova coluna, é necessário definir o **Nome**, **Tipo de Dados** e as propriedades da expressão.

Depois de fechar o assistente, pressionando o botão **Concluir**, você retornará à janela **Assistente de Dados**.

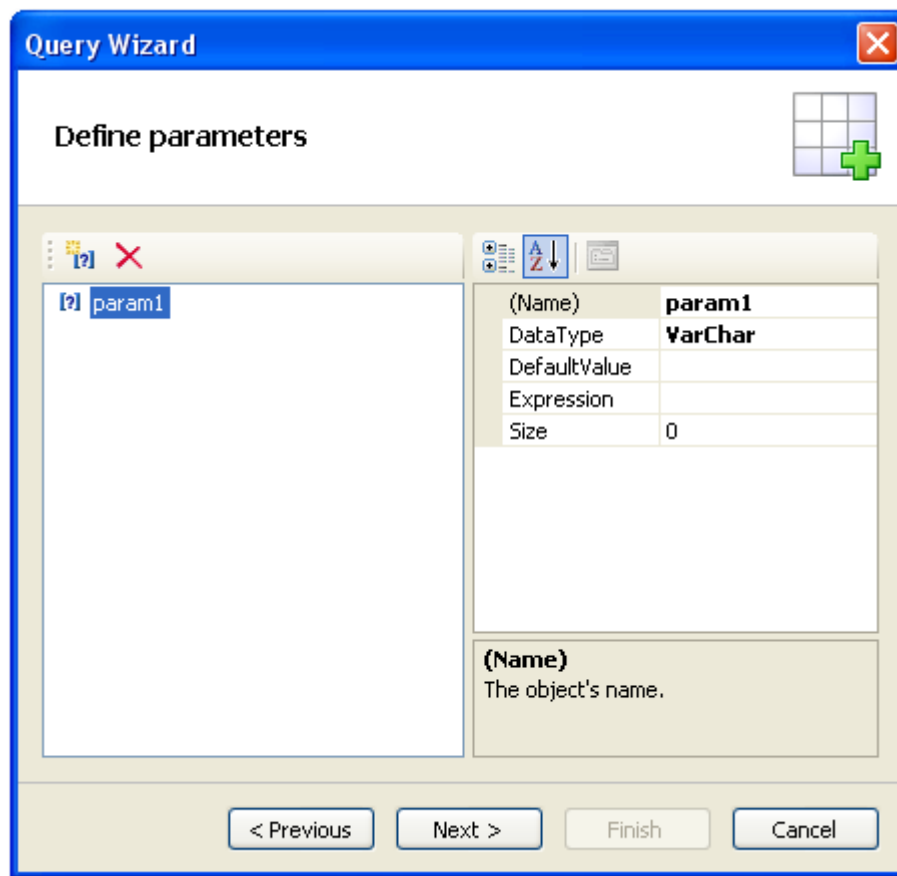
### Parâmetros da *query*

Pode haver parâmetros no texto da *query*. Vamos ver a seguinte *query*:

```
select * from DVDs
where Title = @param1
```

Isto é a *query* para a base de demonstração MS SQL. O parâmetro com o nome param1 é definido em uma *query*. Aqui deve observado o seguinte: o método de descrição dos parâmetros em uma *query* difere para diferentes DBMS. Para o MS SQL um parâmetro é marcado pelo símbolo @, para o MS Access os parâmetros não possuem nomes e são marcado pelo símbolo ?. Já para o Oracle, os parâmetros são precedidos por .:

Para trabalhar com parâmetros você precisa determinar os seus valores ao menos. Isto pode ser feito no terceiro passo do **Query Wizard**. Para criar um parâmetro, pressione o botão **Adicionar parâmetro**. Um novo parâmetro será criado:



As seguintes propriedades dos parâmetros devem ser definidas na direita da janela.

| Propriedade  | Descrição  |
|--------------|--|
| Name         | Nome do parâmetro. Aqui você deve indicar o mesmo nome que utilizou no texto da <i>query</i> . Alguns MSDBs (como por exemplo, o MS Access) não suportam parâmetros nomeados. Neste caso, não altere esta propriedade. |
| DataType     | Tipo de dado do parâmetro.   |
| DefaultValue | Valor que será utilizado se a propriedade <b>Expression</b> não for especificada, ou se a mesma não pode ser calculada (por exemplo, quando forem utilizadas <i>queries</i> no modo de criação de relatórios).         |



|            |  |
|------------|--|
| Expression | Expressão que retorna o valor do parâmetro. Esta será processada quando você rodar o relatório. Você pode indicar qualquer expressão nesta propriedade (veja maiores detalhes no capítulo <b>Expressões</b> ). |
| Size       | Tamanho dos dados do parâmetro. Esta propriedade deve ser indicada se um parâmetro possui o tipo <i>string</i> .   |

Se você definir as propriedades dos parâmetros incorretamente, você obterá um erro ao ir para a última página do assistente.

### Passando um valor para o parâmetro

Parâmetros são freqüentemente usados para pedir um valor para o usuário. Vamos ver duas maneiras para passar um valor para um parâmetro da *query*.

Na primeira maneira, você passa o valor programaticamente. Já que não há uma maneira fácil de passar um valor diretamente para o parâmetro da *query*, você precisa usar o parâmetro do relatório, o qual pode ser facilmente definido por código. Você deve fazer o seguinte:

- Criar o parâmetro do relatório (nós vamos discutir os parâmetros do relatório mais tarde neste capítulo). Defina o mesmo **DataType** para o parâmetro do relatório, já que este é usado no parâmetro da *query*.
- Na propriedade **Expression** do parâmetro da *query*, refira-se para um parâmetro do relatório, por exemplo:

#### [MyReportParameter]

- Passe um valor para o parâmetro do relatório:

```
report1.SetParameterValue("MyReportParameter", 10);
```

Na segunda maneira, você usa **dialogue forms** para fazer isso (veremos mais sobre eles mais tarde). Por exemplo, se você precisar de um valor do tipo *string*, faça o seguinte:

- Adicione um diálogo no seu relatório;
- Coloque o **TextBoxControl** nele. Este controle será usada para colocar o valor da *string*;
- Defina o parâmetro como segue:

```
Name=param1  
DataType=VarChar  
DefaultValue= (empty string)  
Expression=TextBox1.Text  
Size=255
```

Onde o **TextBox1** é um controle que contém um valor digitado pelo usuário.

### Editando uma conexão

A conexão dos dados, que foi criada com a ajuda do **Assistente de Dados**, pode ser editada. Para fazer isso, escolha a conexão dos dados na janela Dados e pressione o botão **Editar** na barra de tarefas. Você verá a janela do assistente de dados que nós vimos antes. Nesta janela, você pode mudar as configurações da conexão, pressionando o botão **Editar Conexão...** O tipo de conexão não pode ser mudado.

Na segunda página do assistente, você pode selecionar as tabelas que você deseja por no seu relatório. Quando você estiver pronto, pressione o botão **Concluir**.

### Editarando a fonte de dados

A fonte de dados que foi criada com a ajuda do **Assistente de Dados**, pode ser editada. Para fazer isso, escolha a fonte de dados na janela Dados e pressione o botão **Editar** na barra de ferramentas. Você verá a janela **Query Wizard** que nós vimos anteriormente. Nesta janela, você pode mudar o texto da *query* SQL, defina os parâmetros da *query* e as colunas de dados.

Para excluir uma fonte de dados, selecione-a e pressione o botão **Delete** na barra de ferramentas. Durante este processo, a deleção física da fonte de dados não ocorre, ela somente fica inacessível. Você pode entrar no menu **Dados > Escolher Dados do Relatório...** e habilite tal fonte de dados. No entanto, isso não deve ser problema para você, porque as fontes de dados deletadas nunca são salvas no arquivo do relatório, e portanto, não são restauradas quando o relatório estiver sendo lido na próxima vez.

### Identificadores

Cada elemento de dados (fontes de dados e colunas) possui um nome próprio. Por padrão, este é o nome definido no banco de dados. Em alguns casos, pode ser difícil entender o que está por trás de tal nome, por exemplo, **ProdID**.

Os elementos de dados possuem um segundo nome – **alias** (identificador). Usando um **alias**, você pode renomear um elemento. Por exemplo, se nós tivermos uma fonte de dados **CATEGORY\_TABLE** e uma coluna chamada **PROD\_ID**, você pode dar a eles os seguintes **alias**:

**CATEGORY\_TABLE** → **Categorias**

**PROD\_ID** → **Product ID**

Você pode se referir a tal coluna de dados da seguinte forma:

**[Categorias.Product ID]**

---

Quando nos referirmos a um elemento de dados, você deve usar o **alias**, se ele tiver sido definido. Nunca se refira à um elemento usando o seu nome original, neste caso.

---

Para renomear um elemento de dados neste caso, escolha a janela Dados, e pressione F2. Também, você pode selecionar o item **Rename** no menu de contexto do objeto. Depois disso, insira o nome requerido e pressione **Enter**.

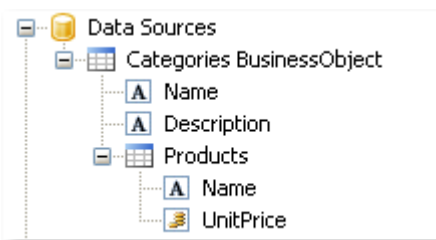
Você também pode renomear um elemento usando a janela Propriedades. Selecione um elemento na janela Dados, troque para a janela Propriedades e mude o valor da propriedade **Alias**.

Para excluir um **alias** (voltar para o nome original), selecione o elemento e escolha o item **Delete alias** no seu menu de contexto.

### Fontes hierárquicas de dados

As fontes de dados que nós vimos são relacionais, ou seja, elas vem de um DBMS relacional (freqüentemente chamado RDBMS). O **CIGAM Report** também suporta outros tipos de dados – as fontes hierárquicas de dados. Tais fontes de dados vem dos então chamados **business objects**, os quais são freqüentemente usados em aplicações para representar fontes de dados relacionais como classes .Net.

A única maneira de adicionar fontes hierárquicas de dados no seu relatório é registrando-o programaticamente. Isto será discutido no **Programmers manual**. Agora nós veremos algumas diferenças entre fontes de dados comuns e hierárquicas. Na figura abaixo você pode ver duas fontes de dados, **Categorias BusinessObject** e **Produtos**. Como você pode ver, a fonte de dados **Produtos** está contida dentro da fonte de dados pai, **Categorias BusinessObject**:



Isto significa que estas duas fontes de dados estão relacionadas uma com a outra e podem ser usadas no relatório do tipo **Mestre-Detalhe**. Você pode também usar cada uma destas fontes de dados separadamente em um relatório simples.

### Relações

Uma relação pode ser estabelecida entre duas fontes de dados. A relação é usada para definir o relacionamento **Mestre-Detalhe**. Por exemplo, um registro na tabela **Categorias** pode ter múltiplas entradas na tabela **Produtos**:

| Categories |              |
|------------|--------------|
| CategoryID | CategoryName |
| 1          | Beverages    |

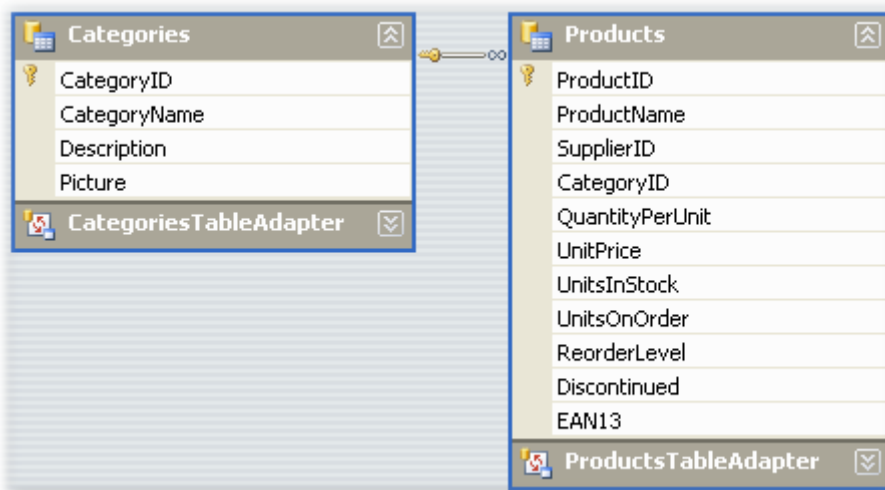
  

| Products  |            |                  |
|-----------|------------|------------------|
| ProductID | CategoryID | Product name     |
| 1         | 1          | Chai             |
| 2         | 1          | Chang            |
| 39        | 1          | Chartreuse verte |
| 38        | 1          | Côte de Blaye    |
| 24        | 1          | Fantástica       |

Para criar uma relação, você precisa indicar o seguinte:

- Tabela pai;
- Tabela filha;
- Conjunto de colunas chaves na tabela pai;
- Conjunto de colunas chaves na tabela filha.

Como um exemplo, iremos olhar as tabelas **Categorias** e **Produtos** de um banco de dados demo. Elas possuem a seguinte estrutura:



| Categories   |
|--------------|
| CategoryID   |
| CategoryName |
| Description  |
| Picture      |

| Products        |
|-----------------|
| ProductID       |
| ProductName     |
| SupplierID      |
| CategoryID      |
| QuantityPerUnit |
| UnitPrice       |
| UnitsInStock    |
| UnitsOnOrder    |
| ReorderLevel    |
| Discontinued    |
| EAN13           |

Ambas as tabelas tem o campo **CategoryID**, no qual o relacionamento pode ser definido. Então, uma categoria pode conter vários produtos.

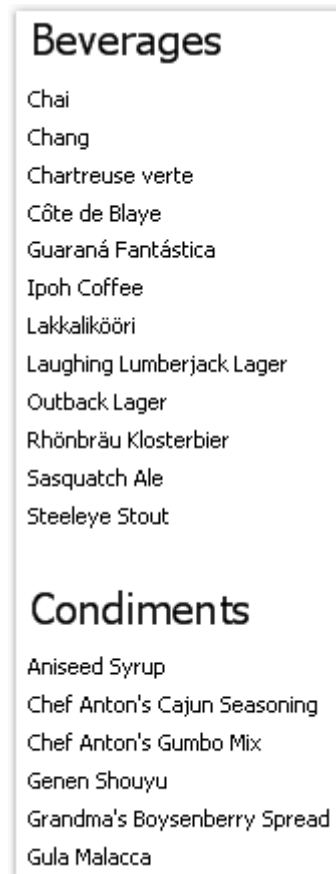


Como as fontes de dados relacionais podem ser usadas no **CIGAM Report**? Há duas maneiras de se fazer isto.

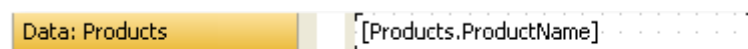
O primeiro método torna possível a construção de relatório do tipo **Mestre-Detalhe**. Para fazer isso, duas bandas Dados são usadas. A banda **Mestre** é conectada à fonte de dados **Mestre**, a banda **Detalhes** à fonte de dados **Detalhes**. Nosso exemplo ficará assim:



Tal relatório, se você gerá-lo, imprimirá uma lista de produtos em cada categoria:



O segundo método permite referir-se ao **Mestre** através da fonte de dados de detalhes. Vamos mostrar isso com um exemplo. Digamos que nós quiséssemos imprimir uma lista de todos os produtos. Para isso, nós precisaríamos de uma banda Dados, a qual seria conectada à tabela **Produtos**:



Tal relatório irá imprimir todos os produtos de todas as categorias. Digamos que, ao lado de cada produto, nós queremos imprimir a categoria que ele pertence. Sem usar relacionamentos, isso teria sido mais difícil. Tudo que nós sabemos sobre a categoria do produto é o seu ID (representado pela coluna **CategoryID** na tabela **Produtos**). O nome da categoria, que nós queremos imprimir, está armazenada na coluna **NomeCategoria** da tabela **Categorias**. Com a ajuda do relacionamento, nós podemos nos referir ao nome da categoria da seguinte forma:

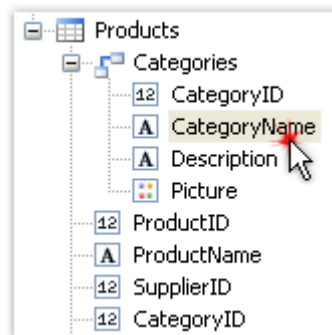
**[Produtos.Categorias.NomeCategoria]**

Para a linha atual da tabela **Produtos**, o **CIGAM Report** irá encontrar a linha correspondente na tabela **Categorias**, e retornará o valor da coluna **NomeCategoria**.

Geralmente, a maneira de referir-se à um campo da tabela pai permite um número ilimitado de tabelas **ancestors**:

**[Child\_table.Its\_parent.Parent\_of\_a\_parent.And\_so\_on.Column\_name]**

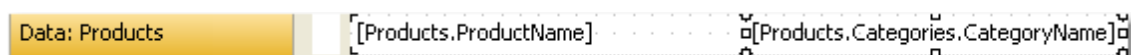
Para adicionar esta coluna ao relatório, abra a tabela **Products** na janela **Dados**. Você verá que entre suas colunas, há um link para a tabela **Categorias**:



Se nós arrastarmos a coluna mostrada acima para dentro do relatório, então nós vamos obter um objeto Texto com um texto:

**[Produtos.Categorias.NomeCategoria]**

Nosso relatório será o seguinte:



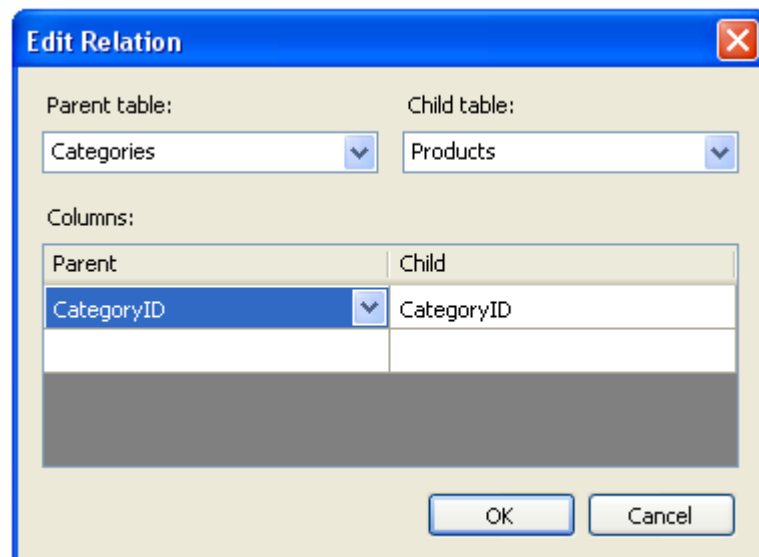
Se nós gerarmos ele, iremos ver o seguinte:

|                   |                |
|-------------------|----------------|
| Alice Mutton      | Meat/Poultry   |
| Aniseed Syrup     | Condiments     |
| Boston Crab Meat  | Seafood        |
| Camembert Pierrot | Dairy Products |
| Carnarvon Tigers  | Seafood        |
| Chai              | Beverages      |
| Chang             | Beverages      |

### Criando uma relação

Para criar uma relação, clique no botão **Ações** na janela Dados, e selecione o item **Novo relation....**

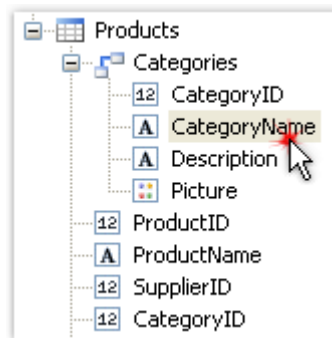
Você verá o editor de relações:



| Parent     | Child      |
|------------|------------|
| CategoryID | CategoryID |

Em primeiro lugar, você deve escolher as tabelas pai e filha. Depois disto, na parte inferior da janela, você precisa escolher colunas de dados relacionadas. As tabelas podem ser relacionadas com a ajuda de uma ou várias colunas. Depois de as colunas terem sido definidas, feche o editor de relações, pressionando o botão Ok.

A relação que você criou pode ser vista na janela Dados, se você escolher a fonte de dados filha e abrir a lista das suas colunas. Ao lado das colunas, você verá o relacionamento com a fonte de dados pai:



A coluna de dados da fonte de dados pai pode ser inserida no relatório usando o método **drag&drop**. Então, se nós escolhermos as colunas mostradas na figura, e arrastá-la para a página do relatório, nós vamos obter um objeto Texto com o seguinte conteúdo:

**[Produtos.Categorias.NomeCategoria]**

### Editando uma relação

Para editar uma relação, abra a lista de colunas da fonte de dados filha, encontre a relação necessária e clique no botão **Editar...** localizado na barra de ferramentas. Isto irá abrir o editor de relações, que nós vimos antes.

### Variáveis de sistema

Há uma lista de variáveis de sistema que podem ser usadas no relatório:

| Variável          | Descrição   |
|-------------------|---|
| <b>Date</b>       | Data e hora do início do relatório.   |
| <b>Page</b>       | Número da página atual.   |
| <b>TotalPages</b> | Número total de páginas no relatório. Para usar esta variável, você precisa habilitar passa duas vezes do relatório.                          |
| <b>PageN</b>      | Número da página na forma: Page N.  |
| <b>PageNofM</b>   | Número da página na forma: Page N of M  |
| <b>Row#</b>       | O número da linha de dados dentro do grupo. Este valor é reiniciado no início de um novo grupo.   |
| <b>AbsRow#</b>    | Número absoluto da linha de dados. Este valor nunca reinicia no início de um novo grupo.  |
| <b>Page#</b>      | Número atual da página. Se você juntar vários relatórios preparados em um pacote, esta variável retornará o número atual da página no pacote. |

Esta variável é na verdade um **macro**. O seu valor é substituído quando o componente é visto na janela de **pré-visualização**. Isso significa que você não pode usá-lo em uma expressão.

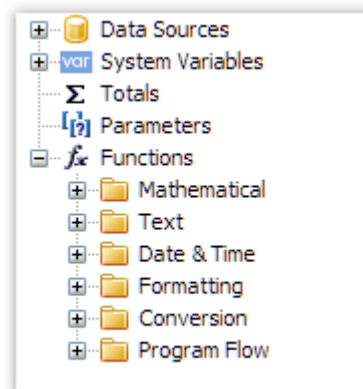
|                    |   |
|--------------------|---|
| <b>TotalPages#</b> | Número total de páginas no seu relatório. Se você juntar vários relatórios preparados em um pacote, esta variável retornará o número de páginas no pacote. Você não precisa usar o passa duas vezes para obter o valor correto. |
|--------------------|---|

esta variável é na verdade um **macro**. Este valor é substituído quando o componente é visto na janela **pré-visualização**. Isso significa que você não pode usá-lo em uma expressão.

**HierarchyLevel** Nível atual de hierarquia em um relatório hierárquico (veja **Printing hierarchy**). O nível maior é igual a 1.

## Funções

O **CIGAM Report.Net** contém um monte de funções pré-prontas (mais de 60). Todas as funções são divididas em várias categorias e são acessíveis através da janela Dados:



Você poderá usar uma função em qualquer expressão, no **script** (veja o capítulo **Script**), ou imprimir o seu valor em um objeto Texto. Por exemplo, o seguinte texto no objeto Texto:

**[Sqrt(4)]**

Será impresso como 2 (raiz quadrada de 4).

A seguinte expressão retornará 4:

**Sqrt(4) + 2**

Vamos ver duas maneiras de inserir uma função no relatório:

- Você pode arrastar e soltar (drag&drop) uma função da janela Dados para uma página do relatório. O objeto Texto será criado, o qual contém o chamado de uma função. Você deve editar o texto para adicionar parâmetros ao chamado da função.
- Você pode arrastar e soltar a função para o código do **script**.
- No editor da expressão, você pode ver uma cópia da janela Dados, a qual age da mesma maneira – você pode arrastar itens dela e soltá-los no texto da expressão.

Abaixo nós iremos descrever cada função em detalhes:



## Matemáticas

### Abs

| Função | Parâmetros    | Valor retornado |
|--------|---------------|-----------------|
| Abs    | Sbyte value   | Sbyte           |
| Abs    | Short value   | Short           |
| Abs    | Int value     | Int             |
| Abs    | Long value    | Long            |
| Abs    | Float value   | Float           |
| Abs    | Double value  | Double          |
| Abs    | Decimal value | decimal         |

Retorna um valor absoluto.

#### Exemplo:

Abs (-2.2) = 2.2

### Acos

| Função | Parâmetros | Valor retornado |
|--------|------------|-----------------|
| Acos   | Double d   | Double          |

Retorna o ângulo (em radiano) cujo cosseno é d. d deve estar em um intervalo entre -1 e 1.

Multiplique o valor de retorno por  $180 / \pi$  para converter de radianos para graus.

#### Exemplo:

Acos (0) \*  $180 / \pi$  = 90

### Asin

| Função | Parâmetros | Valor retornado |
|--------|------------|-----------------|
| Asin   | Double d   | Double          |

Retorna o ângulo (em radianos) cujo seno é d. d deve estar em um intervalo entre -1 e 1.

Multiplique o valor de retorno por  $180 / \pi$  para converter de radianos para graus.



**Exemplo:**

Asin (0) = 0

*Atan*

| Função | Parâmetros | Valor retornado |
|--------|------------|-----------------|
| Atan   | Double d   | Double          |

Retorna o ângulo (em radianos) cuja tangente é d.

Multiplique o valor de retorno por  $180 / \pi$  para converter de radianos para graus.

**Exemplo:**

Atan (1) \*  $180 / \pi$  = 45

*Ceiling*

| Função  | Parâmetros | Valor retornado |
|---------|------------|-----------------|
| Ceiling | Double d   | Double          |
| Ceiling | Decimal d  | decimal         |

Retorna o menor inteiro maior ou igual a d.

**Exemplo:**

Ceiling (1.7) = 2

*Cos*

| Função | Parâmetros | Valor retornado |
|--------|------------|-----------------|
| Cos    | Double d   | Double          |

Retorna o cosseno do ângulo especificado (d). O ângulo deve ser em radianos.

Multiplique o valor de retorno por  $180 / \pi$  para converter de radianos para graus.

**Exemplo:**



$$\text{Cos}(90 * \pi / 180) = 0$$

### Exp

| Função | Parâmetros | Valor retornado |
|--------|------------|-----------------|
| Exp    | Double d   | Double          |

Retorna  $e$  (2.71828), elevado na potência especificada  $d$ .

#### Exemplo:

$$\text{Exp}(1) = 2.71828$$

### Floor

| Função | Parâmetros | Valor retornado |
|--------|------------|-----------------|
| Floor  | Double d   | Double          |
| Floor  | Decimal d  | decimal         |

Retorna o maior inteiro menor ou igual a  $d$ .

#### Exemplo:

$$\text{Floor}(1.7) = 1$$

### Log

| Função | Parâmetros | Valor retornado |
|--------|------------|-----------------|
| Log    | Double d   | Double          |

Retorna o logaritmo do número especificado  $d$ .

#### Exemplo:

$$\text{Log}(2.71828) = 1$$

### Maximum

| Função  | Parâmetros           | Valor retornado |
|---------|----------------------|-----------------|
| Maximum | Int val1, int val2   | Int             |
| Maximum | Long val1, long val2 | Long            |





|                |                            |         |
|----------------|----------------------------|---------|
| <b>Maximum</b> | Float val1, float val2     | Long    |
| <b>Maximum</b> | Double val1, double Val2   | Double  |
| <b>Maximum</b> | Decimal val1, decimal val2 | decimal |

Retorna o maior valor entre val1 e val2.

