

# Capítulo 1

## Introdução ao QuickReport

Módulo I

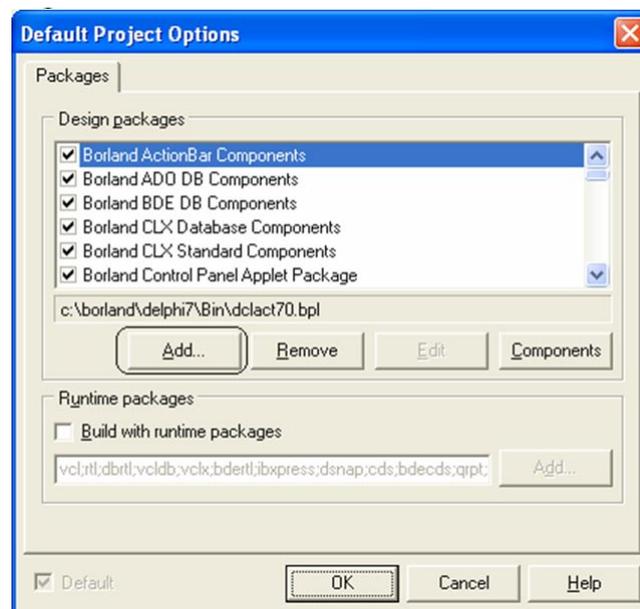


O QuickReport está presente no Delphi desde a versão 2 da ferramenta. O QuickReport foi desenvolvido e é mantido pela empresa QuSoft ([www.qusoft.com](http://www.qusoft.com)). Na versão 7 do Delphi, o QuickReport deixou de ser a ferramenta padrão para geração de relatórios, dando lugar ao Rave Reports, que será abordado no Módulo II. Apesar de não estar instalado por padrão na IDE do Delphi, o QuickReport continua sendo distribuído com a instalação do Delphi 7.

## Instalando o QuickReport no Delphi 7

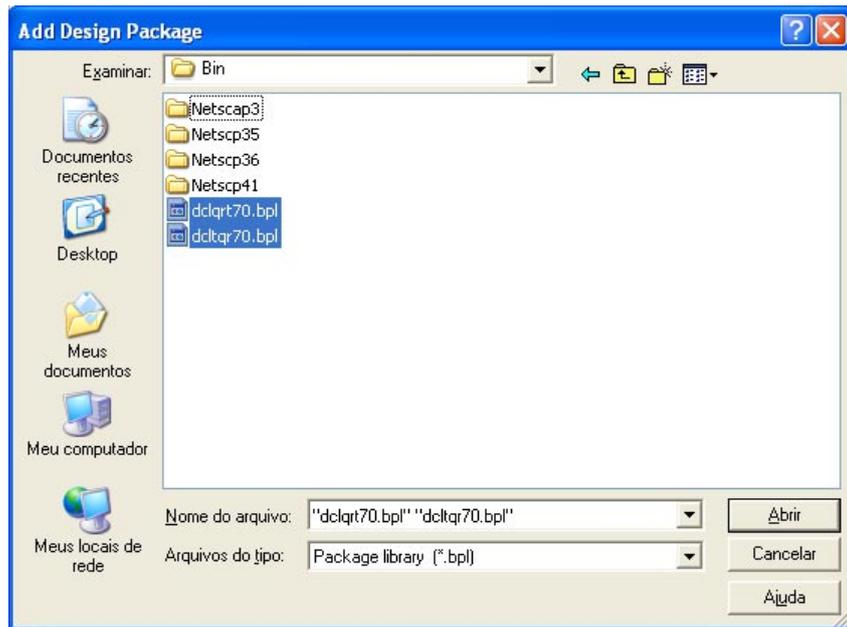
Os *packages* de instalação do QuickReport encontram-se no diretório *Bin* da instalação do Delphi. O arquivo *dclqrt70.bpl* possui os componentes padrões da paleta *QReport*. Já o arquivo *dcltqr70.bpl* possui o componente *QRChart* para a criação de gráficos dentro dos relatórios. Para fazer a instalação dos componentes, acesse o menu *Component|Install Packages*.

Na janela *Default Project Options*, dentro da aba *Packages*, clique no botão *Add* como mostra a **Figura 1.1**.



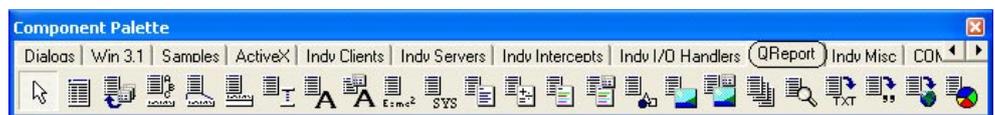
**Figura 1.1.** Janela Default Project Options para instalação de packages

Selecione os arquivos *dclqrt70.bpl* e *dcltqr70.bpl* (dentro da pasta *Bin*), como mostra a **Figura 1.2**.

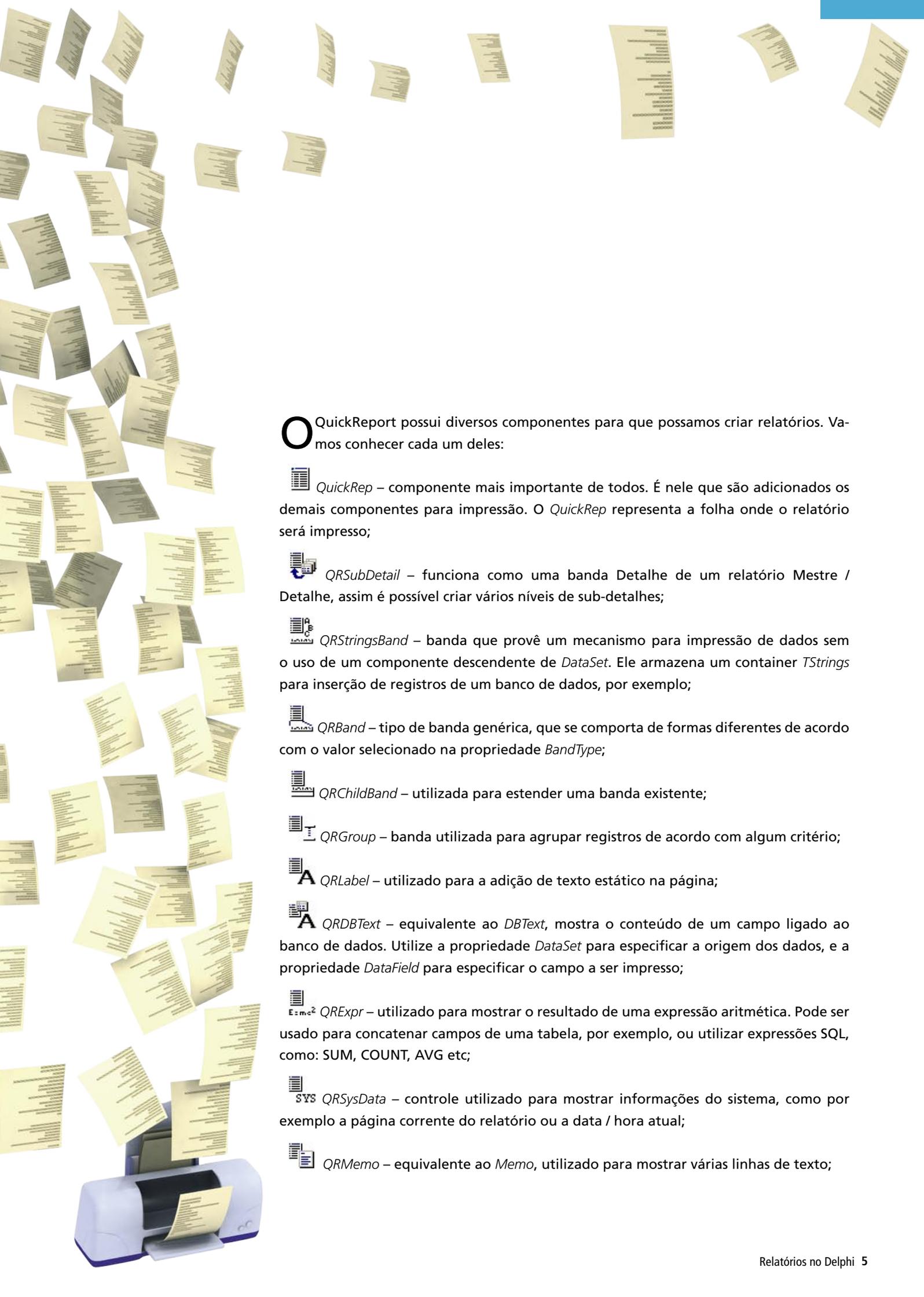


**Figura 1.2.** Selecionando os arquivos para a instalação dos componentes do QuickReport

Os componentes do QuickReport serão instalados na IDE do Delphi. Clique no botão OK para fechar a janela e confirmar a instalação dos pacotes. Todos os componentes do QuickReport são instalados dentro da paleta *QReport*, como mostra a **Figura 1.3**.







**O** QuickReport possui diversos componentes para que possamos criar relatórios. Vamos conhecer cada um deles:



**QuickRep** – componente mais importante de todos. É nele que são adicionados os demais componentes para impressão. O *QuickRep* representa a folha onde o relatório será impresso;



**QRSubDetail** – funciona como uma banda Detalhe de um relatório Mestre / Detalhe, assim é possível criar vários níveis de sub-detalhes;



**QRStringsBand** – banda que provê um mecanismo para impressão de dados sem o uso de um componente descendente de *DataSet*. Ele armazena um container *TStrings* para inserção de registros de um banco de dados, por exemplo;



**QRBand** – tipo de banda genérica, que se comporta de formas diferentes de acordo com o valor selecionado na propriedade *BandType*;



**QRChildBand** – utilizada para estender uma banda existente;



**QRGroup** – banda utilizada para agrupar registros de acordo com algum critério;



**QRLabel** – utilizado para a adição de texto estático na página;



**QRDBText** – equivalente ao *DBText*, mostra o conteúdo de um campo ligado ao banco de dados. Utilize a propriedade *DataSet* para especificar a origem dos dados, e a propriedade *DataField* para especificar o campo a ser impresso;



**QRExpr** – utilizado para mostrar o resultado de uma expressão aritmética. Pode ser usado para concatenar campos de uma tabela, por exemplo, ou utilizar expressões SQL, como: SUM, COUNT, AVG etc;



**QRSysData** – controle utilizado para mostrar informações do sistema, como por exemplo a página corrente do relatório ou a data / hora atual;



**QRMemo** – equivalente ao *Memo*, utilizado para mostrar várias linhas de texto;



*QRExprMemo* – uma composição dos componentes *QRExpr* e *QRMemo*. Pode ser utilizado para a inclusão de expressões em múltiplas linhas;



*QRRichText* – equivalente ao *RichText*, utilizado para mostrar várias linhas de texto com formatação RTF. Pode ser utilizado para imprimir o conteúdo de um componente *RichText*;



*QRDBRichText* – versão Data-Aware do *QRRichText*, utilizado para imprimir informações armazenadas em campos BLOB ou que possuam muito texto;



*QRShape* – equivalente ao *Shape*, utilizado no layout de relatórios, como criação de linhas divisórias e retângulos para agrupamentos.



*QRImage* – componente utilizado para mostrar imagens. Suporta diversos formatos, e pode mostrar a imagem em tempo de designer através da propriedade *Picture*;



*QRDBImage* – versão Data-Aware do *QRImage*, utilizado para imprimir imagens armazenadas em campos BLOB de uma tabela do banco de dados;



*QRCompositeReport* – utilizado para unir dois ou mais relatórios;



*QRPreview* – utilizado para a criação de um preview personalizado para visualização dos relatórios;



*QRTextFilter* – habilita a exportação do relatório para arquivo texto, formato ASCII;



*QRCSVFilter* – habilita a exportação do relatório para arquivo texto, formato CSV;



*QRHTMLFilter* – habilita a exportação do relatório para arquivo no formato HTML;



*QRChart* – equivalente ao *Chart*, é utilizado para criação de gráficos nos relatórios.

## O componente QuickRep

O *QuickRep* funciona como um container para os demais componentes da paleta *QReport*. Crie uma nova aplicação Delphi em *File|New>Application* e adicione um *QuickRep* (primeiro componente da esquerda para a direita na paleta *QReport*) ao formulário (Figura 2.1).





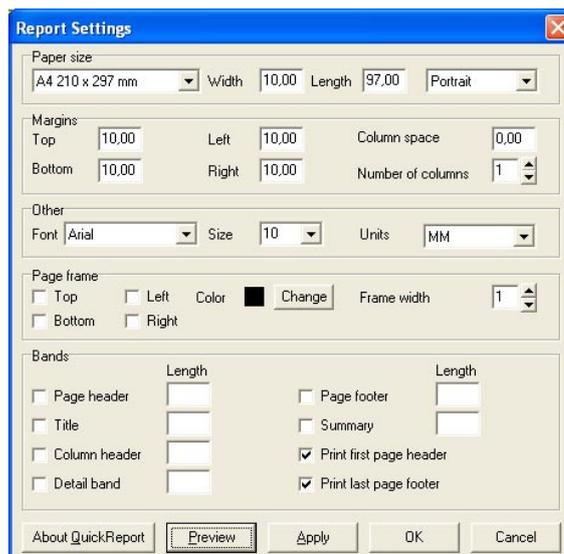
**Figura 2.1.** Componente QuickRep para criação de relatórios

Para ajustar o componente dentro do formulário, você pode utilizar as opções *Zoom in* e *Zoom out* disponíveis a partir do menu de contexto.



Menu de contexto refere-se ao menu de opções que aparece ao clicar com o botão direito do mouse sobre um determinado componente ou área.

Dando um duplo clique sobre o componente, ou selecionando a opção *Report Settings* a partir do menu de contexto, é possível acessar a janela *Report Settings* do componente, como mostra a **Figura 2.2**. Essa janela possui diversas opções relacionadas à página do relatório, como tamanho do papel, margens e fonte. No botão *Preview* localizado na parte inferior, é possível visualizar o relatório em tempo de *designer*, através da janela de visualização padrão do QuickReport.



**Figura 2.2.** Janela Report Settings para configuração do componente QuickRep



Na seção *Paper size* é possível definir o tamanho da folha, largura e comprimento, como também a disposição, se estará na vertical ou na horizontal. Na seção *Margins*, podemos definir o tamanho das margens da folha, o número de colunas que ela irá possuir e o espaçamento entre as colunas.

Na seção *Page frame*, é possível definir se a folha possuirá bordas e onde elas se localizarão. É possível ainda definir a largura e a cor da borda para a folha. Na seção *Bands* podemos incluir diversos tipos de bandas para serem usadas no relatório. Clique no botão OK para fechar a janela e confirmar as alterações.



Você pode adicionar bandas sem a necessidade de abrir a janela Report settings, para isso utilize a propriedade Bands do QuickRep.

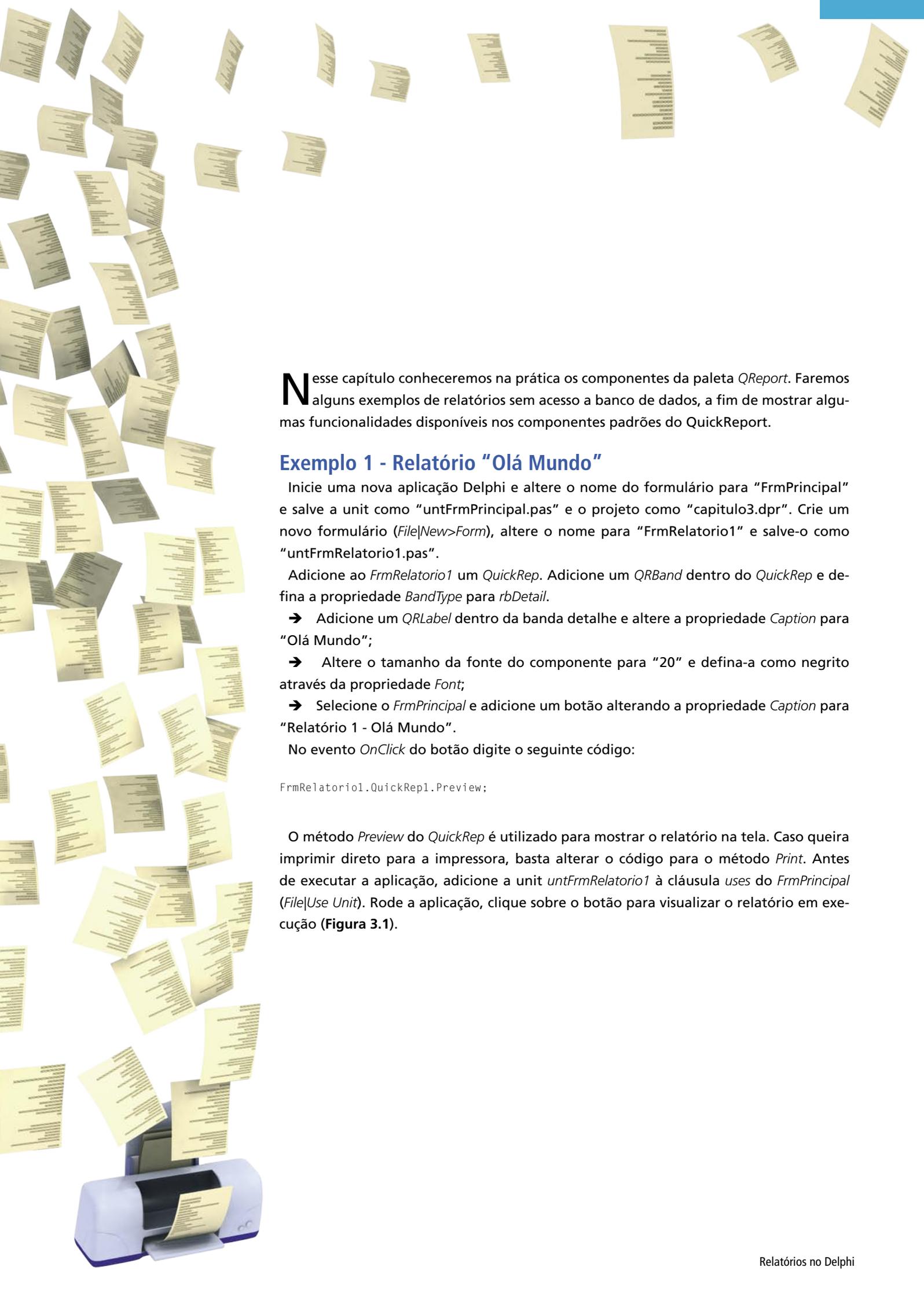


# Capítulo 3

Construindo relatórios  
sem acesso a dados

Módulo I





Nesse capítulo conheceremos na prática os componentes da paleta *QReport*. Faremos alguns exemplos de relatórios sem acesso a banco de dados, a fim de mostrar algumas funcionalidades disponíveis nos componentes padrões do QuickReport.

### Exemplo 1 - Relatório "Olá Mundo"

Inicie uma nova aplicação Delphi e altere o nome do formulário para "FrmPrincipal" e salve a unit como "untFrmPrincipal.pas" e o projeto como "capitulo3.dpr". Crie um novo formulário (*File|New>Form*), altere o nome para "FrmRelatorio1" e salve-o como "untFrmRelatorio1.pas".

Adicione ao *FrmRelatorio1* um *QuickRep*. Adicione um *QRBand* dentro do *QuickRep* e defina a propriedade *BandType* para *rbDetail*.

→ Adicione um *QRLabel* dentro da banda detalhe e altere a propriedade *Caption* para "Olá Mundo";

→ Altere o tamanho da fonte do componente para "20" e defina-a como negrito através da propriedade *Font*;

→ Selecione o *FrmPrincipal* e adicione um botão alterando a propriedade *Caption* para "Relatório 1 - Olá Mundo".

No evento *OnClick* do botão digite o seguinte código:

```
FrmRelatorio1.QuickRep1.Preview;
```

O método *Preview* do *QuickRep* é utilizado para mostrar o relatório na tela. Caso queira imprimir direto para a impressora, basta alterar o código para o método *Print*. Antes de executar a aplicação, adicione a unit *untFrmRelatorio1* à cláusula *uses* do *FrmPrincipal* (*File|Use Unit*). Rode a aplicação, clique sobre o botão para visualizar o relatório em execução (**Figura 3.1**).



Figura 3.1. Relatório em tempo de execução

## Exemplo 2 - Formas geométricas

Crie um novo formulário, altere o nome para "FrmRelatorio2" e salve-o como "untFrmRelatorio2.pas". Adicione um *QuickRep* ao formulário e também uma banda de detalhe utilizando o *QRBand*. Aponte a propriedade *BandType* do *QRBand* para *rbDetail*.

Nessa banda iremos criar algumas formas geométricas utilizando o *QRShape*. Aumente a altura da banda detalhe para que possamos adicionar os componentes. Quando um *QRShape* é adicionado a um relatório, por padrão ele vem definido como um retângulo.

Seu formato pode ser alterado de acordo com a opção selecionada na propriedade *Shape*. Adicione seis *QRShape* dentro da banda detalhe e altere sua propriedade *Shape* para cada opção disponível da lista, como mostra a **Figura 3.2**.

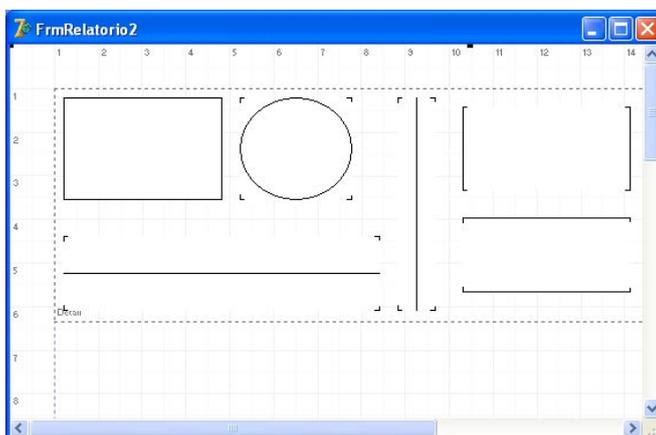


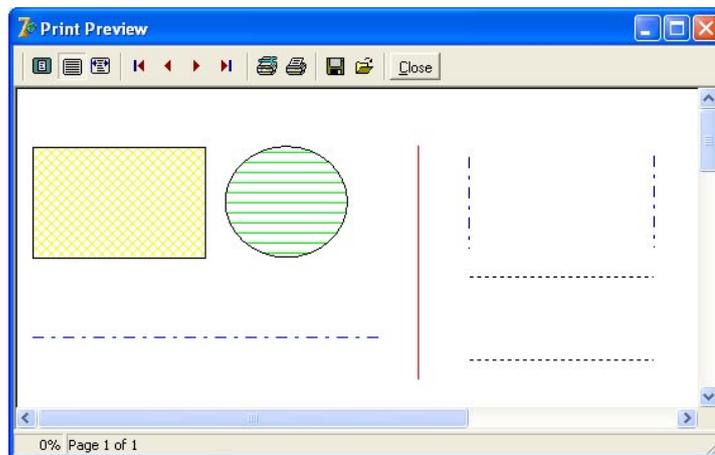
Figura 3.2. Definindo formas geométricas com o componente QRShape

Você pode alterar a cor e o estilo das formas geométricas através das propriedades *Brush.Color* e *Brush.Style*. Para alterar a cor e o estilo das bordas ou linhas utilize as propriedades *Pen.Color* e *Pen.Style*. Para executar o relatório, adicione um novo botão ao *FrmPrincipal*, altere a propriedade *Caption* para "Relatório 2 - Formas Geométricas" e no evento *OnClick* digite o seguinte código:

```
FrmRelatorio2.QuickRep1.Preview;
```



Adicione a unit do *FrmRelatorio2* ao *FrmPrincipal*. Rode a aplicação, clique sobre o botão adicionado anteriormente para visualizar o relatório em execução (**Figura 3.3**).



**Figura 3.3.** Relatório de formas geométricas em execução

## O componente *QRSysData*

O *QRSysData* é utilizado para mostrar o valor de uma variável de sistema. O valor a ser mostrado pelo componente depende da opção selecionada na propriedade *Data*, que possui as seguintes opções:

- *qrsDate*: data corrente;
- *qrsDateTime*: data e hora corrente;
- *qrsDetailCount*: número total de registros impressos na banda detalhe;
- *qrsDetailNo*: número corrente do registro impresso na banda detalhe;
- *qrsPageNumber*: número da página;
- *qrsReportTitle*: valor digitado na propriedade *ReportTitle* do *QuickRep*;
- *qrsTime*: hora corrente.

Essa propriedade pode ser selecionada tanto em tempo de *design* (a partir do *Object Inspector*), como em tempo de execução, como no código a seguir:

```
QRSysData1.Data := qrsDetailCount;
```

Além da propriedade *Data*, o *QRSysData* possui outra propriedade bastante importante, a propriedade *Text*. Essa propriedade é do tipo *string* e o valor nela digitado é impresso como um prefixo para o valor da propriedade *Data*, ou seja, primeiramente é impresso o valor de *Text* seguido do valor de *Data*.

## Exemplo 3 - Imprimindo informações do sistema

Crie um novo formulário, altere o nome para "*FrmRelatorio3*" e salve-o como "*untFrmRelatorio3.pas*". Adicione um *QuickRep* ao formulário e também um *QRBand* e altere a propriedade *BandType* para *rbDetail*.

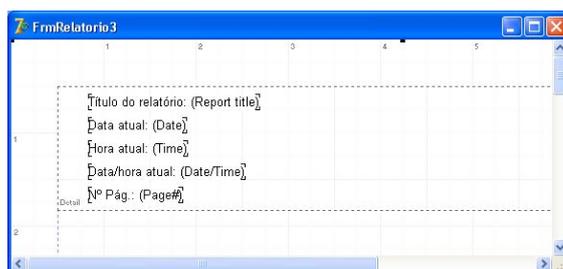
Nessa banda, iremos imprimir algumas informações do sistema utilizando o *QRSysData*. Primeiramente, digite "Exemplo 3 - Imprimindo informações do sistema" na propriedade *ReportTitle* do *QuickRep*, que será o título do relatório. Adicione cinco *QRSysData* dentro da banda detalhe e configure-os de acordo com a **Tabela 3.1**.



Componente	Data	Text
QRSysData1	<i>qrsReportTitle</i>	Título do relatório:
QRSysData2	<i>qrsDate</i>	Data atual:
QRSysData3	<i>qrsTime</i>	Hora atual:
QRSysData4	<i>qrsDateTime</i>	Data/hora atual:
QRSysData5	<i>qrsPageNumber</i>	Nº Pág.:

**Tabela 3.1.** Configurando o QRSysData para imprimir informações do sistema

Veja na **Figura 3.4**, como ficou o relatório.

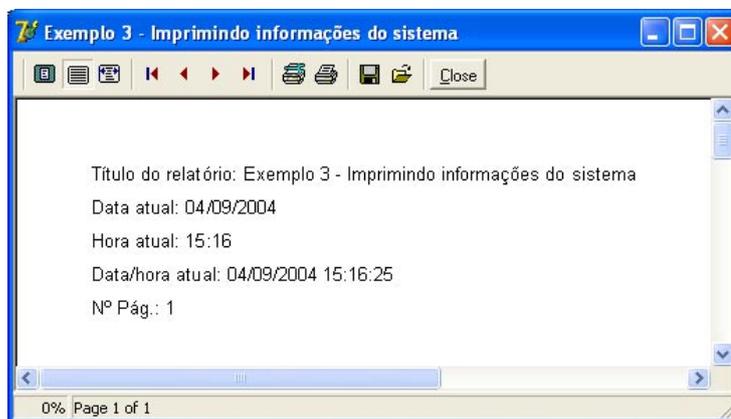


**Figura 3.4.** Opções disponíveis para o componente QRSysData

Para executar o relatório, adicione um novo botão ao *FrmPrincipal*, altere a propriedade *Caption* para "Relatório 3 - Informações do sistema" e digite o seguinte código para o evento *OnClick* do botão:

```
FrmRelatorio3.QuickRep1.Preview;
```

Adicione a *unit FrmRelatorio3* ao *FrmPrincipal*. Rode a aplicação, clique sobre o botão adicionado anteriormente para visualizar o relatório em execução (**Figura 3.5**).



**Figura 3.5.** Relatório com informações de sistema em tempo de execução



Repare que o valor informado na propriedade *ReportTitle* do *QuickRep* aparece na barra de título do relatório.



## O componente QRRichText

O *QRRichText* é utilizado para mostrar textos no formato RTF (*Rich Text Formatted*). O texto a ser impresso pelo *QRRichText* pode ser definido de várias maneiras. Podemos utilizar a propriedade *Lines*, do tipo *TStrings*, para adicionar o texto, tanto em tempo de *design* como em tempo de execução. O texto pode ser lido de um arquivo RTF externo utilizando o método *LoadFromFile* da propriedade *Lines*.

O *QRRichText* pode também mostrar o texto de um *RichEdit (Win32)* do Delphi, utilizando a propriedade *ParentRichEdit*. Uma outra propriedade importante do *QRRichText* é a *AutoStretch*. Essa propriedade é do tipo *Boolean (True ou False)*, e caso receba o valor *True*, o componente passa a controlar sua altura (propriedade *Height*) automaticamente para encaixar o texto a ser impresso.

## Exemplo 4 - Imprimindo textos RTF com QRRichText

Crie um novo formulário, altere o nome para "FrmRelatorio4" e salve-o como "untFrmRelatorio4.pas". Adicione um *QuickRep* ao formulário e uma banda detalhe (*QRBand* configurado como *rbDetail* na propriedade *BandType*) ao *QuickRep*.

Altere a propriedade *Page.Orientation* do *QuickRep* para *poLandscape*. Dessa forma, o relatório será mostrado no formato paisagem. Adicione três *QRRichText* dentro da banda detalhe. Para cada componente utilizaremos uma forma diferente de mostrar o texto a ser impresso. Para o primeiro *QRRichText* entre com um texto qualquer na propriedade *Lines* utilizando o *Object Inspector (Figura 3.6)*.

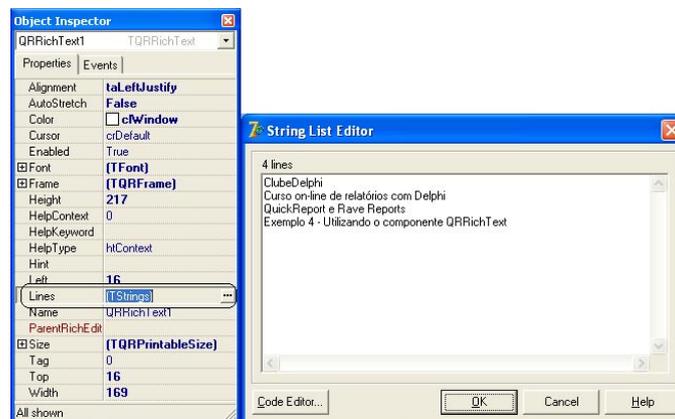


Figura 3.6. Propriedade Lines do componente QRRichText para entrada de texto

Para o segundo componente, iremos carregar o texto a partir de um arquivo RTF externo. Para isso, basta utilizar o método *LoadFromFile*. Esse método possui um parâmetro de entrada (*FileName*) do tipo *string*, onde deve ser informado o caminho completo para o arquivo RTF a ser carregado. No evento *OnCreate* do *FrmRelatorio4* digite o seguinte código (altere <caminho> pelo *path* do arquivo RTF):

```
QRRichText2.Lines.LoadFromFile(  
    '<caminho>:\arquivo.rtf');
```



Arquivos RTF podem ser criados a partir do Microsoft Word ou WordPad. Você pode criar um arquivo RTF com a formatação que desejar, ou utilizar o arquivo *clubedelphi.rtf* da sub-pasta *Capítulo III* do *Módulo I* disponível no CD.

Para o terceiro, e último *QRRichText* faremos a captura do texto a ser impresso a partir de um *RichText*. Selecione *FrmPrincipal* e adicione um *RichText* da paleta *Win32* (Figura 3.7).



**Figura 3.7.** RichText disponível na paleta Win32 do Delphi

Entre com um texto na propriedade *Lines* do componente. No evento *OnCreate* do *FrmRelatorio4* digite:

```
...
QRRichText3.ParentRichEdit :=
    FrmPrincipal.RichEdit1;
```

Adicione a *unit untFrmPrincipal* à cláusula *uses* do *FrmRelatorio4*. Para executar o relatório, adicione um novo botão ao *FrmPrincipal*, altere a propriedade *Caption* para "Relatório 4 - Componente QRRichText", e digite o seguinte código para o evento *OnClick* do botão:

```
FrmRelatorio4.QuickRep1.Preview;
```

Adicione a *unit FrmRelatorio4* ao *FrmPrincipal*. Rode a aplicação, clique sobre o botão adicionado anteriormente e visualize o relatório em execução (Figura 3.8).



**Figura 3.8.** Relatório com impressão de textos RTF utilizando o QRRichText





O QuickReport possui também um QRRichText Data-Aware (QRDBRichText) para acesso a informações armazenadas em banco de dados. Utilize o componente para imprimir textos RTF armazenados em campos BLOB de uma tabela do banco de dados.

## Utilizando o componente QRStringsBand

O *QRStringsBand* é um tipo de banda utilizada para impressão de dados armazenados em memória ou em arquivos texto. O *QRStringsBand* possui a propriedade *Items* do tipo *TStrings*, onde podemos carregar, por exemplo, informações de num *StringList* ou de um arquivo texto, através do método *LoadFromFile* da propriedade.

Cada linha inserida na propriedade *Items* diz respeito a um registro. O *QRStringsBand* é impresso uma vez para cada item inserido em sua propriedade *Items*.



Todo componente inserido dentro de uma banda *QRStringsBand* é impresso repetidamente junto com ela.

Para mostrar os dados na propriedade *Items*, devemos utilizar o *QRExpr*.



O *QRExpr* é mostrado em detalhes no capítulo *Trabalhando com Expressões* deste curso.

## Exemplo 5 - Trabalhando com dados em memória

Crie um novo formulário, altere o nome para "FrmRelatorio5" e salve-o como "untFrmRelatorio5.pas". Adicione um *QuickRep* ao formulário e coloque um *QRStringsBand* dentro do *QuickRep*. Vamos utilizar um *StringList* para inserir algumas informações em memória para que possamos visualizá-las através do *QRStringsBand*.



A classe *StringList* do Delphi é utilizada para armazenar e manipular uma lista de strings. Através do método *Add* da classe é possível inserir uma nova string a lista.



Na seção *private* da *unit untFrmRelatorio5* declare a variável *lista* do tipo *TStringList* como mostra o código a seguir:

```
private
  lista: TStringList;
```

No evento *OnCreate* do *FrmRelatorio5* digite o seguinte código:

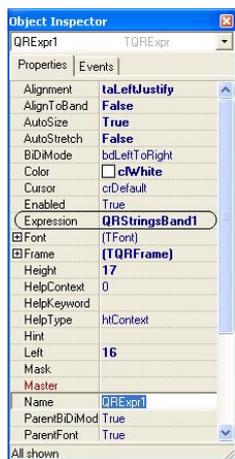
```
procedure TFrmRelatorio5.FormCreate(Sender: TObject);
var
  i: Integer;
begin
  { Cria a lista }
  lista := TStringList.Create;
  { Insere 100 registros dentro da lista }
  for i := 1 to 100 do
    lista.Add('Registro ' + IntToStr(i));
  { Atribui a variável lista a propriedade Items
  do QRStringsBand }
  QRStringsBand1.Items := lista;
end;
end;
```

No código anterior inserimos 100 registros dentro da variável *lista* e atribuímos esses registros a propriedade *Items* do *QRStringsBand*, para que possamos visualizá-los no relatório. Caso você queira trazer os registros a partir de um arquivo texto, basta substituir o código anterior por:

```
QRStringsBand1.Items.LoadFromFile(
  '<caminho>\arquivotexto.txt');
```

Para finalizar, precisamos mostrar esses registros dentro da banda *QRStringsBand* utilizando o *QRExpr* (somente esse componente pode exibir o conteúdo da propriedade *Items*).

Adicione um *QRExpr* dentro da banda *QRStringsBand* e na propriedade *Expression* entre com o nome da banda *QRStringsBand* como mostra a **Figura 3.9**.



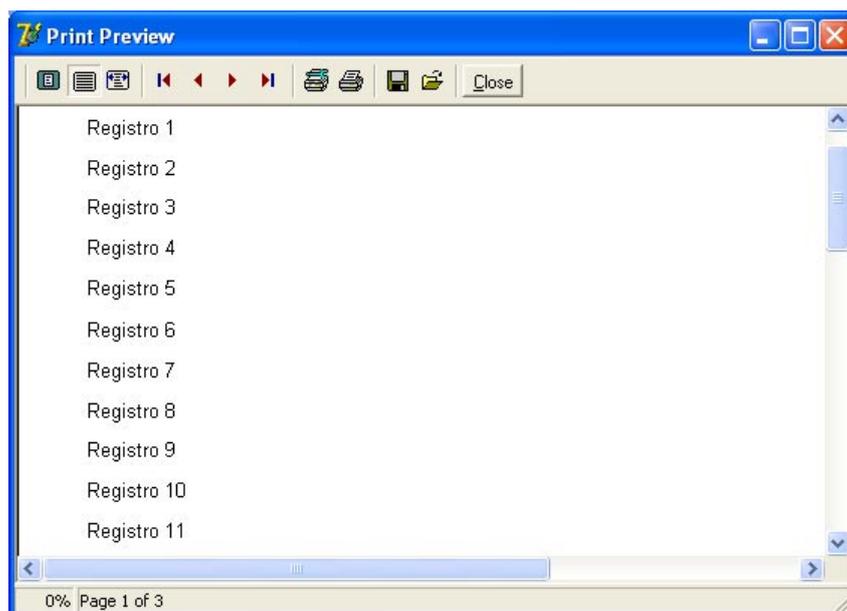
**Figura 3.9.** Configuração do QRExpr para visualização das informações armazenadas no QRStringsBand



Para executar o relatório, adicione um novo botão ao *FrmPrincipal*, altere a propriedade *Caption* para "Relatório 5 - Componente QRStringsBand", e digite o seguinte código para o evento *OnClick* do botão:

```
FrmRelatorio5.QuickRep1.Preview;
```

Adicione a *unit FrmRelatorio5* ao *FrmPrincipal*. Rode a aplicação, clique sobre o botão adicionado anteriormente e visualize o relatório em execução (**Figura 3.10**).



**Figura 3.10.** Relatório com impressão de dados em memória utilizando o StringsBand

## Exercícios

Exercício 1 - Crie um novo relatório utilizando o *QuickReport* com o título "Imprimindo informações gerais do sistema". Adicione uma banda detalhe ao relatório e exiba as seguintes informações:

- Título do relatório (alterando a fonte para negrito e tamanho "16");
- Data e Hora atual do sistema;
- Número da página atual.

Exercício 2 - Crie um arquivo texto contendo uma lista de nomes e utilize um relatório *QuickReport* para imprimir essa lista na tela.

## Solução

### Exercício 1

Crie uma nova aplicação, altere o nome do formulário para "FrmPrincipal" e salve-o como "untFrmPrincipal.pas" e o projeto como "Exercicio1.dpr". Adicione um novo formulário (*File|New>Form*), dê o nome de "FrmExercicio1" e salve-o como "untFrmExercicio1.pas". Adicione um *QuickRep* ao formulário e também um *QRBand*, alterando a propriedade *BandType* para *rbDetail*.



Na propriedade *ReportTitle* do *QuickRep* digite "Imprimindo informações gerais do sistema". Adicione três *QRSysData* dentro da banda detalhe e configure-os de acordo com a Tabela 3.2.

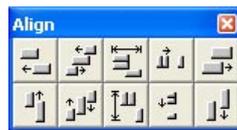
Componente	Data	Text
QRSysData1	<i>qrsReportTitle</i>	Título do relatório:
QRSysData2	<i>qrsDateTime</i>	Data e Hora atual:
QRSysData3	<i>qrsPageNumber</i>	Nº da Página atual:

**Tabela 3.2.** Configuração dos componentes *QRSysData*

Altere a propriedade *Font* do *QRSysData1* para negrito e o tamanho para "16".



Utilize a paleta de alinhamento do Delphi a partir do menu View|Alignment Palette para alinhar os componentes do relatório na página (Figura 3.11).



**Figura 3.11.** Paleta de alinhamento disponível no Delphi

Terminado as configurações dos componentes, o relatório deve estar parecido com a Figura 3.12.



**Figura 3.12.** Relatório em tempo de designer

Para executar o relatório, adicione um botão ao *FrmPrincipal*, altere a propriedade *Caption* para "Exercício 1", e digite o seguinte código para o evento *OnClick* do botão:

```
FrmExercicio1.QuickRep1.Preview;
```

Adicione a *unit FrmExercicio1* ao *FrmPrincipal*. Rode a aplicação e clique sobre o botão para visualizar o relatório em execução (Figura 3.13).



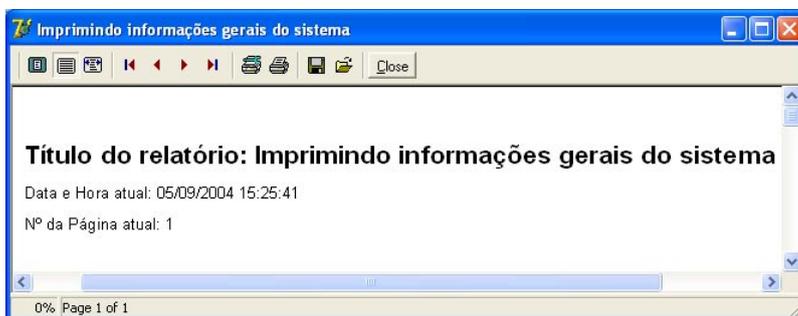


Figura 3.13. Relatório em tempo de execução

## Exercício 2

Abra o Bloco de Notas do Windows, crie um novo arquivo texto e adicione uma lista com dez nomes (Figura 3.14). Salve o arquivo com o nome de "clientes.txt", ou utilize o arquivo disponível na pasta *Módulo II/Capítulo III/ExercícioII* no CD do curso.

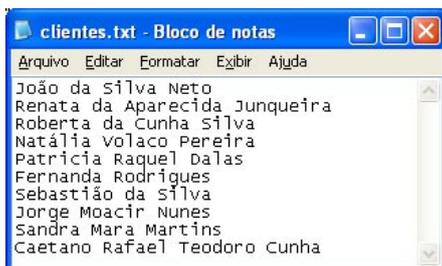


Figura 3.14. Arquivo texto com a lista de clientes

Crie uma nova aplicação, altere o nome do formulário para "FrmPrincipal" e salve-o como "untFrmPrincipal.pas" e o projeto como "Exercicio2.dpr". Adicione um novo formulário (*File|New>Form*), dê o nome de "FrmExercicio2" e salve-o como "untFrmExercicio2.pas".

Adicione um *QuickRep* ao formulário e coloque um *QRStringsBand* dentro do *QuickRep*. Para carregar os registros a partir do arquivo *clientes.txt*, no evento *OnCreate* do *FrmExercicio2* digite o seguinte código:

```
QRStringsBand1.Items.LoadFromFile(  
    '<caminho>\clientes.txt');
```

Adicione um *QRExpr* dentro da banda *QRStringsBand* e na propriedade *Expression* digite o nome da banda, nesse caso "QRStringsBand1". Para executar o relatório, adicione um botão ao *FrmPrincipal*, altere a propriedade *Caption* para "Exercício 2", e digite o seguinte código para o evento *OnClick* do botão:

```
FrmExercicio2.QuickRep1.Preview;
```



Adicione a *unit FrmExercicio2* ao *FrmPrincipal*. Rode a aplicação e clique sobre o botão para visualizar o relatório em execução (Figura 3.15).

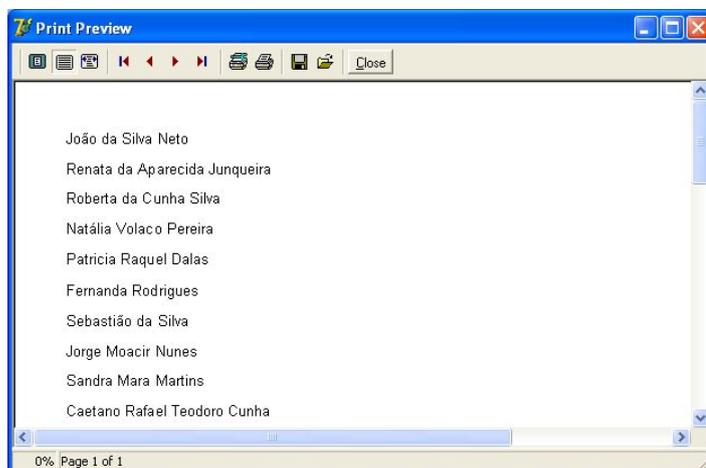


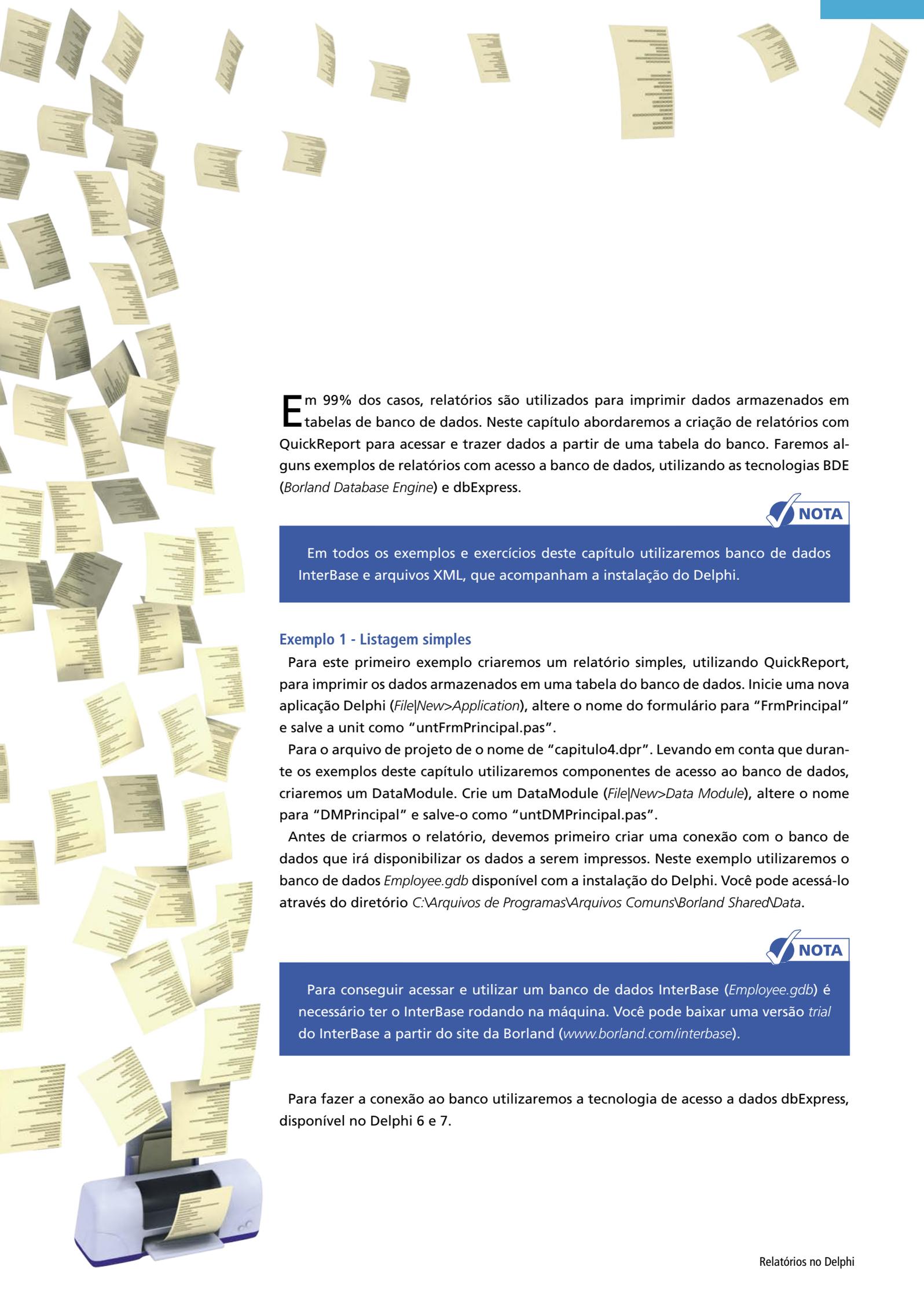
Figura 3.15. Relatório de listagem de clientes



# Capítulo 4

## Construindo relatórios com acesso a dados





**E**m 99% dos casos, relatórios são utilizados para imprimir dados armazenados em tabelas de banco de dados. Neste capítulo abordaremos a criação de relatórios com QuickReport para acessar e trazer dados a partir de uma tabela do banco. Faremos alguns exemplos de relatórios com acesso a banco de dados, utilizando as tecnologias BDE (*Borland Database Engine*) e dbExpress.