**Exemplo:**

Maximum (1,2) = 2

### Minimum

| Função         | Parâmetros                 | Valor retornado |
|----------------|----------------------------|-----------------|
| <b>Minimum</b> | Int val1, int val2         | Int             |
| <b>Minimum</b> | Long val1,, long val2      | Long            |
| <b>Minimum</b> | Float val1, float val2     | Long            |
| <b>Minimum</b> | Double val1, double Val2   | Double          |
| <b>Minimum</b> | Decimal val1, decimal val2 | decimal         |

Retorna o menor valor entre val1 e val2.

**Exemplo:**

Minimum (1,2) = 1

### Round

| Função       | Parâmetros | Valor retornado |
|--------------|------------|-----------------|
| <b>Round</b> | Double d   | Double          |
| <b>Round</b> | Decimal d  | decimal         |

Arredonda d para o inteiro mais próximo.

**Exemplo:**

Round (1.47) = 1

| Função       | Parâmetros            | Valor retornado |
|--------------|-----------------------|-----------------|
| <b>Round</b> | Double d, int digits  | Double          |
| <b>Round</b> | Decimal d, int digits | decimal         |

Arredonda d para uma precisão especificada no parâmetro **digits**



**Example:**

Round (1.478, 2) = 1.48

*Sin*

| Função | Parâmetros | Valor retornado |
|--------|------------|-----------------|
| Sin    | Double d   | Double          |

Retorna o seno do ângulo especificado (d). O ângulo deve ser em radianos.

Multiplique o valor de retorno por  $180 / \pi$  para converter de radianos para graus.

**Exemplo:**

Sin (90 \*  $\pi$  / 180) = 1

*Sqrt*

| Função | Parâmetros | Valor retornado |
|--------|------------|-----------------|
| Sqrt   | Double d   | Double          |

Retorna a raiz quadrada de d.

**Exemplo:**

Sqrt (4) = 2

*Tan*

| Função | Parâmetros | Valor retornado |
|--------|------------|-----------------|
| Tan    | Double d   | Double          |

Retorna a tangente do do ângulo especificado (d). O ângulo deve ser em radianos.

Multiplique o valor de retorno por  $180 / \pi$  para converter de radianos para graus.

**Exemplo:**

Tan (45 \*  $\pi$  / 180) = 1



*Truncate*

| Função   | Parâmetros | Valor retornado |
|----------|------------|-----------------|
| Truncate | Double d   | Double          |
| Truncate | Decimal d  | decimal         |

Calcula a parte inteira de d.

**Exemplo:**

Truncate (1.77) = 1



## Texto

Nota:

- Estas funções não modificam o valor inserido na *string*. Ao invés disso, elas retornam uma nova *string* modificada;
- O primeiro caractere em uma *string* tem índice 0. Tenha isso em mente quando trabalhar com funções que pegam o índice do caractere, como o **Insert**.

## Asc

| Função | Parâmetros | Valor retornado |
|--------|------------|-----------------|
| Asc    | Char c     | int             |

Retorna um valor Inteiro, representando o código do caractere correspondente ao caractere.

### Exemplo:

Asc (A) = 65

## Chr

| Função | Parâmetros | Valor retornado |
|--------|------------|-----------------|
| Chr    | Int i      | char            |

Retorna o caractere associado com o código do caractere específico.

### Exemplo:

Asc (65) = A

## Insert

| Função | Parâmetros                             | Valor retornado |
|--------|--|-----------------|
| Insert | String s, int startIndex, string value | String          |

Inserir uma *substring* valor na *string* s em uma posição de índice especificada **startIndex** e retorna uma nova *string*.

### Exemplo:

Insert (ABC, 1, 12) = A12BC



### Length

| Função | Parâmetros | Valor retornado |
|--------|------------|-----------------|
| Length | String s   | Int             |

Retorna o tamanho do s.

#### Exemplo:

Length (ABC) = 3

### LowerCase

| Função    | Parâmetros | Valor retornado |
|-----------|------------|-----------------|
| LowerCase | String s   | String          |

Converte todos os caracteres de s para minúsculo e retorna um resultado.

#### Exemplo:

LowerCase (ABC) = abc

### PadLeft

| Função  | Parâmetros               | Valor retornado |
|---------|--------------------------|-----------------|
| PadLeft | String s, int totalWidth | String          |

Alinha pela direita os caracteres na string s, colando espaços na esquerda para uma largura total especificada no parâmetro **totalWidth**.

#### Exemplo:

PadLeft (ABC, 5) =     ABC

| Função  | Parâmetros                                       | Valor retornado |
|---------|--|-----------------|
| PadLeft | String s,<br>int totalWidth,<br>char paddingChar | String          |

Alinha pela direita os caracteres na string s, colocando espaços com os caracteres **paddingChar** na esquerda, para uma largura total especificada no parâmetro **totalWidth**.

**Example:**

Padleft (ABC, 5, 0) = 00ABC

**PadRight**

| Função   | Parâmetros                  | Valor retornado |
|----------|-----------------------------|-----------------|
| PadRight | String s,<br>int totalWidth | String          |

Alinha pela esquerda os caracteres na string s, colando espaços na direita para uma largura total especificada no parâmetro **totalWidth**.

**Exemplo:**

Padright (ABC, 5) = ABC

| Função   | Parâmetros                                       | Valor retornado |
|----------|--|-----------------|
| PadRight | String s,<br>int totalWidth,<br>char paddingChar | String          |

Alinha pela esquerda os caracteres na string s, colocando espaços na direita com os caracteres **paddingChar**, para uma largura total especificada no parâmetro **totalWidth**.

**Example:**

Padright (ABC, 5, 0) = ABC00

**Remove**

| Função | Parâmetros                  | Valor retornado |
|--------|-----------------------------|-----------------|
| Remove | String s,<br>Int startIndex | String          |

Deleta todos os caracteres da string s começando na posição **startIndex** e continuando até a última posição.

**Example:**

Remove (ABCD, 3) = ABC

| Função | Parâmetros | Valor retornado |
|--------|------------|-----------------|
|--------|------------|-----------------|



| Função | Parâmetros                                | Valor retornado |
|--------|---|-----------------|
| Remove | String s,<br>Int startIndex,<br>Int count | String          |

Deleta o número de caracteres especificados no parâmetro **count** da string **s**, começando na posição **startIndex**.

**Exemplo:**

Remove (A00BC, 1, 2) = ABC

**Replace**

| Função  | Parâmetros                                       | Valor retornado |
|---------|--|-----------------|
| Replace | String s,<br>String oldValue,<br>String newValue | String          |

Retorna uma string **s** na qual uma **substring oldValue** especificada tenha sido substituída com outra **substring newValue**.

**Exemplo:**

Replace (A00, 00, BC) = ABC

**Substring**

| Função    | Parâmetros                  | Valor retornado |
|-----------|-----------------------------|-----------------|
| Substring | String s,<br>Int startIndex | String          |

Recupera uma **substring** de uma string **s**. A **substring** inicia na posição de caractere especificada no parâmetro **startIndex**.

**Exemplo:**

Substring (ABCDEF, 4) = EF

| Função    | Parâmetros                                 | Valor retornado |
|-----------|--|-----------------|
| Substring | String s,<br>Int startIndex,<br>Int length | String          |



Recupera uma *substring* de uma string *s*. A *substring* inicia na posição de caractere especificada no parâmetro **startIndex** e tenha o tamanho especificado no parâmetro **length**.

**Exemplo:**

Substring (ABCDEF, 1, 3) = BCD

*TitleCase*

| Função    | Parâmetros | Valor retornado |
|-----------|------------|-----------------|
| TitleCase | String s   | String          |

Converte a string especificada para título, onde as primeiras letras de cada palavra ficam maiúsculas.

**Example:**

TitleCase (john smith) = John Smith

*Trim*

| Função | Parâmetros | Valor retornado |
|--------|------------|-----------------|
| Trim   | String s   | String          |

Remove todos os espaços em branco do início ao fim da string *s*

**Example:**

Trim ( ABC ) = ABC

*UpperCase*

| Função    | Parâmetros | Valor retornado |
|-----------|------------|-----------------|
| UpperCase | String s   | String          |

Converte todos os caracteres da string *s* para maiúsculo e retorna um resultado.

**Example:**

UpperCase(abc) = ABC





Universidade Corporativa CIGAM

[www.cigam.com.br](http://www.cigam.com.br)

[treinamento@cigam.com.br](mailto:treinamento@cigam.com.br)



## Data & Hora

### AddDays

| Função  | Parâmetros                     | Valor retornado |
|---------|--------------------------------|-----------------|
| AddDays | DateTime date,<br>Double value | DateTime        |

Adiciona o número especificado de dias (value) à data date e retorna uma nova data.

#### Exemplo:

AddDays (#6/9/2010#, 1) - #6/10/2010#

### AddHours

| Função   | Parâmetros                     | Valor retornado |
|----------|--------------------------------|-----------------|
| AddHours | DateTime date,<br>Double value | DateTime        |

Adiciona o número especificado de horas (value) à data date e retorna uma nova data.

#### Exemplo:

AddHours (#6/10/2010 1:30#, 1) = #6/10/2010 2:30#

### AddMinutes

| Função            | Parâmetros                     | Valor retornado |
|-------------------|--------------------------------|-----------------|
| Adicionar Minutes | DateTime date,<br>Double value | DateTime        |

Adiciona o número especificado de minutos (value) à data date e retorna uma nova data.

#### Exemplo:

Adicionar Minutes (#6/10/2010 1:32#, 1) = #6/10/2010 1:33#

### AddMonths

| Função    | Parâmetros     | Valor retornado |
|-----------|----------------|-----------------|
| AddMonths | DateTime date, | DateTime        |



Double value

Adiciona o número especificado de meses (value) à data date e retorna uma nova data.

**Exemplo:**

AddMonths (#5/10/2010#, 1) - #6/10/2010#

*AddYears*

| Função   | Parâmetros                  | Valor retornado |
|----------|-----------------------------|-----------------|
| AddYears | DateTime date,<br>int value | DateTime        |

Adiciona o número especificado de anos (value) à data date e retorna uma nova data.

**Exemplo:**

AddYears (#6/10/2009#, 1) - #6/10/2010#

*DateDiff*

| Função   | Parâmetros                        | Valor retornado |
|----------|-----------------------------------|-----------------|
| DateDiff | DateTime date1,<br>DateTime date2 | TimeSpan        |

Retorna o intervalo (número de dias, horas, minutos, segundo) entre duas datas.

**Exemplo:**

DateDiff (#1/2/2009#, #1/1/2009#) = 1.00:00:00

*DateSerial*

| Função     | Parâmetros                         | Valor retornado |
|------------|------------------------------------|-----------------|
| DateSerial | Int year,<br>Int month,<br>Int day | DateTime        |

Cria um novo valor DateTime do ano, mês e dia especificados.



**Exemplo:**

DateSerial (2009, 7, 29) = #7/29/2009#

*Day*

| Função | Parâmetros    | Valor retornado |
|--------|---------------|-----------------|
| Day    | DateTime date | int             |

Obtém o dia do mês (1-31) apresentado pela data especificada.

**Exemplo:**

Day (#6/10/2010#) = 10

*DayOfWeek*

| Função    | Parâmetros    | Valor retornado |
|-----------|---------------|-----------------|
| DayOfWeek | DateTime date | String          |

Obtém o nome do dia da semana apresentado pela data especificada.

**Exemplo:**

DayOfWeek (#6/10/2010#) = **quinta-feira (DESCOBRIR SE OBTEM EM PT OU EN)**

*DayOfYear*

| Função    | Parâmetros    | Valor retornado |
|-----------|---------------|-----------------|
| DayOfYear | DateTime date | int             |

Obtém o dia do ano (1-365) apresentado pela data especificada.

**Exemplo:**

DayOfYear (#7/29/2009#) = 210

*DaysInMonth*

| Função      | Parâmetros             | Valor retornado |
|-------------|------------------------|-----------------|
| DaysInMonth | Int year,<br>Int month | int             |



Retorna o número de dias em um mês e ano especificados.

**Exemplo:**

DaysInMonth (2010, 6) = 30

*Hour*

| Função | Parâmetros    | Valor retornado |
|--------|---------------|-----------------|
| Hour   | DateTime date | int             |

Obtém a hora (0-23) apresentada pela data especificada.

**Exemplo:**

Hour (#6/10/2010 1:30#) = 1

*Minute*

| Função | Parâmetros    | Valor retornado |
|--------|---------------|-----------------|
| Minute | DateTime date | int             |

Obtém os minutos (0-59) apresentados pela data especificada.

**Exemplo:**

Minute (#6/10/2010 1:30#) = 30

*Month*

| Função | Parâmetros    | Valor retornado |
|--------|---------------|-----------------|
| Month  | DateTime date | int             |

Obtém o mês (1-12) apresentado pela data especificada.

**Exemplo:**

Month (#6/10/2010#) = 6



### MonthName

| Função | Parâmetros    | Valor retornado |
|--------|---------------|-----------------|
| Month  | DateTime date | int             |

Obtêm o nome do mês especificado (1-12).

#### Exemplo:

MonthName (1) = Janeiro **(DESCOBRIR SE RETORNA EM PT OU EM)**

### Second

| Função | Parâmetros    | Valor retornado |
|--------|---------------|-----------------|
| Second | DateTime date | int             |

Obtêm os segundos (0-59) apresentados pela data especificada.

#### Exemplo:

Second (#6/10/2010 1:30:05#) = 5

### Year

| Função | Parâmetros    | Valor retornado |
|--------|---------------|-----------------|
| Year   | DateTime date | int             |

Obtêm o ano apresentado pela data especificada.

#### Exemplo:

Year (#6/10/2010#) = 2010

## Formatação

### Format

| Função | Parâmetros                              | Valor retornado |
|--------|---|-----------------|
| Format | String format,<br>Params object [] args | string          |

Substitui o formato do item para uma *string* format especificada com o valor da instancia do objeto correspondente em um vetor de args especificados.

Por exemplo, o seguinte chamado de função:

```
Format("Name = {0}, hours = {1:hh}", myName, DateTime.Now)
```

Contem os seguintes itens formatados: {0} e {1:hh}. Eles serão substituídos com valores do **myName** e do **DateTime.Now**. O resultado se parecerá com o seguinte:

**Nome = Thomas, horas = 12**

Cada item formatado utiliza a seguinte forma:

```
{índice[,alignment][:formatString]}
```

- index – um inteiro baseado em zero que indica qual elemento em uma lista deve ser formatado;
- alignment – um inteiro opcional indicando a largura mínima da região que contém um valor formatado. Se a largura do valor formatado é menor que o alinhamento, então a região é preenchida com espaços. Se o alinhamento é negativo, o valor formatado é justificado pela esquerda na região; se o alinhamento é positivo, o valor formatado é justificado pela direita;
- formatString – uma *string* opcional de especificadores de formato.

A seguinte tabela descreve as *strings* padrões de formatos numéricos.

| Especificador do formato | Nome    | Descrição   |
|--------------------------|---------|---|
| C ou c                   | Moeda   | O número é convertido para uma <i>string</i> que representa a quantidade de dinheiro:<br><br>Format({0:C}, 10) = \$10,00 (TESTAR)                           |
| D ou d                   | Decimal | este formato é suportado somente por números inteiros. O número é convertido para uma <i>string</i> de dígitos decimais (0-9)<br><br>Format({0:D}, 10) = 10 |

|               |             |  |
|---------------|-------------|--|
| <b>E ou e</b> | Científico  | este número é convertido para uma <i>string</i> da forma - d.ddd...E+ddd ou -d.ddd...e+DDD, onde cada d indica um dígito de (0-9).<br>Format({0:E}, 10) = 1,000000E+001  |
| <b>F ou f</b> | Ponto fixo  | O número é convertido para uma <i>string</i> da forma - ddd.ddd.... onde cada d indica um dígito (0-9).<br>Format({0:F}, 10) = 10.00   |
| <b>G ou g</b> | Geral       | O número é convertido para a notação mais compacta.<br><br>Format({0:G}, 10) = 10  |
| <b>N ou n</b> | Número      | O número é convertido para uma <i>string</i> da forma - d,DDD,DDD.ddd..., onde cada d indica um dígito (0-9).<br><br>Format({0:N}, 1234.56) = 1,234.56   |
| <b>P ou p</b> | Por cento   | O número é convertido para uma <i>string</i> que representa um percentual. O número convertido é multiplicado por 100 para ser apresentado com uma porcentagem.<br><br>Format({0:P}, 0.15) = 15.00%  |
| <b>X ou x</b> | Hexadecimal | O número é convertido para uma <i>string</i> de dígitos hexadecimais. O tamanho da letra do especificador do formato indica quando usar caracteres maiúsculos ou minúsculos para dígitos hexadecimais maiores do que 9. Por exemplo, use o X para obter ABCDEF, e x para obter abcdef.<br><br>Format({0:X}, 26) = 1A |

Se você formatar os valores de ponto flutuante, você deve indicar um número de casas decimais depois do formato da *string*:

**Format ({0:C1}, 12.23) = \$12.2**

Se os especificadores padrões de formato numérico não possuem o tipo de formatação que você deseja, você pode usar as *strings* de formatação personalizadas:

| Especificador do formato | Nome                                      | Exemplo                                |
|--------------------------|---|--|
| <b>d</b>                 | Padrão de data curta                      | 8/9/2009                               |
| <b>D</b>                 | Padrão de data longa                      | Domingo, 9 de Agosto de 2009           |
| <b>f</b>                 | Padrão de data/hora completo (hora curta) | Domingo, 9 de Agosto de 2009 2:44PM    |
| <b>F</b>                 | Padrão de data/hora completo (hora longa) | Domingo, 9 de Agosto de 2009 2:44:01PM |
| <b>g</b>                 | Padrão geral de data/hora (hora curta)    | 8/9/2009 2:44 PM                       |
| <b>G</b>                 | Padrão geral de data/hora (hora longa)    | 8/9/2009 2:44:01 PM                    |





|          |                      |            |
|----------|----------------------|------------|
| <b>t</b> | Padrão de hora curta | 2:44 PM    |
| <b>T</b> | Padrão de hora longa | 2:44:01 PM |

A seguinte tabela descreve os especificadores customizáveis de data/hora e os resultados que eles produzem.

| <b>Especificador do formato</b> | <b>Descrição</b>   |
|---------------------------------|--|
| <b>d</b>                        | Exibe o dia atual do mês, medido como um número entre 1 e 31, inclusivamente. Se o dia possuir somente um dígito (1-9), ele será exibido somente com um único dígito.    |
| <b>dd</b>                       | Exibe o dia atual do mês, medido como um número entre 1 e 31, inclusivamente. Se o dia possuir somente um dígito (1-9), será colocado um 0 antes dele (01-09).           |
| <b>ddd</b>                      | Exibe o nome do dia abreviado.   |
| <b>dddd</b>                     | Exibe o nome do dia completo.  |
| <b>f ou F</b>                   | Exibe o dígito mais significativo da fração dos segundos.  |
| <b>h</b>                        | Exibe a hora em um intervalo de 1-12. Se a hora possuir somente um dígito (1-9), ela será exibida com um dígito somente.   |
| <b>hh</b>                       | Exibe a hora em um intervalo de 1-12. Se a hora possuir somente um dígito (1-9), será colocado um 0 antes dele (01-09).  |
| <b>H</b>                        | Exibe a hora em um intervalo de 0-23. Se a hora possuir somente um dígito (1-9), ela será exibida com um dígito somente.   |
| <b>HH</b>                       | Exibe a hora em um intervalo de 0-23. Se a hora possuir somente um dígito (1-9), será colocado um 0 antes dele (01-09).  |
| <b>m</b>                        | Exibe o minuto em um intervalo de 0-59. Se o minuto possuir somente um dígito (0-9), ele será exibido com um dígito somente.   |
| <b>mm</b>                       | Exibe o minuto em um intervalo de 0-59. Se o minuto possuir somente um dígito (0-9), será colocado um 0 antes dele.  |
| <b>M</b>                        | Exibe um mês, mensurado como um número entre 1 e 12, inclusive. Se o mês possui somente um dígito (1-9), ele será exibido com um dígito somente.                         |
| <b>MM</b>                       | Exibe um mês, mensurado como um número entre 1 e 12, inclusive. Se o mês possui somente um dígito (1-9), será colocado um 0 antes dele.                                  |
| <b>MMM</b>                      | Exibe o nome abreviado do mês.   |
| <b>MMMM</b>                     | Exibe o nome completo do mês.  |
| <b>s</b>                        | Exibe os segundos em um intervalo de 0-59. Se o segundo possuir somente um dígito (0-9), ele será exibido com um dígito somente.   |
| <b>ss</b>                       | Exibe os segundos em um intervalo de 0-59. Se o segundo possuir somente um dígito (0-9), será colocado um 0 antes dele.  |
| <b>t</b>                        | Exibe o primeiro caractere da designação A.M./P.M.   |
| <b>tt</b>                       | Exibe a designação A.M./P.M.   |
| <b>y</b>                        | Exibe o ano com no máximo dois dígitos. Os dois primeiros dígitos do ano são omitidos. Se o ano possuir somente um dígito (1-9), ele será exibido com um dígito somente. |



|             |   |
|-------------|---|
| <b>yy</b>   | Exibe o ano com no máximo dois dígitos. Os dois primeiros dígitos do ano são omitidos. Se o ano possuir somente um dígito (1-9), será colocado um 0 antes dele (01-09).   |
| <b>yyyy</b> | Exibe o ano, incluindo o século. Se o ano possuir menos do que quatro dígitos, então ele será completado com zeros até possuir 4 dígitos. Exemplo: 99 (0099);1(0001).   |
| <b>z</b>    | Exibe o deslocamento do fuso horário em relação ao fuso horário atual, apenas em horas inteiras. O deslocamento é sempre exibido com um sinal à esquerda (zero é exibido como +0, indicando horas a leste de Greenwich (+) ou a oeste (-). O intervalo dos valores é -12 até +13. Se o deslocamento for somente um dígito (0-9), ele será exibido com um dígito somente mais o sinal à esquerda correto (+1). |
| <b>zz</b>   | Exibe o deslocamento do fuso horário em relação ao fuso horário atual, apenas em horas inteiras. O deslocamento é sempre exibido com um sinal à esquerda (zero é exibido como +00, indicando horas a leste de Greenwich (+) ou a oeste (-). O intervalo dos valores é -12 até +13. Se o deslocamento for somente um dígito (0-9), será colocado um 0 antes mais o sinal correto (+01).                        |
| <b>zzz</b>  | Exibe o deslocamento do fuso horário em relação ao fuso horário atual, apenas em horas inteiras. O deslocamento é sempre exibido com um sinal à esquerda (zero é exibido como +00:00, indicando horas a leste de Greenwich (+) ou a oeste (-). O intervalo dos valores é -12:00 até +13:00. Se o deslocamento for somente um dígito (0-9), será colocado um 0 antes mais o sinal correto (+01).               |
| <b>:</b>    | Separador de tempo.   |
| <b>/</b>    | Separador de data.  |

Exemplos de uso:

```
Format("{0:$#,##0.00}", 1024.25) = "$1,024.25"
```

```
Format("{0:00%}", 0.25) = "25%"
```

```
Format("{0:$#,##0.00;($#,##0.00);Zero}", 1024.25) = "$1,024.25"
```

```
Format("{0:$#,##0.00;($#,##0.00);Zero}", -1024.25) = "($1,024.25)"
```

```
Format("{0:$#,##0.00;($#,##0.00);Zero}", 0) = "Zero"
```

### *FormatCurrency*

| Função                | Parâmetros   | Valor retornado |
|-----------------------|--------------|-----------------|
| <b>FormatCurrency</b> | object value | string          |

Formata um valor especificado como moeda, usando as configurações regionais do Windows.

**Exemplo:**

```
FormatCurrency(1.25) = $1.25
```

| Função | Parâmetros | Valor retornado |
|--------|------------|-----------------|
|--------|------------|-----------------|



|                       |                                    |        |
|-----------------------|------------------------------------|--------|
| <b>FormatCurrency</b> | object value,<br>int decimalDigits | string |
|-----------------------|------------------------------------|--------|

Formata o valor especificado como moeda. O parâmetro **decimalDigits** indica quantas casas serão exibidas a direita do decimal.

**Exemplo:**

FormatCurrency(1.25, 1) = \$1.3

### *FormatDateTime*

| <b>Função</b>         | <b>Parâmetros</b> | <b>Valor retornado</b> |
|-----------------------|-------------------|------------------------|
| <b>FormatDateTime</b> | DateTime value    | string                 |

Formata o valor especificado como data/hora, usando as configurações regionais do Windows. Esta função não inclui valores neutros na *string* resultante.

**Exemplo:**

FormatDateTime(#1/1/2009#) – 01/01/2009

FormatDateTime(#1/1/2009 1:20#) = 01/01/2009 1:20:00 AM

FormatDateTime(#1:30#) = 1:30:00 AM.

| <b>Função</b>         | <b>Parâmetros</b>                | <b>Valor retornado</b> |
|-----------------------|----------------------------------|------------------------|
| <b>FormatDateTime</b> | DateTime value,<br>string format | string                 |

Formata o valor especificado como data/hora, usando o formato nomeado especificado no *format*. Os valores válidos para este parâmetro são:

Long Date

Short Date

Long Time

Short Time

**Exemplo:**

FormatDateTime(#1/1/2009 1:30#, Long Date) = Thursday, January 01, 2009



`FormatDateTime(#1/1/2009#, Short Date) = 01/01/2009`

`FormatDateTime(#1:30#, Short Time) = 1:30:00 AM`

`FormatDateTime(#1:30#, Long Time) = 1:30:00 AM`

### *FormatNumber*

| Função              | Parâmetros   | Valor retornado |
|---------------------|--------------|-----------------|
| <b>FormatNumber</b> | object value | string          |

Formata o valor especificado como um número, usando as configurações regionais do Windows.

#### **Example:**

`FormatNumber (1234.56) = 1,234.56`

| Função              | Parâmetros                         | Valor retornado |
|---------------------|------------------------------------|-----------------|
| <b>FormatNumber</b> | object value,<br>int decimalDigits | string          |

Formata o valor especificado como um número. O parâmetro `decimalDigits` indica quantas casas serão exibidas após a vírgula.

#### **Exemplo:**

`FormatNumber(1234.56, 1) = 1,234.6`

### *FormatPercent*

| Função               | Parâmetros   | Valor retornado |
|----------------------|--------------|-----------------|
| <b>FormatPercent</b> | object value | string          |

`FormatPercent(0.15) = 15.00%`

| Função               | Parâmetros                         | Valor retornado |
|----------------------|------------------------------------|-----------------|
| <b>FormatPercent</b> | object value,<br>int decimalDigits | string          |

Formata o valor especificado com uma porcentagem. O parâmetro `decimalDigits` indica quantas casas serão exibidas após a vírgula.

`FormatPercent(0.15) = 15%`



Universidade Corporativa CIGAM

[www.cigam.com.br](http://www.cigam.com.br)

[treinamento@cigam.com.br](mailto:treinamento@cigam.com.br)



## Conversão

### *ToBoolean*

| Função    | Parâmetros   | Valor retornado |
|-----------|--------------|-----------------|
| ToBoolean | object value | bool            |

Converte o valor especificado para **booleano**.

#### **Exemplo:**

ToBoolean(1) = true

ToBoolean(0) = false

### *ToByte*

| Função | Parâmetros   | Valor retornado |
|--------|--------------|-----------------|
| ToByte | object value | byte            |

Converte o valor especificado para byte.

#### **Exemplo:**

ToByte(55) = 55

### *ToChar*

| Função | Parâmetros   | Valor retornado |
|--------|--------------|-----------------|
| ToChar | object value | char            |

Converte o valor especificado para caractere.

#### **Exemplo:**

ToChar(65) = A

### *ToDateTime*

| Função     | Parâmetros   | Valor retornado |
|------------|--------------|-----------------|
| ToDateTime | object value | DateTime        |



Converte o valor especificado para data/hora.0

**Exemplo:**

ToDateTime(1/1/2009) = #1/1/2009#

*ToDecimal*

| Função    | Parâmetros   | Valor retornado |
|-----------|--------------|-----------------|
| ToDecimal | object value | decimal         |

Converte o valor especificado para decimal.

**Exemplo:**

ToDecimal(1) = 1m

ToDecimal(1) = 1m

*ToDouble*

| Função   | Parâmetros   | Valor retornado |
|----------|--------------|-----------------|
| ToDouble | object value | double          |

Converte o valor especificado para **double**.

**Exemplo:**

ToDouble(1) = 1

ToDouble(1) = 1

*ToInt32*

| Função  | Parâmetros   | Valor retornado |
|---------|--------------|-----------------|
| ToInt32 | object value | int             |

Converte o valor especificado para inteiro.

**Exemplo:**

ToInt32(1f) = 1



ToInt32(1) = 1

#### *ToRoman*

| Função  | Parâmetros   | Valor retornado |
|---------|--------------|-----------------|
| ToRoman | object value | string          |

Converte o valor especificado para sua representação em romano. O valor deve estar entre 1-3998.

#### **Exemplo:**

ToRoman(9) = IX

#### *ToSingle*

| Função   | Parâmetros   | Valor retornado |
|----------|--------------|-----------------|
| ToSingle | object value | float           |

Converte o valor especificado para **float**.

#### **Exemplo:**

ToSingle(1m) = 1f

ToSingle(1) = 1f

#### *ToStrings*

| Função   | Parâmetros   | Valor retornado |
|----------|--------------|-----------------|
| ToString | object value | string          |

Converte o valor especificado para *string*.

#### **Exemplo:**

ToString(false) = False

ToString(DateTime.Now) = 08/09/2009 4:45:00 PM

#### *ToWords*

| Função  | Parâmetros   | Valor retornado |
|---------|--------------|-----------------|
| ToWords | object value | string          |





Converte o valor especificado para palavras.

**Exemplo:**

ToWords(1024.25) = One thousand and twenty-four dollars and 25 cents (**DESCOBRIR SE É EM PT OU EN**)

| Função  | Parâmetros                           | Valor retornado |
|---------|--------------------------------------|-----------------|
| ToWords | object value,<br>string currencyName | string          |

Converte a moeda especificada para palavras. O parâmetro **currencyName** indica a moeda. Os valores válidos para este parâmetro são:

USD

EUR

GBP

**Exemplo:**

ToWords(1024.625, EUR) = One thousand.... euros... (**DESCOBRIR SE É PT OU EN**)

| Função  | Parâmetros                                  | Valor retornado |
|---------|---|-----------------|
| ToWords | object value,<br>string one,<br>string many | string          |

Converte o valor inteiro especificado para palavras. O parâmetro **one** contém o nome na forma singular; o parâmetro **many** contém o nome na forma plural.

**Example:**

ToWords(124, page, pages) = One hundred and twenty-four pages

ToWords(1, page, pages) = One page

**ToWordsEnGb**

| Função      | Parâmetros   | Valor retornado |
|-------------|--------------|-----------------|
| ToWordsEnGb | object value | string          |



Converte o valor da moeda especificada para palavras em inglês britânico. Há as seguintes diferenças entre esta função e a ToWords:

- a moeda GBP é usada por padrão;
- palavras diferentes usadas quando converte-se para milhares e trilhares.

**Exemplo:**

ToWords(121) = One hundred and twenty-one pounds and 00 pence

| Função      | Parâmetros                           | Valor retornado |
|-------------|--------------------------------------|-----------------|
| ToWordsEnGb | object value,<br>string currencyName | string          |

Converte a moeda especificada para palavras em inglês britânico. O parâmetro **currencyName** indica a moeda. Os valores válidos para este parâmetro são:

USD

EUR

GBP

**Exemplo:**

ToWordsEnGb(1024.25, EUR) = One thousand and twenty-four euros and 25 cents

| Função      | Parâmetros                                  | Valor retornado |
|-------------|---|-----------------|
| ToWordsEnGb | object value,<br>string one,<br>string many | string          |

Converte o valor inteiro especificado para palavras em inglês britânico. O parâmetro **one** contém o nome no singular; o parâmetro **many** contém o nome no plural.

*ToWordsRu*

| Função    | Parâmetros   | Valor retornado |
|-----------|--------------|-----------------|
| ToWordsRu | object value | string          |

Converte o valor da moeda especificada para palavras em russo.

**Exemplo:**

ToWordsRu(1024.25) = 25 (**DESCOBRIR SE É EM PT OU EN**)

| Função    | Parâmetros                           | Valor retornado |
|-----------|--------------------------------------|-----------------|
| ToWordsRu | object value,<br>string currencyName | string          |

Converte a moeda especificada para palavras em russo. O parâmetro **currencyName** indica a moeda. Os valores válidos para este parâmetro são:

RUR

UAH

USD

EUR

**Exemplo:**

ToWordsRu(1024.625, EUR) = One thousand.... euros...

| Função    | Parâmetros   | Valor retornado |
|-----------|--|-----------------|
| ToWordsRu | object value,<br>boot male,<br>string one,<br>string two,<br>string many | string          |

Converte o valor inteiro especificado para palavras em russo. O parâmetro **male** indica o gênero do nome. Os parâmetros **one**, **two** e **Five** contém uma forma do nome usado com os números 1, 2 e 5.

**Example:**

//the ...

//the ...

**Program Flow****Choose**

| Função | Parameters    | Return value |
|--------|---------------|--------------|
| Choose | double index, | object       |

**params object [] choice**

Retorna um elemento do vetor **choice** com o índice especificado no parâmetro **index**. O primeiro elemento do vetor tem o index 1.

**Exemplo:**

Choose(2, one, two, three) = two

**IIf**

| Função | Parameters   | Return value |
|--------|--|--------------|
| IIf    | bool expression,<br>object truePart,<br>object falsePart | object       |

Retorno o valor truePart, se a expression é verdadeira. Caso contrário, retorna o valor falsePart .

**Exemplo:**

IIf(2 < 5, true, false) = false

**Switch**

| Função | Parameters                   | Return value |
|--------|------------------------------|--------------|
| Switch | params object [] expressions | object       |

O argumento suprimido das *expressions* consiste de pares de expressões e valores. A função **Switch** avalia as expressões ímpares do menor ao maior índice, e retorna o valor par associado com a primeira expressão que for *true*.

```
// Retorna um dos valores seguintes: a maior do que 0,  
// a menor do que 0, a igual a 0, dependendo do valor de "a".  
Switch(  
a > 0, "à greater than 0",  
a < 0, "à less than 0",  
a == 0, "à equals to 0")
```

**Totais**

Em muitos relatórios, poderá haver necessidade de nós termos que mostrar informações totais: soma de um grupo, número de linhas em uma lista, entre outros. O **CIGAM Report** usa os totais para realizar estas tarefas. Para um total, você precisa indicar os seguintes parâmetros:



- o tipo da função total;
- a expressão, que deve ser calculada. Para a função Count, você não precisa indicar a expressão;
- a condição. A função será calculada se a condição for satisfeita. Não é obrigatório definir a condição;
- a banda de dados, para a qual a função será processada;
- a banda, na qual o valor total será impresso.

A lista de funções de total é dada abaixo:

| Função  | Descrição                          |
|---------|------------------------------------|
| Sum     | Calcula a soma da expressão        |
| Min     | Calcula o menor valor da expressão |
| Max     | Calcula o maior valor da expressão |
| Average | Calcula o valor médio da expressão |
| Count   | Retorna o número de linhas         |

### Criando um Total

Nós iremos ver o uso da função total com um exemplo. Vamos criar um relatório **Mestre-Detalhe** que utiliza duas tabelas – **Categorias** e **Produtos**.



O relatório preparado será o seguinte:



## Beverages

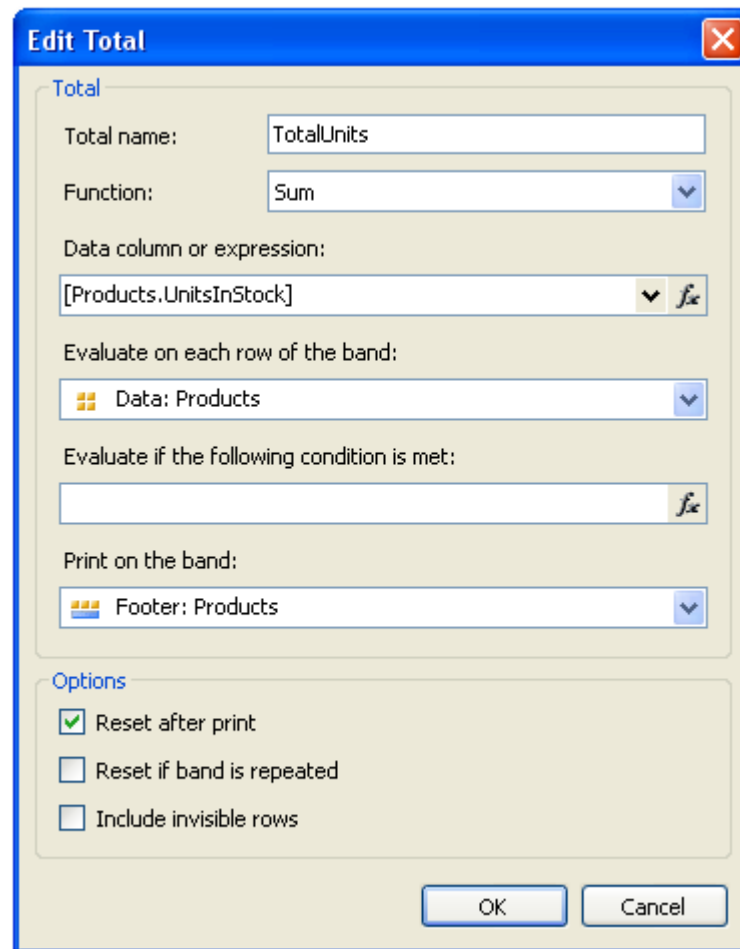
| Product Name              | Units In Stock |
|---------------------------|----------------|
| Chai                      | 39             |
| Chang                     | 17             |
| Chartreuse verte          | 69             |
| Côte de Blaye             | 17             |
| Guaraná Fantástica        | 20             |
| Ipoh Coffee               | 17             |
| Lakkalikööri              | 57             |
| Laughing Lumberjack Lager | 52             |
| Outback Lager             | 15             |
| Rhönbräu Klosterbier      | 125            |
| Sasquatch Ale             | 111            |
| Steeleye Stout            | 20             |

## Condiments

| Product Name                     | Units In Stock |
|----------------------------------|----------------|
| Aniseed Syrup                    | 13             |
| Chef Anton's Cajun Seasoning     | 53             |
| Chef Anton's Gumbo Mix           | 0              |
| Genen Shouyu                     | 39             |
| Grandma's Boysenberry Spread     | 120            |
| Gula Malacca                     | 27             |
| Louisiana Fiery Hot Pepper Sauce | 76             |
| Louisiana Hot Spiced Okra        | 4              |
| Northwoods Cranberry Sauce       | 6              |
| Original Frankfurter grüne Soße  | 32             |
| Sirop d'érable                   | 113            |
| Vegie-spread                     | 24             |

Vamos adicionar um total neste relatório, o qual irá imprimir a quantidade total de unidades em estoque por cada categoria – a soma da coluna de dados **UnitsInStock**. O total será impresso na banda **Rodapé de Dados**.

Para imprimir um valor total, você precisa criá-lo primeiro. Para isso, pressione o botão **Action** na janela Dados e escolha o item **Novo Total**. Outro método é clicar com o botão direito no elemento **Totais** na **árvore de dados** e escolha o item de menu **Novo Total**. Você verá a janela de edição de total.



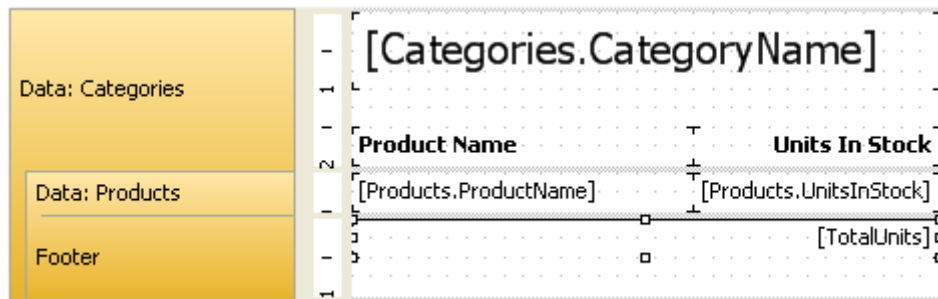
Primeiramente, você deverá indicar o nome do total. Você se referirá ao total pelo seu nome, então nomeie-o de tal forma que seja fácil entender o que ele calcula. Vamos chamar o nosso total de **TotalUnits**.

Então, escolhemos a função **Sum** para o total.

Agora nós temos que indicar o intervalo de dados em que será calculado o total. Para isso, no campo **Avaliar em cada banda de dados:**, nós escolheremos a banda na qual o total será impresso, ou seja, a banda **Rodapé de Dados**.

Feche o editor pressionando o botão OK. Você verá o novo total aparecer na janela Dados.

Agora você pode arrastá-lo para dentro do relatório:



Quando nós gerarmos o relatório, veremos o seguinte:

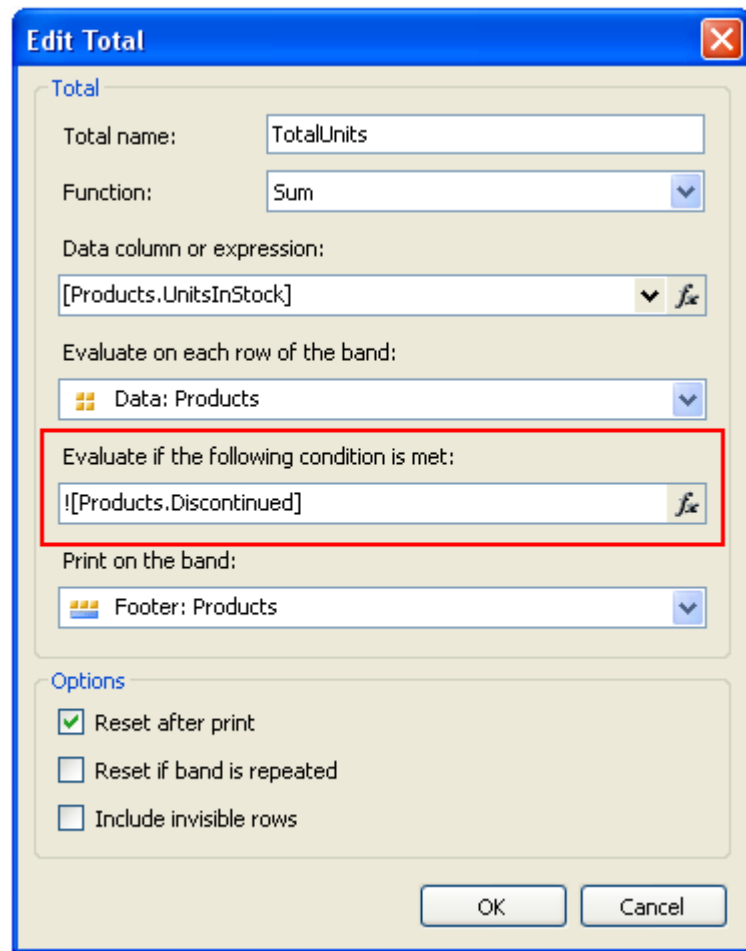
| Beverages                 |                | Condiments                       |                |
|---------------------------|----------------|----------------------------------|----------------|
| Product Name              | Units In Stock | Product Name                     | Units In Stock |
| Chai                      | 39             | Aniseed Syrup                    | 13             |
| Chang                     | 17             | Chef Anton's Cajun Seasoning     | 53             |
| Chartreuse verte          | 69             | Chef Anton's Gumbo Mix           | 0              |
| Côte de Blaye             | 17             | Genen Shouyu                     | 39             |
| Guaraná Fantástica        | 20             | Grandma's Boysenberry Spread     | 120            |
| Ipoh Coffee               | 17             | Gula Malacca                     | 27             |
| Lakkalikööri              | 57             | Louisiana Fiery Hot Pepper Sauce | 76             |
| Laughing Lumberjack Lager | 52             | Louisiana Hot Spiced Okra        | 4              |
| Outback Lager             | 15             | Northwoods Cranberry Sauce       | 6              |
| Rhönbräu Klosterbier      | 125            | Original Frankfurter grüne Soße  | 32             |
| Sasquatch Ale             | 111            | Sirop d'érable                   | 113            |
| Steeleye Stout            | 20             | Vegie-spread                     | 24             |
|                           | 559            |                                  | 507            |

### Totais condicionais

No exemplo anterior, o total foi calculado para todas as linhas de dados. Nós podemos limitar o intervalo indicando uma condição no editor de total. O total será calculado para somente aquelas linhas, as quais a condição retornar **true**.

Por exemplo, nós podemos definir as seguintes condições:





Isto significa que o total deverá ser calculado para aqueles produtos os quais a **flag Discontinued** não estiver definida.

### Totais contínuos

No nosso exemplo, os totais foram reiniciados depois de imprimir a banda **Rodapé de Dados**. Isto ocorreu porque nós indicamos, no editor de total, que é necessário reiniciá-lo após imprimi-lo. Como resultados, cada categoria imprimiu seu próprio valor total.

Se nós desmarcarmos o *checkbox* **Reset after print**, o total não será reiniciado após a impressão. Isso é o que chamamos de execução total.

Se você precisar imprimir dois tipos de totais ao mesmo tempo – totais comuns e totais contínuos – crie mais um total com configurações similares e desmarque a **flag Reset after print**.

## Totais de página

Para criar um total que será impresso no rodapé da página, você terá que indicar o rodapé da página no campo **Imprimir na banda:** .

## Parâmetros do relatório

Você pode definir os parâmetros no relatório. Parâmetro é uma variável e seu valor pode ser definido tanto dentro do relatório quanto fora (um programa, abrir um relatório pode transferir valores de parâmetros para ele. Veja detalhes no **Programmers manual**). Um parâmetro pode ser usada em expressões e ser mostrado nos objetos do relatório como um objeto Texto.

Os métodos mais comuns de utilizar parâmetros:

- A filtragem de dados por condições definidas no parâmetro;
- Imprimir o valor do parâmetro no relatório.

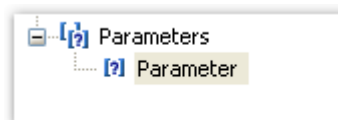
Um parâmetro possui as seguintes propriedades:

| Propriedade | Descrição  |
|-------------|--|
| Name        | O nome do parâmetro pode ter qualquer símbolo, exceto o ponto .  |
| DataType    | Tipo de dados do parâmetro.  |
| Expression  | Expressões que retornam o valor do parâmetro. Mais detalhes sobre expressões podem ser encontrados no capítulo <b>Expressions</b> . Esta expressão será processada quando chamarmos o parâmetro. |
| Value       | Valor do parâmetro. Esta propriedade não está disponível no <b>editor</b> e pode ser preenchida programaticamente.   |

Você definiu as propriedades **Name** e **DataType**. A propriedade **Expression** pode ficar vazia. Neste caso o valor do parâmetro deve ser passado programaticamente.

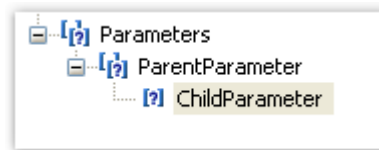
## Criando um parâmetro

Para criar um parâmetro, selecione o elemento **Parameters** na janela Dados. Clique com o botão direito nele e escolha o item **Novo parâmetro** no menu de contexto:



Pressione F2 e dê o nome do parâmetro, então vá à janela Propriedades e defina a propriedade **DataType** do parâmetro.

Parâmetros não podem ser aninhados. Para criar um parâmetro aninhado, selecione o parâmetro pai, clique com o botão direito sobre ele e selecione o item **Novo parâmetro** no menu de contexto:



Você pode se referir à ambos, parâmetro pai e aninhado. O nível de aninhamento não é limitado.

### Usando parâmetros no relatório

Você pode se referir à um parâmetro de uma expressão usando colchetes:

**[Parameter name]**

Para um parâmetro aninhado você precisa se referir usando este método:

**[Parent parâmetro.Child parâmetro]**

Uma vez que o parâmetro tem um tipo definitivo (ele é dado na propriedade **DataType**), então com parâmetros, você pode realizar aquelas ações que são permitidas para tipos de dados. Então, os parâmetros do tipo *string* pode ser usados em expressões da seguinte maneira:

**[StringParameter].Substring(0,2)**

Vamos ver um exemplo do uso de parâmetros. Imagine que nós tenhamos um relatório que imprima a tabela **Employees**. Nós queremos modificar o relatório para imprimir informações sobre o empregado com o número indicado. Para fazer isso, nós precisamos filtrar os dados na coluna de dados **EmployeeID**. Crie um parâmetro com o nome **EmployeeID**. Indique o tipo do parâmetro – **Int32**, já que exatamente este tipo tem a coluna de dados **EmployeeID**. Para filtrar um empregado com o ID indicado, nós precisaremos entrar no editor da banda Dados e indicar a seguinte expressão na guia **Filter**:

**[Employees.EmployeeID] == [EmployeeID]**

Para passar o valor de um parâmetro do seu programa para o relatório, use o seguinte código:

**report1.SetParameterValue[EmployeeID, 2]**

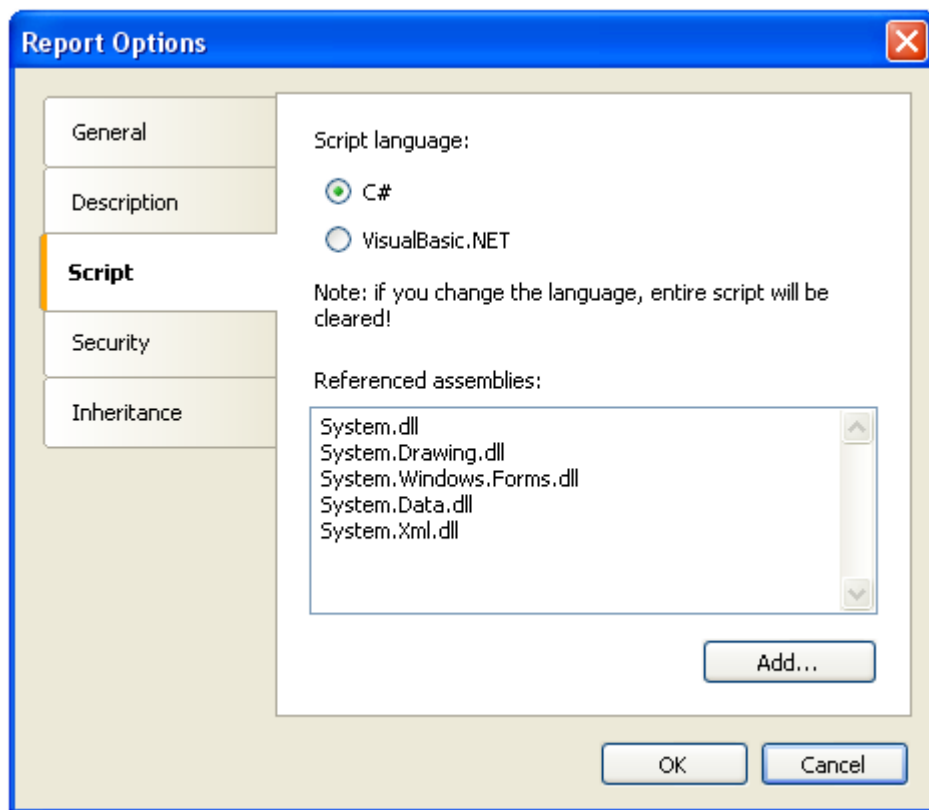
## Expressões

As expressões são utilizadas em muitos lugares no **CIGAM Report**. Por exemplo, o objeto Texto pode conter expressões entre colchetes.

Uma expressão é um código em C# ou linguagem VB.Net, a qual retorna qualquer valor. Por exemplo:

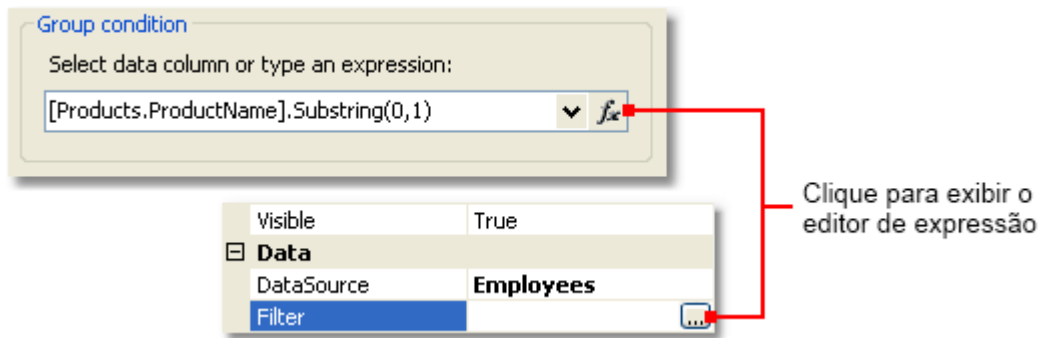
**2+2**

Uma expressão deve ser escrita em uma linguagem escolhida, assim como um **script** no relatório. Por padrão, é C#. Você pode mudar a linguagem no menu **Relatório > Opções...**, escolhendo o elemento **Script** na janela.

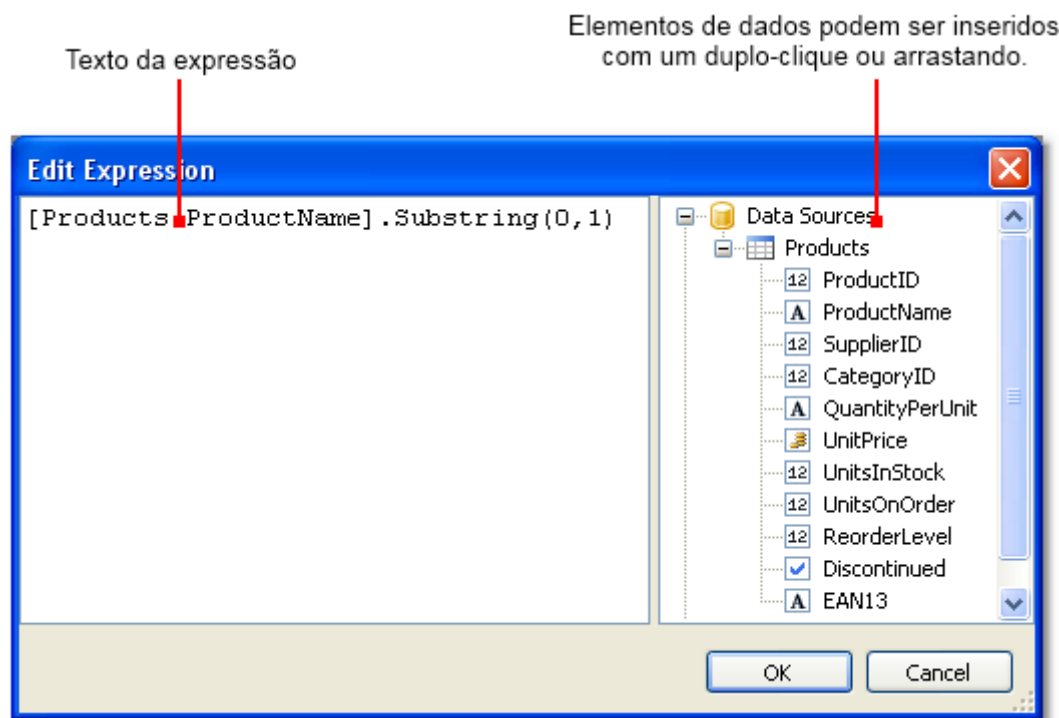


## Editor de expressão

Para escrever uma expressão rapidamente, use o editor de expressão. Ele pode ser aberto em alguns lugares da interface do **CIGAM Report**, onde você pode digitar uma expressão:



O editor de expressão apresenta uma janela onde você pode escrever uma expressão e inserir alguns elementos de dados:



## Referência para objetos do relatório

Para referir-se a objetos do relatório, use o nome do objeto. O seguinte exemplo retornará a altura do objeto **Text1**:

**Text1.Height**

Para se referir a propriedades do relatório, use a variável **Report**. O seguinte exemplo retorna o nome do arquivo do qual o relatório foi carregado.