Em todos os exemplos e exercícios deste capítulo utilizaremos banco de dados InterBase e arquivos XML, que acompanham a instalação do Delphi.

### Exemplo 1 - Listagem simples

Para este primeiro exemplo criaremos um relatório simples, utilizando QuickReport, para imprimir os dados armazenados em uma tabela do banco de dados. Inicie uma nova aplicação Delphi (*File|New>Application*), altere o nome do formulário para "FrmPrincipal" e salve a unit como "untFrmPrincipal.pas".

Para o arquivo de projeto de o nome de "capitulo4.dpr". Levando em conta que durante os exemplos deste capítulo utilizaremos componentes de acesso ao banco de dados, criaremos um DataModule. Crie um DataModule (*File|New>Data Module*), altere o nome para "DMPrincipal" e salve-o como "untDMPrincipal.pas".

Antes de criarmos o relatório, devemos primeiro criar uma conexão com o banco de dados que irá disponibilizar os dados a serem impressos. Neste exemplo utilizaremos o banco de dados *Employee.gdb* disponível com a instalação do Delphi. Você pode acessá-lo através do diretório *C:\Arquivos de Programas\Arquivos Comuns\Borland Shared\Data*.



Para conseguir acessar e utilizar um banco de dados InterBase (*Employee.gdb*) é necessário ter o InterBase rodando na máquina. Você pode baixar uma versão *trial* do InterBase a partir do site da Borland ([www.borland.com/interbase](http://www.borland.com/interbase)).

Para fazer a conexão ao banco utilizaremos a tecnologia de acesso a dados dbExpress, disponível no Delphi 6 e 7.





Caso você esteja utilizando uma versão anterior a do Delphi 6, você poderá fazer a conexão ao banco InterBase utilizando o BDE (*Borland Database Engine*). Veremos no segundo exemplo deste capítulo como criar a conexão utilizando a tecnologia BDE.

Adicione um *SQLConnection (dbExpress)* ao *DMPPrincipal* e dê o nome de “conCapitulo4” (Figura 4.1).



Figura 4.1. Paleta dbExpress disponível no Delphi 6 e 7

Utilizaremos o *SQLConnection* para fazer a conexão ao banco de dados. Dando um duplo clique sobre o componente, ou selecionando a opção *Edit Connection Properties* a partir do menu de contexto, é possível acessar o editor de conexões (*dbExpress Connections*) como mostra a Figura 4.2.

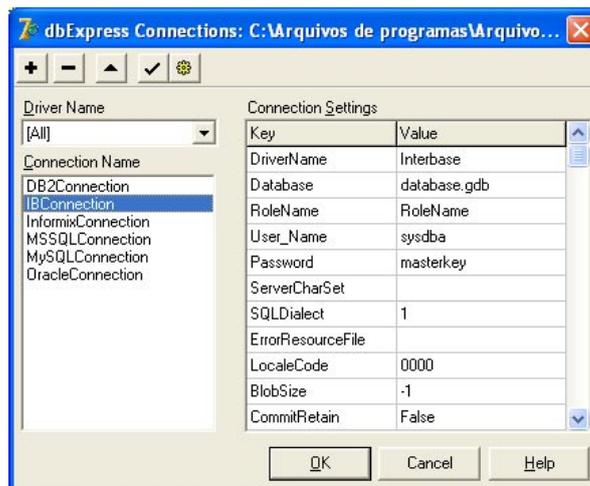


Figura 4.2. Janela para criação e configuração da conexão com o banco de dados

Clique sobre o botão *Add Connection*  para criar uma nova conexão. Configure a janela *New Connection* de acordo com a Figura 4.3.



Figura 4.3. Especificando o driver e o nome da conexão



Após criada a conexão, ela é adicionada em *Connection Name* no editor de conexões. Devemos agora indicar o caminho do banco de dados, neste exemplo o arquivo *Employee.gdb*. Na opção *Connection Settings*, digite o caminho do banco no campo *Value* da chave *Database* (Figura 4.4).

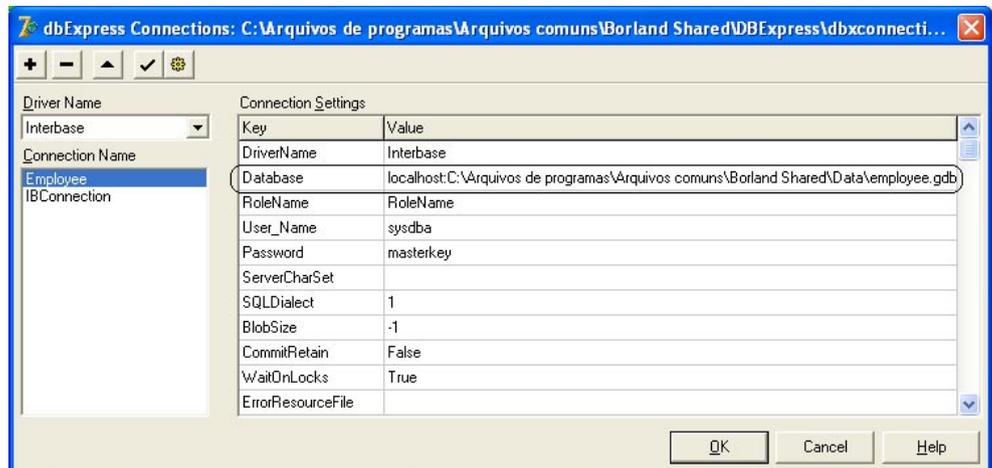


Figura 4.4. Configurando a nova conexão ao banco de dados Employee.gdb



Por padrão o InterBase traz um usuário (SYSDBA) e uma senha (*masterkey*) para acesso ao servidor. Caso você utilize outro usuário ou senha, basta alterar as chaves *User\_Name* e *Password*.



Altere o *localhost* pelo nome do servidor mais o caminho onde esta o banco de dados (GDB), caso o Interbase esteja instalado em um servidor e não na máquina local.

Para testar a nova conexão clique no botão *Test Connection* . Clique no botão OK para confirmar as alterações. Como a conexão já possui um usuário e senha configurado, podemos marcar como *False* a propriedade *LoginPrompt* do *SQLConnection*.

Adicione ao *DMPrincipal* um *SQLDataSet* (*dbExpress*), um *DataSetProvider* e um *ClientDataSet* (*Data Access*). Utilizaremos esses componentes para selecionar os dados do banco (Figura 4.5).



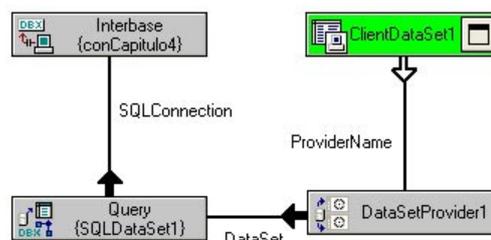


**Figura 4.5.** Data Module com os componentes de acesso a dados

Selecione o *SQLDataSet1* e aponte a propriedade *SQLConnection* para *conCapitulo4*. No *SQLDataSet1* é onde definimos a instrução SQL para selecionar os dados que desejamos. Na propriedade *CommandText* digite a seguinte instrução:

```
select * from CUSTOMER
```

Selecione o *DataSetProvider1* e aponte a propriedade *DataSet* para *SQLDataSet1*. Para o *ClientDataSet1*, aponte a propriedade *ProviderName* para *DataSetProvider1*. Os componentes devem estar configurados como mostra o diagrama da **Figura 4.6**.



**Figura 4.6.** Configuração dos componentes de acesso a dados

Terminada essa etapa, vamos criar o relatório. Crie um novo formulário (*File|New>Form*), altere o nome para "FrmRelatorio1" e salve-o como "untFrmRelatorio1.pas". Adicione ao *FrmRelatorio1* um *QuickRep* e também duas bandas, através do *QRBand*.

➔ Altere a propriedade *BandType* da primeira banda para *rbTitle* e da segunda para *rbDetail*;

➔ Adicione um *QRLabel* na banda título e altere a propriedade *Caption* para "Listagem de Clientes";

➔ Altere o tamanho da fonte do *QRLabel* para "22" e defina-a como negrito através da propriedade *Font*.

Para que o relatório possa mostrar os dados a serem impressos, através do *ClientDataSet*, adicione a unit *untDMPrincipal* a cláusula *uses* do *FrmRelatorio1*. Selecione o *QuickRep1* e aponte a propriedade *DataSet* para *DMPPrincipal.ClientDataSet1*.

Para mostrar os registros devemos utilizar o *QRDBText* (equivalente ao *DBEdit* da paleta *Data Controls*). Cada *QRDBText* é utilizado para mostrar o dado armazenado em um campo da tabela do banco de dados.

Vamos utilizar esse componente para mostrar o código, nome, telefone e cidade de cada cliente. Adicione quatro *QRDBText* dentro da banda detalhe e configure-os conforme a **Tabela 4.1**



Componente	DataSet	DataField
QRDBText1	<i>DMPrincipal.ClienteDataSet1</i>	CUST_NO
QRDBText2	<i>DMPrincipal.ClienteDataSet1</i>	CUSTOMER
QRDBText3	<i>DMPrincipal.ClienteDataSet1</i>	PHONE_NO
QRDBText4	<i>DMPrincipal.ClienteDataSet1</i>	CITY

**Tabela 4.1.** Configurações dos componentes QRDBText

O relatório deverá ficar como mostra a **Figura 4.7**.



**Figura 4.7.** Definindo a estrutura do relatório de listagem simples

O *QRDBText* possui duas propriedades essenciais: *DataSet* e *DataField*. A propriedade *DataSet* é utilizada para especificar o componente (descendente de *DataSet*) que possui os registros a serem impressos. A propriedade *DataField* é utilizada para especificar o campo da tabela que deverá ter seus valores impressos a partir do *QRDBText*.

Antes de fazer a chamada ao relatório devemos “abrir” o componente (neste exemplo o *ClientDataSet*) que possui os registros a serem impressos. No evento *OnCreate* do *FrmRelatorio1* digite o seguinte código:

```
DMPrincipal.ClienteDataSet1.Open;
```



Poderíamos ativar normalmente o *ClientDataSet* em tempo de *design*, alterando a propriedade *Active* para *True*.

Selecione o *FrmPrincipal* e adicione um botão, alterando a propriedade *Caption* para “Relatório 1 - Listagem simples”. Para o evento *OnClick* do botão digite:

```
FrmRelatorio1.QuickRep1.Preview;
```

Adicione a unit *untFrmRelatorio1* a cláusula *uses* do *FrmPrincipal*. Rode a aplicação, clique sobre o botão e visualize o relatório em execução na **Figura 4.8**.



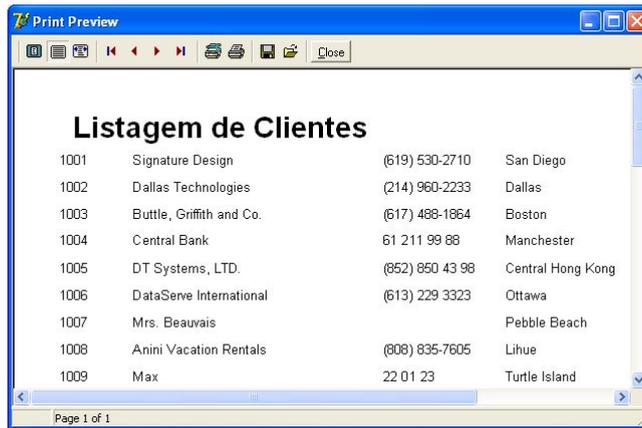


Figura 4.8. Relatório de listagem simples em tempo de execução

Para finalizar o exemplo, vamos criar uma nova banda para mostrar o título das colunas que estão sendo impressas. Adicione um novo *QRBand* ao *QuickRep* e altere a propriedade de *BandType* para *rbColumnHeader*. Adicione quatro *QRLabel* a nova banda e configure-os de acordo com a Tabela 4.2.

Componente	Caption	Font
QRLabel2	Código	<i>Negrito</i>
QRLabel3	Nome do Cliente	<i>Negrito</i>
QRLabel4	Telefone	<i>Negrito</i>
QRLabel5	Cidade	<i>Negrito</i>

Tabela 4.2. Configuração dos rótulos dos registros

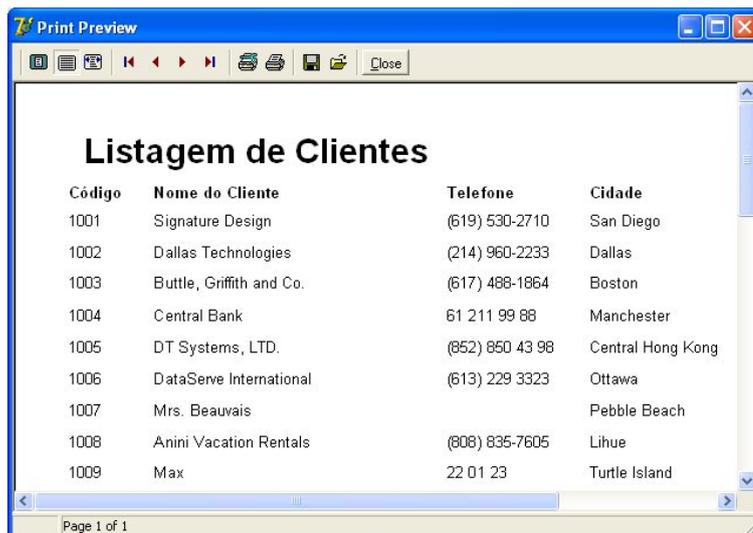
O relatório deverá ficar como mostra a Figura 4.9.



Figura 4.9. Definindo o título das colunas do relatório

Terminado as configurações na nova banda de cabeçalho das colunas, rode novamente a aplicação. Veja o relatório em execução na Figura 4.10.





**Print Preview**

**Listagem de Clientes**

Código	Nome do Cliente	Telefone	Cidade
1001	Signature Design	(619) 530-2710	San Diego
1002	Dallas Technologies	(214) 960-2233	Dallas
1003	Buttle, Griffith and Co.	(617) 488-1864	Boston
1004	Central Bank	61 211 99 88	Manchester
1005	DT Systems, LTD.	(852) 850 43 98	Central Hong Kong
1006	DataServe International	(613) 229 3323	Ottawa
1007	Mrs. Beauvais		Pebble Beach
1008	Anini Vacation Rentals	(808) 835-7605	Lihue
1009	Max	22 01 23	Turtle Island

Page 1 of 1

Figura 4.10. Relatório de listagem simples com título nas colunas impressas

## Exemplo 2 - Relatório colunado

Como no primeiro exemplo, criaremos um relatório para imprimir dados de uma tabela do banco de dados, porém, dessa vez os dados serão impressos em forma de colunas dentro do relatório, onde normalmente é usado para imprimir etiquetas. Para este exemplo faremos a conexão ao banco InterBase utilizando a tecnologia de acesso a dados BDE (*Borland Database Engine*).

Antes de adicionarmos os componentes de acesso a dados, devemos criar um alias (conexão) com o banco de dados *Employee.gdb*. Para isso, abra o BDE a partir do menu Iniciar do Windows (*Iniciar>Programas>Borland Delphi>BDE Administrator*). Veja o BDE em execução na Figura 4.11.

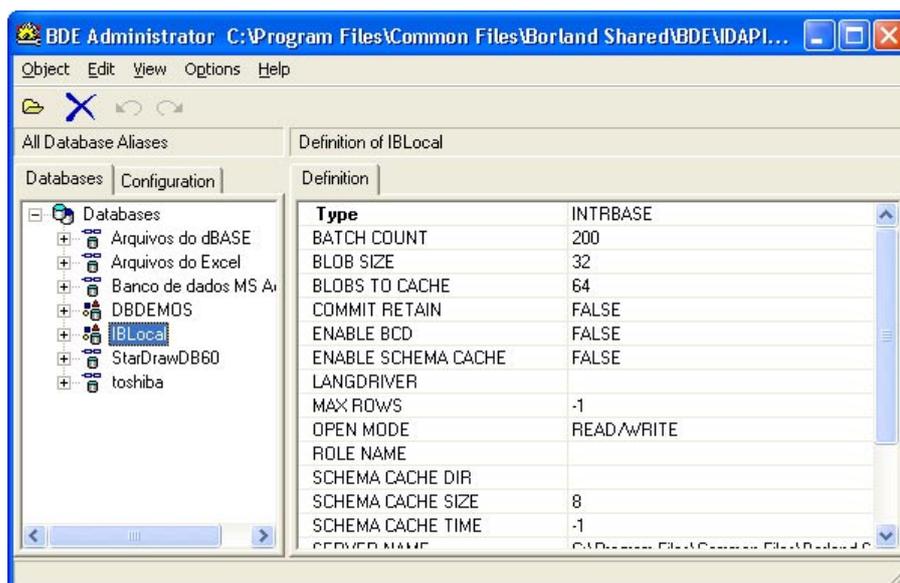


Figura 4.11. O Borland Database Engine em execução



Através do BDE é possível criar conexões para vários tipos de banco de dados, para isso selecione o menu *Object|New*, e na janela *New Database Alias* escolha o nome do *driver* a ser utilizado. Neste exemplo, selecione o *driver* para o InterBase (Figura 4.12).



Figura 4.12. Selecionando o nome do driver para o novo alias

Na aba *Databases* (painel direito) será criado um novo alias para o banco InterBase. Altere o nome do alias para "Employee" (opção *Rename* no menu de contexto). Ao selecionar o alias no painel esquerdo do BDE, serão mostradas as informações e parâmetros no painel direito.

No parâmetro *SERVER NAME* entre com o caminho do banco de dados *Employee.gdb* (que por padrão é: *C:\Arquivos de Programas\Arquivos Comuns\Borland Shared\Data\Employee.gdb*). Clique no botão *Apply* a partir da barra de ferramentas para salvar as configurações do novo alias.

Para testar a conexão, selecione o alias *Employee* e clique na opção *Open* a partir do menu de contexto. Na janela *Database Login* entre com o usuário (*SYSDBA*) e senha (*masterkey*). Caso a conexão tenha sido estabelecida com sucesso, o alias *Employee* deverá estar parecido com a Figura 4.13.

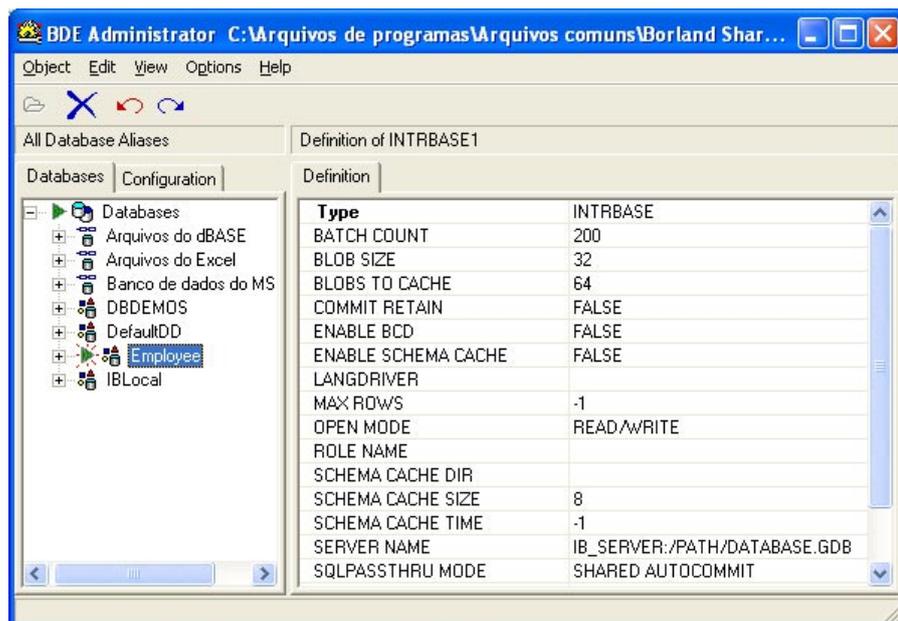


Figura 4.13. Alias Employee conectado ao banco de dados Employee.gdb



Terminado a criação do alias, feche o BDE. A aplicação Delphi utilizará esse alias para fazer a conexão ao banco de dados. Selecione o *DMPrincipal* e adicione um *Database (BDE)* (Figura 4.14).

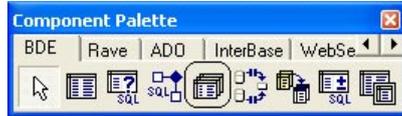


Figura 4.14. Paleta de componentes BDE



Para versões anteriores a do Delphi 6, os componentes do BDE estão localizados na paleta *Data Access*.

A propriedade *AliasName* do *Database* mostra todos os alias definidos dentro do *BDE Administrator*. Altere a propriedade *AliasName* do *Database* para *Employee*. Para a propriedade *DatabaseName* digite "DBEmployee". Altere a propriedade *Connected* do *Database* para *True*, será aberta a janela de login para a inserção do usuário e senha.

Para que essa janela não seja chamada a cada conexão com o banco de dados, selecione o *Database* e escolha a opção *Database Editor* a partir do menu de contexto. Na janela de configuração, configure os parâmetros em *Parameter overrides* e *Options*, de acordo com a Figura 4.15.

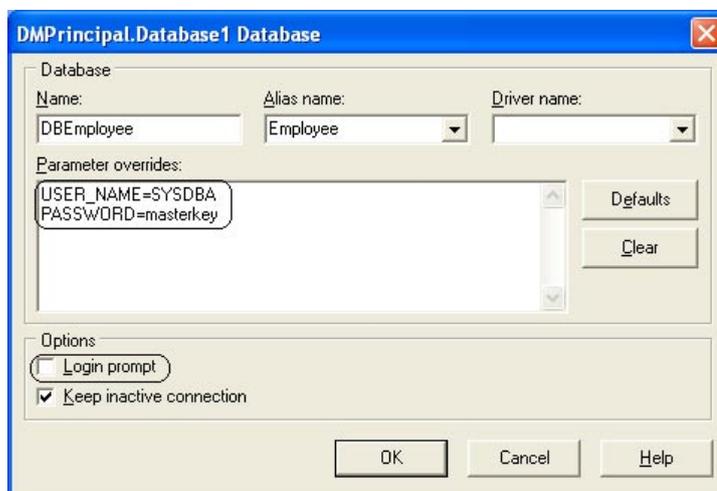


Figura 4.15. Definindo o usuário e senha padrão para acesso ao banco de dados

Adicione ao *DMPrincipal* um *Query (BDE)*. Através dele podemos passar instruções SQL para selecionar registros no banco de dados. Aponte a propriedade *DatabaseName* do *Query* para *DBEmployee*. Na propriedade *SQL* digite a seguinte instrução:

```
select DEPT_NO, DEPARTMENT, LOCATION
from DEPARTMENT
```

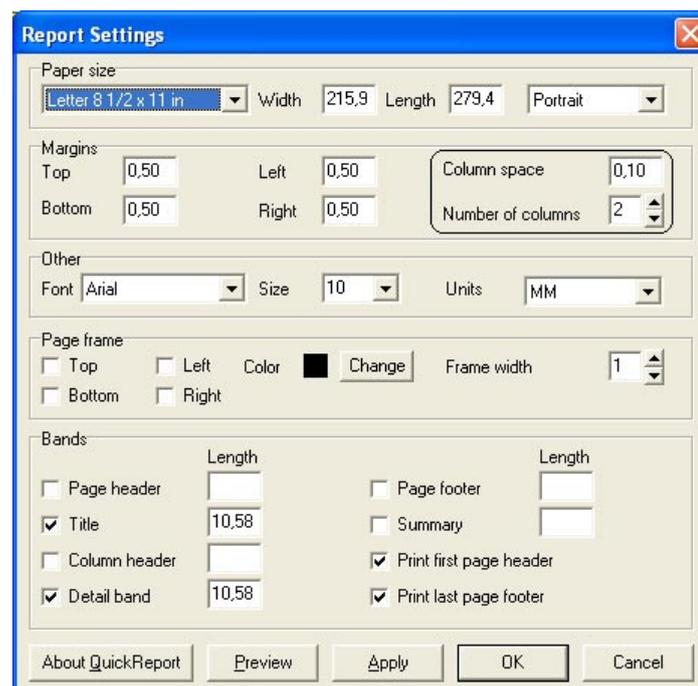


Terminada essa etapa, vamos criar o relatório para exibir os registros disponibilizados pelo *Query*. Crie um novo formulário, altere o nome para "FrmRelatorio2" e salve-o como "untFrmRelatorio2.pas". Adicione ao *FrmRelatorio2* um *QuickRep*. Vamos agora criar duas bandas ao relatório, adicionando dois *QRBand* dentro do *QuickRep*.