**Report.FileName**

Além disso, você pode se referir a propriedades de objetos aninhados. O seguinte exemplo retornará o nome de um relatório:

## Report.ReportInfo.Name

### Usando funções .Net

Você pode qualquer objeto .Net nas expressões. O seguinte exemplo demonstra o uso da função Max:

**Math.Max(5,10)**

Por padrão o relatório usa as seguintes bibliotecas .Net:

**System.dll**

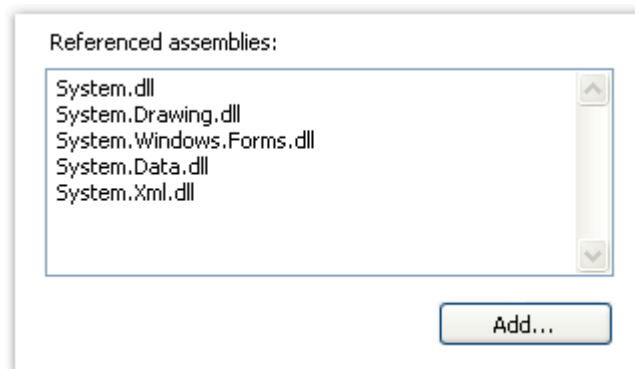
**System.Drawing.dll**

**System.Windows.Forms.dll**

**System.Data.dll**

**System.Xml.dll**

Você tem acesso a todos os objetos .Net declarados nestas bibliotecas. Se você precisar aceitar outra biblioteca, adicione o seu nome na lista de bibliotecas do relatório. Você pode fazer isso através do menu **Relatório > Opções...**, escolhendo o elemento **Script** em uma janela:



Por exemplo, se você quiser usar uma função no seu relatório que tenha sido declarada na sua aplicação, adicione a biblioteca da aplicação (.exe ou .dll) na lista de bibliotecas do relatório. Depois disso, você pode chamar a função usando o **namespace** da sua aplicação. Por exemplo, se a seguinte função é definida na aplicação:

`namespace Demo`

```
{  
    public static class MyFunctions  
    {  
        public static string Func1()  
    }  
}
```



```
{  
    return "Hello!";  
}  
}
```

Você pode usá-la no seu relatório da seguinte forma:

**Demo.MyFunctions.Func1()**

Se você adicionar a linha **using Demo** no topo do script do relatório, isso permitirá que você diminua a sintaxe:

**MyFunctions.Func1()**

Para se referir à função ou à variável que foi definida em um script, somente use o seu nome:

**myPrivateVariableThatIHaveDeclaredInScript**

**MyScriptFunction()**

---

Você pode usar em uma função somente aquelas funções que retornam um valor.

---

## Referência a elementos de dados

Além dos elementos padrões da linguagem, você pode usar os seguintes elementos do relatório nas expressões:

- colunas de fontes de dados;
- variáveis de sistema;
- valores totais;
- parâmetros do relatório.

Todos estes elementos estão contidos na janela Dados. Veja mais detalhes no capítulo **Data**. Qualquer um destes elementos pode ser usado em uma expressão, colocando entre colchetes. Por exemplo:

**[Page] + 1**

esta expressão retorna o número da próxima página impressa. Uma variável de sistema **Page**, que retorna o número da página atual do relatório, é usada na expressão. Ele fica entre colchetes.

## Referência a fontes de dados

Para referência as colunas das fontes de dados, o seguinte formato é usado:



### [DataSource.Column]

O nome da fonte é separado do nome da coluna por um ponto. Por exemplo:

### [Employees.FirstName]

O nome da fonte pode ser em maiúsculo, se nós nos referirmos a fonte de dados usando uma relação. Veja mais detalhes na seção **Data**. Por exemplo, é assim que você pode se referir para uma coluna relacionada da fonte de dados:

### [Produtos.Categorias.NomeCategoria]

Vamos ver o seguinte exemplo do uso de colunas em uma expressão:

### [Employees.FirstName] + + [Employees.LastName]

Aqui deve ser notado que: toda a coluna tem um tipo definitivo de dados, o qual é definido na propriedade **DataType** (você pode vê-la na janela Propriedades se você tiver escolhido a coluna de dados na janela Dados anteriormente). A maneira como uma coluna pode ser utilizada em uma expressão depende do seu tipo. Para instância, no exemplo mencionado acima, ambas colunas – primeiro nome e sobrenome – são do tipo string e é por isso que elas podem ser usadas de uma certa maneira. No exemplo seguinte, nós iremos tentar usar a coluna **Employees.Age** do tipo número, para gerar um erro:

### [Employee.FirstName] + + [Employees.Age]

O erro ocorre porque você nunca deve misturar strings com números. Para isso, você precisa converter o número para uma string:

### [Employee.FirstName] + + [Employees.Age].ToString()

Neste caso, nos referimos a coluna **Employees.Age** como se ela fosse uma variável inteira. E ela é. Nós sabemos que todas as expressões são compiladas. Todas as coisas não padrões (como referir-se a colunas de dados), do ponto de vista do compilador, são convertidas para outro tipo, o qual é não é compreendido pelo compilador. Então, a última expressão será mudada da seguinte forma:

```
(string)(Report.GetColumnValue("Employees.FirstName")) + " " +  
(int)(Report.GetColumnValue("Employees.Age")).ToString()
```

Como visto, o **CIGAM Report** muda a referência às colunas de dados da seguinte maneira:

```
[Employees.FirstName] -->  
(string) (Report.GetColumnValue("Employees.FirstName"))  
[Employees.Age] --> (int) (Report.GetColumnValue("Employees.Age"))
```



Ou seja, nós podemos usar a coluna de dados em uma expressão como se ela fosse uma variável com um tipo definitivo. Por exemplo, a seguinte expressão retornará a primeira letra do nome do empregado:

`[Employees.FirstName].Substring(0,1)`

## Referência às variáveis de sistema

Você pode usar nas expressões as seguintes variáveis de sistema (elas são acessíveis na janela Dados):

| Variável          | Tipo de dados .Net | Descrição   |
|-------------------|--------------------|---|
| <b>Date</b>       | DateTime           | Data e hora do início do relatório.   |
| <b>Page</b>       | int                | Número da página atual.   |
| <b>TotalPages</b> | int                | Número total de páginas no relatório. Para usar esta variável, você precisa habilitar a opção do relatório passa duas vezes. Você pode fazer isso no menu <b>Relatório &gt; Properties...</b> |
| <b>PageN</b>      | string             | Número da página na forma: Page N.  |
| <b>PageNofM</b>   | string             | Número da página na forma: Page N of M.   |
| <b>Row#</b>       | int                | Número da linha de dados dentro do grupo. Este valor é reiniciado no início de um novo grupo.   |
| <b>AbsRow#</b>    | int                | Número absoluto da linha de dados. Este valor nunca é reiniciado no início de um novo grupo.  |

Toda variável tem um tipo definitivo de dados. E isso depende de como ele será usado na expressão. Aqui temos um exemplo de uma expressão onde a data está sendo utilizada:

`[Date].Year`

esta expressão retorna o ano atual. Como a variável **Date** é do tipo DateTime, nós podemos nos referir a sua propriedade **Year**. Nós podemos igualmente obter o mês atual (`[Date].Month`).

O **CIGAM Report** converte a referência a variáveis de sistema da seguinte forma (por exemplo, a variável **Date**):

`((DateTime)Report.GetVariableValue("Date"))`

## Referência a valores totais

Para se referir a um valor total, use o seu nome:

`[TotalSales]`

O **CIGAM Report** converte a referência aos totais da seguinte forma:



```
Report.GetTotalValue("TotalSales")
```

Como você pode ver o tipo de dados não é usado aqui. Isto é porque o valor total é do tipo **FastReport.Variant**. Isto pode ser usado diretamente em quaisquer expressões porque ele é automaticamente convertido para qualquer tipo. Por exemplo:

```
[TotalSales] * 0.2f
```

## Referência aos parâmetros do relatório

Para se referir aos parâmetros do relatório, você deve usar os nomes deles:

```
[Parameter1]
```

Os parâmetros podem ser aninhados. Neste caso, você deve usar ambos nomes dos parâmetros pai e filho da seguinte forma:

```
[ParentParameter.ChildParameter]
```

Os parâmetros possuem um tipo definitivo de dados. Ele é definido na propriedade **DataType** do parâmetro. A maneira como eles podem ser usados em uma expressão depende do tipo de dados dos parâmetros.

O **CIGAM Report** converte a referência aos parâmetros do relatório da seguinte forma:

```
((string)Report.GetParameterValue("Parameter1"))
```



## Script

*Scripts* são uma programação de linguagem de alto nível, e parte do relatório. *Scripts* podem ser escritos em uma das seguintes **linguagens .NET**:

- C#;
- VisualBasic.NET.

Um *script* é aplicável em muitos lugares. Utilizando-os, você pode:

- Manipular dados, o que não pode ser feito através dos meios regulares da *engine*<sup>4</sup> do CIGAM Report;
- Controlar a impressão de páginas do relatório e de bandas na página;
- Controlar a interação entre elementos em *forms* de diálogo;
- Controlar a formação de objetos de tabelas dinâmicas;
- Diversas outras funções baseadas em *scripts*, dependendo do seu conhecimento da linguagem utilizada.

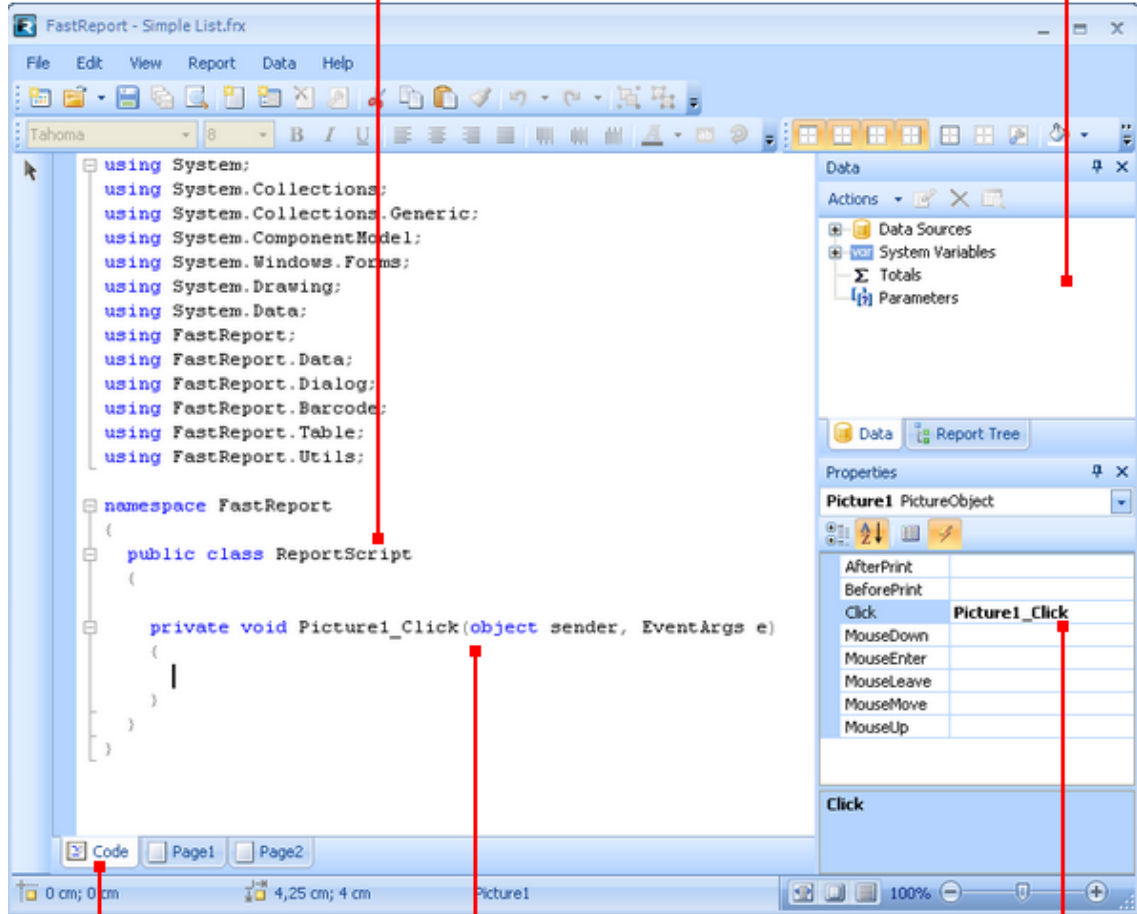
Para visualizar o *script* de um relatório, troque para a aba **Código** no editor:

---

<sup>4</sup> **Engine**: a *engine* de um programa é o núcleo do mesmo. São as rotinas responsáveis pela execução das funcionalidades de um programa, e são distintas para cada aspecto do programa, como a aparência, forma de processamento dos dados, etc. O termo é utilizado geralmente para referenciar de maneira geral o que, falando especificamente, seriam objetos de programação, bibliotecas (*libraries*), SDKs, plataformas, dentre outros componentes de um programa.

Classe Principal

Elementos de dados podem ser arrastados para um script

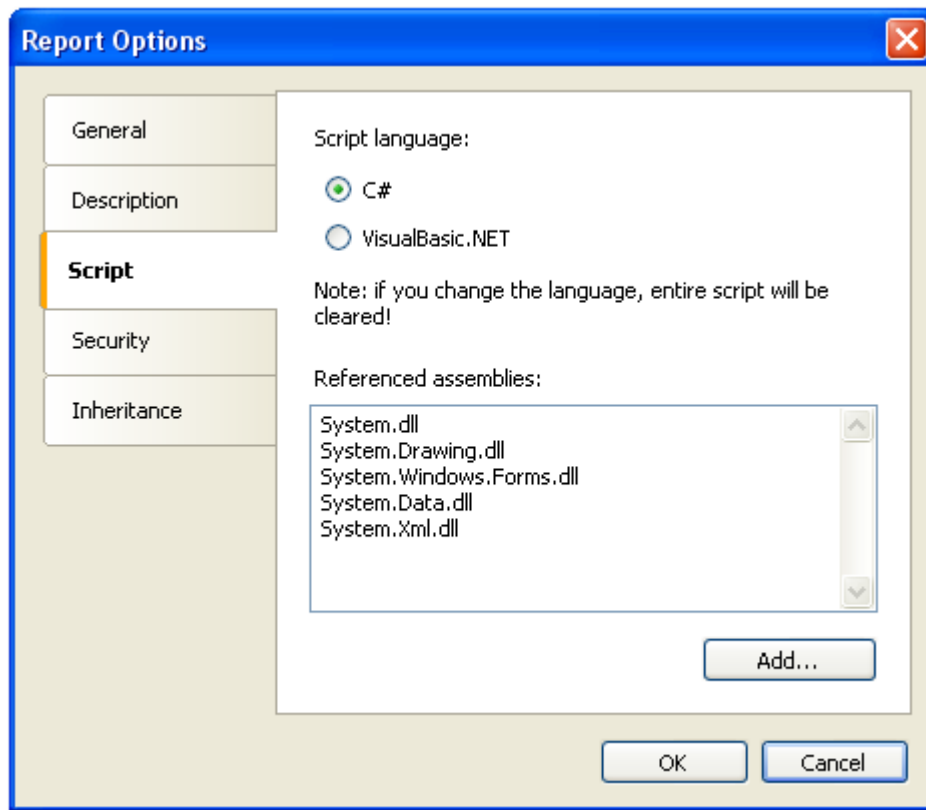


Aba Código.

Manipulador de eventos que foi criado na janela Propriedades.

Na janela Propriedades, você pode criar manipuladores de eventos com um duplo-clique.

A linguagem de *script* pode ser definida no menu **Relatório > Opções**. Isto deve ser feito logo após criar um novo relatório, pois quando é alterada a linguagem, o *script* existente é excluído.



## Informações Gerais

Ao contrário de outros geradores de relatórios, o CIGAM Report contém somente o que você escreveu. No *script*, você pode:

- Adicionar suas variáveis, métodos e propriedades à classe principal do *script*;
- Criar um manipulador de objetos do relatório;
- Adicionar novas classes ao *script*, se necessário. Uma classe pode ser adicionada antes da classe principal **ReportScript** ou depois desta.

Você não pode:

- Excluir, renomear ou alterar a área de visibilidade da classe principal ReportScript ;
- Renomear um espaço no qual a classe principal esteja localizada.

Quando o relatório está sendo executado, o seguinte ocorre:

- O CIGAM Report adiciona ao *script* uma lista de variáveis, cujos nomes correspondem aos nomes dos objetos de relatório. Isto é feito antes da compilação do *script* e permite a você referenciar os objetos de relatório por seus nomes;
- Um manipulador de expressões é adicionado ao *script*, que manipula todas as expressões encontradas no relatório;
- Um *script* é compilado, se o mesmo não está vazio;




Universidade Corporativa CIGAM

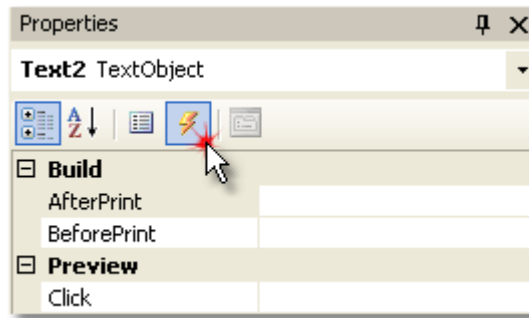
[www.cigam.com.br](http://www.cigam.com.br)

[treinamento@cigam.com.br](mailto:treinamento@cigam.com.br)

- Uma classe de *script* é inicializada;
- O relatório é executado.

## Manipuladores de Eventos

Um *script* é principalmente utilizado para criar os manipuladores de eventos dos objetos. Para criar um manipulador de evento selecione o objeto desejado. Na janela **Propriedades**, pressione o botão , para ativar a lista de eventos:



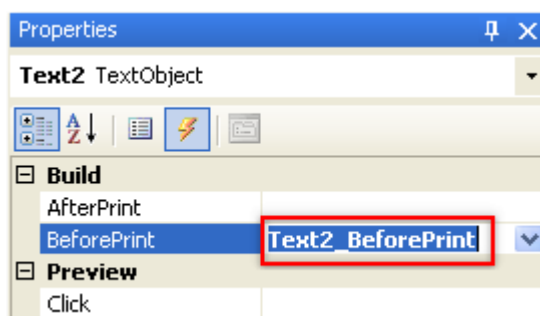
Dê um duplo clique no evento desejado, e o CIGAM Report irá adicionar um manipulador de evento em branco ao código do relatório:

```
private void Text2_BeforePrint(object sender, EventArgs e)
{
}
```

O objeto **Report** também possui eventos. Este objeto pode ser selecionado da seguinte maneira:

- Selecione **Report** na janela **Árvore de Relatório**;
- Selecione **Report** na lista *drop-down* da janela **Propriedades**;

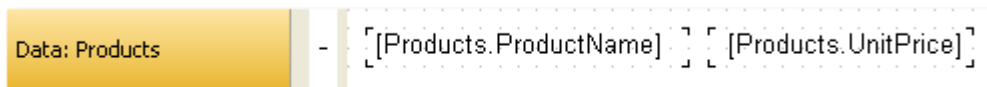
Para excluir o manipulador de evento, escolha um evento na janela **Propriedades**, selecione o texto e pressione a tecla **Delete**:



## Eventos de Relatório

Para controlar um relatório com o máximo de flexibilidade, todo objeto de relatório possui diversos eventos. Por exemplo, em um manipulador conectado à banda **Dados**, você pode filtrar registros, ou seja, exibir ou esconder a banda, dependendo de certas condições.

Vamos considerar os eventos que são disparados durante o processo de geração do relatório. Como exemplo, iremos pegar um relatório simples, contendo uma página, uma banda **Dados** e dois objetos **Texto** nesta banda.



No início do relatório, o objeto **Report** dispara o evento **StartReport**. Antes da formação da página do relatório, o evento **StartPage** é disparado. Este evento é disparado toda vez que é gerada uma nova página no relatório. Neste caso, independente de quantas páginas foram configuradas no relatório preparado, o evento é disparado uma vez, já que o relatório só possui uma página.

**Não confunda página configurada com página impressa! O evento StartPage é disparado a cada nova página do relatório configurado, e não a cada página que será impressa.**

Além disto, a impressão da banda **Dados** é iniciada. Isto ocorre da seguinte maneira:

1. O evento **BeforePrint** da banda é disparado;
2. O evento **BeforePrint** de todos os objetos da banda é disparado;
3. Todos os objetos são preenchidos com dados;
4. O evento **AfterData** de todos os objetos da banda é disparado;
5. O evento **BeforeLayout** da banda é disparado;
6. Os objetos são dispostos na banda, a altura da banda é calculada e a mesma é estendida (se puder, ver **propriedade CanGrow**);
7. O evento **AfterLayout** da banda é disparado;
8. Se a banda não pode se encaixar em um espaço livre na página atual, é gerada uma nova página;
9. A banda e todos os seus objetos são exibidos em uma página preparada do relatório;
10. O evento **AfterPrint** da banda é disparado;
11. O evento **AfterPrint** de todos os objetos é disparado.

A impressão da banda ocorre enquanto houverem dados na fonte. Após, a formação do relatório de nosso exemplo é finalizada. São disparados os eventos **ConcluirPage** de uma página e, por fim, o evento **ConcluirReport** do objeto **Report**.

Como pode ver, utilizando eventos de diferentes objetos, você pode controlar cada passo da formação de um relatório. Para utilizar corretamente os eventos, é importante um entendimento completo do processo de impressão das bandas, exposto nos 11 passos acima.



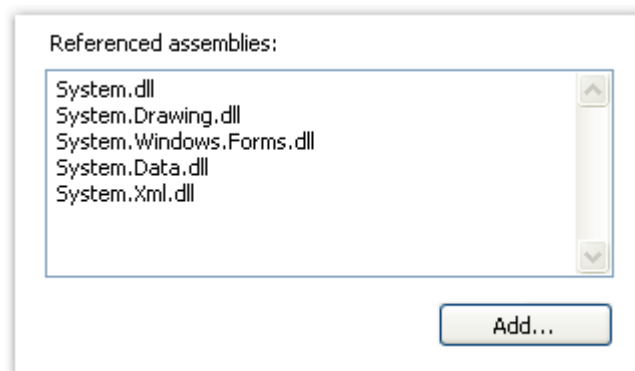
Muitas das operações podem ser feitas apenas utilizando o evento **BeforePrint** da banda: qualquer alteração feita ao objeto também será exibida no resultado final. Mas neste evento, não é possível analisar em qual página a banda será impressa se a mesma for estendida, pois a altura desta é calculada no passo 6. Isto pode ser feito com ajuda dos eventos **AfterLayout** no passo 7 ou **AfterPrint** no passo 10, mas neste último caso, a banda já foi impressa e operações com objetos não terão resultado. Em outras palavras, você deve determinar com clareza em que momento cada evento é disparado e utilizar aqueles que correspondem à tarefa definida.

## Utilizando objetos .NET

Em um *script*, podem ser utilizados quaisquer objetos .NET, que são definidos nas seguintes *assemblies*<sup>5</sup>:

- System.dll;
- System.Drawing.dll;
- System.Windows.Forms.dll;
- System.Data.dll;
- System.Xml.dll.

À parte disto, você pode utilizar qualquer objeto definido na *assembly* do CIGAM Report. Se você necessita acestar qualquer outra *assembly*, adicione-a à lista do CIGAM Report. Isto pode ser feito no menu **Relatório > Opções**, selecionando a aba **Script**:



Por exemplo, se você deseja utilizar uma função em seu relatório que já tenha sido declarada em uma aplicação, adicione a *assembly* da aplicação (.exe ou .dll) à lista de *assemblies* de um relatório. Após isto, você pode chamar a função utilizando o **namespace** de sua aplicação. Por exemplo, se a seguinte função é definida na aplicação:

`namespace Demo`

---

<sup>5</sup> **Assembly**: Um *assembly* é um arquivo (classes, imagens, textos, recursos, executáveis, dlls, etc) que representa uma unidade de distribuição, controle de versão, ou outros, em linguagem .NET. Toda aplicação .NET é composta de uma ou mais *assemblies*.



```
{  
  
    public static class MyFunctions  
    {  
  
        public static string Func1()  
        {  
  
            return Hello!;  
  
        }  
  
    }  
  
}
```

esta função pode ser chamada no *script* da seguinte maneira:

```
string hello = Demo.MyFunctions.Func1();
```

### Referências a objetos do relatório

Para referenciar objetos do relatório (por exemplo, um objeto Texto), utilize o nome do objeto. O exemplo a seguir retorna a altura do objeto **Texto1**:

```
float height = Texto1.Height;
```

Note que a unidade de medida nativa do relatório é em **pixels**. Tenha isto em mente quando for utilizar as propriedades do objeto, como **Left**, **Top**, **Width** e **Height**. Para converter pixels em centímetros (e vice-versa), utilize as constantes, definidas na classe **Units**:

```
float heightInPixels = Texto1.Height;
```

```
float heightInCM = heightInPixels / Units.Centimeters;
```

```
Texto1.Height = Units.Centimeters * 5; // 5cm
```



## Report and Engine objects

Além dos objetos, que estão contidos no relatório, há duas variáveis definidas no *script*: **Report** e **Engine**.

A variável **Report** referencia o relatório atual. Na lista abaixo, uma lista dos métodos do objeto Report:

| Método  | Descrição  |
|---|--|
| <code>object Calc(<br/>  string expression)</code>                                | Calcula uma expressão e retorna o valor. Quando este método é chamado pela primeira vez, uma expressão é compilada (o que demanda algum tempo).  |
| <code>object GetColumnValue(<br/>  string complexName)</code>                     | Retorna o valor da coluna de dados. O nome deve ser apresentado no formato FonteDados.Coluna. Se a coluna possui o valor <b>null</b> , este é convertido em um valor padrão (0, <i>string</i> vazia, falso). |
| <code>object GetColumnValueNullable(<br/>  string complexName)</code>             | Retorna o valor da coluna de dados. Ao contrário do método anterior, este pode ser nulo ( <b>null</b> ) e não é convertido.  |
| <code>Parameter GetParameter(<br/>  string complexName)</code>                    | Retorna o parâmetro de relatório com o nome indicado. O nome pode ser composto quando referenciando um parâmetro aninhado: ParamPrincipal.ParamAninhado.   |
| <code>object GetParameterValue(<br/>  string complexName)</code>                  | Retorna o valor do parâmetro de relatório com o nome indicado.   |
| <code>void SetParameterValue(<br/>  string complexName<br/>  object value)</code> | Define o valor do parâmetro de relatório como o nome indicado.   |
| <code>object GetVariableValue(<br/>  string complexName)</code>                   | Retorna o valor da variável de sistema, como por exemplo, <b>Date</b> .  |
| <code>object GetTotalValue(<br/>  string name)</code>                             | Retorna o valor do total definido na janela <b>Dados</b> pelo próprio nome.  |
| <code>DataSourceBase GetDataSource(<br/>  string alias)</code>                    | Retorna a fonte de dados definida no relatório pelo próprio nome.  |

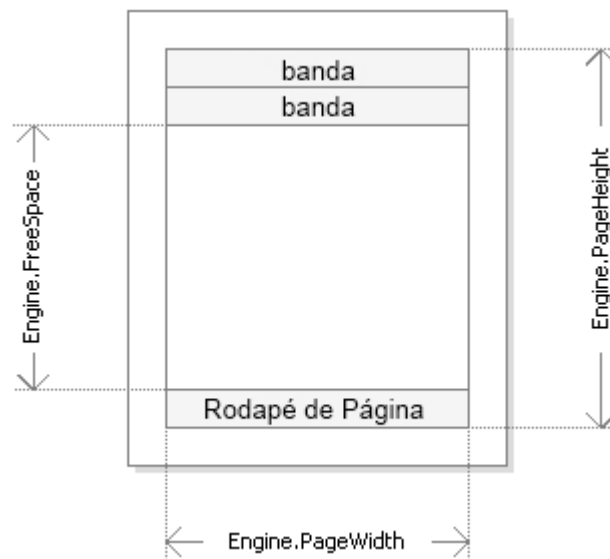
O objeto **Engine** é a *engine* que controla a criação de relatórios. Utilizando os métodos e propriedades da *engine*, você pode gerenciar o processo de colocação das bandas na página. Você pode usar as seguintes propriedades do objeto Engine:

| Método | Descrição |
|--------|-----------|
|--------|-----------|



|                                       |  |
|---------------------------------------|--|
| <code>float</code> CurX               | Coordenadas atuais do eixo X. Um valor pode ser atribuído a esta propriedade, para deslocar o objeto impresso.   |
| <code>float</code> CurY               | Coordenadas atuais de impressão do eixo Y. Um valor pode ser atribuído a esta propriedade, para deslocar o objeto impresso.  |
| <code>int</code> CurColumn            | Número da coluna atual em um relatório de múltiplas colunas. A primeira coluna tem o número 0.   |
| <code>int</code> CurPage              | Número da página sendo impressa. Este valor pode ser recebido da variável de sistema <b>Page</b> .   |
| <code>float</code> PageWidth          | Largura da página (já subtraído o tamanho das margens laterais).   |
| <code>float</code> PageHeight         | Altura da página (já subtraído o tamanho das margens superior e inferior).   |
| <code>float</code> PageFooterHeight   | Altura do rodapé da página (e todas as suas bandas-filhas).  |
| <code>float</code> ColumnFooterHeight | Altura do rodapé da coluna (e todas as suas bandas-filhas).  |
| <code>float</code> FreeSpace          | Tamanho do espaço livre na página.   |
| <code>bool</code> FirstPass           | Retorna o valor <b>true</b> se a primeira (ou única) passagem do relatório está sendo executada. O número de passagens pode ser obtida da propriedade <code>Report.DoublePass</code> . |
| <code>bool</code> FinalPass           | Retorna o valor <b>true</b> se a última (ou única) passagem do relatório está sendo executada.   |

Na imagem abaixo, você pode ver o significado de algumas das propriedades listadas acima.



As propriedades **Engine.PageWidth** e **Engine.PageHeight** determinam o tamanho da área de impressão, que é quase sempre menor do que o tamanho real da página. O tamanho da área impressa é determinado pelas margens da página, que por sua vez são retiradas das propriedades de página **LeftMargin**, **TopMargin**, **RightMargin** e **BottomMargin**.

A propriedade **Engine.FreeSpace** determina a altura do espaço livre na página. Se a banda **Rodapé de Página** está na página, sua altura é considerada quando é calculada a propriedade **FreeSpace**. Note que, após a impressão de uma banda, o espaço livre é reduzido.

Como acontece a formação de uma página de relatório preparada? A *engine* do CIGAM Report exibe bandas na página enquanto houver espaço para inseri-las. Quando não há mais espaço livre, a banda **Rodapé de Relatório** é impressa e uma nova página é formada. A exibição de uma banda inicia da posição atual, que é determinada pelas coordenadas X e Y. Esta posição é reajustada pelas propriedades **Engine.CurX** e **Engine.CurY**. Após a impressão de uma banda, a **CurY** automaticamente aumenta de acordo com a altura da banda impressa. Após a formação de uma nova página, a posição da propriedade **CurY** é definida como **0**. A posição da **CurX** é alterada quando é impresso um relatório com múltiplas colunas.

As propriedades **Engine.CurX** e **Engine.CurY** são acessíveis não somente para leitura, mas também para escrita. Isto significa que você pode alterar uma banda manualmente utilizando um dos eventos adequados. Exemplos de utilização destas propriedades podem ser vistos na seção **Exemplos**.

---

Quando trabalhando com propriedades que retornam o **tamanho** ou **posição**, lembre-se que estas propriedades são **medidas em pixels**.

---

No objeto **Engine**, os seguintes métodos são definidos:

| Método | Descrição |
|--------|-----------|
|--------|-----------|

|  |   |
|--|---|
| <code>void Adicionar Outline(<br/>string text)</code>  | Adiciona um elemento no esquema de relatório (leia mais na seção <b>Relatórios Interativos</b> ) e define a posição atual do elemento adicionado. |
| <code>void OutlineRoot()</code>                        | Define a posição atual na raiz do esquema de relatório.   |
| <code>void OutlineUp()</code>                          | Altera a posição atual para um elemento de nível mais alto no esquema de relatório.   |
| <code>void Adicionar Marcação(<br/>string name)</code> | Adiciona uma marcação (leia mais na seção <b>Relatórios Interativos</b> ).  |
| <code>void GetBookmarkPage(<br/>string name)</code>    | Retorna o número da página onde a marcação com o nome indicado está localizada.   |

Através da utilização dos métodos **Adicionar Outline**, **OutlineRoot** e **OutlineUp**, você pode formar o esquema do relatório manualmente. Geralmente, isto é feito automaticamente com a ajuda da propriedade **OutlineExpression**, presente em cada banda e página do relatório.

O método **Adicionar Outline** adiciona um elemento-filho ao elemento atual do esquema, e faz com que o elemento-filho se torne o elemento atual. A página atual do relatório e a posição atual na página são associados com o novo elemento. Se você chamar o método **Adicionar Outline** diversas vezes, então você terá o seguinte esquema:

```
Item 1  
  Item 2  
    Item 3
```

Para controlar o elemento atual, existem os métodos **OutlineUp** e **OutlineRoot**. O primeiro método move o ponteiro do elemento, localizado em um nível superior. Veja o exemplo:

```
Engine.AddOutline (Item1);  
Engine.AddOutline (Item1);  
Engine.AddOutline (Item1);  
Engine.OutlineUp ();  
Engine.AddOutline (Item1);
```

O *script* acima gera o seguinte esquema:

```
Item 1  
  Item 2  
  Item 3  
  Item 4
```

O método **OutlineRoot** move o elemento atual para a raiz do esquema. Como neste exemplo:

```
Engine.AddOutline (Item1);  
Engine.AddOutline (Item1);  
Engine.AddOutline (Item1);  
Engine.OutlineRoot ();  
Engine.AddOutline (Item1);
```

este *script* gera o esquema abaixo:

```
Item 1  
  Item 2  
    Item 3  
Item 4
```

Para trabalhar com marcações, os métodos **Adicionar Marcação** e **GetBookmarkPage** do objeto **Engine** são utilizados. Geralmente, marcações são adicionadas automaticamente quando utilizando a propriedade **Marcação**, que está presente em todos os objetos.

Utilizando o método **Adicionar Marcação**, você pode adicionar uma marcação programaticamente. Este método cria uma marcação na página e posição atual.

O método **GetBookmarkPage** retorna o número da página onde a marcação se encontra. Este método é geralmente utilizado na criação de um índice, para exibir números de página. Neste caso, o relatório deve ser configurado para realizar um passe duplo.

## Referências a fontes de dados

Ao contrário das expressões do CIGAM Report (maiores informações no capítulo **Expressões**), nunca utilize colchetes no *script* para referenciar fontes de dados. Ao invés disso, utilize o método **GetColumnValue** do objeto **Report**, que retorna o valor da coluna:

```
string productName = (string)Report.GetColumnValue(Produto.Nome);
```

Assim, você precisa indicar o nome da fonte e sua coluna. O nome da fonte pode ser composto, caso estejamos nos referindo à fonte de dados através do uso de uma relação. Detalhes sobre as relações podem ser encontrados no capítulo **Dados**. Por exemplo, você pode referenciar uma coluna da fonte relacionada da seguinte maneira:

```
string categoryName =  
(string)Report.GetColumnValue(Produto.Categorias.NomeCategoria);
```

Para facilitar o trabalho, utilize a janela **Dados**. Dela, você pode arrastar elementos de dados para o *script*, durante este processo o aplicativo cria um código para referenciar o elemento.

Para referenciar a própria fonte de dados, utilize o método **GetDataSource** do objeto **Report**:

```
DataSourceBase ds = Report.GetDataSource (Produtos);
```



Ajuda nos métodos da classe `DataSourceBase` podem ser recebidos através do sistema de ajuda de referências da classe **FastReport.Net**. Este objeto é utilizado no *script* da seguinte maneira:

```
//pegar referência da fonte de dados

DataSourceBase ds = Report.GetDataSource(Produtos);

//inicializar a referência

Ds.Init();

//enumerar as linhas
while (ds.HasMoreRows)
{
    //pegar o valor da coluna de dados da linha atual
    string productName = (string)Report.GetColumnValue(Produto.Nome);
    //faça algo com ele...
    //...

    //vá para a próxima linha de dados
    Ds.Next();
}
```

## Referência a variáveis de sistema

Para referenciar variáveis de sistema, utilize o método **GetVariableValue** do objeto **Report**:

```
DateTime date = (DateTime)Report.GetVariableValue (Date);
```

Uma lista das variáveis de sistema pode ser vista na janela **Dados**. A partir desta, você pode arrastar uma variável para um *script*, e o aplicativo cria automaticamente o código.

## Referência a valores totais

Para referenciar um valor total, utilize o método **GetTotalValue** do objeto **Report**:

```
DateTime date = (DateTime)Report.GetVariableValue (Date);
```

Uma lista dos totais pode ser vista na janela **Dados**. A partir desta, você pode arrastar uma variável para um *script*, e o aplicativo cria automaticamente o código.

Um valor total possui o tipo **FastReport.Variant**. Pode ser utilizado diretamente em qualquer expressão, pois este tipo é automaticamente convertido. Por exemplo:

```
float tax = Report.GetTotalValue (TotalVendas) * 0.2f;
```



A referência ao valor total pode ser feita no momento em que este está sendo processado. Geralmente o total está pronto para utilização no momento da impressão da banda.

## Reference to report parameters

Para referenciar parâmetros de relatório, utilize o método **GetParameterValue** do objeto **Report**:

```
int myParam = (int)Report.GetParameterValue (MeuParametro);
```

Parâmetros podem ser aninhados. Neste caso, marque o nome do parâmetro pai, e após o período, o nome do parâmetro-filho:

```
Report.GetParameterValue (ParametroPai.ParametroFilho);
```

Parâmetros possuem um tipo definido de dados. Este é fornecido através da propriedade **DataType** do parâmetro. Você deve levar isto em conta quando fizer referência a parâmetros. Você pode ver uma lista de parâmetros na janela **Dados**: a partir desta, você pode arrastar parâmetros para o *script*, e o aplicativo cria automaticamente o código.

Para alterar o valor de um parâmetro, utilize o método **SetParameterValue** do objeto **Report**:

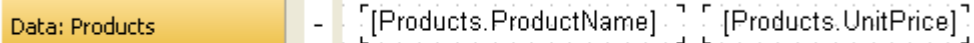
```
Report.SetParameterValue (MeuParametro, 10);
```

## Exemplos

### Exemplo 1. Alterando a aparência de um objeto

Neste exemplo iremos mostrar como alterar a cor do texto dependendo do valor impresso no objeto. Iremos utilizar o evento de banda **BeforePrint** e algumas referências à coluna de dados a partir do *script*.

Crie um relatório simples conforme o exemplo abaixo:



Selecione o objeto, que imprime a coluna **PrecoUnitario** e crie um manipulador de evento **BeforePrint**:

```
private void Text2_BeforePrint(object sender, EventArgs e)
{
    if (((Decimal)Report.GetColumnValue(Produto.PrecoUnitario)) > 20)
        Text2.TextColor = Color.Red;
}
```

Para inserir a coluna de dados **Produto.PrecoUnitario** no *script*, arraste-a da janela **Dados**. A seguinte *string* será adicionada ao *script*:

```
((Decimal)Report.GetColumnValue(Produtos.PrecoUnitario))
```

Se gerarmos o relatório, veremos que todos os produtos com o preço acima de 20 serão demarcados em vermelho:

|                              |       |
|------------------------------|-------|
| Chai                         | 18,00 |
| Chang                        | 19,00 |
| Aniseed Syrup                | 10,00 |
| Chef Anton's Cajun Seasoning | 22,00 |
| Chef Anton's Gumbo Mix       | 21,35 |
| Grandma's Boysenberry Spread | 25,00 |

O mesmo efeito pode ser alcançado com a ajuda de realce condicional (você pode ler mais sobre isto na seção **Realce condicional** do capítulo **Criação de Relatórios**).

### Exemplo 2. Realçando linhas de número par na banda

Neste exemplo iremos mostrar como mudar a cor de preenchimento das linhas de número par na banda **Dados**. Iremos utilizar o evento de banda **BeforePrint** e uma referência à variável de sistema **Row#**, a partir do *script*.

Crie um relatório simples conforme o exemplo abaixo:

|                |   |                        |                      |
|----------------|---|------------------------|----------------------|
| Data: Products | - | [Products.ProductName] | [Products.UnitPrice] |
|----------------|---|------------------------|----------------------|

Crie um manipulador de evento **BeforePrint** para a banda:

```
private void Data1_BeforePrint(object sender, EventArgs e)
{
    if (((Int32)Report.GetVariableValue(Row#)) % 2 == 0)
        Data1.FillColor = Color.Gainsboro;
}
```

A variável de sistema **Row#** retorna o número da linha impressa na banda. Para inserir uma referência a esta variável no *script*, arraste-a da janela **Dados**. A seguinte *string* será adicionada ao *script*:

```
((Int32)Report.GetVariableValue(Row#))
```

Se gerarmos o relatório, veremos que as linhas de número par serão realçadas na cor cinza:

|                              |       |
|------------------------------|-------|
| Chai                         | 18,00 |
| Chang                        | 19,00 |
| Aniseed Syrup                | 10,00 |
| Chef Anton's Cajun Seasoning | 22,00 |
| Chef Anton's Gumbo Mix       | 21,35 |

O mesmo efeito pode ser alcançado com a ajuda da propriedade **EvenStyle** da banda **Dados**. Você pode ler mais sobre isto na seção **Realçando linhas de dados pares e ímpares** do capítulo **Criação de Relatórios**).

### Exemplo 3. Filtragem de dados

Neste exemplo, iremos mostrar como ocultar a linha da banda **Dados** dependendo de certas condições. Iremos utilizar o evento de banda **BeforePrint** e uma referência à fonte de dados, a partir do *script*.

Crie um relatório simples conforme o exemplo abaixo:

Data: Products - [[Products.ProductName]] [[Products.UnitPrice]]

Crie um manipulador de evento **BeforePrint** para a banda:

```
private void Data1_BeforePrint(object sender, EventArgs e)
{
    if (((Decimal)Report.GetColumnValue(Row#)) > 20)
        Data1.Visible = false;
}
```

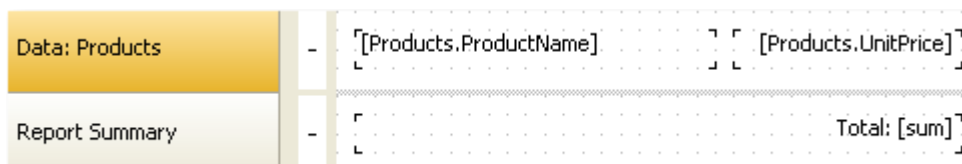
|               |       |
|---------------|-------|
| Chai          | 18,00 |
| Chang         | 19,00 |
| Aniseed Syrup | 10,00 |
| Konbu         | 6,00  |
| Genen Shouyu  | 15,50 |
| Pavlova       | 17,45 |

O mesmo efeito pode ser alcançado com a utilização do filtro de dados que pode ser definido no editor da banda **Dados**.

#### Exemplo 4. Calculando um total

Neste exemplo, iremos mostrar como calcular a soma utilizando métodos de programação. Iremos utilizar o evento de banda **BeforePrint**, uma referência à coluna de dados a partir do *script*, e uma variável local cujo valor será impresso no relatório.

Crie um relatório simples conforme o exemplo abaixo:



|                |   |                        |                      |
|----------------|---|------------------------|----------------------|
| Data: Products | - | [Products.ProductName] | [Products.UnitPrice] |
| Report Summary | - | Total: [sum]           |                      |

No *script*, declare a variável soma e crie um manipulador de evento **BeforePrint** pertencendo à banda:

```
public class ReportScript
{
    private decimal sum;
    private void Data1_BeforePrint(object sender, EventArgs e)
    {
        sum += (Decimal)Report.GetColumnValue(Produtos.PrecoUnitario);
    }
}
```

A coluna de dados **Produtos.PrecoUnitario** pode ser inserida no *script* arrastando-a da janela **Dados**.

Se você rodar o relatório, você verá o seguinte:

|                                 |         |
|---------------------------------|---------|
| Röd Kaviar                      | 15,00   |
| Longlife Tofu                   | 10,00   |
| Rhönbräu Klosterbier            | 7,75    |
| Lakkalikööri                    | 18,00   |
| Original Frankfurter grüne Soße | 13,00   |
| Total:                          | 2222,71 |

O mesmo efeito pode ser alcançado com a utilização de totais.

### Exemplo 5. Alterando a posição de impressão

Neste exemplo, iremos mostrar como alterar manualmente a posição de uma banda, utilizando o objeto **Engine**. Iremos utilizar o evento de banda **BeforePrint** e o objeto **Engine**.

Crie um relatório simples conforme o exemplo abaixo:

```
Data: Products - [Products.ProductName] [Products.UnitPrice]
```

Crie um manipulador de evento **BeforePrint** pertencendo à banda:

```
private void Data1_BeforePrint(object sender, EventArgs e)
{
    Engine.CurX = ((Int32)Report.GetVariableValue(Row#)) * 10;
}
```

Se você rodar o relatório, você verá o seguinte:

|                              |       |
|------------------------------|-------|
| Chai                         | 18,00 |
| Chang                        | 19,00 |
| Aniseed Syrup                | 10,00 |
| Chef Anton's Cajun Seasoning | 22,00 |
| Chef Anton's Gumbo Mix       | 21,35 |



Universidade Corporativa CIGAM

[www.cigam.com.br](http://www.cigam.com.br)

[treinamento@cigam.com.br](mailto:treinamento@cigam.com.br)


---

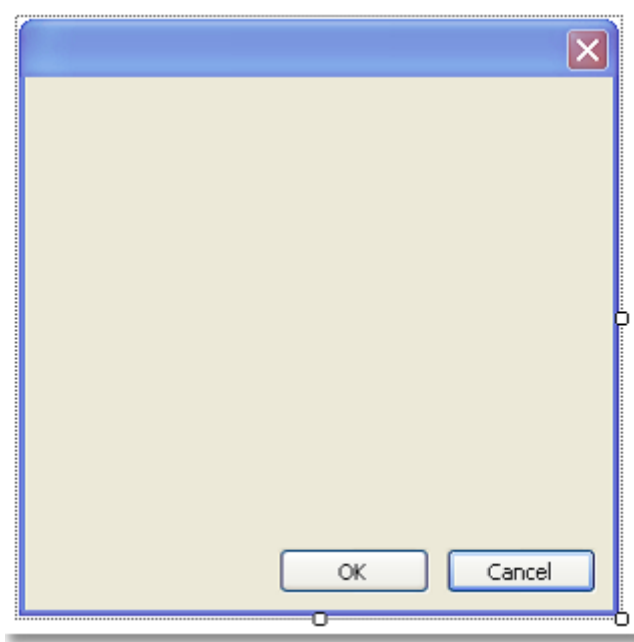
O mesmo efeito pode ser alcançado com a utilização de totais.

---

## Forms de diálogo

A parte de uma página comum, um relatório pode conter um ou mais *forms* de diálogo. Tais *forms* serão exibidos quando o relatório é iniciado. No diálogo, você pode informar qualquer tipo de dados necessários para criação de um relatório. Um diálogo também pode ser utilizado para filtragem de dados, o que é exibido no relatório.

Para adicionar um diálogo em um relatório, pressione o botão  na barra de ferramentas do editor. Um novo diálogo tem a seguinte aparência:























Um relatório, tendo um ou mais diálogos, funciona da seguinte maneira:

- Quando iniciado, o relatório exibe o primeiro diálogo;
- Se o diálogo é fechado com ajuda do botão **OK**, então o próximo diálogo é exibido. Se não, o relatório encerra o funcionamento;
- Após todos os diálogos terem sido exibidos, o relatório é construído.

## Controles

Em um *form* de diálogo, os seguintes controles podem ser utilizados:

| Ícone   | Nome                  | Descrição   |
|---|-----------------------|---|
|  | ButtonControl         | Botão no padrão Windows.  |
|  | CheckBoxControl       | Checkbox no padrão Windows.   |
|  | CheckedListBoxControl | Exibe uma <i>listbox</i> na qual uma <i>checkbox</i> é exibida à esquerda de cada item. |
|  | ComboBoxControl       | Controle <i>combo box</i> no padrão Windows.  |

|   |                       |   |
|---|-----------------------|---|
|    | DataGridViewControl   | Exibe os dados em uma grade customizável.   |
|    | DataSelectorControl   | Exibe duas listas e permite a realocação de itens entre estas.  |
|    | DateTimePickerControl | Controle de Windows que permite ao usuário selecionar uma data e horário, além de exibir ambos em um formato especificado.          |
|    | GroupBoxControl       | Controle Windows que exibe uma moldura ao redor de um grupo de controles, com uma legenda opcional.                                 |
|    | LabelControl          | Etiqueta padrão de Windows.   |
|    | ListBoxControl        | Controle de Windows para exibir uma lista de itens.   |
|    | ListViewControl       | Controle de visualização de listas de Windows, que exibe um conjunto de itens que podem ser exibidos de quatro maneiras diferentes. |
|    | MaskedTextBoxControl  | Utiliza uma máscara para distinguir a maneira apropriada de exibir a informação.  |
|    | MonthCalendarControl  | Controle de Windows que permite ao usuário selecionar uma data utilizando um seletor visual de data mensal.                         |
|  | NumericUpDownControl  | Controle de Windows do tipo <i>spin Box</i> para selecionar e exibir números.   |
|  | PanelControl          | Utilizado para agrupar conjuntos de controles.  |
|  | PictureBoxControl     | Caixa para exibição de uma imagem.  |
|  | RadioButtonControl    | Permite ao usuário selecionar uma única opção de um grupo de escolhas quando agrupado com outros controles do mesmo tipo.           |
|  | RichTextBoxControl    | Controle de caixa de texto com opções para edição em formato RTF (semelhante ao editor de texto <b>Word</b> ).                      |
|  | TextBoxControl        | Controle de caixa de texto simples.   |
|  | TreeViewControl       | Exibe um conjunto hierárquico dos itens etiquetados.  |

Todos os controles, à exceção do **DataSelectorControl**, são uma analogia completa dos controles do **Windows.Forms padrão**. O nome dos elementos possui um **sufixo Control** para evitar nomes duplicados. Portanto, o **ButtonControl** do CIGAM Report corresponde ao controle **Button** padrão.

### Referenciando um controle no script

Uma referência a um controle pode ser feita no *script* utilizando o seu nome:

```
TextBoxControl11.Text = meu texto;
```

Na verdade, o controle do CIGAM Report é apenas uma embalagem para o controle .NET padrão. Este controle inclui muitas, porém não todas as propriedades do controle padrão. Se você precisa de





alguma propriedade que não é implementada pelo controle do **CIGAM Report**, é possível acestar um controle padrão da seguinte maneira:

- Utilizando a propriedade **Control**, que é do tipo **System.Windows.Forms.Control**:

```
(TextBox1.Control as TextBox) .ShortcutsEnabled = false;
```

- Utilizando a propriedade que possui o mesmo nome que o próprio controle, mas sem o sufixo **Control**. Por exemplo, o **TextBoxControl** possui a propriedade **TextBox**, que é do tipo **System.Windows.Forms.TextBox** e retorna o controle **TextBox** embalado:

```
TextBox1.TextBox.ShortcutsEnabled = false;
```

Maiores informações sobre propriedades e métodos dos controles podem ser obtidas na **MSDN** (Microsoft Developer Network).

## Filtragem de dados

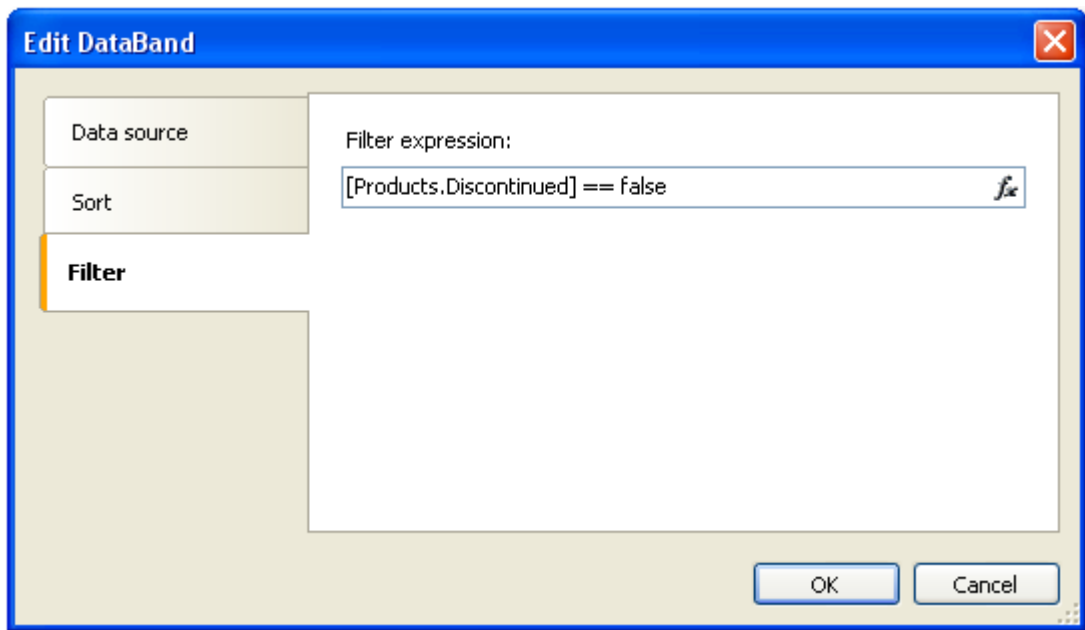
Diálogos podem ser utilizados para filtrar os dados que serão impressos em um relatório. Por exemplo: você tem um relatório que imprime a lista de todos os colaboradores. Utilizando um diálogo, você pode escolher um ou vários deste relatório, e quando construir o relatório, os dados são filtrados para que somente os colaboradores escolhidos sejam exibidos.