- Altere a propriedade *BandType* da primeira banda para *rbTitle* e da segunda para *rbDetail*;
- Adicione um *QLabel* dentro da banda título e altere a propriedade *Caption* para "Relatório de Departamentos";
- Altere o tamanho da fonte do *QLabel* para "22" e defina-a como negrito através da propriedade *Font*.

Para que possamos imprimir os dados da tabela de departamento (DEPARTMENT) em colunas dentro do relatório, precisamos configurar o *QuickRep*. Selecione o componente e dê um duplo clique sobre ele, ou selecione a opção *Report settings* a partir do menu de contexto para acessar a janela *Report Settings*.

Configure as propriedades *Column space* e *Number of columns* de acordo com a **Figura 4.16**.



**Figura 4.16.** Definindo o número de colunas do relatório

Através do item *Number of columns* é possível definir o número de colunas que o relatório irá possuir. Em *Column space* é determinado o espaçamento entre as colunas. Clique em OK para fechar a janela e confirmar as alterações.

Repare que a banda detalhe é ajustada de acordo com o número de colunas que informamos. Para que o relatório possa acessar o *Query1*, adicione a unit *untDMPrincipal* a cláusula *uses* do *FrmRelatorio2*.

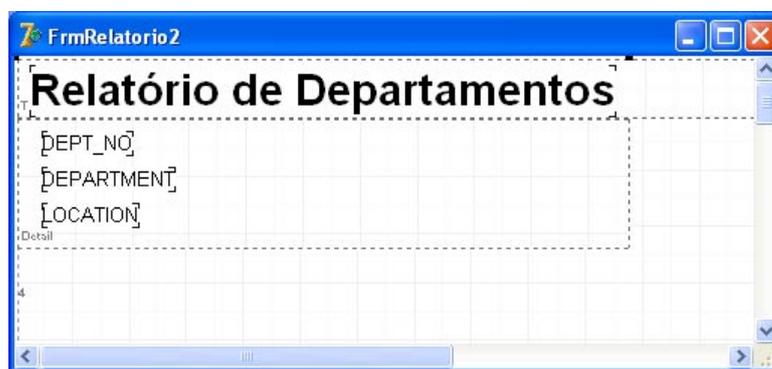


Selecione o *QuickRep1* e aponte a propriedade *DataSet* para *DMPrincipal.Query1*. Para mostrar os registros que estão armazenados no *Query1*, devemos utilizar o *QRDBText*. Adicione três *QRDBText* dentro da banda detalhe e configure-os conforme a **Tabela 4.3**.

Componente	DataSet	DataField
QRDBText1	<i>DMPrincipal.Query1</i>	DEPT_NO
QRDBText2	<i>DMPrincipal.Query1</i>	DEPARTMENT
QRDBText3	<i>DMPrincipal.Query1</i>	LOCATION

**Tabela 4.3.** Configurações dos componentes *QRDBText* no relatório de colunas

Distribua os componentes na banda de detalhe conforme a **Figura 4.17**.



**Figura 4.17.** Relatório colunado em tempo de design

Antes de fazer a chamada ao relatório devemos “abrir” o componente que possui os dados a serem impressos, neste exemplo o *Query*. No evento *OnCreate* do *FrmRelatorio2* digite o seguinte código:

```
DMPrincipal.Query1.Open;
```



Poderíamos ativar normalmente o *Query* em tempo de *design*, alterando a propriedade *Active* para *True*.

Selecione o *FrmPrincipal* e adicione um botão alterando a propriedade *Caption* para “Relatório 2 - Relatório Colunado”. Para o evento *OnClick* do botão digite o seguinte código:

```
FrmRelatorio2.QuickRep1.Preview;
```

Adicione a unit *untFrmRelatorio2* a cláusula *uses* do *FrmPrincipal*. Rode a aplicação, clique sobre o botão adicionado anteriormente e visualize o relatório (**Figura 4.18**).



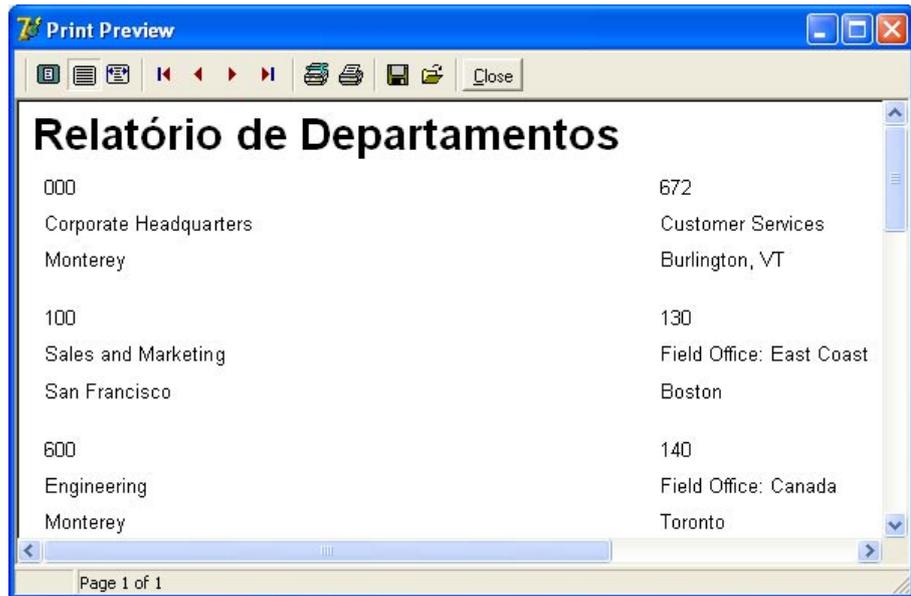


Figura 4.18. Relatório colunado em tempo de execução

### Exemplo 3 - Relatório com impressão de imagens

Nesse exemplo criaremos um relatório para imprimir dados armazenados em um arquivo XML. O arquivo utilizado é o *biolife.xml* disponível com a instalação do Delphi. O objetivo aqui é mostrar a impressão de imagens em relatórios do QuickReport.

Selecione o *DMPrincipal* e adicione um *ClientDataSet* (Data Access). Altere a propriedade *Name* para "cdsBiolife". Utilizaremos o *ClientDataSet* pra carregar o arquivo XML em memória para que possamos acessar e imprimir seus registros. Selecione o *cdsBiolife* e escolha a opção *Load from MyBase table* a partir do menu de contexto. Na janela *Open* selecione o arquivo *biolife.xml* (que por padrão está em *C:\Arquivos de Programas\Arquivos Comuns\Borland Shared\Data*). Repare que a propriedade *Active* do *ClientDataSet* é alterada para *True*. Isso indica que os dados armazenados no arquivo XML foram carregados em memória (*cache*).

Crie um novo formulário, altere o nome para "FrmRelatorio3" e salve-o como "untFrmRelatorio3.pas". Adicione ao *FrmRelatorio3* um *QuickRep* e três *QRBand*.

- Altere a propriedade *BandType* da primeira banda para *rbTitle*, da segunda para *rbColumnHeader* e da terceira para *rbDetail*;
- Adicione um *QRLabel* dentro da banda título e altere a propriedade *Caption* para "Relatório Biolife".
- Altere o tamanho da fonte do *QRLabel* para "18" e defina-a como negrito através da propriedade *Font*.

Para que o relatório possa acessar o *cdsBiolife* adicione a unit *untDMPrincipal* a cláusula *uses* do *FrmRelatorio3*. Selecione o *QuickRep1* e aponte a propriedade *DataSet* para *DMPrincipal.cdsBiolife*.

Na banda de título das colunas (*rbColumnHeader*) adicione quatro *QRLabel*. Vamos configurar esses componentes para mostrar o título dos campos do relatório. Configure-os de acordo com a Tabela 4.4.



Componente	Caption	Font
QRLabel1	Categoria	<i>Negrito</i>
QRLabel2	Nome	<i>Negrito</i>
QRLabel3	Espécie	<i>Negrito</i>
QRLabel4	Imagem	<i>Negrito</i>

**Tabela 4.4.** Configurações dos componente QRLabel no relatório

O relatório deve ficar como mostra a **Figura 4.19**.



**Figura 4.19.** Definição do título dos campos que serão impressos no relatório

Para mostrar os registros que estão armazenados no *cdsBiolife*, utilizaremos o *QRDBText*. Adicione três *QRDBText* dentro da banda detalhe. Configure-os de acordo com a **Tabela 4.5**.

Componente	DataSet	DataField
QRDBText1	<i>DMPrincipal.cdsBiolife</i>	Category
QRDBText2	<i>DMPrincipal.cdsBiolife</i>	Common_Name
QRDBText3	<i>DMPrincipal.cdsBiolife</i>	Species Name

**Tabela 4.5.** Configurações do QRDBText no relatório

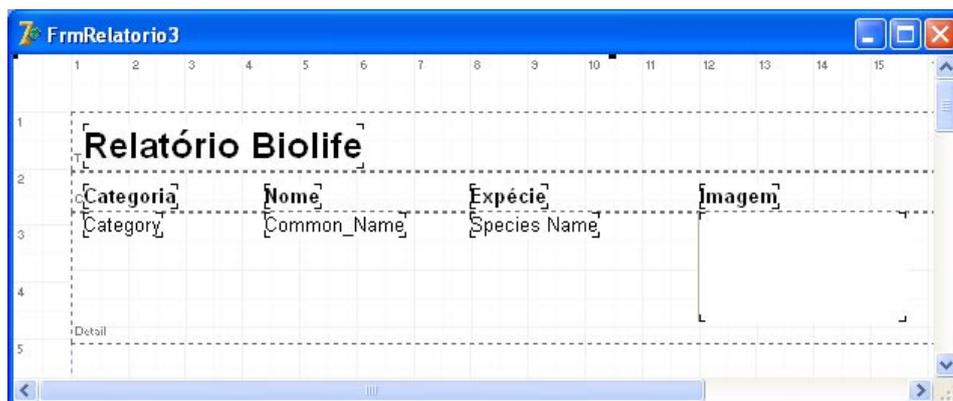
Para imprimir as imagens, devemos utilizar o *QRDBImage* da paleta *QuickRep*, que tem a finalidade de imprimir imagens armazenadas em banco de dados ou arquivos XML. Como o *QRDBText*, o *QRDBImage* também possui as propriedades *DataSet* e *DataField*. Adicione um *QRDBImage* e aponte a propriedade *DataSet* para *DMPrincipal.cdsBiolife* e *DataField* para *Graphic*. Para terminar a configuração do *QRDBImage* altere sua propriedade *Stretch* para *True*.



A propriedade *Stretch* do *QRDBImage* é utilizada para ajustar o tamanho da imagem automaticamente dentro do componente.



Veja como ficou o layout do relatório na **Figura 4.20**.

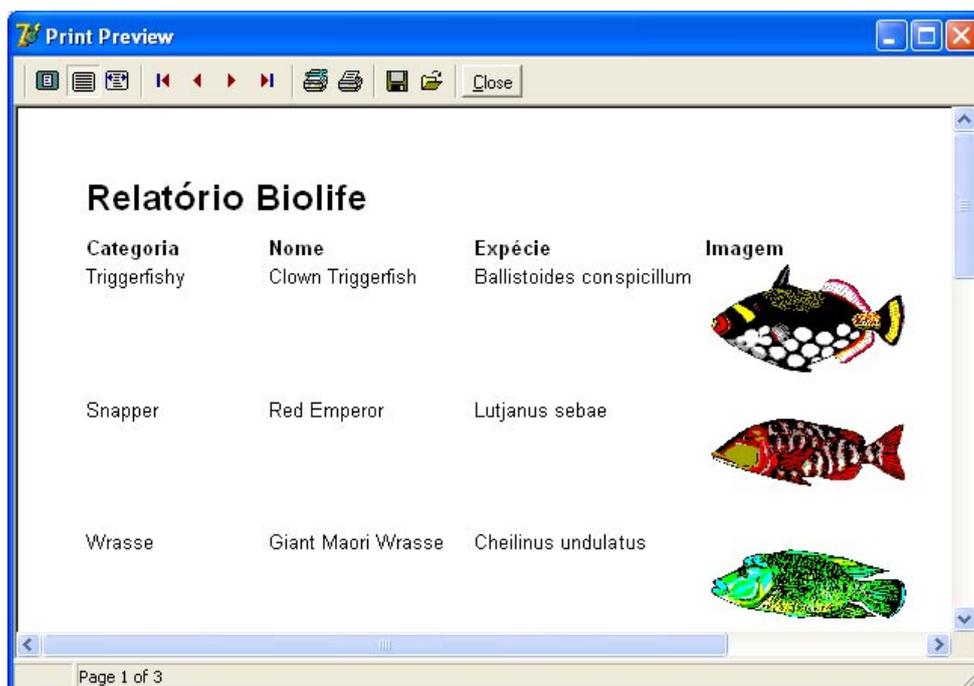


**Figura 4.20.** Relatório com impressão de imagens em tempo de design

Selecione o *FrmPrincipal* e adicione um botão, alterando a propriedade *Caption* para “Relatório 3 - Relatório com Impressão de Imagens”. No evento *OnClick* do botão digite o seguinte código:

```
FrmRelatorio3.QuickRep1.Preview;
```

Adicione a unit *untFrmRelatorio3* a cláusula *uses* do *FrmPrincipal* e verifique se o *cdsBiolife* possui a propriedade *Active* definida como *True*. Rode a aplicação, clique sobre o botão adicionado anteriormente e visualize o relatório em execução na **Figura 4.21**.



**Figura 4.21.** Relatório com impressão de imagens



Para terminar nosso exemplo, vamos mostrar no final do relatório o número total de registros impressos. Adicione uma nova banda (*QRBand*) ao relatório, definindo a propriedade *BandType* para *rbSummary*.



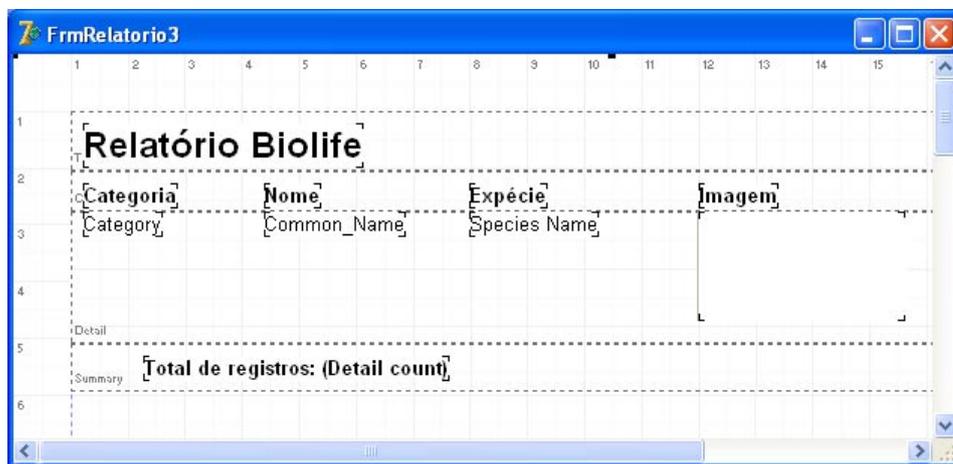
A banda do tipo sumário (*rbSummary*) é impressa sempre no final do relatório, após todos os registros terem sido impressos.

Adicione um *QRSysData* dentro da banda sumário. Altere a propriedade *Text* para "Total de registros: " e a propriedade *Data* para *qrsDetailCount*.



A opção *qrsDetailCount* é utilizada para contar o número de vezes que a banda detalhe é impressa dentro do relatório, ou seja, a opção retorna a quantidade de registros impressos na banda detalhe.

Altere a fonte do *QRSysData* para negrito (propriedade *Font*). Veja como ficou o relatório em tempo de *design* na **Figura 4.22**.



**Figura 4.22.** Configuração da banda do tipo sumário

Rode a aplicação novamente e veja o relatório em execução na **Figura 4.23**, mostrando a quantidade de registros impressos no relatório.



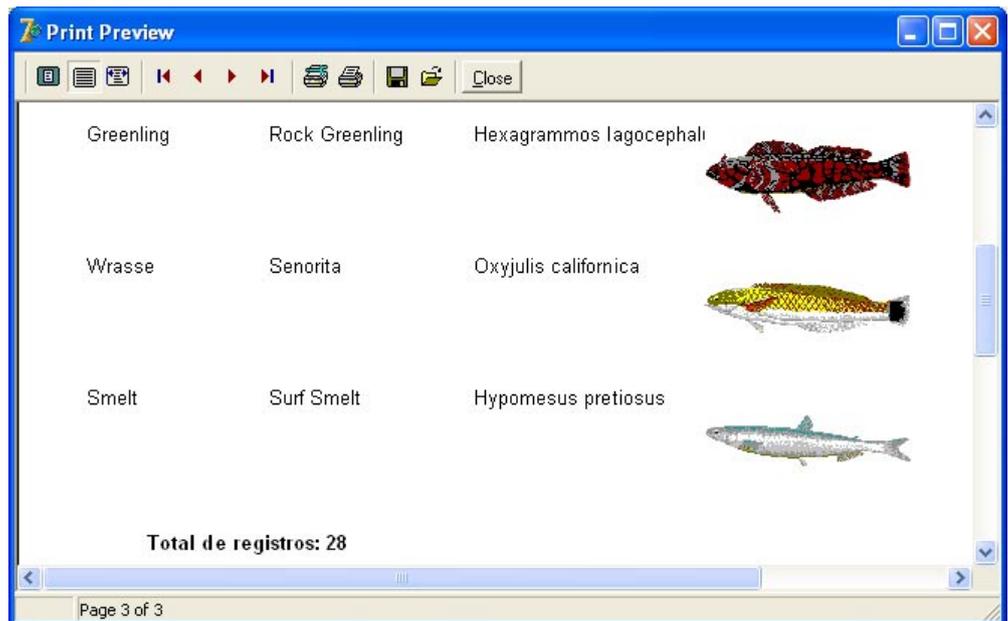


Figura 4.23. Banda sumário com o total de registros impressos no relatório

## Exercícios

Exercício 1 – Crie um relatório utilizando o QuickReport com o título “Lista de Empregados”. Adicione três bandas ao relatório: título, cabeçalho das colunas e detalhe.

- Vincule o relatório a tabela EMPLOYEE do banco de dados *Employee.gdb* (utilizando dbExpress para o acesso aos dados).
- Para a banda título imprima o título do relatório e a data / hora atual do sistema.
- Para a banda de cabeçalho das colunas, adicione os seguintes títulos: Código, Primeiro Nome, Último Nome, Telefone e Salário.
- Na banda detalhe imprima os dados dos seguintes campos: EMP\_NO, FIRST\_NAME, LAST\_NAME, PHONE\_EXT e SALARY.

### OBSERVAÇÃO

Para esse exercício utilize a conexão *Employee* criada no exemplo 1 deste capítulo.

Exercício 2 – Crie um relatório utilizando o QuickReport para imprimir a lista de vendas efetuadas por todos os empregados. Para o título do relatório coloque “Listagem de Vendas Efetuadas”. Adicione quatro bandas ao relatório de vendas: título, cabeçalho das colunas, detalhe e sumário.

- Construa uma instrução SQL para buscar dados contidos na tabela SALES, CUSTOMER e EMPLOYEE do banco de dados *Employee.gdb* que acompanha o InterBase. Utilize o BDE para o acesso aos dados;
- Para a primeira banda do relatório, a banda de título, entre com o título do relatório;
- Para a banda cabeçalho das colunas adicione os seguintes títulos para as colunas: Nº Venda, Data Venda, Vendedor, Cliente, Quant. e Valor;
- Na banda detalhe imprima os dados dos seguintes campos das tabelas referenciadas na instrução SQL do relatório: PO\_NUMBER, ORDER\_DATE, FIRST\_NAME, LAST\_NAME, CUSTOMER, QTY\_ORDERED e TOTAL\_VALUE;



- Na banda sumário imprima a quantidade total e o valor total de todas as vendas impressas no relatório.

## Solução

### Exercício 1

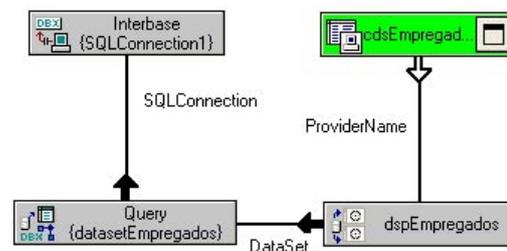
Crie uma nova aplicação no Delphi, dê o nome de "FrmPrincipal" ao formulário e salve-o como "untFrmPrincipal.pas", para o projeto salve como "Exercicio1.dpr". Adicione o DataModule *DMPrincipal* do projeto *capitulo4.dpr*, através do menu *Project\Add to Project*.

Adicione ao *DMPrincipal* um *SQLDataSet (dbExpress)*, um *DataSetProvider* e um *ClientDataSet (Data Access)*.

- ➔ Altere a propriedade *Name* do *SQLDataSet* para "datasetEmpregados";
- ➔ Altere a propriedade *Name* do *DataSetProvider* para "dspEmpregados";
- ➔ Altere a propriedade *Name* do *ClientDataSet* para "cdsEmpregados";
- ➔ Ligue o *datasetEmpregados* ao *SQLConnection1*, através da propriedade *SQLConnection*. Para a propriedade *CommandText* digite a seguinte instrução SQL:

```
select * from EMPLOYEE
```

Configure os outros componentes de acordo com o diagrama da **Figura 4.24**.



**Figura 4.24.** Configuração dos componentes

Crie um novo formulário, altere o nome para "FrmExercicio1" e salve-o como "untFrmExercicio1.pas". Adicione ao *FrmExercicio1* um *QuickRep*. Altere a propriedade *ReportTitle* do *QuickRep* para "Lista de Empregados". Adicione três bandas (*QRBand*) ao relatório.

- ➔ Altere a propriedade *BandType* da primeira banda para *rbTitle*, da segunda para *rbColumnHeader* e da terceira para *rbDetail*;
- ➔ Adicione um *QRSysData* dentro da banda título e altere a propriedade *Data* para *qrsReportTitle*;
- ➔ Altere o tamanho da fonte do *QRSysData* para "20" e defina-a como negrito através da propriedade *Font*.

Para acessar o *cdsEmpregados* adicione a unit *untDMPrincipal* a cláusula *uses* do *FrmExercicio1*. Selecione o *QuickRep1* e aponte a propriedade *DataSet* para *DMPrincipal.cdsEmpregados*. Na banda de título das colunas (*rbColumnHeader*) adicione cinco *QRLabel*. Configure os componentes, que irão mostrar o título das colunas do relatório, conforme a **Tabela 4.6**.



Componente	Caption	Font
QRLabel1	Código	<i>Negrito</i>
QRLabel2	Primeiro nome	<i>Negrito</i>
QRLabel3	Último nome	<i>Negrito</i>
QRLabel4	Telefone	<i>Negrito</i>
QRLabel5	Salário	<i>Negrito</i>

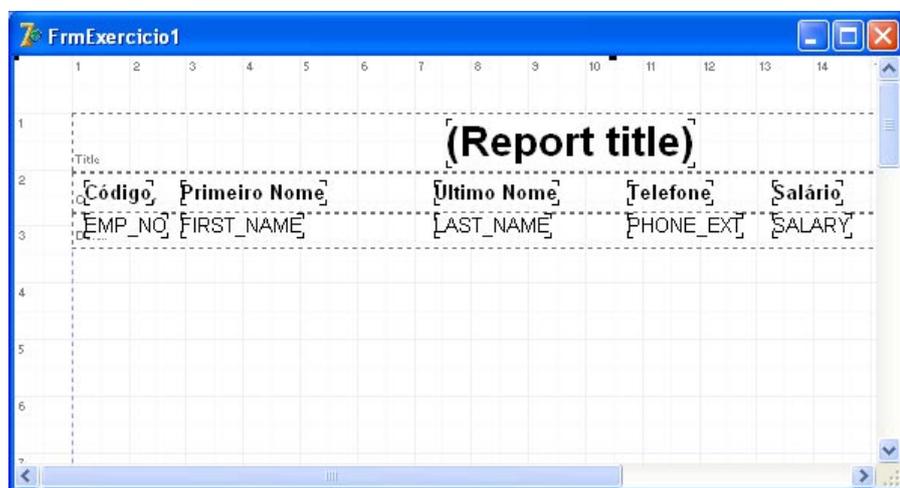
**Tabela 4.6.** Configurações dos componentes de título das colunas

Adicione cinco *QRDBText* dentro da banda detalhe. Configure-os de acordo com a **Tabela 4.7**.

Componente	DataSet	DataField
QRDBText1	<i>DMPrincipal.cdsEmpregados</i>	EMP_NO
QRDBText2	<i>DMPrincipal.cdsEmpregados</i>	FIRST_NAME
QRDBText3	<i>DMPrincipal.cdsEmpregados</i>	LAST_NAME
QRDBText4	<i>DMPrincipal.cdsEmpregados</i>	PHONE_EXT
QRDBText5	<i>DMPrincipal.cdsEmpregados</i>	SALARY

**Tabela 4.7.** Configuração dos componentes de acesso aos dados

Veja como ficou o layout do relatório na **Figura 4.25**.



**Figura 4.25.** Layout do relatório de lista de empregados

Antes de fazer a chamada ao relatório devemos “abrir” o componente que possui os dados a serem impressos, neste exemplo o *ClientDataSet*. No evento *OnCreate* do *FrmExercicio1* digite o seguinte código:

```
DMPrincipal.cdsEmpregados.Open;
```



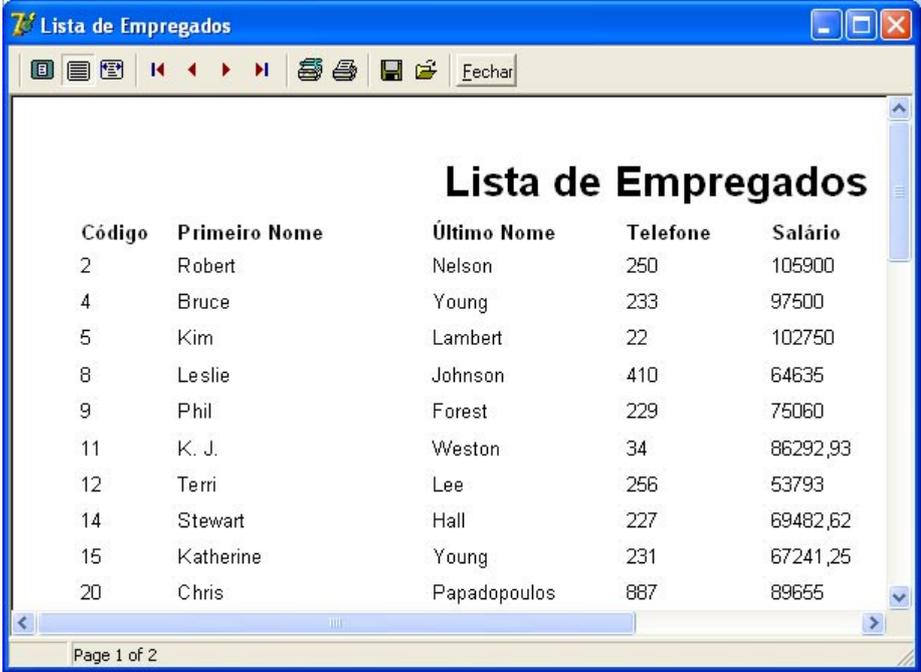


Poderíamos ativar normalmente o *ClientDataSet* em tempo de *design*, alterando a propriedade *Active* para *True*.

Adicione um botão no *FrmPrincipal*, alterando a propriedade *Caption* para "Exercício 1 - Lista de Empregados". Para o evento *OnClick* do botão digite o seguinte código:

```
FrmExercicio1.QuickRep1.Preview;
```

Adicione a unit *untFrmExercicio1* a cláusula *uses* do *FrmPrincipal*. Rode a aplicação, clique sobre o botão adicionado anteriormente, para visualizar o relatório em execução (Figura 4.26).



Código	Primeiro Nome	Último Nome	Telefone	Salário
2	Robert	Nelson	250	105900
4	Bruce	Young	233	97500
5	Kim	Lambert	22	102750
8	Leslie	Johnson	410	64635
9	Phil	Forest	229	75060
11	K. J.	Weston	34	86292,93
12	Terri	Lee	256	53793
14	Stewart	Hall	227	69482,62
15	Katherine	Young	231	67241,25
20	Chris	Papadopoulos	887	89655

Figura 4.26. Relatório em tempo de execução

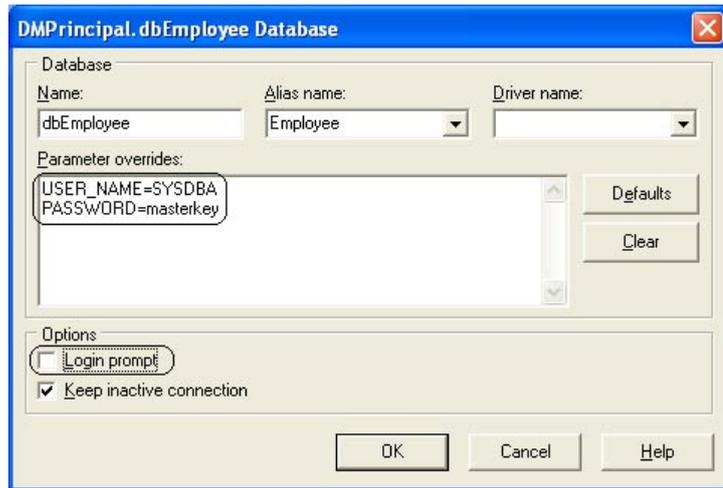
## Exercício 2

Crie uma nova aplicação no Delphi, dê o nome de "FrmPrincipal" ao formulário e salve-o como "untFrmPrincipal.pas", para o projeto salve como "Exercicio2.dpr". Crie um *DataModule*, dê o nome de "DMPrincipal" e salve-o como "untDMPrincipal". Adicione ao *DMPrincipal* um *DataBase* (BDE) e um *Query* (BDE).

➔ Altere a propriedade *Name* do *DataBase* para "dbEmployee";

Com o *dbEmployee* selecionado, aponte a propriedade *AliasName* para *Employee*, e altere a propriedade *DataBaseName* para "dbEmployee". Na janela de configuração do *dbEmployee* (clique com o botão direito e escolha a opção *Database Editor*, no menu de contexto), configure os parâmetros em *Parameter overrides* e *Options*, de acordo com a Figura 4.27.





**Figura 4.27.** Definindo o usuário e senha para acesso ao banco de dados

- ➔ Altere a propriedade *Name* do *Query* para "qryVendas";
- ➔ Ligue o *qryVendas* ao *dbEmployee*, através da propriedade *DatabaseName*. Para a propriedade *SQL* do *qryVendas* digite a seguinte instrução SQL:

```
select PO_NUMBER, ORDER_DATE,
       FIRST_NAME || ' ' || LAST_NAME AS VENDEDOR,
       CUSTOMER, QTY_ORDERED, TOTAL_VALUE
from SALES, CUSTOMER, EMPLOYEE
where SALES.SALES_REP = EMPLOYEE.EMP_NO and
       SALES.CUST_NO = CUSTOMER.CUST_NO
order by FIRST_NAME, LAST_NAME, CUSTOMER
```

Crie um novo formulário, altere o nome para "FrmExercicio2" e salve-o como "untFrmExercicio2.pas". Adicione ao formulário um *QuickRep*. Altere a propriedade *ReportTitle* para "Listagem de Vendas Efetuadas", e adicione quatro bandas (*QRBand*) ao relatório.

- ➔ Altere a propriedade *BandType* da primeira banda para *rbTitle*, da segunda para *rbColumnHeader*, da terceira para *rbDetail* e da quarta para *rbSummary*;
- ➔ Adicione um *QRSysData* dentro da banda título e altere a propriedade *Data* para *qrsReportTitle*;

Altere o tamanho da fonte do *QRSysData* para "20" e defina-a como negrito através da propriedade *Font*.

Para acessar o *qryVendas* adicione a unit *untDMPrincipal* a cláusula *uses* do *FrmExercicio2*. Selecione o *QuickRep1* e aponte a propriedade *DataSet* para *DMPrincipal.qryVendas*. Na banda de título das colunas (*rbColumnHeader*) adicione seis *QRLabel*. Configure os componentes, que irão mostrar o título das colunas do relatório, conforme a **Tabela 4.8**.

Componente	Caption	Font
QRLabel1	Nº Venda	<i>Negrito</i>
QRLabel2	Data Venda	<i>Negrito</i>
QRLabel3	Vendedor	<i>Negrito</i>
QRLabel4	Cliente	<i>Negrito</i>
QRLabel5	Quant.	<i>Negrito</i>
QRLabel6	Valor	<i>Negrito</i>

**Tabela 4.8.** Configurações dos componentes de título das colunas



Adicione seis *QRDBText* dentro da banda detalhe. Configure-os de acordo com a **Tabela 4.9**.

Componente	DataSet	DataField	AutoSize
QRDBText1	<i>DMPrincipal.qryVendas</i>	PO_NUMBER	<i>False</i>
QRDBText2	<i>DMPrincipal.qryVendas</i>	ORDER_DATE	<i>False</i>
QRDBText3	<i>DMPrincipal.qryVendas</i>	VENDEDOR	<i>False</i>
QRDBText4	<i>DMPrincipal.qryVendas</i>	CUSTOMER	<i>False</i>
QRDBText5	<i>DMPrincipal.qryVendas</i>	QTY_ORDERED	<i>False</i>
QRDBText6	<i>DMPrincipal.qryVendas</i>	TOTAL_VALUE	<i>False</i>

**Tabela 4.9.** Configurações dos componentes que irão mostrar os dados

Adicione dois *QRExpr* dentro da banda sumário e configure-os de acordo com a **Tabela 4.10**.

Componente	Expression	Alignment	Font	AutoSize
QRExpr1	<i>SUM(QTY_ORDERED)</i>	<i>taRightJustify</i>	<i>Negrito</i>	<i>False</i>
QRExpr2	<i>SUM(TOTAL_VALUE)</i>	<i>taRightJustify</i>	<i>Negrito</i>	<i>False</i>

**Tabela 4.10.** Configurando os componentes QRExpr para somatório de colunas

Veja como ficou o layout do relatório na **Figura 4.28**.



**Figura 4.28.** Layout do relatório de lista de empregados

Antes de fazer a chamada ao relatório devemos “abrir” o componente que possui os dados a serem impressos, neste exemplo o *Query*. No evento *OnCreate* do *FrmExercicio2* digite o seguinte código:

```
DMPrincipal.qryVendas.Open;
```





Poderíamos ativar normalmente o *Query* em tempo de *design*, alterando a propriedade *Active* para *True*.

Adicione um botão no *FrmPrincipal*, alterando a propriedade *Caption* para “Exercício 2 - Listagem de vendas efetuadas”. Para o evento *OnClick* do botão digite o seguinte código:

```
FrmExercicio2.QuickRep1.Preview;
```

Adicione a unit *untFrmExercicio2* a cláusula *uses* do *FrmPrincipal*. Rode a aplicação, clique sobre o botão adicionado anteriormente, para visualizar o relatório em execução (Figura 4.29).



N° Venda	Data Venda	Vendedor	Cliente	Quant.	Valor
V93C0120	22/03/1993	Claudia Sutherland	DataServe International	1	47
V93C0990	09/08/1993	Claudia Sutherland	DataServe International	40	399960
V9324200	09/08/1993	Claudia Sutherland	Signature Design	1000	560000
V93F2051	18/12/1993	Jacques Glon	3D-Pad Corp.	1	999
V93F3088	27/08/1993	Jacques Glon	3D-Pad Corp.	10	10000
V93F2030	12/12/1993	Jacques Glon	3D-Pad Corp.	15	450000
V93B1002	20/09/1993	Jacques Glon	Dyno Consulting	1	100
V93N5822	18/12/1993	Jacques Glon	GeoTech Inc.	2	1500
V92F3004	15/10/1992	K. J. Weston	3D-Pad Corp.	3	2000
V9346200	31/12/1993	K. J. Weston	Buttle, Griffith and Co.	3	0
V9345200	11/11/1993	K. J. Weston	Buttle, Griffith and Co.	900	27000
V92E0340	15/10/1992	K. J. Weston	Central Bank	7	70000
V91E0210	04/03/1991	K. J. Weston	Central Bank	10	5000
V9333005	03/02/1993	K. J. Weston	Dallas Technologies	2	600

Figura 4.29. Relatório em tempo de execução



# Capítulo 5

## Trabalhando com Grupos

Módulo I





Neste capítulo veremos como criar relatórios para imprimir dados agrupados a partir de um ou mais critérios. Por exemplo, você pode listar todos os clientes cadastrados no sistema, agrupando os registros de acordo com a cidade onde moram.



Em todos os exemplos e exercícios deste capítulo utilizaremos banco de dados *Employee.gdb* que acompanha o InterBase.

### Exemplo 1 - Agrupamento simples

Neste exemplo criaremos um relatório para imprimir os dados de uma tabela do banco de dados. Porém, esses dados serão agrupados e ordenados a partir de um campo da tabela. Inicie uma nova aplicação Delphi, altere o nome do formulário para "FrmPrincipal" e salve-o como "untFrmPrincipal.pas". Para o projeto dê o nome de "capitulo5.dpr". Crie um DataModule (*File|New>Data Module*), altere o nome para "DMPrincipal" e salve-o como "untDMPrincipal.pas".

Antes de construir o relatório, precisamos fazer a conexão com o banco de dados. Para isso, adicione um *SQLConnection* ao *DMPrincipal*. Altere a propriedade *Name* para "conCapitulo5". Utilizaremos o *SQLConnection* para fazer a conexão ao banco de dados, por isso, selecione na propriedade *ConnectionString* a conexão *Employee*.



A conexão *Employee* foi criada durante o primeiro exemplo do capítulo 4.

Para finalizar a configuração do *SQLConnection*, altere a propriedade *LoginPrompt* para *False*. Neste exemplo faremos a impressão dos dados contidos na tabela *DEPARTMENT*, agrupando-as a partir do campo *LOCATION*.

Adicione um *SQLDataSet* ao *DMPrincipal*, altere a propriedade *Name* para "datasetDepartamentos" e aponte a propriedade *SQLConnection* para *conCapitulo5*. Para a propriedade *CommandText* digite a seguinte instrução SQL:

```
select * from DEPARTMENT
order by LOCATION, DEPARTMENT
```



Repare que adicionamos a cláusula `ORDER BY` a instrução SQL do `SQLDataSet`. Utilizando esse comando informamos ao `SQLDataSet` para trazer os registros ordenados pelos campos `LOCATION` e `DEPARTMENT`.



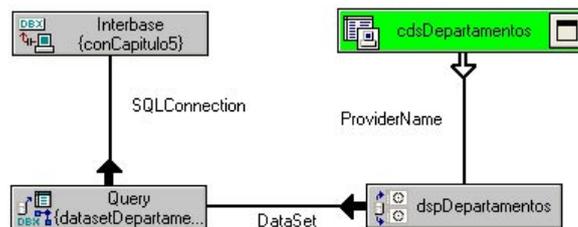
A cláusula `ORDER BY` é essencial para que possamos fazer o agrupamento dos departamentos pela sua localização.

Ainda no `DMPrincipal` adicione um `DataSetProvider` e um `ClientDataSet` (*Data Access*).

➔ Altere a propriedade `Name` do `DataSetProvider` para "`dspDepartamentos`";

➔ Altere a propriedade `Name` do `ClientDataSet` para "`cdsDepartamentos`".

Configure os componentes de acordo com o diagrama da **Figura 5.1**.



**Figura 5.1.** Configuração dos componentes para acesso aos dados

Crie um novo formulário, altere o nome para "`FrmRelatorio1`" e salve-o como "`untFrmRelatorio1.pas`". Adicione ao `FrmRelatorio1` um `QuickRep` e altere a propriedade `ReportTitle` para "`Lista de Departamentos agrupados por localização`". Adicione três bandas (`QRBand`) ao relatório.

➔ Altere a propriedade `BandType` da primeira banda para `rbTitle`, da segunda para `rbColumnHeader` e da terceira para `rbDetail`;

➔ Adicione um `QRSysData` dentro da banda título e altere a propriedade `Data` para `qrsReportTitle`;

➔ Altere o tamanho da fonte do `QRSysData` para "`16`" e defina-a como negrito através da propriedade `Font`.

Para que possamos fazer o agrupamento dos departamentos em relação a sua localização precisaremos utilizar mais uma banda no relatório. O `QuickReport` possui um tipo de banda, chamada `QRGroup`, criada exclusivamente para trabalhar com agrupamentos dentro do relatório.

Adicione um `QRGroup` ao relatório. Veja como ficou disponibilizado as bandas do relatório na **Figura 5.2**.





**Figura 5.2.** Bandas do relatório de agrupamento simples

Para que o relatório possa acessar o `cdsDepartamentos` do `DMPPrincipal`, adicione a unit `untDMPPrincipal` a cláusula `uses` do `FrmRelatorio1`. Selecione o `QuickRep1` e aponte a propriedade `DataSet` para `DMPPrincipal.cdsDepartamentos`.

Na banda de título das colunas (`rbColumnHeader`) adicione três `QRLabel`. Vamos configurar esses componentes para mostrar o título dos campos do relatório, de acordo com a Tabela 5.1.

Componente	Caption	Font
QRLabel1	Localização	<i>Negrito</i>
QRLabel2	Departamento	<i>Negrito</i>
QRLabel3	Telefone	<i>Negrito</i>

**Tabela 5.1.** Configurando os componentes para exibir o rótulo dos campos

Veja como ficou o relatório na Figura 5.3.



**Figura 5.3.** Definição do título dos campos que serão impressos no relatório



Antes de adicionar os *QRDBText* a banda detalhe do relatório, vamos adicionar na banda *Group Header (QRGroup)* um *QRDBText* para mostrar o campo *LOCATION* do *cdsDepartamentos*, para configurar o agrupamento no relatório.

- ➔ Altere a propriedade *DataSet* para *DMPrincipal.cdsDepartamentos* e *DataField* para *LOCATION* no *QRDBText*;
- ➔ Selecione a propriedade *Font* e altere a cor da fonte para azul e defina-a como negrito.

Selecione a banda *Group Header* e altere a propriedade *Expression* para *LOCATION* (Figura 5.4).

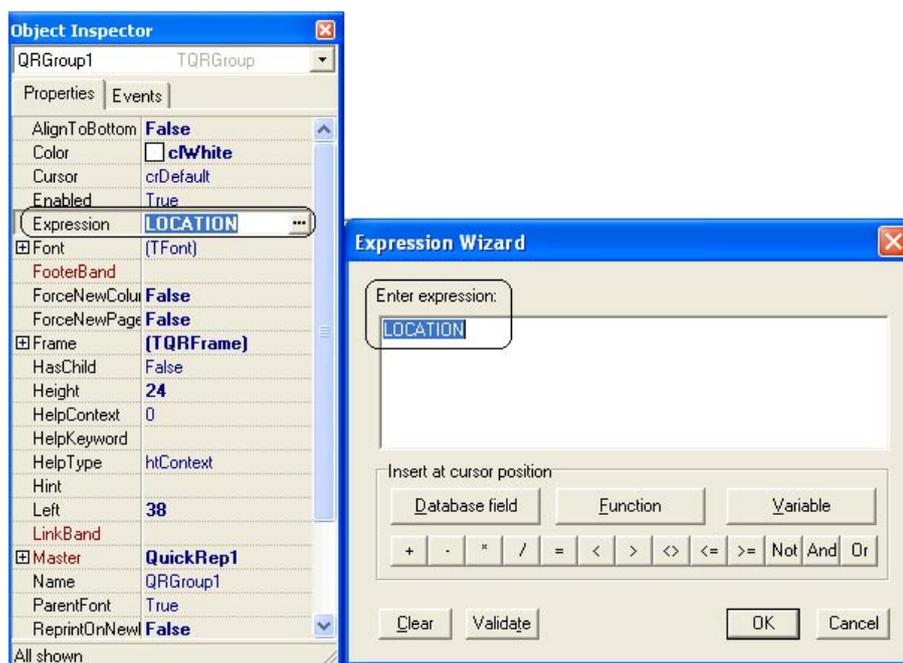


Figura 5.4. Configurando a propriedade *Expression* da banda *QRGroup*

A propriedade *Expression* é responsável em fazer a “quebra” dos grupos de acordo com o valor nela definido.



Expressões podem ser variáveis de banco de dados, variáveis de sistemas, funções ou constantes.

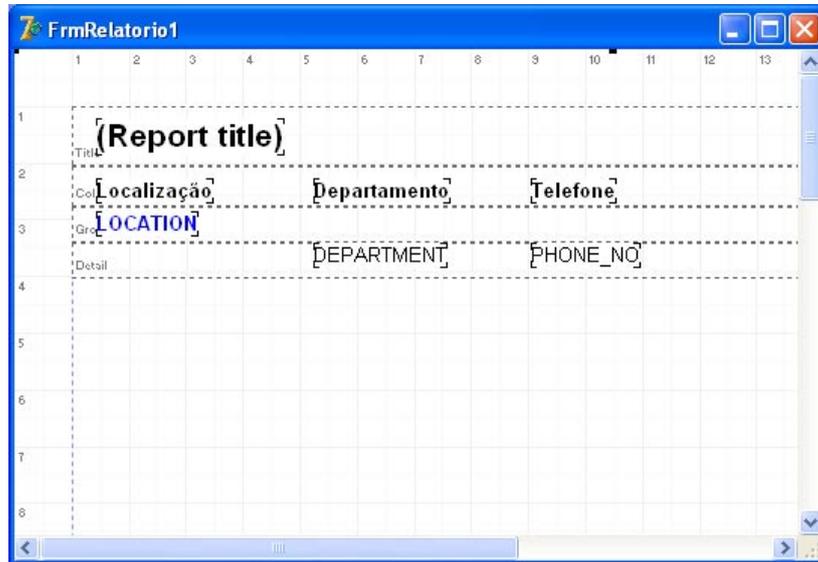
Para a banda detalhe, adicione dois *QRDBText* e configure-os de acordo com a Tabela 5.2.

Componente	DataSet	DataField
QRDBText1	<i>DMPrincipal.cdsDepartamentos</i>	DEPARTMENT
QRDBText2	<i>DMPrincipal.cdsDepartamentos</i>	PHONE_NO

Tabela 5.2. Configurando os *QRDBTexts*



Ajuste a posição dos componentes *QRDBText* de acordo com a **Figura 5.5**.



**Figura 5.5.** Layout do relatório de Listagem de Departamentos

Antes de fazer a chamada ao relatório devemos abrir o componente que possui os dados a serem impressos, nesse caso o *ClientDataSet*. No evento *OnCreate* do *FrmRelatorio1* digite o seguinte código:

```
DMPrincipal.cdsDepartamentos.Open;
```



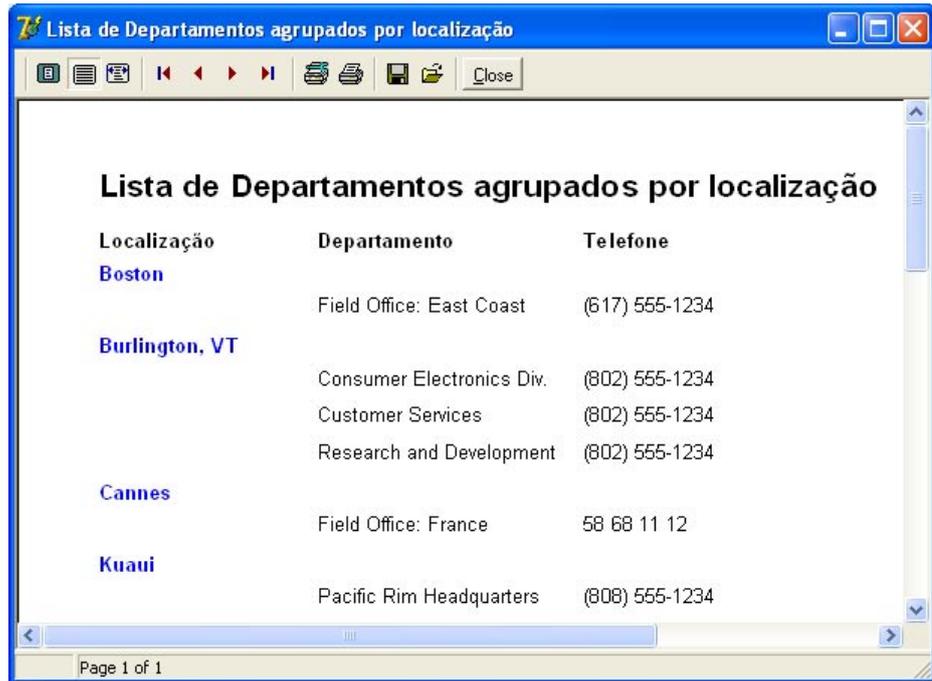
Poderíamos ativar normalmente o *ClientDataSet* em tempo de *design*, alterando a propriedade *Active* para *True*.