---

Para utilizar a filtragem de dados, é necessário que o relatório inicial contenha todos os dados. O nome filtro, por si só, assume que dados desnecessários não serão impressos quando o relatório é gerado.

---

O método mais simples para organizar a filtragem de dados é utilizando a propriedade **Filter** na banda **Dados**. No editor de bandas, você pode indicar a expressão do filtro, como no exemplo abaixo:



Utilizando o diálogo, você pode solicitar um valor ao usuário, e utilizar este na expressão de filtragem. Veja o exemplo **Filtro Simples**, na seção **Exemplos** no final deste capítulo.

este método pode ser utilizado se um valor simples é necessário. Se a tarefa é exibir uma lista de variáveis e retirar uma ou mais destas, a implementação deste recurso se torna mais complicado. Pode até parecer uma tarefa simples – exibir uma lista dos colaboradores no elemento **ListBoxControl** e escolher um ou mais valores. Para implementar isto, você deve utilizar o *script*, que faz o seguinte:












- Acesta a fonte de dados;
- Reúne os dados solicitados;
- Preenche o **ListBoxControl** com os dados da fonte;
- Após a escolha do(s) colaborador(es), constrói uma expressão de filtro que será utilizada na banda **Dados**.

O CIGAM Report pode fazer isto automaticamente. Para tanto, é utilizada a filtragem automática, que é o nosso próximo tópico neste capítulo.

### Filtragem automática – como funciona

O controle é conectado à coluna de dados utilizando a propriedade **DataColumn**. Se o controle pode exibir uma lista de valores (como o **ListBoxControl**, por exemplo), este é preenchido com valores da coluna de dados indicada. Isto acontece automaticamente quando o diálogo é exibido. Além disso, o usuário pode trabalhar com o diálogo, selecionando um ou mais itens no controle e fechando o diálogo. Neste momento, a fonte de dados que foi indicada na propriedade **DataColumn** é filtrada automaticamente.

A vantagem deste método é que você pode utilizá-lo em qualquer relatório **sem a necessidade de script**.

| Ícone   | Nome                  |
|---|-----------------------|
|  | CheckBoxControl       |
|  | CheckedListBoxControl |
|  | ComboBoxControl       |
|  | DataSelectorControl   |
|  | DateTimePickerControl |
|  | ListBoxControl        |
|  | MaskedTextBoxControl  |
|  | MonthCalendarControl  |
|  | NumericUpDownControl  |
|  | RadioButtonControl    |
|  | TextBoxControl        |

## Operações de Filtro

Por padrão, o CIGAM Report filtra as linhas de dados que contém um valor igual ao valor do controle. Este comportamento é definido na propriedade **FilterOperation** do controle. Você pode utilizar as seguintes operações:

| Operação                  | Equivalente | Efeito   |
|---------------------------|-------------|--|
| <b>Equal</b>              | =           | Filtra o valor se este é igual ao valor do controle.       |
| <b>NotEqual</b>           | <>          | Filtra o valor se este é diferente do valor do controle.   |
| <b>LessThan</b>           | <           | Filtra o valor se este é inferior ao do controle.          |
| <b>LessThanOrEqual</b>    | <=          | Filtra o valor se este é igual ou inferior ao do controle. |
| <b>GreaterThan</b>        | >           | Filtra o valor se este é maior que o do controle.          |
| <b>GreaterThanOrEqual</b> | >=          | Filtra o valor se este é maior ou igual ao do controle.    |

Por exemplo, se a propriedade **FilterOperation** é definida como **LessThanOrEqual**, e você utilizar o valor **5** no controle, então todas as linhas de dados com valores iguais ou menores do que 5 serão filtradas.

Para dados do tipo *string*, você pode utilizar algumas operações extras:

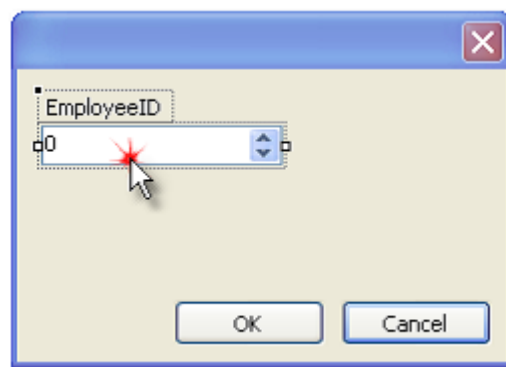
| Operação        | Efeito   |
|-----------------|--|
| <b>Contains</b> | Filtra o valor se este contém o valor do controle. |

|                      |   |
|----------------------|---|
| <b>NotContains</b>   | Filtra o valor se este não contém o valor do controle.      |
| <b>StartsWith</b>    | Filtra o valor se este inicia com o valor do controle.      |
| <b>NotStartsWith</b> | Filtra o valor se este não inicia com o valor do controle.  |
| <b>EndsWith</b>      | Filtra o valor se este encerra com o valor do controle.     |
| <b>NotEndsWith</b>   | Filtra o valor se este não encerra com o valor do controle. |

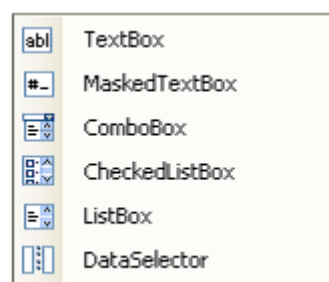
Por exemplo, se a propriedade **FilterOperation** do controle é definida como **StartsWith** e você utilizar a *string* **A**, então todas as linhas de dados cujos valores se iniciam em **A** serão filtradas.

### Adicionando um filtro ao relatório

O editor de diálogos do CIGAM Report possui algumas facilidades para adição de controles que suportam filtragem de dados. Para isto, arraste e solte a coluna de dados da janela **Dados** para o *form* de diálogo. Durante este procedimento, o CIGAM Report cria o cabeçalho (controle **LabelControl**) e um controle que pode ser utilizado para filtragem de dados:



O tipo de controle depende do tipo de coluna de dados. Se a coluna é do tipo *string*, então após a inserção desta, você receberá a opção de escolher o tipo de controle:



Se você possui dois controles similares conectados à mesma coluna de dados, o CIGAM Report automaticamente configura o intervalo de dados com a ajuda da propriedade **FilterOperation**. O primeiro controle terá a propriedade **FilterOperation** igual a **GreaterThanOrEqual**, e o segundo, **LessThanOrEqual**. Isto será feito caso você insira uma coluna de dados cujos valores não sejam do tipo *string*.

Dito isto, para adicionar filtragem de dados a qualquer relatório, você deve fazer o seguinte:

- Adicionar um novo diálogo ao relatório;
- Soltar uma coluna de dados no diálogo, a partir da qual você deseja filtrar o relatório.

### Filtrando dados em intervalos

este método de filtragem é simples de utilizar quando estamos trabalhando com valores que possuam uma características quantitativa, como por exemplo o custo. Você pode filtrar produtos que tenham custo abaixo ou acima do que é dado. Para indicar como o valor será interpretado, utilize a propriedade **FilterOperation**, citada acima.

Utilizando dois controles que são conectados à mesma coluna de dados e possuam diferentes configurações na propriedade **FilterOperation**, você pode indicar o início e o fim do intervalo de dados. Para o primeiro controle, você deve indicar a **FilterOperation** como **GreaterThanOrEqualTo**, e para o segundo, **LessThanOrEqualTo**.

### Filtrando em uma coluna de dados relacionada

Como sabemos, uma relação pode ser estabelecida entre duas fontes de dados. Veja mais detalhes no capítulo **Dados**. Com a ajuda da relação, é possível filtrar os dados na fonte, utilizando uma coluna de dados de uma fonte diferente.

Presumindo que você tenha colocado no diálogo um **ListBoxControl** e indicado a seguinte coluna de dados na propriedade **DataColumn**:

```
Produtos.Categorias.NomeCategoria;
```

- Quando preenchendo o controle com valores, a coluna **NomeCategoria** da fonte de dados **Categorias** será utilizada;
- O filtro será aplicado à tabela **Produtos**. Serão filtradas as linhas de dados para as quais a seguinte condição esteja correta:

```
O [Produtos.Categorias.NomeCategoria] contém um dos valores selecionados pelo usuário;
```

### Controlando a filtragem a partir do script

Mesmo que a filtragem automática seja o suficiente para muitos dos casos, você tem a possibilidade de gerenciar a mesma manualmente. Para isto, os seguintes métodos e propriedades são utilizados:

A propriedade **AutoFill** controla o preenchimento dos controles com dados. É utilizada pelos controles que podem criar uma lista de valores, como o **ListBoxControl**. Antes de exibir um diálogo, o CIGAM Report preenche estes controles com dados. Por padrão, a propriedade é definida como **true**. Se desabilitada, o controle não será preenchido, e você deve fazer isto manualmente, chamando o método **FillData**:


```
ListBox1.FillData();
```

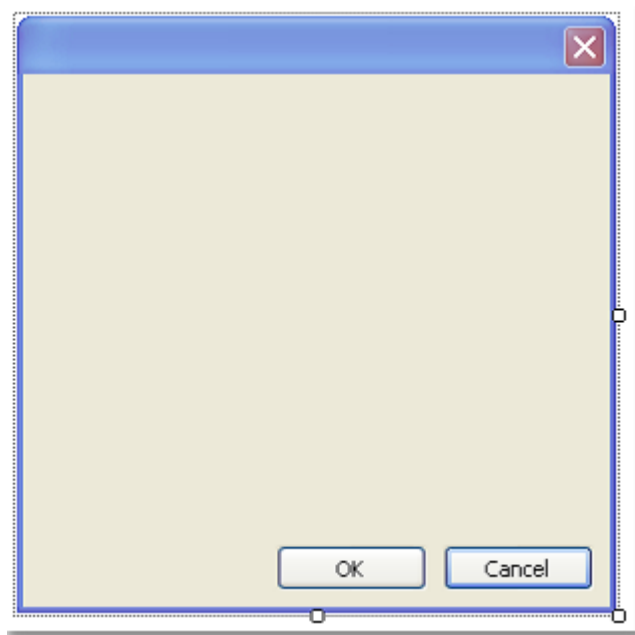
A propriedade **AutoFilter** controla a filtragem de dados, e é utilizada por todos os controles. Após fechar o diálogo com o botão **OK**, o aplicativo aplica o filtro de dados automaticamente. Por padrão, esta propriedade também é definida como **true**. Se for desabilitada, a filtragem não ocorrerá, e você deve fazer isto manualmente, chamando o método **FilterData**:

```
ListBox1.FilterData();
```

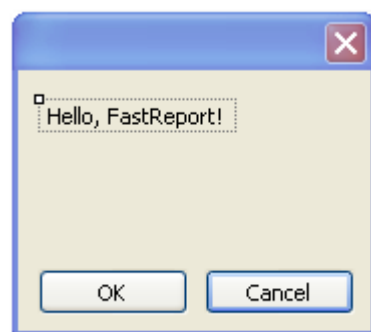
## Exemplos

### Exemplo 1. Olá, CIGAM Report!

Neste exemplo, tudo que faremos é criar um diálogo que irá exibir uma saudação. Crie um novo relatório e adicione um *form* de diálogo ao mesmo. Para isto, clique no botão  na barra de ferramentas:



No diálogo, insira um **LabelControl** e defina sua propriedade **Texto** na janela **Propriedades**:

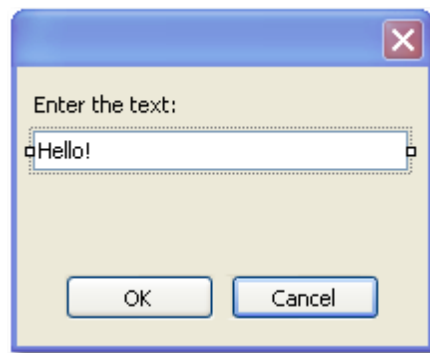


Ao executar o relatório, você verá o diálogo. Feche-o com o botão **OK**, e o relatório será gerado. Se o diálogo for fechado utilizando os botões **Cancel** ou **X**, o relatório será encerrado.

### Exemplo 2. Solicitar um texto para o usuário

Neste exemplo, iremos criar um diálogo que irá solicitar um texto do usuário e, mais tarde, imprimir este no relatório.

Crie um novo relatório e adicione um *form* de diálogo a este. No diálogo, insira os controles **LabelControl** e **TextBoxControl**:



No exemplo, o valor informado é armazenado na propriedade **Texto** do controle **TextBoxControl**. Para imprimir este valor no relatório, adicione um novo objeto **Texto** na banda **Título do Relatório** e escreva o seguinte neste:

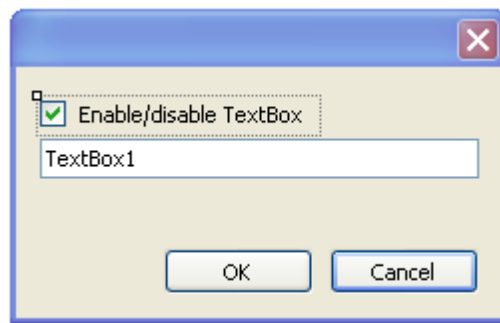
```
Você digitou: [TextBox1.Text]
```

No exemplo, TextBox1 é o nome do controle **TextBoxControl**.

### Exemplo 3. Manipulando controles de diálogo

Utilizando *script* e eventos dos controles, você pode manipular os controles exatamente como é feito no Visual Studio. Iremos mostrar um exemplo de como o **CheckBoxControl** pode manipular a acessibilidade do **TextBoxControl**.

Crie um novo relatório e adicione um *form* de diálogo a este. No diálogo, insira os controles **CheckBoxControl** e **TextBoxControl**:



Agora selecione o **CheckBoxControl**, abra a janela **Propriedades** e clique no botão ⚡. Dê um duplo clique no evento **CheckedChanged**, que é disparado na alteração do status da *checkbox*. O aplicativo irá criar um manipulador vazio para este evento. Insira o seguinte código neste manipulador:

```
private void CheckBox1_CheckedChanged (object sender, EventArgs e)
{
    TextBox1.Enabled = CheckBox1.Checked;
}
```

Ao executarmos o relatório, podemos habilitar ou desabilitar o **TextBoxControl** através da *checkbox*.

### Exemplo 4. Manipulando objetos de relatório

Vejam um exemplo de relatório, que imprime uma lista de categorias e produtos em cada categoria:

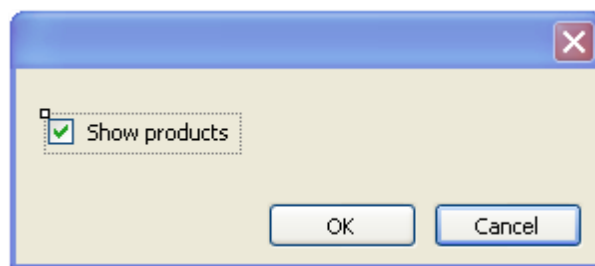


| Beverages                                   |            |
|---|------------|
| Soft drinks, coffees, teas, beers, and ales |            |
| Product name                                | Unit price |
| Chai  | 18,00      |
| Chang                                       | 19,00      |
| Chartreuse verte                            | 18,00      |
| Côte de Blaye                               | 263,50     |
| Guaraná Fantástica                          | 4,50       |
| Ipoih Coffee                                | 46,00      |
| Lakkalikööri                                | 18,00      |
| Laughing Lumberjack Lager                   | 14,00      |
| Outback Lager                               | 15,00      |
| Rhinbräu Klosterbier                        | 7,75       |
| Sasquatch Ale                               | 14,00      |
| Steeleye Stout                              | 18,00      |

| Condiments   |            |
|--|------------|
| Sweet and savory sauces, relishes, spreads, and seasonings |            |
| Product name   | Unit price |
| Aniseed Syrup  | 10,00      |

Iremos mostrar como parar a impressão dos produtos e imprimir somente as categorias, com a ajuda de um diálogo. Para isto, adicione um diálogo ao relatório:



Dê um duplo clique no botão **OK**. O aplicativo irá criar um manipulador vazio para o evento **Click**. Insira o seguinte código neste manipulador:

```
private void btnOk_Click(object sender, EventArgs e)
{
    Data2.Visible = CheckBox1.Checked;
}
```

Iremos controlar a visibilidade da banda que imprime a lista de produtos. Em nosso exemplo, esta banda tem o nome de **Data2**. Se o relatório for gerado com a *checkbox* desmarcada, teremos o seguinte resultado:

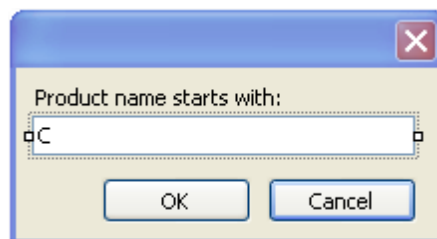


### Exemplo 5. Filtro simples

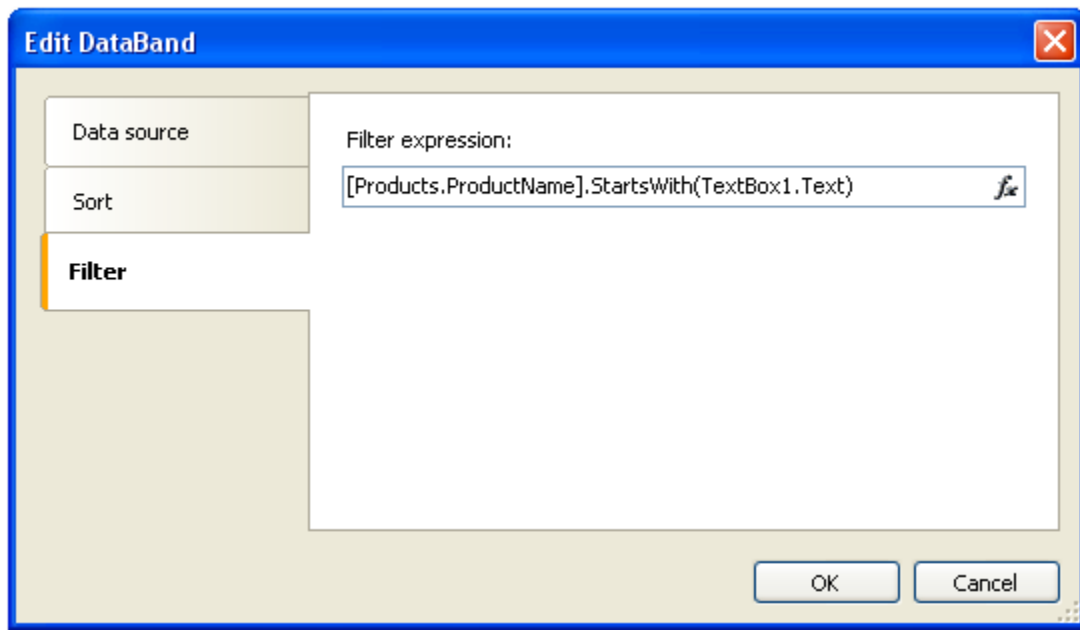
Veja o seguinte exemplo de relatório, que imprime uma lista de produtos:

| Header         | Product name           |
|----------------|------------------------|
| Data: Products | [Products.ProductName] |

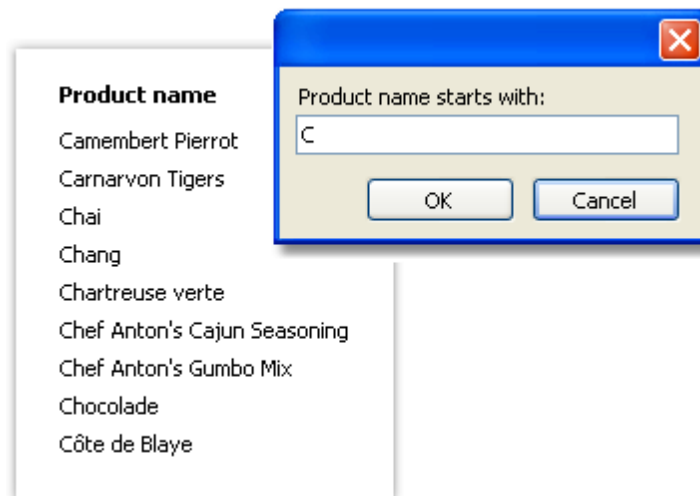
Iremos mostrar como filtrar lista dos produtos de acordo com a primeira letra do nome do produto. Durante este procedimento, não iremos utilizar os recursos de filtragem automática. Para isto, adicione um diálogo ao relatório e adicione dois controles ao mesmo: **LabelControl** e **TextBoxControl**:



Agora abra o editor da banda **Dados** e indique a seguinte expressão de filtro:



Execute o relatório e certifique-se de que tudo está funcionando:



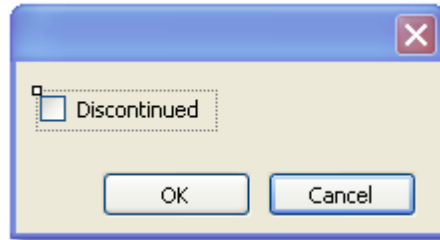
### Exemplo 6. Filtragem automática

Neste exemplo iremos mostrar como adicionar um filtro ao relatório que imprime uma lista de produtos a partir da tabela **Produto**. Iremos filtrar os dados de acordo com a coluna **Produtos.Descontinuados**.

O relatório tem aproximadamente esta aparência:

| Header         | Product name           | Discontinued                        |
|----------------|------------------------|-------------------------------------|
| Data: Products | [Products.ProductName] | <input checked="" type="checkbox"/> |

Adicione um novo diálogo ao relatório, e arraste a coluna **Produtos.Descontinuados** da janela **Dados** para o *form* de diálogos:



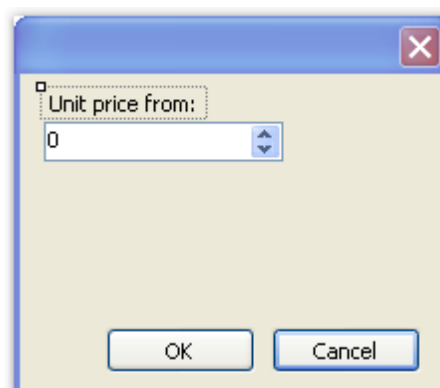
Isto é tudo que precisamos fazer, com apenas dois cliques. O aplicativo automaticamente conecta o controle à coluna de dados.

Execute o relatório e marque a *flag* **Descontinuado**. Após, clique no botão **OK**, e você verá um relatório que contém só os produtos com a *flag* **Descontinuado**:

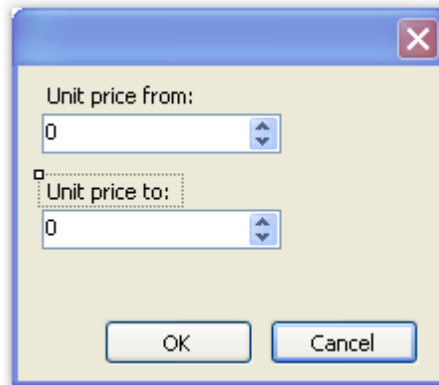
| Product name                  | Discontinued |
|-------------------------------|--------------|
| Alice Mutton                  | ✓            |
| Chef Anton's Gumbo Mix        | ✓            |
| Guaraná Fantástica            | ✓            |
| Mishi Kobe Niku               | ✓            |
| Perth Pasties                 | ✓            |
| Rössle Sauerkraut             | ✓            |
| Singaporean Hokkien Fried Mee | ✓            |
| Thüringer Rostbratwurst       | ✓            |

### Exemplo 7. Filtragem automática por intervalo

Neste exemplo iremos mostrar em um relatório anterior como imprimir produtos que possuem um custo dentro do intervalo indicado. Para isto iremos adicionar um novo diálogo ao relatório, e arrastar a coluna **Produtos.PrecoUnitario** para o *form*. Após isto iremos corrigir o texto da *label*:



Agora, da mesma maneira, iremos adicionar mais uma coluna **Produtos.PrecoUnitario** e corrigir o seu *label*:



Isto é tudo que precisamos fazer, o resto foi feito automaticamente pelo aplicativo: conectar os controles à coluna de dados e definir as suas propriedades **FilterOperation**. O primeiro controle possui a **FilterOperation** igual a **GreaterThanOrEqualTo**, e o segundo, igual a **LessThanOrEqualTo**.

Execute o relatório e indique os valores, por exemplo de 20 a 30. Quando clicar no botão **OK**, o relatório será gerado contendo produtos cujos valores estejam no intervalo indicado:

| Product name                     | UnitPrice |
|----------------------------------|-----------|
| Chef Anton's Cajun Seasoning     | 22,00     |
| Chef Anton's Gumbo Mix           | 21,35     |
| Fløtemysost                      | 21,50     |
| Grandma's Boysenberry Spread     | 25,00     |
| Gravad lax                       | 26,00     |
| Gustaf's Knäckebröd              | 21,00     |
| Louisiana Fiery Hot Pepper Sauce | 21,05     |
| Maxilaku                         | 20,00     |
| Nord-Ost Matjeshering            | 25,89     |
| Pâté chinois                     | 24,00     |
| Queso Cabrales                   | 21,00     |
| Sirop d'érable                   | 28,50     |
| Tofu                             | 23,25     |
| Uncle Bob's Organic Dried Pears  | 30,00     |

### Exemplo 8. Filtragem por coluna de dados relacionada

Neste exemplo, iremos utilizar uma coluna de uma fonte de dados relacionadas para fazer a filtragem dos dados.