Selecione o *FrmPrincipal* e adicione um botão alterando a propriedade *Caption* para "Relatório 1 - Agrupamento Simples". Para o evento *OnClick* do botão digite o seguinte código:

```
FrmRelatorio1.QuickRep1.Preview;
```

Adicione a unit *untFrmRelatorio1* a cláusula *uses* do *FrmPrincipal*. Rode a aplicação, clique sobre o botão adicionado anteriormente e visualize o relatório em execução na **Figura 5.6**.





Localização	Departamento	Telefone
<b>Boston</b>		
	Field Office: East Coast	(617) 555-1234
<b>Burlington, VT</b>		
	Consumer Electronics Div.	(802) 555-1234
	Customer Services	(802) 555-1234
	Research and Development	(802) 555-1234
<b>Cannes</b>		
	Field Office: France	58 68 11 12
<b>Kuaui</b>		
	Pacific Rim Headquarters	(808) 555-1234

Figura 5.6. Relatório com agrupamento simples em tempo de execução

## Exemplo 2 - Vendas por cliente

Neste exemplo criaremos um relatório para imprimir os dados de uma tabela de vendas agrupadas pelo cliente. Utilizaremos os dados das tabelas SALES e CUSTOMER, agrupando-as pelo campo CUST\_NO. Adicione um *SQLDataSet* ao *DMPrincipal*, altere a propriedade *Name* para "datasetVendasCli" e aponte a propriedade *SQLConnection* para *conCapitulo5*. Na propriedade *CommandText* digite a seguinte instrução SQL:

```
select CUSTOMER.CUSTOMER, SALES.PO_NUMBER,  
       ORDER_DATE, QTY_ORDERED, TOTAL_VALUE  
from CUSTOMER, SALES  
where SALES.CUST_NO = CUSTOMER.CUST_NO  
order by CUSTOMER.CUSTOMER, PO_NUMBER
```

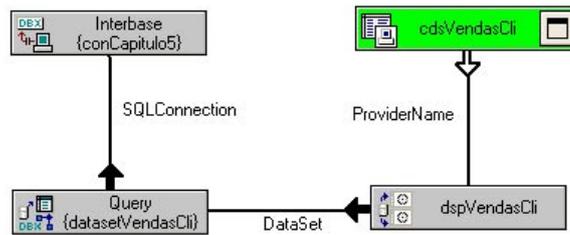
A instrução anterior buscará todas as vendas da tabela SALES ordenadas pelo nome do cliente. Como a tabela SALES não possui o campo com o nome do cliente (CUSTOMER) e sim, apenas o campo com o código (CUST\_NO), referenciamos a tabela CUSTOMER no SQL e criamos uma ligação entre as duas tabelas na cláusula WHERE para trazer o nome do cliente.

Ainda no *DMPrincipal* adicione um *DataSetProvider* e um *ClientDataSet* (Data Access).

- ➔ Altere a propriedade *Name* do *DataSetProvider* para "dspVendasCli";
- ➔ Altere a propriedade *Name* do *ClientDataSet* para "cvsVendasCli".

Configure os componentes de acordo com o diagrama da Figura 5.7.





**Figura 5.7.** Configuração dos componentes para acesso aos dados

Crie um novo formulário, altere o nome para "FrmRelatorio2" e salve-o como "untFrmRelatorio2.pas". Adicione um *QuickRep* e altere a propriedade *ReportTitle* para "Vendas por Cliente". Adicione três bandas (*QRBand*) ao relatório.

- Altere a propriedade *BandType* da primeira banda para *rbTitle*, da segunda para *rbColumnHeader*, e da terceira para *rbDetail*;
- Adicione um *QRSysData* dentro da banda título e altere a propriedade *Data* para *qrsReportTitle*;
- Altere o tamanho da fonte do *QRSysData* para "18" e defina-a como negrito através da propriedade *Font*.

Para que possamos fazer o agrupamento das vendas em relação ao nome do cliente precisaremos utilizar mais uma banda no relatório, para isso adicione um *QRGroup* ao *QuickRep*. Neste exemplo, utilizaremos ainda mais uma banda para imprimir o total de vendas, e o valor total das vendas de cada cliente. Adicione uma banda (*QRBand*) ao relatório, e altere a propriedade *BandType* para *rbGroupFooter*.

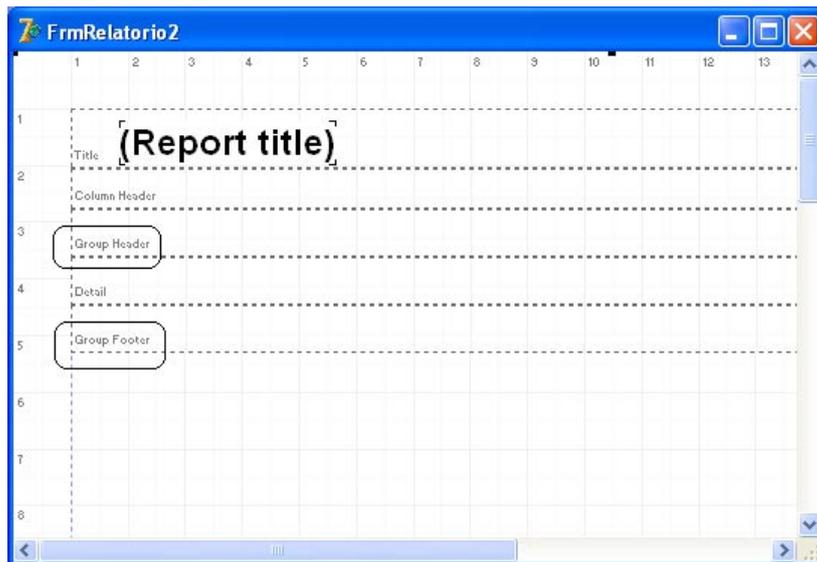
Selecione a banda *Group Header* e aponte a propriedade *FooterBand* para a banda *Group Footer (QRBand4)*. A propriedade *FooterBand* é usada para definir qual será a banda rodapé do grupo.



Bandas *Group Footer* são usadas em conjunto com *QRGroup* para imprimir dados no final de cada agrupamento, como por exemplo, sub-totais.

Veja como ficou o layout do relatório na **Figura 5.8**.





**Figura 5.8.** Bandas Group Header e Group Footer do relatório

Para o relatório acessar o `cdsVendasCli` do `DMPrincipal`, adicione a unit `untDMPrincipal` a cláusula `uses` do `FrmRelatorio2`. Selecione o `QuickRep1` e aponte a propriedade `DataSet` para `DMPrincipal.cdsVendasCli`.

Na banda de título das colunas (`rbColumnHeader`) adicione cinco `QRLabel`. Vamos configurar esses componentes para imprimir o título dos campos do relatório, conforme a Tabela 5.3.

Componente	Caption	Font
QRLabel1	Cliente	<i>Negrito</i>
QRLabel2	Nº da Venda	<i>Negrito</i>
QRLabel3	Data	<i>Negrito</i>
QRLabel4	Quantidade	<i>Negrito</i>
QRLabel5	Valor	<i>Negrito</i>

**Tabela 5.3.** Configuração dos rótulos dos registros a serem impressos

Adicione um `QRDBText` a banda `Group Header`.

→ Altere a propriedade `DataSet` para `DMPrincipal.cdsVendasCli` e `DataField` para `CUSTOMER`;

→ Selecione a propriedade `Font` e altere a cor para verde e defina-a como negrito.

Selecione a banda `Group Header` e altere a propriedade `Expression` para `"CUSTOMER"`. Para a banda detalhe, adicione quatro `QRDBText` e configure-os de acordo com a Tabela 5.4.



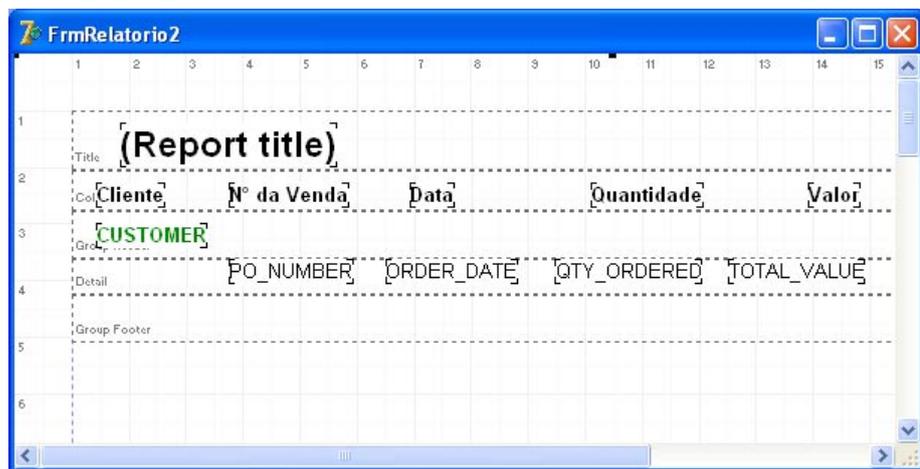
Componente	DataSet	DataField	Alignment
QRDBText2	DMPrincipal.cdsVendasCli	PO_NUMBER	taLeftJustify
QRDBText3	DMPrincipal.cdsVendasCli	ORDER_DATE	taLeftJustify
QRDBText4	DMPrincipal.cdsVendasCli	QTY_ORDERED	taRightJustify
QRDBText5	DMPrincipal.cdsVendasCli	TOTAL_VALUE	taRightJustify

**Tabela 5.4.** Configuração dos componentes que mostrarão os dados



A propriedade *Alignment* é utilizada para alinhar o texto impresso no *QRDBText*. Normalmente campos numéricos são alinhados da direita para a esquerda.

Ajuste a posição dos *QRDBText* de acordo com a **Figura 5.9**.



**Figura 5.9.** Layout do relatório Vendas por Cliente

Para finalizar a montagem do relatório, vamos imprimir na banda *Group Footer* a quantidade total e o valor total das vendas de cada cliente. Adicione dois *QRExpr* na banda *Group Footer* e configure-os de acordo com a **Tabela 5.5**.

Componente	Expression	ResetAfterPrint	Alignment
QRExpr1	<i>SUM(QTY_ORDERED)</i>	<i>True</i>	<i>taRightJustify</i>
QRExpr2	<i>SUM(TOTAL_VALUE)</i>	<i>True</i>	<i>taRightJustify</i>

**Tabela 5.5.** Configurações dos somatórios dos relatórios



A propriedade *ResetAfterPrint* é usada para zerar o valor do *QRExpr* após a sua impressão, para que traga corretamente a soma dos valores de cada grupo. Caso essa propriedade não seja ativada, o valor impresso no final de cada grupo será acumulativo.



Selecione os dois *QRExpr* e defina a fonte para negrito, através da propriedade *Font*. No evento *OnCreate* do *FrmRelatorio2* digite o seguinte código:

```
DMPPrincipal.cdsVendasCli.Open;
```

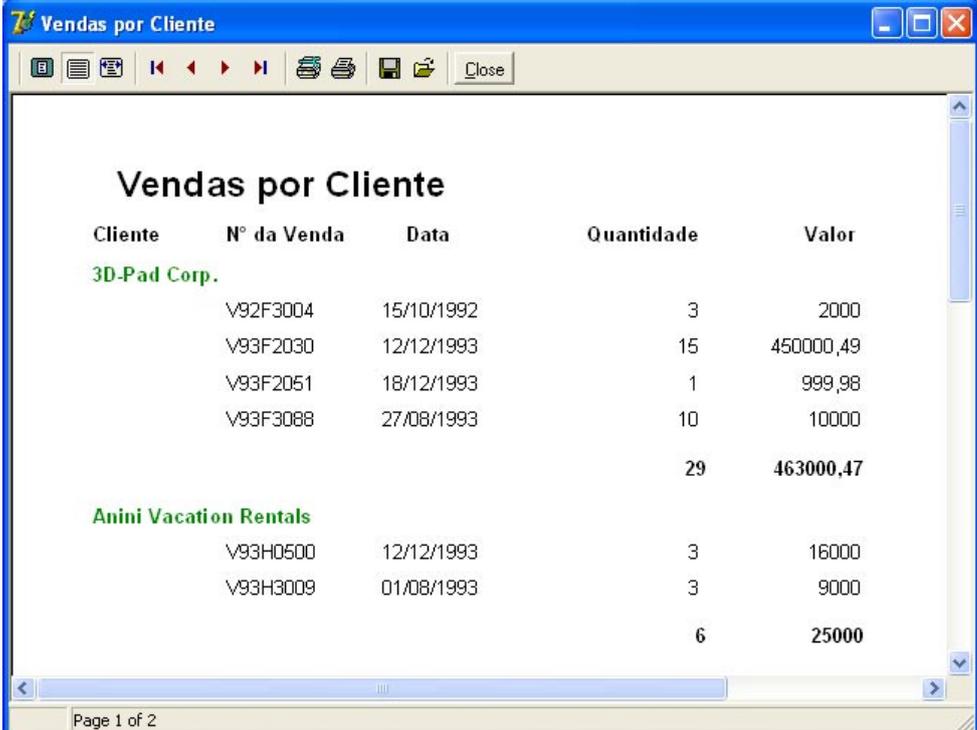


A propriedade *Alignment* é utilizada para alinhar o texto impresso no *QRDBText*. Normalmente campos numéricos são alinhados da direita para a esquerda.

Selecione o *FrmPrincipal* e adicione um botão alterando a propriedade *Caption* para "Relatório 2 - Vendas por Cliente". Para o evento *OnClick* do botão digite o seguinte código:

```
FrmRelatorio2.QuickRep1.Preview;
```

Adicione a unit *untFrmRelatorio2* a cláusula *uses* do *FrmPrincipal*. Rode a aplicação, clique sobre o botão adicionado anteriormente e visualize o relatório em execução na Figura 5.10.



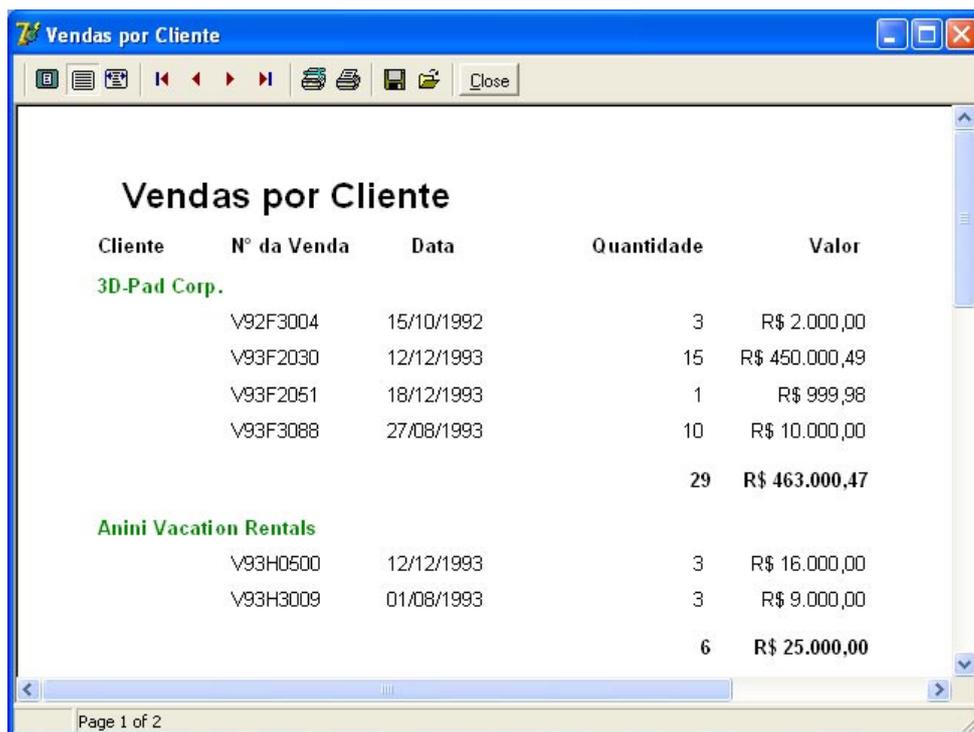
Cliente	Nº da Venda	Data	Quantidade	Valor
<b>3D-Pad Corp.</b>				
	V92F3004	15/10/1992	3	2000
	V93F2030	12/12/1993	15	450000,49
	V93F2051	18/12/1993	1	999,98
	V93F3088	27/08/1993	10	10000
			<b>29</b>	<b>463000,47</b>
<b>Anini Vacation Rentals</b>				
	V93H0500	12/12/1993	3	16000
	V93H3009	01/08/1993	3	9000
			<b>6</b>	<b>25000</b>

Figura 5.10. Relatório de Vendas por Cliente em tempo de execução





Caso você queira formatar o campo *Valor* para adicionar o símbolo de moeda e o separador de milhar, basta alterar a propriedade *Mask* do *QRDBText* (banda detalhe) e *QRExpr* (banda *Group Footer*) para "R\$ ###,##0.00". Lembre-se que para formatação de máscaras você deve trocar os pontos por vírgulas e a vírgula por ponto. Veja na **Figura 5.11** o relatório com o campo *Valor* formatado.



Cliente	Nº da Venda	Data	Quantidade	Valor
<b>3D-Pad Corp.</b>				
	V92F3004	15/10/1992	3	R\$ 2.000,00
	V93F2030	12/12/1993	15	R\$ 450.000,49
	V93F2051	18/12/1993	1	R\$ 999,98
	V93F3088	27/08/1993	10	R\$ 10.000,00
			<b>29</b>	<b>R\$ 463.000,47</b>
<b>Anini Vacation Rentals</b>				
	V93H0500	12/12/1993	3	R\$ 16.000,00
	V93H3009	01/08/1993	3	R\$ 9.000,00
			<b>6</b>	<b>R\$ 25.000,00</b>

**Figura 5.11.** Relatório de Vendas por Cliente com formatação no campo Valor

## Exercícios

Exercício 1 – Crie um novo relatório utilizando o QuickReport com o título "Status das Vendas". Esse relatório precisa imprimir as vendas agrupadas por status. O número total de vendas e o valor total das vendas de cada status deve ser impresso no final de cada agrupamento. Adicione cinco bandas ao relatório: título, cabeçalho das colunas, cabeçalho do grupo, detalhe e rodapé do grupo.

- Vincule o relatório a tabela SALES do banco de dados *Employee.gdb* (utilizando dbExpress para fazer o acesso aos dados);
- Para a banda título imprima o título do relatório;
- Para a banda de cabeçalho das colunas adicione os seguintes títulos: Status, Nº Venda, Data e Valor;
- Na banda *Group Header* imprima o campo ORDER\_STATUS;
- Na banda detalhe imprima os dados dos seguintes campos da tabela SALES: PO\_NUMBER, ORDER\_DATE e TOTAL\_VALUE;
- Na banda *Group Footer* imprima o número total de vendas e o valor total das vendas de cada *Status* utilizando o *QRExpr*.



Exercício 2 – Após a construção do relatório do exercício 1, altere o *FrmPrincipal*, adicionando no exercício, uma opção para que o usuário possa informar um *Status* específico para filtrar o relatório de “Status por Vendas”.

## OBSERVAÇÃO

Utilize um *ComboBox (Standard)* para listar os Status disponíveis.

## Solução Exercício 1

Crie uma nova aplicação no Delphi, dê o nome de “FrmPrincipal” ao formulário, salve-o como “untFrmPrincipal.pas” e o projeto salve como “Exercicio1.dpr”. Adicione o *DMPrincipal* do projeto *capitulo5.dpr*, através do menu *Project\Add to Project*.

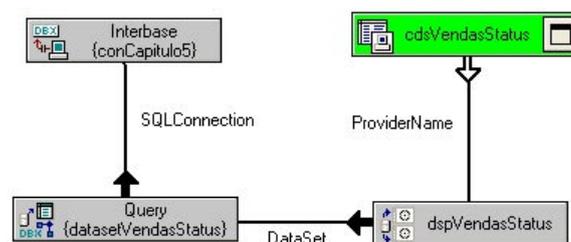
Adicione ao *DMPrincipal* um *SQLDataSet (dbExpress)*, um *DataSetProvider* e um *ClientDataSet (Data Access)*.

- ➔ Altere a propriedade *Name* do *SQLDataSet* para “datasetVendasStatus”;
- ➔ Altere a propriedade *Name* do *DataSetProvider* para “dspVendasStatus”;
- ➔ Altere a propriedade *Name* do *ClientDataSet* para “cdsVendasStatus”;

Ligue o *datasetVendasStatus* com o *conCapitulo5*, através da propriedade *SQLConnection*. Na propriedade *CommandText* digite a seguinte instrução SQL:

```
select ORDER_STATUS, PO_NUMBER, ORDER_DATE,
       TOTAL_VALUE
from SALES
order by ORDER_STATUS, PO_NUMBER
```

Configure os componentes de acordo com o diagrama da **Figura 5.12**.



**Figura 5.12.** Configuração dos componentes

Crie um novo formulário, altere o nome para “Exercicio1” e salve-o como “untExercicio1.pas”. Adicione ao *Exercicio1* um *QuickRep*, e altere a propriedade *ReportTitle* para “Status das Vendas”. Adicione três bandas (componente *QRBand*) ao relatório.

- ➔ Altere a propriedade *BandType* da primeira banda para *rbTitle*, da segunda para *rbColumnHeader* e da terceira para *rbDetail*;
- ➔ Adicione um *QRSysData* dentro da banda título e altere a propriedade *Data* para *qrsReportTitle*;
- ➔ Altere o tamanho da fonte do *QRSysData* para “18” e defina-a como negrito através da propriedade *Font*.



Adicione ao *QuickRep* um *QRGroup*. Adicione também uma banda (*QRBand*) ao relatório, e altere a propriedade *BandType* para *rbGroupFooter*. Selecione a banda *Group Header* e aponte a propriedade *FooterBand* para a banda *Group Footer (QRBand4)*.

Para acessar o *cdsVendasStatus* adicione a unit *untDMPrincipal* a cláusula *uses* do *FrmExercicio1*. Selecione o *QuickRep1* e aponte a propriedade *DataSet* para *DMPrincipal.cdsVendasStatus*. Na banda de título das colunas (*rbColumnHeader*) adicione quatro *QRLabel*. Configure os componentes de acordo com a **Tabela 5.6**.

Componente	Caption	Font
QRLabel1	Status	<i>Negrito</i>
QRLabel2	Nº Venda	<i>Negrito</i>
QRLabel3	Data	<i>Negrito</i>
QRLabel4	Valor	<i>Negrito</i>

**Tabela 5.6.** Configuração dos rótulos dos registros a serem impressos

Adicione um *QRDBText* a banda *Group Header*.

→ Altere a propriedade *DataSet* para *DMPrincipal.cdsVendasStatus* e *DataField* para *ORDER\_STATUS*;

→ Selecione a propriedade *Font* e altere a cor para azul e defina-a como negrito.

Selecione a banda *Group Header* e altere a propriedade *Expression* para *ORDER\_STATUS*. Para a banda detalhe, adicione três *QRDBText* e configure-os de acordo com a **Tabela 5.7**.

Componente	DataSet	DataField	Alignment
QRDBText1	<i>DMPrincipal.cdsVendasStatus</i>	<i>PO_NUMBER</i>	<i>taLeftJustify</i>
QRDBText2	<i>DMPrincipal.cdsVendasStatus</i>	<i>ORDER_DATE</i>	<i>taLeftJustify</i>
QRDBText3	<i>DMPrincipal.cdsVendasStatus</i>	<i>TOTAL_VALUE</i>	<i>taRightJustify</i>

**Tabela 5.7.** Configuração dos registros a serem impressos

Ajuste a posição dos *QRDBText* de acordo com a **Figura 5.13**.