Iremos manipular um relatório de lista simples, que apenas imprime uma lista de produtos. O nome da categoria é impresso ao lado de cada produto. Isto é feito com ajuda da relação:

```
[Produtos.Categorias.NomeCategoria]
```

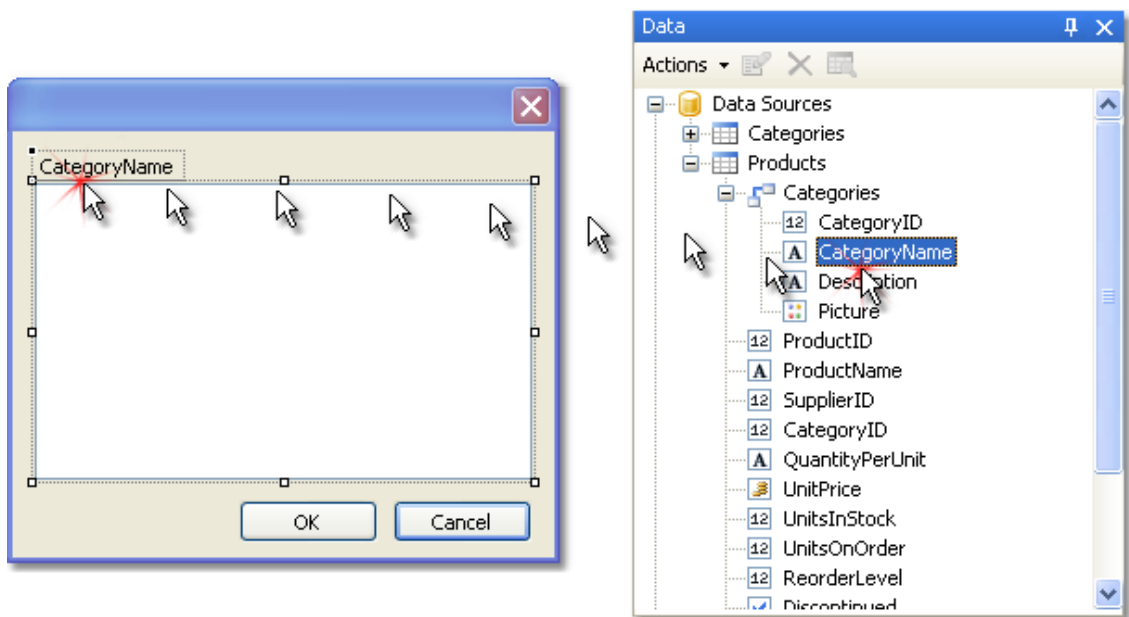
O relatório será como abaixo:

| Report Title   | PRODUCT CATALOG        |                                    |                      |
|----------------|------------------------|------------------------------------|----------------------|
| Header         | Product name           | Category name                      | Unit price           |
| Data: Products | [Products.ProductName] | [Products.Categories.CategoryName] | [Products.UnitPrice] |

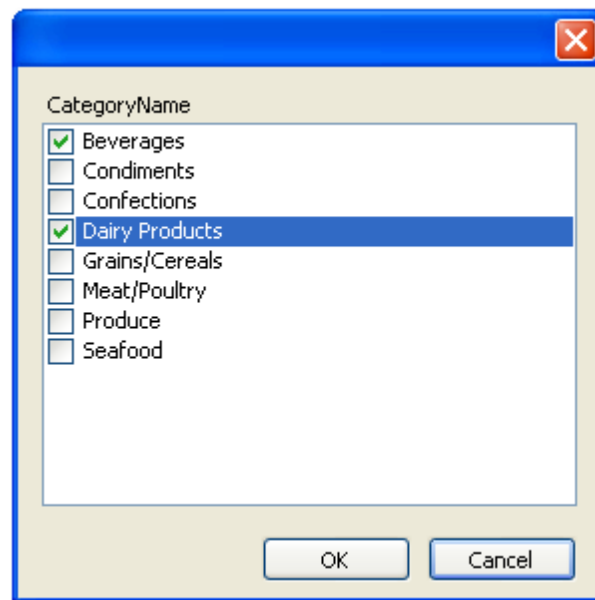
Quando executarmos o relatório, veremos o seguinte:

| PRODUCT CATALOG              |                |            |
|------------------------------|----------------|------------|
| Product name                 | Category name  | Unit price |
| Alice Mutton                 | Meat/Poultry   | 39,00      |
| Aniseed Syrup                | Condiments     | 10,00      |
| Boston Crab Meat             | Seafood        | 18,40      |
| Camembert Pierrot            | Dairy Products | 34,00      |
| Carnarvon Tigers             | Seafood        | 62,50      |
| Chai                         | Beverages      | 18,00      |
| Chang                        | Beverages      | 19,00      |
| Chartreuse verte             | Beverages      | 18,00      |
| Chef Anton's Cajun Seasoning | Condiments     | 22,00      |
| Chef Anton's Gumbo Mix       | Condiments     | 21,35      |
| Chocolade                    | Confections    | 12,75      |

Vamos adicionar uma filtragem pelo nome de categoria. Para isto, adicione um novo diálogo e arraste a coluna **Produtos.Categorias.NomeCategoria** para o *form*.



Quando é criado o controle, será solicitado que você escolha o tipo. Escolha **CheckedListBoxControl**. Ao executar o relatório, veremos o seguinte diálogo:



Escolha diversas categorias e clique no botão **OK**. Os dados serão filtrados e o seguinte relatório será exibido:



## PRODUCT CATALOG

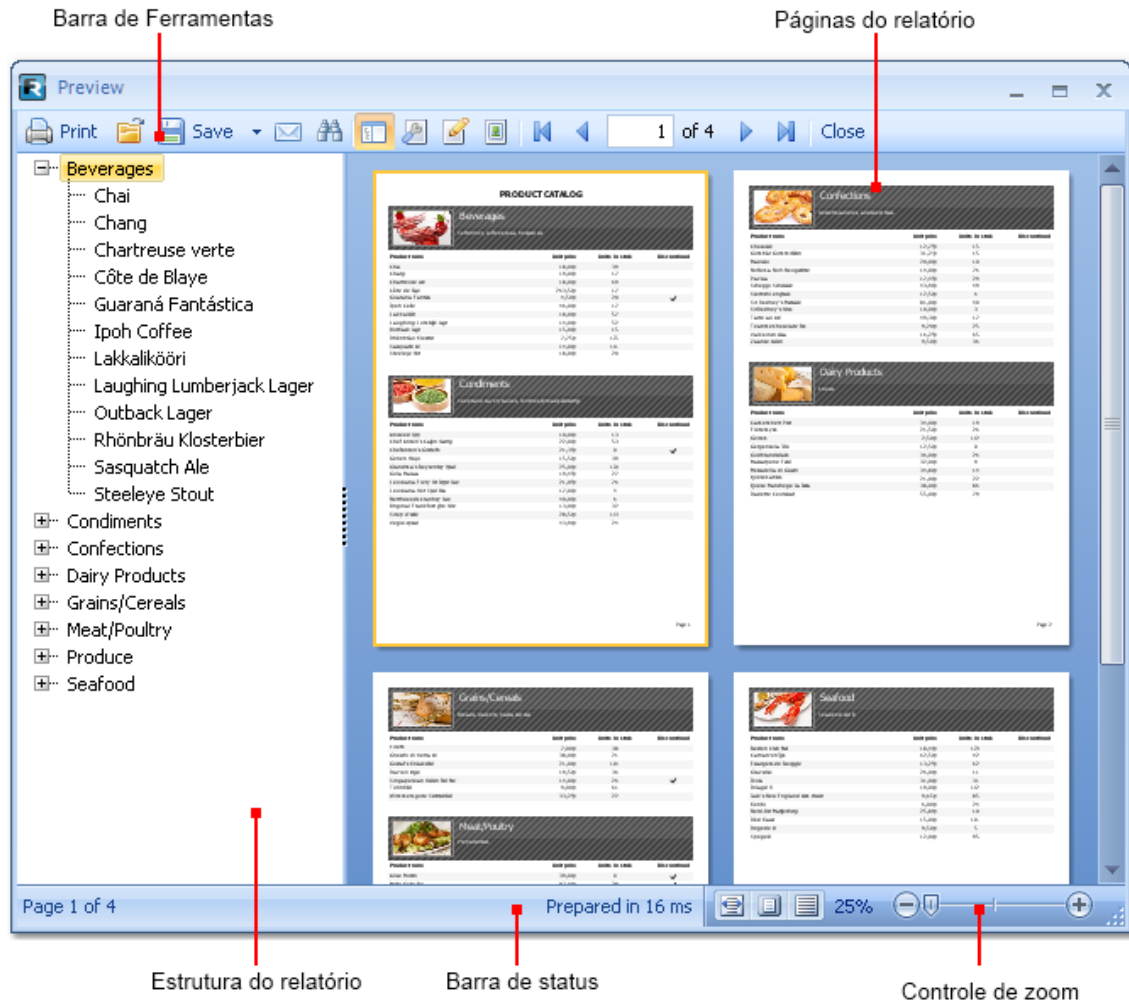
| Product name       | Category name  | Unit price |
|--------------------|----------------|------------|
| Camembert Pierrot  | Dairy Products | 34,00      |
| Chai               | Beverages      | 18,00      |
| Chang              | Beverages      | 19,00      |
| Chartreuse verte   | Beverages      | 18,00      |
| Côte de Blaye      | Beverages      | 263,50     |
| Fløtemysost        | Dairy Products | 21,50      |
| Geitost            | Dairy Products | 2,50       |
| Gorgonzola Telino  | Dairy Products | 12,50      |
| Guaraná Fantástica | Beverages      | 4,50       |
| Gudbrandsdalsost   | Dairy Products | 36,00      |
| Ipoh Coffee        | Beverages      | 46,00      |
| Lebkuchen          | Beverages      | 18,00      |

Como pode ser visto, só foram mantidos os produtos das categorias selecionadas.







## Pré-visualização, impressão e exportação










Um relatório montado pode ser mostrado na tela, impresso na impressora ou exportado para um dos formatos suportados. Tudo isso pode ser feito na janela de pré-visualização:



Na barra de ferramentas, você pode encontrar os seguintes botões:




| Botões  | Descrição   |
|---|---|
|  | Imprime o relatório.                                  |
|  | Abre o arquivo do relatório preparado no formato FPX. |
|  | Salva o relatório em um dos formatos suportados.      |
|  | Manda o relatório por e-mail.                         |

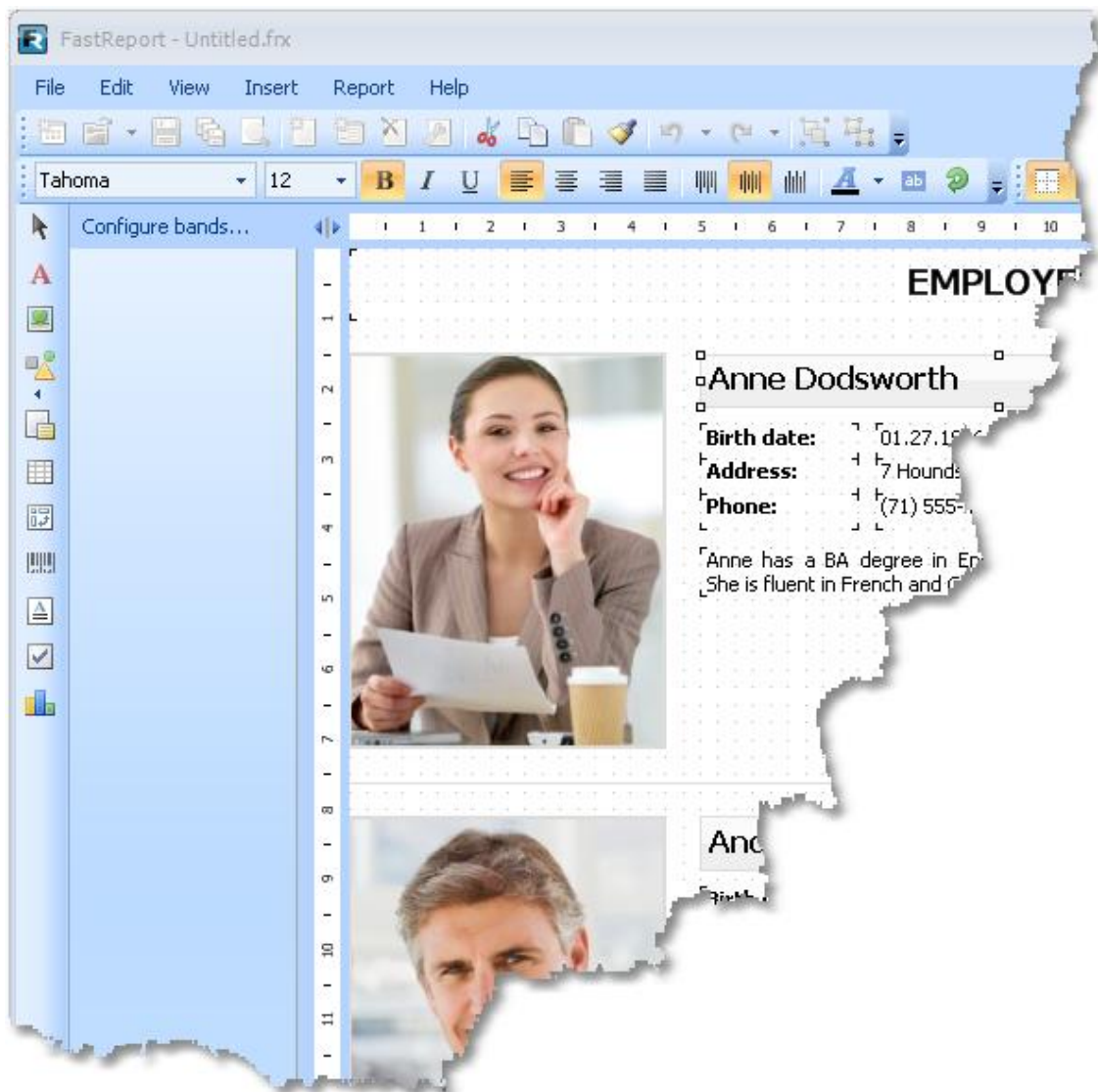
|   |   |
|---|---|
|  | Procura texto no relatório.   |
|  | Mostra ou oculta o esquema do relatório.                            |
|  | Configurações da página   |
|  | Editar a página atual do relatório                                  |
|  | Configurações de marca d'água.                                      |
|  | Ir para a primeira página   |
|  | Ir para a página anterior   |
| 1   | Ir para a página indicada. Insira o número da página e tecle enter. |
|  | Ir para a próxima página.   |
|  | Ir para a última página.  |

Você pode usar os seguintes atalhos do teclado:

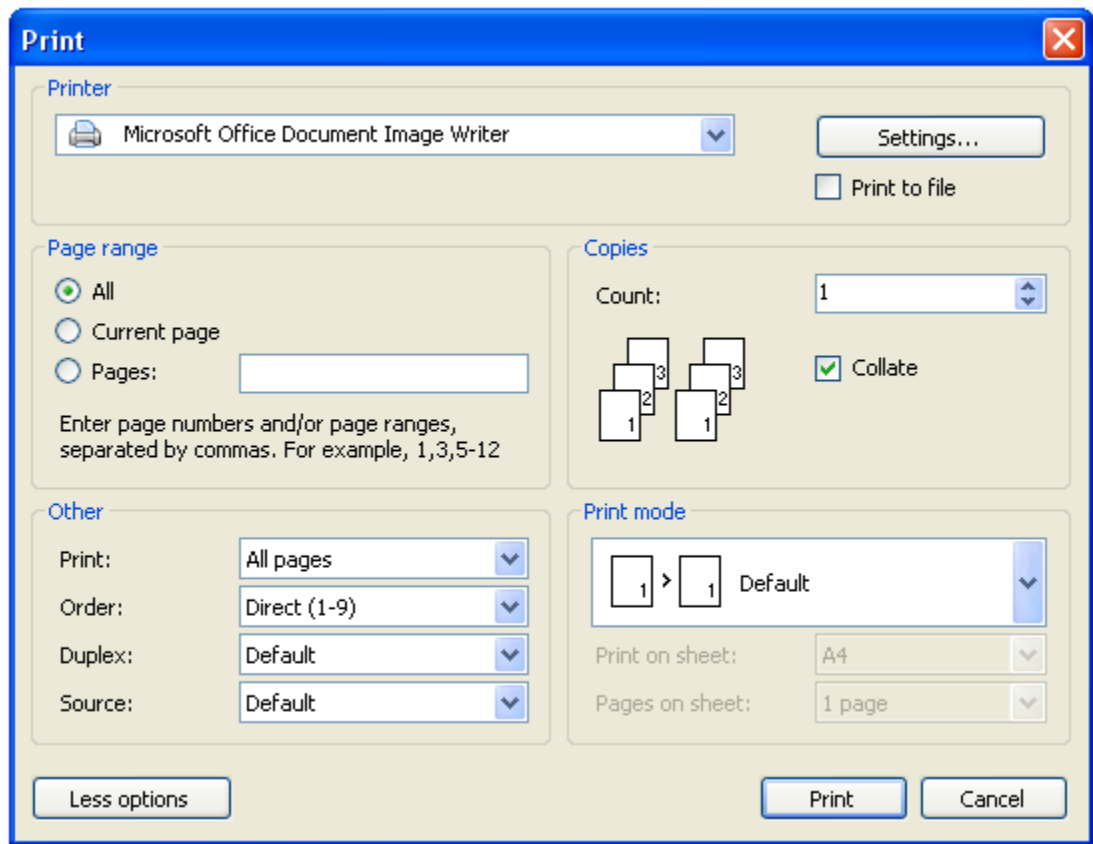
| Tecla                       | Descrição                                |
|-----------------------------|--|
| <b>Ctrl + P</b>             | Imprime o relatório.                     |
| <b>Ctrl + F</b>             | Procura texto.                           |
| <b>Setas</b>                | Desloca a página para a direção da seta. |
| <b>PageUp,<br/>PageDown</b> | Ir para cima/baixo da página.            |
| <b>Home</b>                 | Ir para a primeira página                |
| <b>End</b>                  | Ir para a última página                  |
| <b>Esc</b>                  | Fecha a janela pré-visualização.         |

## Editarando o relatório

Para editar a página dor relatório preparado, clique no botão  na janela de pré-visualização. Durante isto, a página atual será carregada no editor de relatório, onde você pode fazer o que quiser com ele:



Depois de editar, aperte o botão  (ou pressione Ctrl + P). Você verá a janela de impressão:



Vamos ver as configurações disponíveis nesta janela. O botão More/Less permite mostrar toda a janela ou somente as configurações básicas. Por padrão, a janela é exibida de uma forma simplificada.

Grupo **Printer**: aqui, você escolhe a impressora, muda suas configurações (botão **Settings...**) e escolhe imprimir para o arquivo.

Grupo **Page**: aqui você pode escolher quais páginas imprimir (todas, atual ou o número dado de páginas).

Grupo **Copy**: aqui você pode definir o número de cópias e escolher a ordem das páginas nas cópias (**Collate**).

Grupo **Outros**: aqui você pode escolher quais páginas imprimir (todas, pares, ímpares), escolher a ordem de impressão (direto, ao contrário), definir a impressão dos dois lados (se sua impressora suporta isso) e escolher a fonte de papel.

O grupo **Print mode** permite escolher um dos modos de impressão:

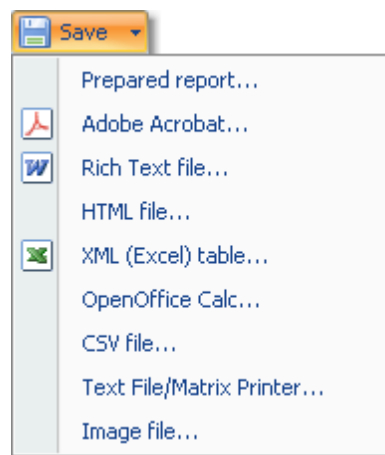
| Modo    | Descrição  |
|---------|--|
| Default | A impressora imprime em um papel, indicado no relatório. Uma página do relatório corresponde à uma folha impressa. |

|                          |  |
|--------------------------|--|
| <b>Splitte big pages</b> | Use este modo se você precisa imprimir relatórios do tamanho A3 em folhas do tamanho A4. Uma página do relatório irá gerar duas folhas impressas. Quando usar este modo, você deve escolher o tamanho do papel na lista <b>Print on sheet</b> .  |
| <b>Scale</b>             | Use este modo se você precisar imprimir um relatório A4 tendo uma folha do tamanho A3. Em uma folha de impressão, você pode imprimir 1,2,4 ou 8 páginas do relatório. Quando usar este modo, você precisa escolher o tamanho do papel no qual você quer imprimir, da lista <b>Printing on sheet</b> , e também indicar o número de páginas na lista <b>Pages on list</b> . |

Depois de pressionar o botão **Print**, a impressão do relatório iniciará. Se a opção **Print to file** estiver marcada, então o nome do arquivo será pedido e o relatório será salvo naquele arquivo (arquivo com extensão PRN).

## Exportando o relatório

O **CIGAM Report** permite exportar o relatório montado para diferentes formatos. No momento, exporta para 8 formatos: PDF, RTF, HTML, Excel (XML), OpenOffice Calc, CSV, TXT, arquivo de imagem. Para escolher a exportação, pressione o botão salvar na janela de pré-visualização e escolha a exportação desejada:





## Salvando no formato FPX

O formato FPX é o nativo do **CIGAM Report**. As vantagens deste formato são as seguintes:

- Salvar o relatório sem perder a qualidade. Abrindo um arquivo já salvo, você pode fazer todas as operações com ele, como imprimir, exportar, editar;
- Formato compacto baseado em XML, compactado com a ajuda do ZIP;
- Quando necessário, o arquivo do relatório pode ser descompactado por qualquer compactador/descompactador que suporte o formato ZIP e corrigido manualmente em qualquer editor de texto.

O único problema com o formato é que, para visualizá-lo, você precisa ter o **CIGAM Report.Net**.

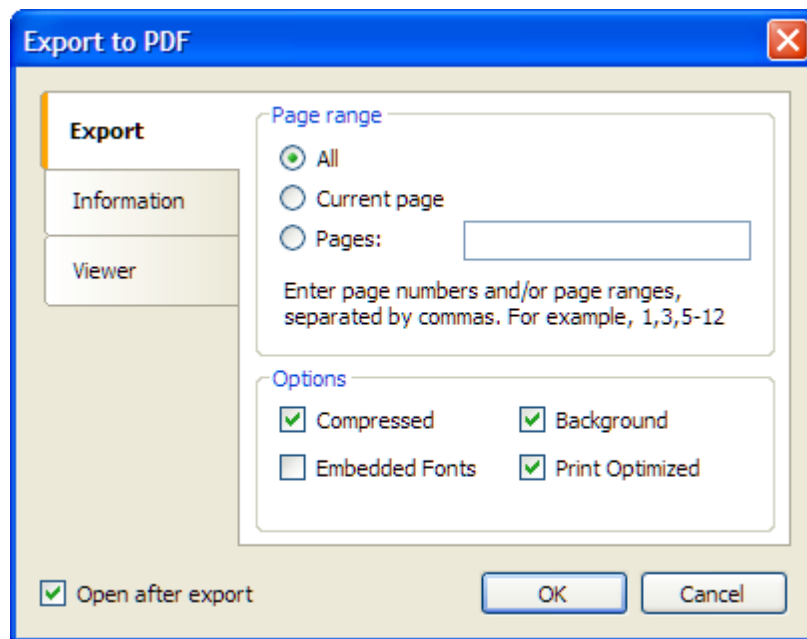
Para salvá-lo no formato FPX, pressione o botão **Save** na janela de pré-visualização e escolha a opção **Prepared Report**. Para abrir arquivos já salvos, pressione o botão **Open**.

## Exportar para o Adobe Acrobat (PDF)

O PDF (Portable Document Format) é um formato de documentos eletrônicos criado pela Adobe Systems. O pacote **free** do Acrobat Reader é usado para visualização. Este formato é bastante flexível – ele permite a inclusão de fontes necessárias, imagens bitmap e vetoriais; ele permite transferência e armazenamento de documentos intencionados à visualização e posterior impressão.

O método de exportação é composto por camadas.

Quando exportarmos para o Excel, haverá uma janela para definir os parâmetros do arquivo que será gerado:

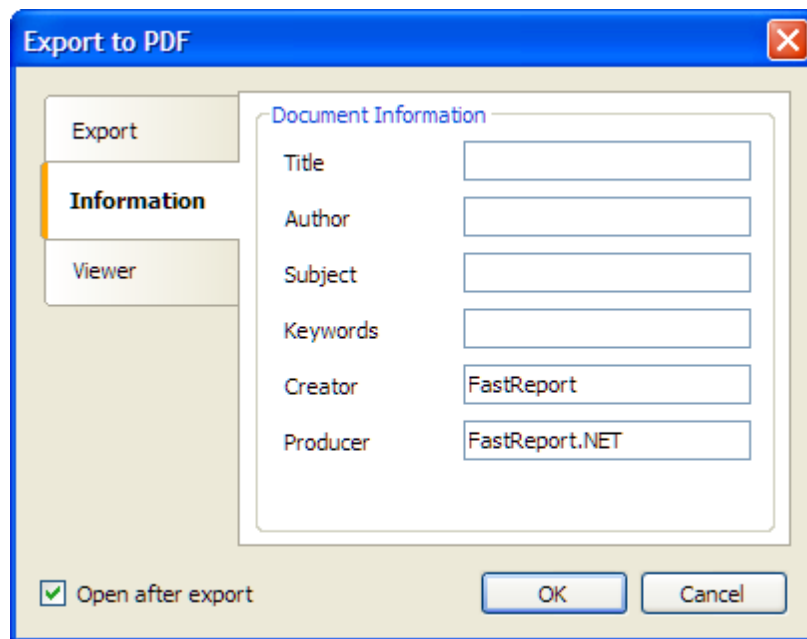


Parâmetros da exportação:

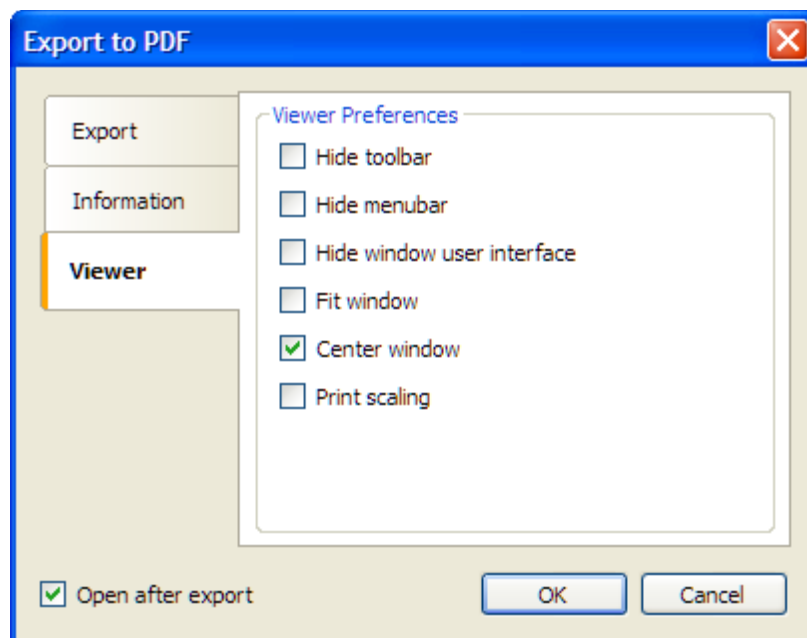
- ComprEted – o arquivo gerado é compactado. Isso reduz o tamanho do arquivo mas aumenta o tempo de exportação;
- Embedded fonts – todas as fontes usadas em um relatório serão incluídas no arquivo PDF. Isso aumenta significativamente o tamanho do arquivo;
- Background – a marca d'água da página será exportada como uma imagem. Isso aumenta significativamente o tamanho do arquivo;
- Print optimized – geração de todos os objetos gráficos (como figuras, gráficos) em alta resolução, para futura impressão.

Na guia **Information**, você deve preencher os campos de informação do documento:





Na guia **Viewer**, você deve definir algumas opções relacionadas ao visualizados de documento Acrobat Reader:



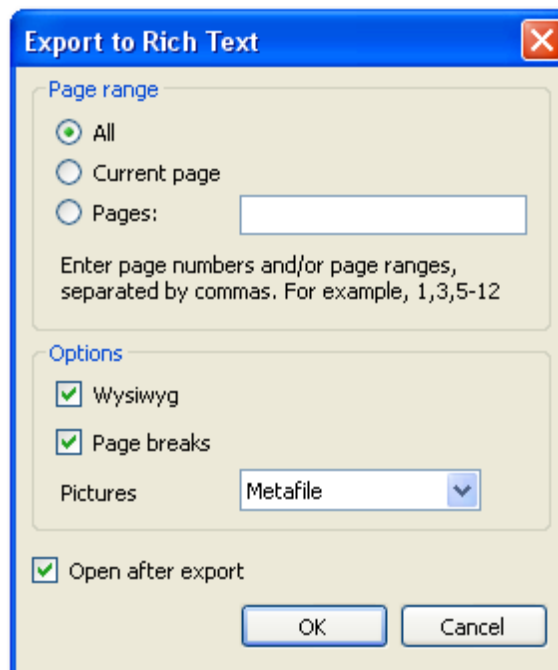
## Exportar para Word (RTF)



O RTF (Rich Texto Format) foi desenvolvido pela Microsoft como um formato padrão para a troca de informações textuais. No momento, documentos RTF são compatíveis com muitos novos editores de texto e sistemas operacionais.

Método de exportação: tabular.

Quando exportamos para RTF, haverá uma janela para definir os parâmetros do arquivo que será gerado:



Parâmetros de exportação:

- Wysiwyg – o resultado será o mais próximo possível ao relatório. Se esta opção é desmarcada, o **CIGAM Report** irá reduzir o número de colunas e linhas no resultado;
- Page breaks – habilita quebras de página no arquivo RTF;
- Pictures – seleciona o formato das imagens no arquivo RTF. Note que o formato **Metafile** é o melhor para exibir tais objetos do relatório como o **MSChartObject** e o **ShapeObject**.

---

A aparência e o tamanho do arquivo resultante dependem do template do relatório (veja na seção **Recommendation on report development**)

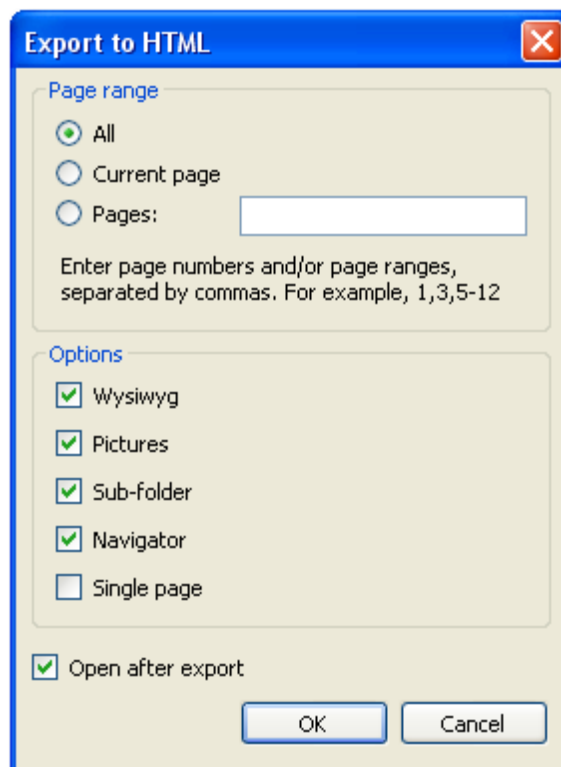
---

## Exportar para HTML

O HTML (HyperText Markup Language) é a linguagem de marcação predominante nas páginas da Web. Ele garante um meio de descrever a estrutura de uma informação baseada em texto em um documento – denotando certos textos como links, cabeçalhos, parágrafos, lista, e outros – e para completar isso, textos com formas interativas, imagens incorporadas, e outros objetos.

Método de exportação: tabular.

Quando exportarmos para HTML, uma janela será exibida para definirmos os parâmetros:



Parâmetros de exportação:

- Wysiwyg – o resultado da exportação será o mais próximo possível do relatório;
- Pictures – habilita a exportação de imagens;
- Sub-folder – todos os arquivos extras são salvos em uma página separada chamada .files;
- Navigator – cria um navegador especial para navegar entre as páginas;
- Single Page – todas as páginas serão salvas em um arquivo.

---

A aparência e o tamanho do arquivo resultante dependem do template do relatório (veja na seção **Recommendation on report development**)

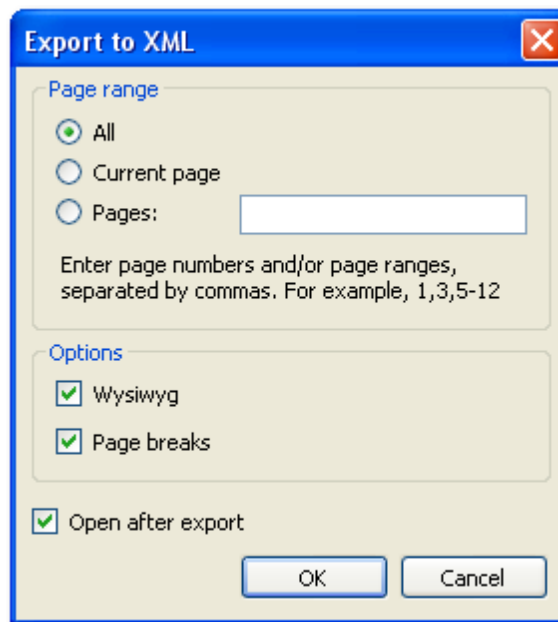
---

Exportar para o Excel (XML)

O Excel é uma aplicação para trabalhar com planilhas eletrônicas. Ele está incluso no Microsoft Office.

Método de exportação: tabular.

Quando exportarmos para o Excel, haverá uma janela para definir os parâmetros do arquivo que será gerado:



Parâmetros de exportação:

- Wysiwyg – o resultado será o mais próximo possível do relatório. Se esta opção estiver desabilitada, o **CIGAM Report** irá reduzir o número de linhas e colunas do arquivo gerado;
- Page breaks – habilita quebras de página no arquivo XLS.

---

A aparência e o tamanho do arquivo resultante dependem do template do relatório (veja na seção **Recommendation on report development**)

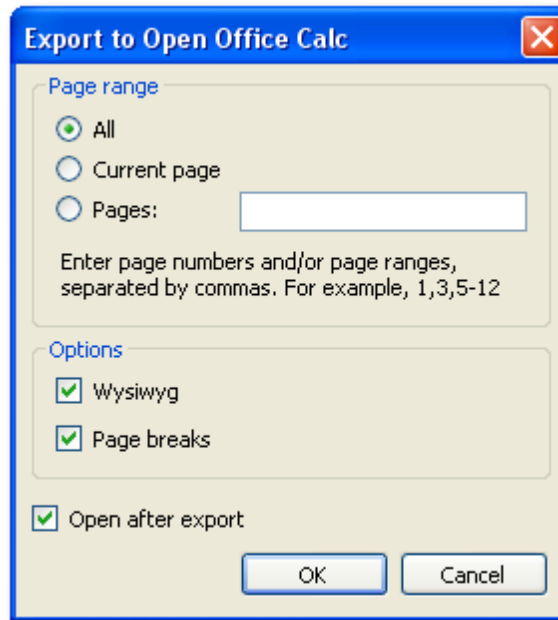
---

## Exportar para o OpenOffice Calc

O OpenDocument Format (ODF, OASIS Open Document Format for Office Application) foi designado pela OASIS e baseado no format XML usado no OpenOffice.

Método de exportação: tabular.

Quando exportarmos para o OpenOffice Calc, haverá uma janela para definir os parâmetros do arquivos que será gerado:



Parâmetros de exportação:

- Wysiwyg – o resultado será o mais próximo possível do relatório. Se esta opção estiver desabilitada, o **CIGAM Report** irá reduzir o número de linhas e colunas do arquivo gerado;
- Page breaks – habilita quebras de página no arquivo que será gerado.

---

A aparência e o tamanho do arquivo resultante dependem do template do relatório (veja na seção **Recommendation on report development**)

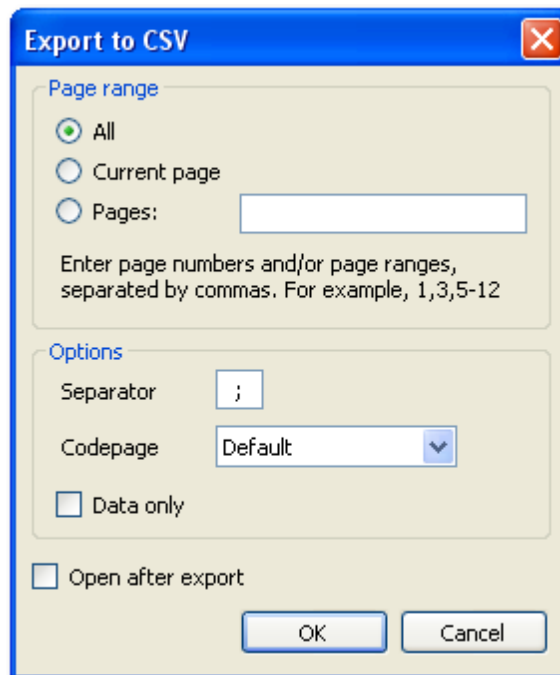
---

## Exportar para CSV

O arquivo CSV é usado para armazenamento digital de dados estruturados na forma de uma lista de tabelas. Cada linha do arquivo CSV corresponde a uma linha na tabela. Dentro de uma linha, os campos são separados por vírgulas, cada campo pertencendo a uma coluna da tabela. Arquivos CSV são freqüentemente usados para mover dados tabulares entre dois programas de computador diferentes, por exemplo entre um programa de banco de dados e um programa de planilhas.

Método de exportação: tabular.

Quando exportarmos para CSV, haverá uma janela para definir os parâmetros do arquivo que será gerado:



Parâmetros de exportação:

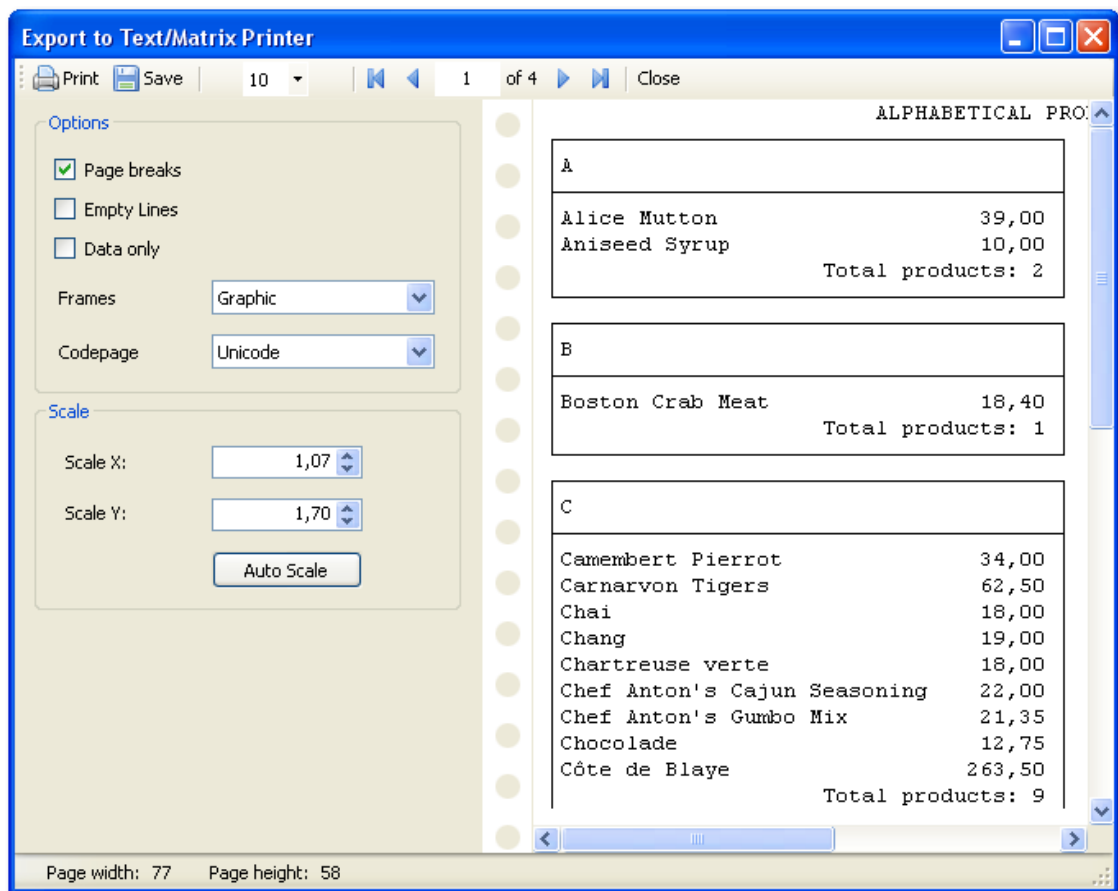
- Separator – o campo separador de caractere;
- Codepage – codificação usada para codificar o texto no arquivo gerado. A codificação Default refere-se ao codepage padrão do Windows. Note que o Excel não suporta a codificação Unicode;
- Data only – habilite este checkbox para exportar objetos que fiquem somente na banda Dados.

## Exportar para TXT

TXT é um arquivo de texto padrão que pode ser aberto por qualquer editor de texto, ou impresso em uma impressora matricial.

Método de exportação: tabular.

Quando exportarmos o TXT, haverá uma janela para definir os parâmetros do arquivo que será gerado:



Parâmetros de exportação:

- Page breaks – habilita quebras de página no arquivo que será gerado;
- Empty lines – habilita linhas em branco no arquivo que será gerado;
- Data only - habilite este checkbox para exportar objetos que fiquem somente na banda Dados;
- Frames – tipo das bordas do objeto. Selecione None se você não quer exportar bordas;
- Codepage – codificação usada para codificar o texto no arquivo que será gerado;
- Scale X- escala horizontal;
- Scale Y – escala vertical;
- Auto Scale – calcula a escala X e a escala Y automaticamente para evitar que dados sejam perdidos.

## Exportar para imagem

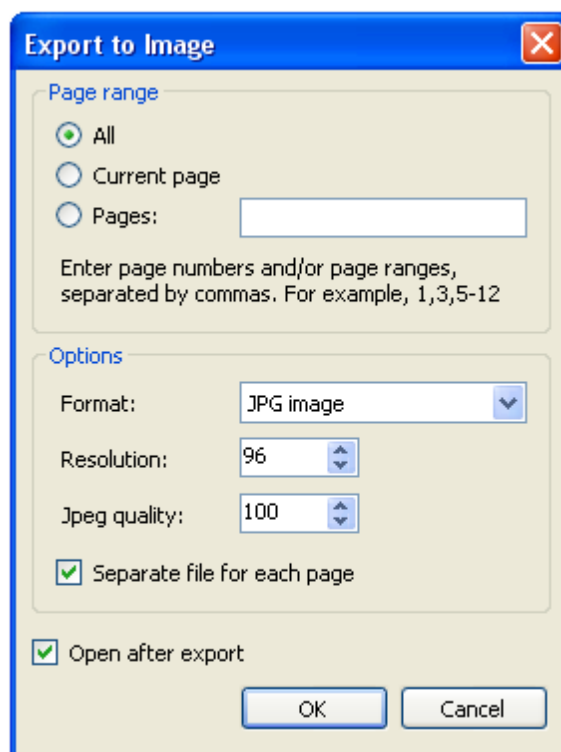
O **CIGAM Report** permite exportar informações para os seguintes formatos gráficos:

- BMP

- PNG
- JPG
- GIF
- TIFF
- Window Metafile (EMF, WMF)

Método de exportação: desenho.

Quando exportarmos um arquivo para imagem, haverá uma janela para definir os parâmetros:



Parâmetros de exportação:

- Separate file for each Page – se a opção estiver habilitada, então a página do relatório será exportada para um arquivo separado. O nome do arquivo será formado com base na página escolhida e no número dado;
- Resolution – resolução da imagem gráfica. Use 96dpi para exibição e 300dpi para impressão;
- Jpeg quality – nível de compactação do arquivo JPG. Esta opção é usada somente quando exportamos para o formato Jpeg.

---

Quando exportamos várias páginas para um arquivo (quando a opção Separate file for each Page estiver desmarcada), a exportação usará muitos recursos da CPU/Memória.

---

## Recomendações para desenvolvimento dos relatórios

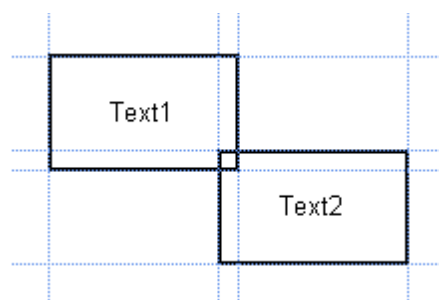
Neste capítulo, serão discutidos requerimentos especiais de desenvolvimento dos relatórios para exportar para outros formatos de dados.

O **CIGAM Report** permite um grande número de maneiras de manipular objetos durante a criação de um relatório. Isto dá vantagens rápidas de desenvolvimento de quaisquer relatórios e suas futuras impressões. O documento impresso ficará igual ao exibido na tela. E esta é a intenção primária do uso do gerador de relatório **CIGAM Report**.

O ponto negativo da liberdade de desenvolvimento é a complexidade da exportação do documento para diferentes formatos, os quais têm seus próprios limites e requerimentos para apresentação da informação, e às vezes são mais complexos. Muitos formatos, como o HTML, XLS ou RTF, usam apresentação de dados com tabelas. Estes formatos não permitem cruzar ou ajustar células nas camadas quando marcam tabelas.

Os filtros de exportação, como regra, levam em conta Estes requerimentos. Isto é calculado por um algoritmo especial que captura objetos sobrepostos e otimiza a disposição destes. Na sobreposição de objetos, há novas colunas e linhas que podem aparecer no relatório. Isto é necessário para manter a máxima semelhança entre o resultado e o relatório configurado originalmente. Um grande número de objetos sobrepostos em um relatório leva a um aumento direto no número de linhas e colunas da tabela resultante, o que afeta o tamanho do arquivo e a complexidade deste.

A qualidade de exportação do relatório depende muito da construção competente do relatório inicial. Vejamos o seguinte exemplo:



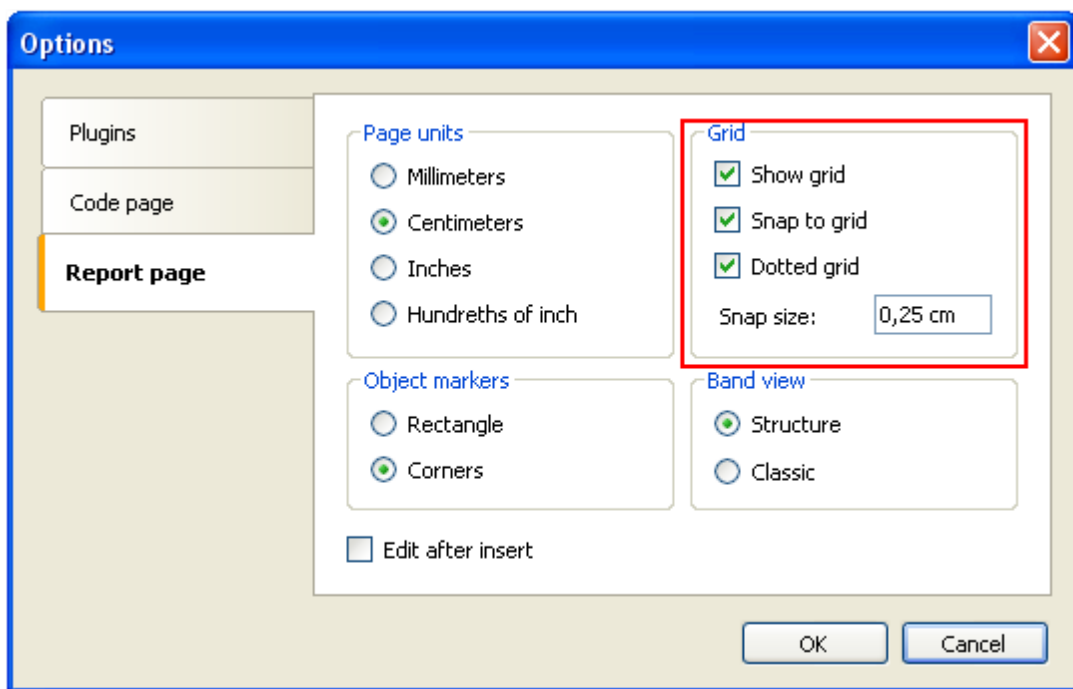
Há uma leve sobreposição de dois objetos na mesma banda. O número de registros na formação do relatório era de 150. Na exportação para formato **RTF**, 450 linhas serão criadas (150 para cada objeto, e 150 para a sobreposição). Se removermos a sobreposição, teremos apenas 300 linhas na tabela resultante. Em relatórios maiores e com um número considerável de objetos, a diferença será gritante. Isto, é claro, afeta o tamanho final do arquivo.

Quando criar tabelas no relatório, preste atenção às bordas da célula-vizinha. É importante que estas não se cruzem e nem se organizem em camadas. O algoritmo de filtragem da exportação irá cortar células, mas o resultado da exportação pode acabar se distanciando muito do desejado (você não



verá exatamente aquilo que queria). Arranje os objetos de maneira que estes estejam devidamente alinhados tanto vertical quanto horizontalmente. Linhas-guia podem ajudá-lo nisto.

O alinhamento de grade também pode ser muito útil no caso de células sobrepostas. Ative o alinhamento por grade nas opções do editor. Para simplificar o alinhamento você pode aumentar o afastamento das unidades da grade. A definição do afastamento e alinhamento da grade podem ser encontrados no menu **Visualizar > Opções**:



Para emoldurar texto é mais apropriado utilizar a borda deste do que objetos gráficos como linhas, retângulos e etc. Tente não utilizar objetos de fundo atrás de objetos de texto transparentes.

A aplicação destas simples regras irá auxiliá-lo a criar um relatório que terá uma aparência perfeitamente de acordo com a configuração inicial após ser exportado para qualquer formato baseado em tabelas.

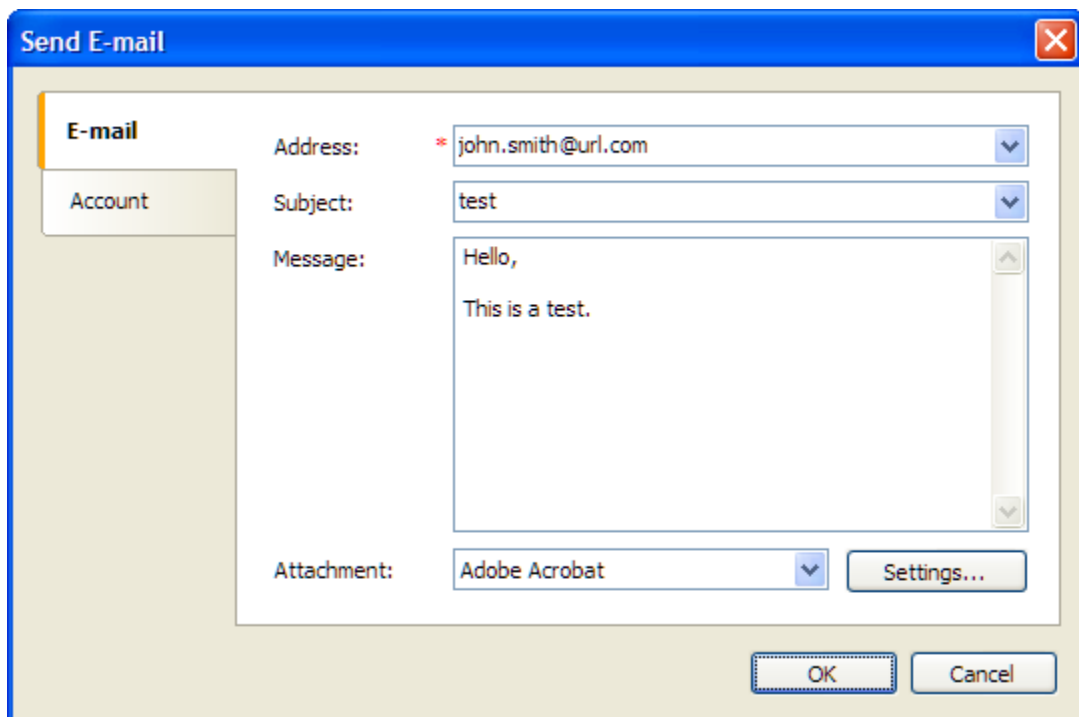
## Enviando o relatório por e-mail

O CIGAM Report permite o envio de um relatório preparado via e-mail. Isto funciona de duas maneiras:

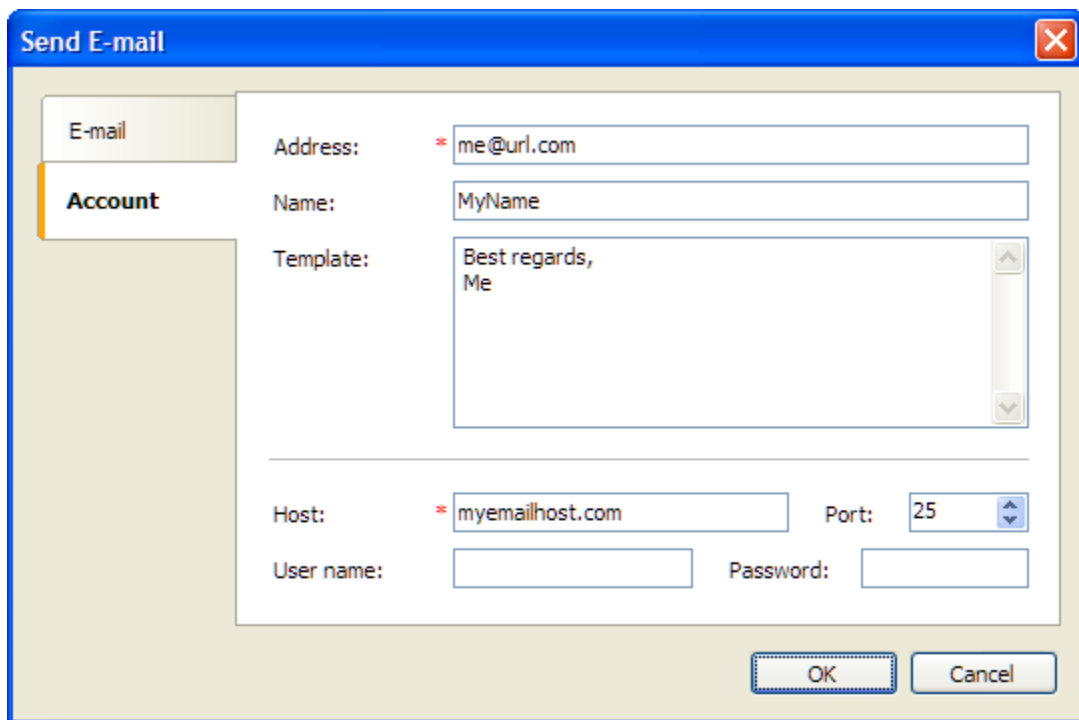
| Modo | Descrição   |
|------|---|
| SMTP | este é o método padrão. Para enviar um e-mail, você não necessita de nenhum programa externo.   |
| MAPI | Você pode ativar este modo, através de <i>script</i> . Para isto, defina a variável <b>Config.EmailSettings.UseMAPI = true</b> , ou, se você utiliza o componente <b>EnvironmentSettings</b> , defina a propriedade <b>EnvironmentSettings.EmailSettings.UseMAPI</b> para <b>true</b> . |

Para enviar um e-mail, o CIGAM Report utiliza o cliente de e-mail padrão, como por exemplo o **Outlook Express**. Este cliente deve suportar o protocolo MAPI.

Para enviar um e-mail, você deve especificar um destinatário. Você também deve especificar o assunto e o corpo do e-mail(embora estes não sejam obrigatórios). Selecione também o formato do anexo que será enviado:



Se você utiliza o modo **SMTP**, você deve configurar uma conta. Só é necessário fazer isto uma vez. Uma vez que isto tenha sido feito, o aplicativo irá salvar os parâmetros no arquivo de configuração. Os parâmetros podem ser encontrados na guia **Conta**. Todos os campos obrigatórios são marcados por um asterisco vermelho:



The image shows a 'Send E-mail' dialog box with a blue title bar and a close button (X) in the top right corner. On the left side, there is a vertical menu with two options: 'E-mail' and 'Account', with 'Account' currently selected. The main area contains several input fields: 'Address:' with the value '\* me@url.com', 'Name:' with the value 'MyName', and 'Template:' with the text 'Best regards, Me'. Below these, there are fields for 'Host:' with the value '\* myemailhost.com', 'Port:' with the value '25', 'User name:', and 'Password:'. At the bottom right, there are 'OK' and 'Cancel' buttons.

Se o seu servidor de e-mail necessita de uma autenticação, você deve preencher os campos **Usuário** e **Senha** também.