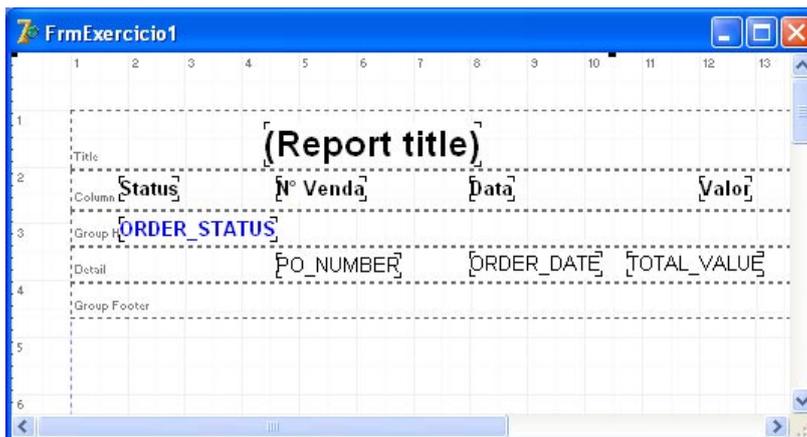


Figura 5.13. Layout do relatório de Status das Vendas

Adicione dois *QRExpr* na banda *Group Footer* e configure-os de acordo com a Tabela 5.8.

Componente	Expression	ResetAfterPrint	Alignment
QRExpr1	COUNT	True	taRightJustify
QRExpr2	SUM(TOTAL_VALUE)	True	taRightJustify

Tabela 5.7. Configuração do somatório dos grupos do relatório

Selecione os dois *QRExpr* e altere a fonte para negrito através da propriedade *Font*. Veja como ficou o relatório na Figura 5.14.

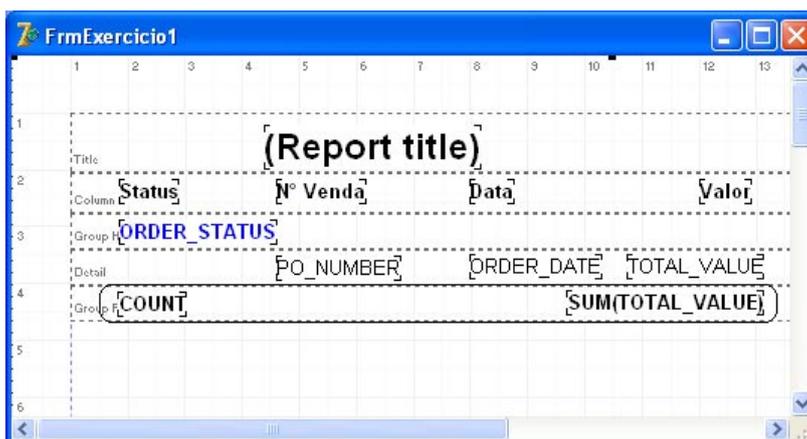


Figura 5.14. Relatório Status das Vendas

Altere a propriedade *Mask* do *QRDBText* que mostra o campo *TOTAL\_VALUE* e o *QRExpr* que mostra o total para "R\$ ###,##0.00".

No evento *OnCreate* do form *FrmExercicio1* digite o seguinte código:

```
DMPrincipal.cdsVendasStatus.Open;
```



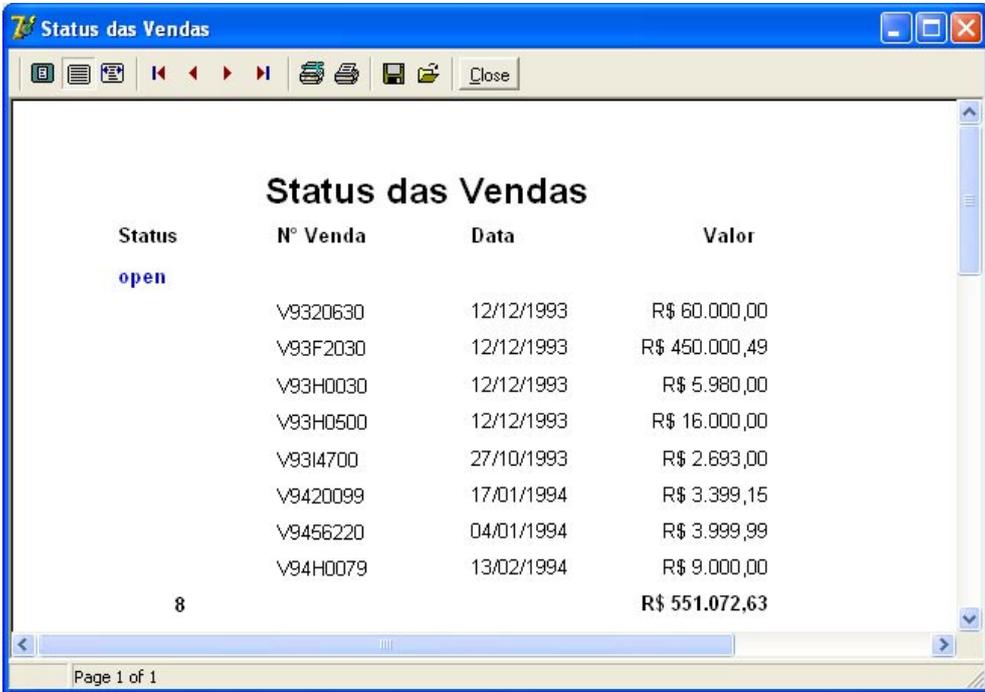


Poderíamos ativar normalmente o *ClientDataSet* em tempo de *design*, alterando a propriedade *Active* para *True*.

Selecione o *FrmPrincipal* e adicione um botão, alterando a propriedade *Caption* para "Exercício 1 - Status das Vendas". Para o evento *OnClick* do botão digite o seguinte código:

```
FrmExercicio1.QuickRep1.Preview;
```

Adicione a unit *untFrmExercicio1* a cláusula *uses* do *FrmPrincipal*. Rode a aplicação, clique sobre o botão e visualize o relatório em execução (**Figura 5.15**).



Status	Nº Venda	Data	Valor
open	V9320630	12/12/1993	R\$ 60.000,00
	V93F2030	12/12/1993	R\$ 450.000,49
	V93H0030	12/12/1993	R\$ 5.980,00
	V93H0500	12/12/1993	R\$ 16.000,00
	V93I4700	27/10/1993	R\$ 2.693,00
	V9420099	17/01/1994	R\$ 3.399,15
	V9456220	04/01/1994	R\$ 3.999,99
	V94H0079	13/02/1994	R\$ 9.000,00
	8		R\$ 551.072,63

**Figura 5.15.** Relatório Status das Vendas em tempo de execução

## Exercício 2

Selecione o formulário *FrmPrincipal* do exercício anterior (*exercicio1.dpr*) e adicione um *ComboBox (Standard)*. Altere a propriedade *Name* para "cbStatus", em *Items* adicione os valores: "shipped", "open" e "waiting". Altere a propriedade *Style* para *csDropDownList*, para bloquear o usuário de digitar valores diferentes dos adicionados na lista.

O formulário deve ficar semelhante à **Figura 5.16**.





Figura 5.16. Layout do formulário

Selecione o *dspVendasStatus* que esta no *DMPrincipal* e altere a propriedade *Options|poAllowCommandText* para *True*.



A propriedade *poAllowCommandText* informa que a instrução SQL será feita pelo *ClientDataSet*, através da propriedade *CommandText*.

Adicione a unit *untDMPrincipal* a cláusula *uses* do *FrmPrincipal* e altere o evento *OnClick* do botão para:

```
with DMPrincipal.cdsVendasStatus do
begin
  Close;
  CommandText := 'Select order_status, po_number, ' +
    'order_date, total_value ' +
    'From Sales Where UPPER(order_status) = ' +
    AnsiUpperCase(QuotedStr(cbStatus.Text)) +
    'order by order_status, po_number';
  Open;
end;
FrmExerciciol.QuickRepl.Preview;
```

No código anterior, alteramos a instrução SQL para que possamos fazer o filtro no campo *ORDER\_STATUS* de acordo com o valor informado pelo usuário. Rode a aplicação, selecione um Status na lista e clique sobre o botão para visualizar o relatório em execução.



Você pode utilizar um *DBLookupComboBox*, quando necessitar mostrar itens de um campo de uma tabela, que podem ser utilizados como filtro.



# Capítulo 6

## Construindo relatórios Mestre/Detalhe

Módulo I





Neste capítulo, veremos como criar relatórios mestre / detalhe com o QuickReport. Relatórios mestre / detalhe são usados para imprimir registros relacionados de duas ou mais tabelas. Por exemplo, você pode imprimir todos os pedidos com seus respectivos itens, onde as informações do pedido seriam as informações mestre, enquanto que os itens seriam o detalhe.



Em todos os exemplos e exercícios deste capítulo utilizaremos banco de dados InterBase *Employee.gdb*.

### Exemplo 1 - Mestre/Detalhe simples

Neste primeiro exemplo criaremos um relatório para imprimir os dados armazenados em duas tabelas do banco. Porém, esses dados serão relacionados (ligados) a partir de um campo comum em cada tabela.

O relatório fará a impressão de todos os departamentos e seus empregados. Inicie uma nova aplicação Delphi. Altere o nome para "FrmPrincipal" e salve a unit como "untFrmPrincipal.pas". Para o arquivo de projeto dê o nome de "capitulo6.dpr". Crie um DataModule, altere o nome para "DMPrincipal" e salve-o como "untDMPrincipal.pas".

Antes de construir o relatório, precisamos fazer a conexão com o banco de dados. Para isso, adicione um *SQLConnection* ao *DMPrincipal* e altere a propriedade *Name* para "conCapitulo6". Altere a propriedade *ConnectionString* para *Employee*.



A conexão *Employee* foi criada durante o primeiro exemplo do capítulo 4.

Para finalizar a configuração, altere a propriedade *LoginPrompt* para *False*. Neste exemplo, faremos à impressão dos dados contidos nas tabelas *DEPARTMENT* e *EMPLOYEE*.

Adicione dois *SQLDataSet* ao *DMPrincipal*, altere a propriedade *Name* do primeiro para "datasetDepartamentos" e do segundo para "datasetEmpregados". Aponte a propriedade *SQLConnection* dos dois *SQLDataSets* para *conCapitulo6*.



O *datasetDepartamentos* será o *DataSet* mestre, enquanto que o *datasetEmpregados* será o detalhe. Para a propriedade *CommandText* do *datasetDepartamentos* digite a seguinte instrução SQL:

```
select * from DEPARTMENT  
order by DEPARTMENT
```

Para o *datasetEmpregados* digite a seguinte instrução SQL na propriedade *CommandText*:

```
select * from EMPLOYEE  
where DEPT_NO = :DEPT_NO  
order by FIRST_NAME
```

O campo *DEPT\_NO* é o campo chave que fará o relacionamento entre a tabela *DEPARTMENT* e *EMPLOYEE*. Repare que no *datasetEmpregados* definimos um parâmetro na cláusula *WHERE* de nome *DEPT\_NO* para o campo *DEPT\_NO*.

Como o *datasetDepartamentos* (mestre) possui um campo de nome *DEPT\_NO*, definindo o nome do parâmetro no *DataSet* detalhe com o mesmo nome do campo chave do *DataSet* mestre, o Delphi automaticamente já faz o controle dos registros, sem a necessidade de utilizarmos por exemplo, o evento *AfterScroll* do *DataSet*.

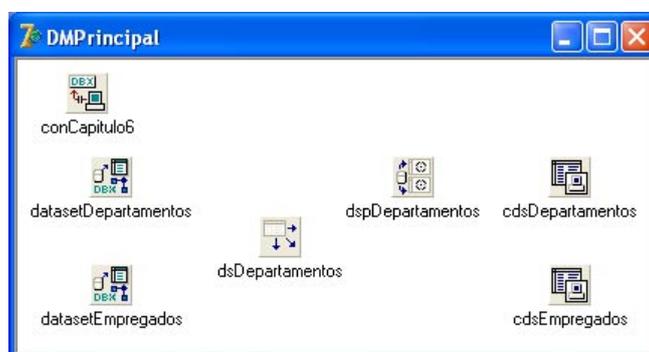
Para que o Delphi consiga controlar os registros do detalhe de acordo com o mestre, precisamos ligar os dois *SQLDataSet* através de um *DataSource*. Adicione um *DataSource* ao *DMPPrincipal*, altere o nome para "dsDepartamentos" e ligue a propriedade *DataSet* para *datasetDepartamentos*.

Selecione o *datasetEmpregados* e aponte a propriedade *DataSource* para *dsDepartamentos*. Dessa forma os dois *SQLDataSet* estarão ligados e será possível o *DataSet* mestre fazer o controle dos registros do *DataSet* detalhe.

Ainda no *DMPPrincipal* adicione um *DataSetProvider* e dois *ClientDataSet* (*Data Access*).

- ➔ Altere a propriedade *Name* do *DataSetProvider* para "dspDepartamentos";
- ➔ Altere a propriedade *Name* do primeiro *ClientDataSet* para "cdsDepartamentos";
- ➔ Altere a propriedade *Name* do segundo *ClientDataSet* para "cdsEmpregados";
- ➔ Ligue o *dspDepartamentos* ao *datasetDepartamentos*, através da propriedade *DataSet* e o *cdsDepartamentos* ao *dspDepartamentos*, através da propriedade *ProviderName*.

Veja como ficou o *DataModule* na **Figura 6.1**.



**Figura 6.1.** Componentes para acesso aos dados



Clique com o botão direito do mouse sobre o *cdsDepartamentos* e escolha a opção *Fields Editor*. Dentro do editor de campos (*Fields Editor*), clique com o botão direito novamente e escolha a opção *Add Fields* (Figura 6.2).



Figura 6.2. Adicionando os campos ao *cdsDepartamentos*

Repare que um campo de nome *datasetEmpregados* foi adicionado ao *cdsDepartamentos*. Esse é um campo do tipo *DataSet*, isso é, será nesse campo que serão guardadas as informações referente ao *DataSet* detalhe (*datasetEmpregados*).



Você pode estar se perguntando o porque de apenas um *DataSetProvider*, e não dois. Como os registros do *DataSet* detalhe serão controlados pelo *DataSet* mestre, não há necessidade de adicionarmos um *DataSetProvider* para o *DataSet* detalhe. Você terá uma idéia melhor sobre o funcionamento da arquitetura mestre / detalhe no decorrer deste capítulo.

Aponte a propriedade *DataSetField* do *cdsEmpregados* para *cdsDepartamentosdatasetEmpregados*.



A propriedade *DataSetField* do *ClientDataSet* é utilizada para ligar o *DataSet* detalhe ao *DataSet* mestre. Por essa razão, não há a necessidade de um *DataSetProvider* para o *DataSet* detalhe. Todo o controle dos registros mestres e detalhes são feitos pelo *DataSet* mestre.

Crie um novo formulário, altere o nome para "FrmRelatorio1" e salve-o como "untFrmRelatorio1.pas". Adicione ao *FrmRelatorio1* um *QuickRep* e altere a propriedade *ReportTitle* para "Mestre/Detalhe simples". Adicione três bandas (*QRBand*) ao relatório.



- ➔ Altere a propriedade *BandType* da primeira banda para *rbTitle*, da segunda para *rbColumnHeader* e da terceira para *rbDetail*;
- ➔ Adicione um *QRSysData* dentro da banda título e altere a propriedade *Data* para *qrsReportTitle*;
- ➔ Altere o tamanho da fonte do *QRSysData* para "16" e defina-a como negrito através da propriedade *Font*.

Utilizaremos a banda detalhe para fazer a impressão dos dados contidos no *DataSet* mestre (*cdsDepartamentos*). Para que possamos imprimir os dados do *DataSet* detalhe (*cdsEmpregados*) precisaremos utilizar um novo tipo de banda, a *QRSubDetail*.

O *QRSubDetail* é utilizado para impressão de bandas detalhes em um relacionamento mestre / detalhe, onde é possível criar vários níveis de sub-detalhes de acordo com a necessidade.

Adicione um *QRSubDetail* ao relatório (Figura 6.3).



Figura 6.3. QRSubDetail para impressão da banda detalhe em um relatório mestre/detalhe

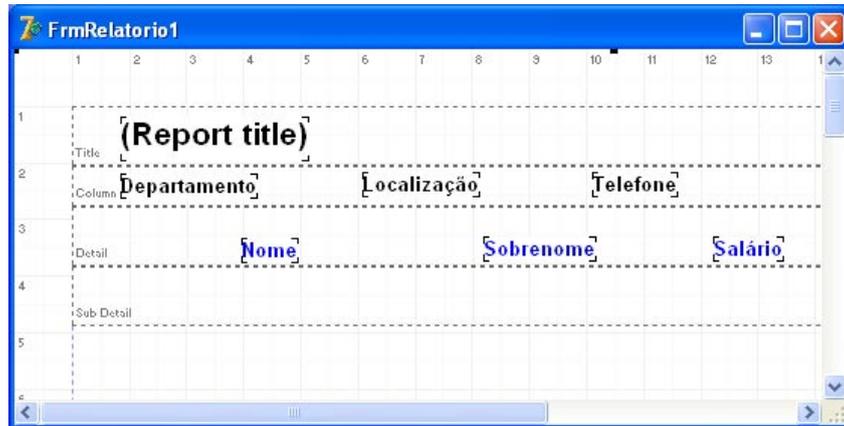
Para que o relatório possa acessar os *ClientDataSets*, adicione a unit *untDMPrincipal* a cláusula *uses* do *FrmRelatorio1*. Selecione o *QuickRep1* e aponte a propriedade *DataSet* para *DMPrincipal.cdsDepartamentos*. Na banda de título das colunas (*rbColumnHeader*) adicione seis *QRLabel*. Vamos configurar esses componentes para imprimir o título dos campos do relatório, de acordo com a Tabela 6.1.

Banda	Componente	Caption	Font	Cor da Fonte
<i>rbColumnHeader</i>	QRLabel1	Departamento	<i>Negrito</i>	<i>Preta</i>
	QRLabel2	Localização	<i>Negrito</i>	<i>Preta</i>
	QRLabel3	Telefone	<i>Negrito</i>	<i>Preta</i>
<i>rbDetail</i>	QRLabel4	Nome	<i>Negrito</i>	<i>Azul</i>
	QRLabel5	Sobrenome	<i>Negrito</i>	<i>Azul</i>
	QRLabel6	Salário	<i>Negrito</i>	<i>Azul</i>

Tabela 6.1 Configuração dos rótulos dos campos no relatório



Veja o layout do relatório na **Figura 6.4**.



**Figura 6.4.** Layout do relatório mestre/detalhe simples

Adicione três *QRDBText* dentro da banda detalhe, e configure-os de acordo com a **Tabela 6.2**.

Componente	DataSet	DataField
QRDBText1	<i>DMPrincipal.cdsDepartamentos</i>	DEPARTMENT
QRDBText2	<i>DMPrincipal.cdsDepartamentos</i>	LOCATION
QRDBText3	<i>DMPrincipal.cdsDepartamentos</i>	PHONE_NO

**Tabela 6.2.** Configuração dos componentes que "imprimem" os dados

Vamos configurar a banda *SubDetail* para que imprima os campos do *DataSet* detalhe (*cdsEmpregados*). Selecione a banda *SubDetail* e aponte a propriedade *DataSet* para *DMPrincipal.cdsEmpregados*. Adicione três *QRDBText* dentro da banda *SubDetail* e configure-os de acordo com a **Tabela 6.3**.

Componente	DataSet	DataField
QRDBText4	<i>DMPrincipal.cdsEmpregados</i>	FIRST_NAME
QRDBText5	<i>DMPrincipal.cdsEmpregados</i>	LAST_NAME
QRDBText6	<i>DMPrincipal.cdsEmpregados</i>	SALARY

**Tabela 6.3.** Configurando a banda *SubDetail*

Veja como ficou o relatório na **Figura 6.5**.



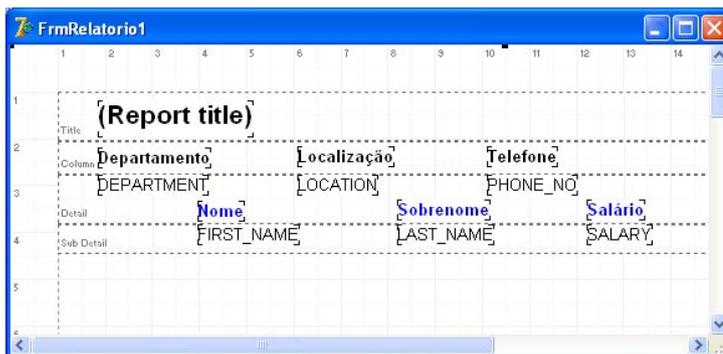


Figura 6.5. Layout final do relatório Mestre/Detalhe simples

Antes de fazer a chamada ao relatório devemos abrir os componentes para o acesso aos dados a serem impressos, neste exemplo os *ClientDataSets*. No evento *OnCreate* do *FrmRelatorio1* digite o seguinte código:

```
DMPrincipal.cdsDepartamentos.Open;
DMPrincipal.cdsEmpregados.Open;
```



Poderíamos ativar normalmente os *ClientDataSets* em tempo de *design*, alterando a propriedade *Active* para *True*.

Para terminar, vamos fazer a chamada ao relatório, para isso, selecione o *FrmPrincipal*, adicione um botão e altere a propriedade *Caption* para "Relatório 1 - Mestre/Detalhe simples". Para o evento *OnClick* do botão digite o seguinte código:

```
FrmRelatorio1.QuickRep1.Preview;
```

Adicione a unit *untFrmRelatorio1* a cláusula *uses* do *FrmPrincipal*. Rode a aplicação, clique sobre o botão e visualize o relatório (Figura 6.6).

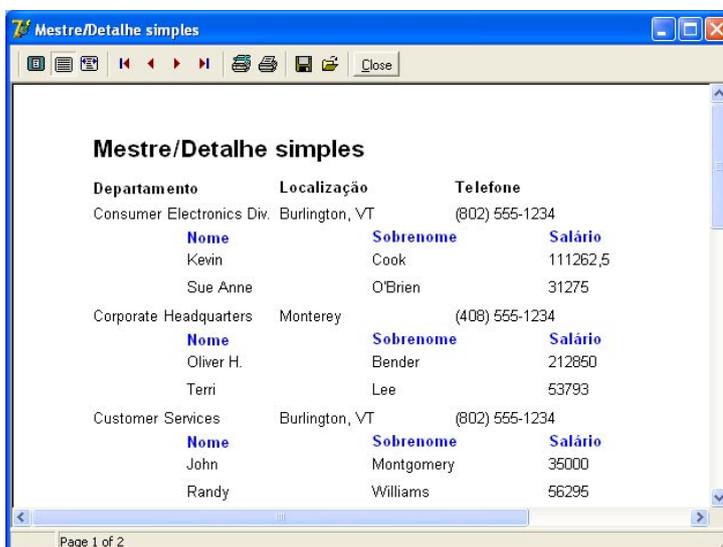


Figura 6.6. Relatório Mestre / Detalhe simples em tempo de execução



## Exemplo 2 - Mestre / Detalhe com 3 níveis

Neste exemplo, criaremos um relatório para imprimir os dados armazenados em três tabelas do banco. O relatório fará a impressão de todos os Países, Clientes e Vendas. Os dados serão obtidos nas tabelas COUNTRY, CUSTOMER e SALES.

Adicione três *SQLDataSet* ao *DMPrincipal*. Altere a propriedade *Name* do primeiro *SQLDataSet* para "datasetPais", do segundo para "datasetClientes" e do terceiro para "datasetVendas". Aponte a propriedade *SQLConnection* dos três *SQLDataSets* para *conCapitulo6*.

O *datasetPais* será o *DataSet* mestre, o *datasetClientes* será o primeiro detalhe, enquanto que o *datasetVendas* será o segundo detalhe. Para a propriedade *CommandText* do *datasetPais* digite a seguinte instrução SQL:

```
select * from COUNTRY
order by COUNTRY
```

Para o *datasetClientes* digite a seguinte instrução SQL na propriedade *CommandText*:

```
select * from CUSTOMER
where COUNTRY = :COUNTRY
order by CUSTOMER
```

Para o *datasetVendas* digite a seguinte instrução SQL na propriedade *CommandText*:

```
select * from SALES
where CUST_NO = :CUST_NO
order by PO_NUMBER
```

O campo COUNTRY é o campo que fará o relacionamento entre as tabelas COUNTRY e CUSTOMER. O campo CUST\_NO é o campo que fará o relacionamento entre as tabelas CUSTOMER e SALES. Adicione dois *DataSource* ao *DMPrincipal* e configure-os como mostra a Tabela 6.4.

Componente	Name	DataSet
DataSource1	dsPais	datasetPais
DataSource2	dsClientes	datasetClientes

Tabela 6.4. Configurando os DataSources

Selecione o *datasetClientes* e aponte a propriedade *DataSource* para *dsPais*. Para o *datasetVendas* aponte a propriedade *DataSource* para *dsClientes*. Terminado as configurações, os três *SQLDataSets* estão ligados a partir dos *DataSources*. Ainda no *DMPrincipal* adicione um *DataSetProvider* e três *ClientDataSet* (*Data Access*).

- ➔ Altere a propriedade *Name* do *DataSetProvider* para "dspPais";
- ➔ Altere a propriedade *Name* do primeiro *ClientDataSet* para "cdsPais";
- ➔ Altere a propriedade *Name* do segundo *ClientDataSet* para "cdsClientes";
- ➔ Para o terceiro *ClientDataSet* dê o nome de "cdsVendas";
- ➔ Ligue o *dspPais* ao *datasetPais*, através da propriedade *DataSet*;
- ➔ Ligue o *cdsPais* ao *dspPais*, através da propriedade *ProviderName*;
- ➔ Adicione os campos (*Fields*), conforme a técnica utilizada no exemplo anterior, para o *cdsPais*;



- Aponte a propriedade *DataSetField* do *cdsClientes* para *cdsPaisdatasetClientes* e adicione os campos (*Fields*), conforme o exemplo anterior;
- Selecione o *cdsVendas* e aponte a propriedade *DataSetField* para *cdsClientesdatasetVendas*.

Vamos criar o relatório para imprimir os dados contidos nos componentes *cdsPais*, *cdsClientes* e *cdsVendas*. Crie um novo formulário, altere o nome para "FrmRelatorio2" e salve-o como "untFrmRelatorio2.pas". Adicione ao *FrmRelatorio2* um *QuickRep*, e altere a propriedade *ReportTitle* para "Mestre/Detalhe - 3 níveis". Adicione três bandas (*QRBand*) ao relatório.

- Altere a propriedade *BandType* da primeira banda para *rbTitle*, da segunda para *rbColumnHeader* e da terceira para *rbDetail*;
- Adicione um *QRSysData* dentro da banda título e altere a propriedade *Data* para *qrsReportTitle*;
- Altere o tamanho da fonte do *QRSysData* para "18" e defina-a como negrito através da propriedade *Font*.

A banda detalhe será utilizada para fazer a impressão dos registros contidos no *DataSet* mestre (*cdsPais*). Para imprimir os dados de *cdsClientes* e *cdsVendas*, utilizaremos dois *QRSubDetail*, para isso, adicione dois *QRSubDetail* ao relatório (Figura 6.7).



Figura 6.7. QRSubDetail do relatório Mestre/Detalhe - 3 níveis

Para que o relatório possa acessar os componentes de acesso a dados, neste exemplo os *ClientDataSets*, adicione a unit *untDMPrincipal* a cláusula *uses* do *FrmRelatorio2*.

Selecione o *QuickRep1* e aponte a propriedade *DataSet* para *DMPrincipal.cdsPais*. Na banda de título das colunas (*rbColumnHeader*) adicione um *QLabel* e digite na propriedade *Caption*: "Nome do País".

Selecione o *QLabel* e altere a fonte para negrito através da propriedade *Font*. Adicione mais cinco *QLabel* a banda detalhe e configure-os de acordo com a Tabela 6.5.



Componente	Caption	Font	Cor da Fonte
QRLabel2	Cód.	Negrito	Azul
QRLabel3	Cliente	Negrito	Azul
QRLabel4	Nº Venda	Negrito	Azul
QRLabel5	Quant.	Negrito	Azul
QRLabel6	Valor	Negrito	Azul

**Tabela 6.5.** Configurando os rótulos dos dados a serem impressos

Veja como ficou o layout do relatório na **Figura 6.8**.



**Figura 6.8.** Layout do relatório Mestre/Detalhe - 3 níveis

Adicione um *QRDBText* dentro da banda detalhe, altere a propriedade *DataSet* para *DMPrincipal.cdsPais* e *DataField* para *COUNTRY*.

Vamos agora configurar a primeira banda *SubDetail* para que imprima os campos do *DataSet* detalhe (*cdsClientes*). Selecione a banda *SubDetail* e aponte a propriedade *DataSet* para *DMPrincipal.cdsClientes*. Adicione dois *QRDBText* dentro da banda *SubDetail* e configure-os de acordo com a **Tabela 6.6**.

Componente	DataSet	DataField	Alignment
QRDBText1	<i>DMPrincipal.cdsClientes</i>	CUST_NO	<i>taRightJustify</i>
QRDBText2	<i>DMPrincipal.cdsClientes</i>	CUSTOMER	<i>taLeftJustify</i>

**Tabela 6.6.** Configurações da primeira banda *SubDetail*

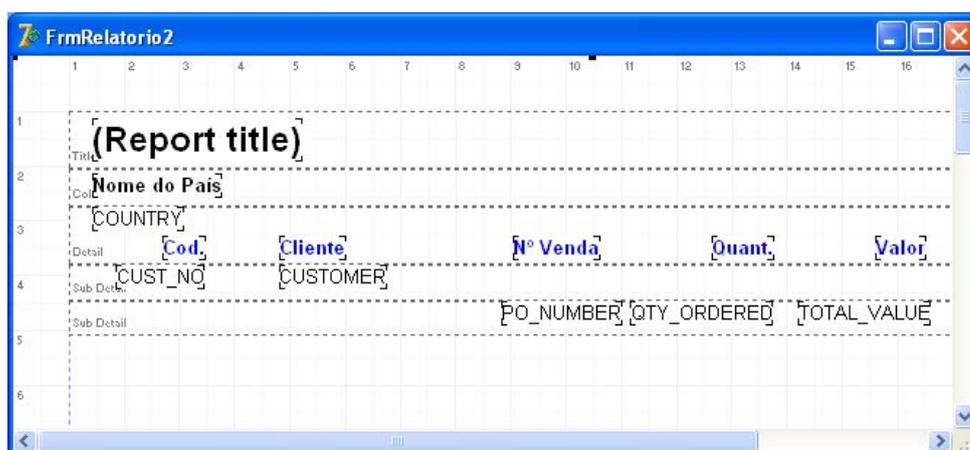
Para a segunda banda *SubDetail* vamos configurá-la, para que imprima os campos do segundo *DataSet* detalhe (*cdsVendas*). Selecione a banda *SubDetail* e aponte a propriedade *DataSet* para *DMPrincipal.cdsVendas*. Adicione três *QRDBText* dentro da banda *SubDetail* e configure-os de acordo com a **Tabela 6.7**.



Componente	DataSet	DataField	Alignment
QRDBText1	DMPPrincipal.cdsVendas	PO_NUMBER	taLeftJustify
QRDBText2	DMPPrincipal.cdsVendas	QTY_ORDERED	taRightJustify
QRDBText3	DMPPrincipal.cdsVendas	TOTAL_VALUE	taRightJustify

**Tabela 6.7.** Configurações da segunda banda SubDetail

Veja como ficou o relatório na **Figura 6.9**.



**Figura 6.9.** Layout do relatório Mestre/Detalhe - 3 níveis

Antes de fazer a chamada ao relatório, devemos abrir os componentes de acesso a dados que possuem os dados a serem impressos, nesse caso os *ClientDataSets*. No evento *OnCreate* do *FrmRelatorio1* digite o seguinte código:

```
DMPPrincipal.cdsPais.Open;
DMPPrincipal.cdsClientes.Open;
DMPPrincipal.cdsVendas.Open;
```



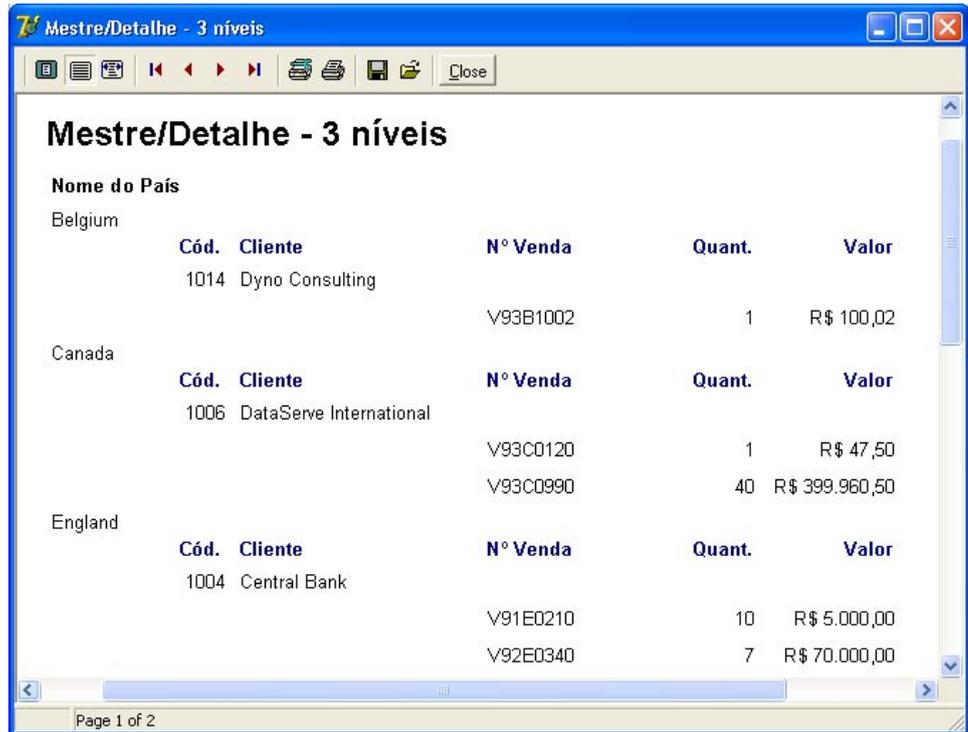
Poderíamos ativar normalmente os *ClientDataSets* em tempo de *design*, alterando a propriedade *Active* para *True*.

No *FrmPrincipal* adicione um botão e altere a propriedade *Caption* para "Relatório 2 - Mestre/Detalhe - 3 níveis". Para o evento *OnClick* do botão digite o seguinte código:

```
FrmRelatorio2.QuickRep1.Preview;
```

Adicione a unit *untFrmRelatorio2* a cláusula *uses* do *FrmPrincipal*. Rode a aplicação, clique sobre o botão adicionado anteriormente e visualize o relatório (**Figura 6.10**).





Mestre/Detailhe - 3 níveis				
Nome do País				
Belgium				
	<b>Cód. Cliente</b>	<b>Nº Venda</b>	<b>Quant.</b>	<b>Valor</b>
	1014 Dyno Consulting	V93B1002	1	R\$ 100,02
Canada				
	<b>Cód. Cliente</b>	<b>Nº Venda</b>	<b>Quant.</b>	<b>Valor</b>
	1006 DataServe International	V93C0120	1	R\$ 47,50
		V93C0990	40	R\$ 399.960,50
England				
	<b>Cód. Cliente</b>	<b>Nº Venda</b>	<b>Quant.</b>	<b>Valor</b>
	1004 Central Bank	V91E0210	10	R\$ 5.000,00
		V92E0340	7	R\$ 70.000,00

Figura 6.10. Relatório Mestre/Detailhe - 3 níveis em tempo de execução



Neste exemplo utilizei a propriedade *Currency* do *TOTAL\_VALUE* para ter o formato "moeda", sem a necessidade de especificar a propriedade *Mask* do *QRDBText*.

## Exercícios

Exercício 1 - Crie um novo relatório utilizando o QuickReport com o título "Histórico de Salários". Esse relatório precisa imprimir o histórico dos salários dos empregados cadastrados. Adicione quatro bandas ao relatório: título, cabeçalho das colunas, detalhe e sub-detalhe.

- Para o relatório, utilize as informações das tabelas *EMPLOYEE* e *SALARY\_HISTORY* do banco de dados *Employee.gdb* (utilizando *dbExpress* para fazer o acesso aos dados);
- Para a banda título imprima o título do relatório;
- Para a banda de cabeçalho das colunas adicione os seguintes títulos: Cód. e Empregado;
- Na banda Detalhe, imprima os campos *EMP\_NO* e *FIRST\_NAME*. Imprima também nessa banda os títulos das colunas: Data Alteração, Salário Antigo e Novo Salário;
- Na banda sub-detalhe imprima os dados dos seguintes campos da tabela *SALARY\_HISTORY*: *CHANGE\_DATE*, *OLD\_SALARY* e *NEW\_SALARY*.



Exercício 2 – Crie um novo relatório mestre / detalhe com o título de “Cargos por País”. Esse relatório fará a impressão dos cargos disponíveis em todos os países cadastrados. Adicione quatro bandas ao relatório: título, cabeçalho das colunas, detalhe e sub-detalhe.

- Para o relatório, utilize os dados das tabelas COUNTRY e JOB do banco de dados *Employee.gdb* (utilizando dbExpress para fazer o acesso aos dados);
- Para a banda título imprima o título do relatório e a data atual do sistema;
- Para a banda de cabeçalho das colunas adicione os seguintes títulos: País e Moeda;
- Na banda Detalhe, imprima os campos COUNTRY e CURRENCY. Imprima também nessa banda os títulos das colunas: País e Moeda.
- Na banda sub-detalhe imprima os dados dos seguintes campos da tabela JOB: JOB\_TITLE, MIN\_SALARY, MAX\_SALARY e JOB\_REQUIREMENT. Imprima também nessa banda os títulos das colunas: Cargo, Salário Mínimo, Salário Máximo e Requisitos;

Exercício 3 – Altere o relatório do exercício anterior (“Cargos por País”) para que imprima os campos em vermelho para todos os cargos que tiverem seu salário mínimo (MIN\_SALARY) inferior a vinte e cinco mil.

## Solução Exercício 1

Crie uma nova aplicação no Delphi, dê o nome ao formulário para “FrmPrincipal” e salve-o como “untFrmPrincipal.pas”, para o projeto dê o nome do “Exercicio1.dpr”. Adicione o DataModule utilizado no capítulo 6, através do menu *Project\Add to project*.

Adicione no *DMPrincipal* dois *SQLDataSets*, altere a propriedade *Name* do primeiro para “datasetEmpregado” e do segundo para “datasetSalario”. Aponte a propriedade *SQLConnection* dos dois *SQLDataSets* para *conCapitulo6*. Para a propriedade *CommandText* do *datasetEmpregado* digite a seguinte instrução SQL:

```
select EMP_NO, FIRST_NAME
from EMPLOYEE
order by FIRST_NAME
```

Para o *datasetSalario* digite a seguinte instrução SQL na propriedade *CommandText*:

```
select * from SALARY_HISTORY
where EMP_NO = :EMP_NO
order by CHANGE_DATE
```

Adicione um *DataSource* ao *DMPrincipal*. Altere o nome para “dsEmpregado” e ligue a propriedade *DataSet* ao *datasetEmpregado*. Selecione o *datasetSalario* e aponte a propriedade *DataSource* para *dsEmpregado*. Ainda no *DMPrincipal* adicione um *DataSetProvider* e dois *ClientDataSet* (*Data Access*).

- ➔ Altere a propriedade *Name* do *DataSetProvider* para “dspEmpregado”;
- ➔ Altere a propriedade *Name* do primeiro *ClientDataSet* para “cdsEmpregado”;
- ➔ Altere a propriedade *Name* do segundo *ClientDataSet* para “cdsSalario”;
- ➔ Ligue o *dspEmpregado*, através da propriedade *DataSet*, ao *datasetEmpregado* e ligue o *cdsEmpregado* ao *dspEmpregado* através da propriedade *ProviderName*;
- ➔ Clique com o botão direito do mouse sobre o *cdsEmpregado* e escolha a opção *Fields Editor*, dentro do editor de campos (*Fields Editor*), clique com o botão direito novamente e escolha a opção *Add all Fields*.



- ➔ Para o *cdsSalario*, aponte a propriedade *DataSetField* para *cdsEmpregadodatasetSalario*. Crie um novo formulário, altere o nome para "FrmExercicio1" e salve-o como "untFrmExercicio1.pas". Adicione ao *FrmExercicio1* um *QuickRep* e altere a propriedade *ReportTitle* para "Histórico de Salários". Adicione três bandas (*QRBand*) ao relatório.
  - ➔ Altere a propriedade *BandType* da primeira banda para *rbTitle*, da segunda para *rbColumnHeader* e da terceira para *rbDetail*;
  - ➔ Adicione um *QRSysData* dentro da banda título e altere a propriedade *Data* para *qrsReportTitle*;
  - ➔ Altere o tamanho da fonte do componente *QRSysData* para "16" e defina-a como negrito através da propriedade *Font*.
- Adicione um *QRSubDetail* ao relatório (Figura 6.11).

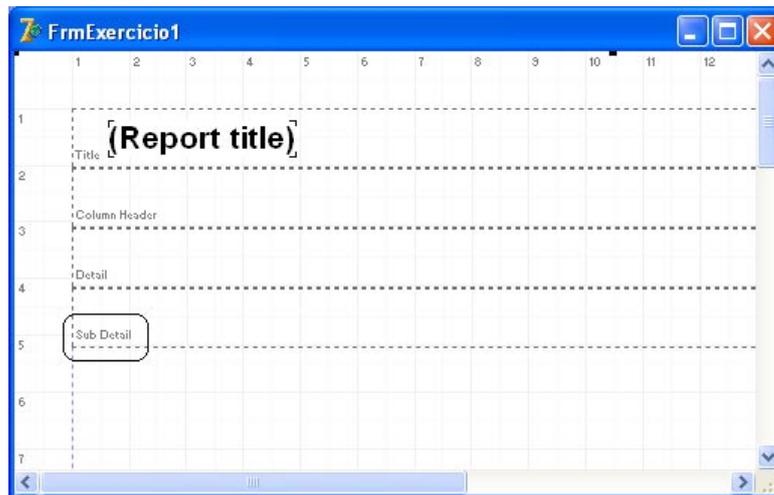


Figura 6.11. SubDetail do relatório Histórico de Salários

Adicione a unit *untDMPrincipal* a cláusula *uses* do *FrmExercicio1*. Selecione o *QuickRep1* e aponte a propriedade *DataSet* para *DMPrincipal.cdsEmpregado*. Na banda de título das colunas (*rbColumnHeader*) adicione dois *QRLabel*. Altere a propriedade dos *QRLabels* para "Cód." e "Empregado".

Altere a fonte dos componentes para negrito através da propriedade *Font*. Adicione mais três *QRLabel* e altere o *Caption* para "Data Alteração", "Salário Antigo" e "Novo Salário". Altere a cor da fonte dos componentes para azul e coloque negrito a partir da propriedade *Font*. Veja como ficou o layout do relatório na Figura 6.12.





Figura 6.12. Layout do relatório Histórico dos Salários

Adicione dois *QRDBText* dentro da banda detalhe. Configure-os de acordo com a Tabela 6.8.

Componente	DataSet	DataField	Alignment
QRDBText1	<i>DMPrincipal.cdsEmpregado</i>	EMP_NO	<i>taRightJustify</i>
QRDBText2	<i>DMPrincipal.cdsEmpregado</i>	FIRST_NAME	<i>taLeftJustify</i>

Tabela 6.8. Configurações da banda Detalhe

Selecione a banda *SubDetail* e aponte a propriedade *DataSet* para *DMPrincipal.cdsSalario*. Adicione três *QRDBText* dentro da banda *SubDetail* e configure-os de acordo com a Tabela 6.9.

Componente	DataSet	DataField	Alignment	Mask
QRDBText3	<i>DMPrincipal.cdsSalario</i>	CHANGE_DATE	<i>taLeftJustify</i>	--
QRDBText4	<i>DMPrincipal.cdsSalario</i>	OLD_SALARY	<i>taRightJustify</i>	R\$ ###,##0.00
QRDBText5	<i>DMPrincipal.cdsSalario</i>	NEW_SALARY	<i>taRightJustify</i>	R\$ ###,##0.00

Tabela 6.9. Configurações da banda SubDetail

Veja como ficou o relatório na Figura 6.13.



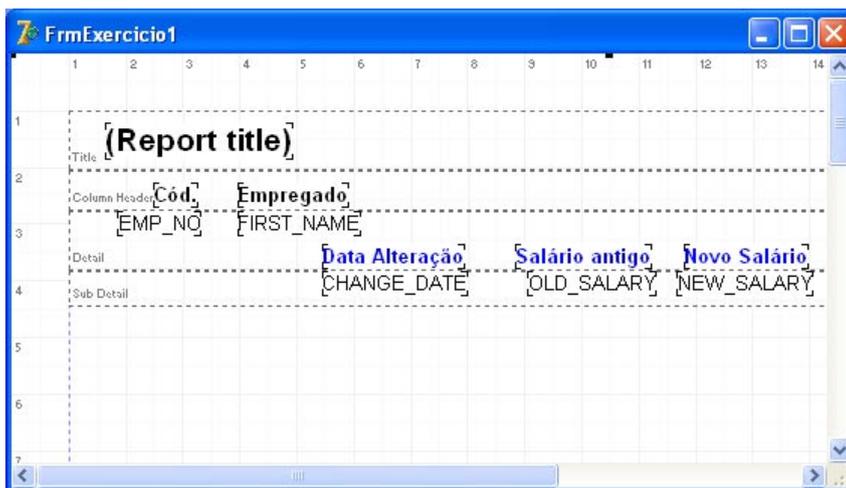


Figura 6.13. Layout do relatório Histórico de Salários

No evento *OnCreate* do *FrmExercicio1* digite o seguinte código:

```
DMPrincipal.cdsEmpregado.Open;
DMPrincipal.cdsSalario.Open;
```



Poderíamos ativar normalmente os *ClientDataSets* em tempo de *design*, alterando a propriedade *Active* para *True*.

No *FrmPrincipal* adicione um botão alterando a propriedade *Caption* para "Exercicio1 - Histórico de Salários". Para o evento *OnClick* do botão digite o seguinte código:

```
FrmExercicio1.QuickRep1.Preview;
```

Adicione a unit *untFrmExercicio1* a cláusula *uses* do *FrmPrincipal*. Rode a aplicação, clique sobre o botão e visualize o relatório em execução (Figura 6.14).



Figura 6.14. Relatório Histórico de Salários em tempo de execução



## Exercício 2

Crie uma nova aplicação no Delphi, dê o nome ao formulário para "FrmPrincipal" e salve-o como "untFrmPrincipal.pas", para o projeto dê o nome do "Exercicio2.dpr". Adicione um DataModule ao projeto e altere seu nome para "DMPrincipal". Salve a unit como "untDMPrincipal.pas".

Adicione no *DMPrincipal* um *SQLConnection (dbExpress)*, altere a propriedade *Name* para "conExercicio2" e configure-o para acessar o banco de dados *Employee.gdb*.



Utilize a conexão *Employee* criada em exercícios anteriores.

Adicione também dois *SQLDataSets*, altere a propriedade *Name* do primeiro para "datasetPaises" e do segundo para "datasetCargos". Aponte a propriedade *SQLConnection* dos dois *SQLDataSets* para *conExercicio2*. Para a propriedade *CommandText* do *datasetPaises* digite a seguinte instrução SQL:

```
select *  
from COUNTRY  
order by COUNTRY
```

Para o *datasetCargos* digite a seguinte instrução SQL na propriedade *CommandText*:

```
select * from JOB  
where JOB_COUNTRY = :COUNTRY  
order by JOB_TITLE
```

Adicione um *DataSource* ao *DMPrincipal*. Altere o nome para "dsPaises" e ligue a propriedade *DataSet* ao *datasetPaises*. Selecione o *datasetCargos* e aponte a propriedade *DataSource* para *dsPaises*. Ainda no *DMPrincipal* adicione um *DataSetProvider* e dois *ClientDataSet (Data Access)*.

- Altere a propriedade *Name* do *DataSetProvider* para "dspPaises";
- Altere a propriedade *Name* do primeiro *ClientDataSet* para "cdsPaises";
- Altere a propriedade *Name* do segundo *ClientDataSet* para "cdsCargos";
- Ligue o *dspPaises*, através da propriedade *DataSet*, ao *datasetPaises* e ligue o *cdsPaises* ao *dspPaises* através da propriedade *ProviderName*;
- Clique com o botão direito do mouse sobre o *cdsPaises* e escolha a opção *Fields Editor*, dentro do editor de campos (*Fields Editor*), clique com o botão direito novamente e escolha a opção *Add all Fields*;
- Para o *cdsCargos*, aponte a propriedade *DataSetField* para *cdsPaisesdatasetCargos*;
- Configure nos campos *MIN\_SALARY* e *MAX\_SALARY* a propriedade *Currency* para *True*.

Crie um novo formulário, altere o nome para "FrmExercicio2" e salve-o como "untFrmExercicio2.pas". Adicione ao *FrmExercicio2* um *QuickRep*, altere a propriedade *ReportTitle* para "Cargos por País" e adicione três bandas (*QRBand*) ao relatório.

- Altere a propriedade *BandType* da primeira banda para *rbTitle*, da segunda para *rbColumnHeader* e da terceira para *rbDetail*;

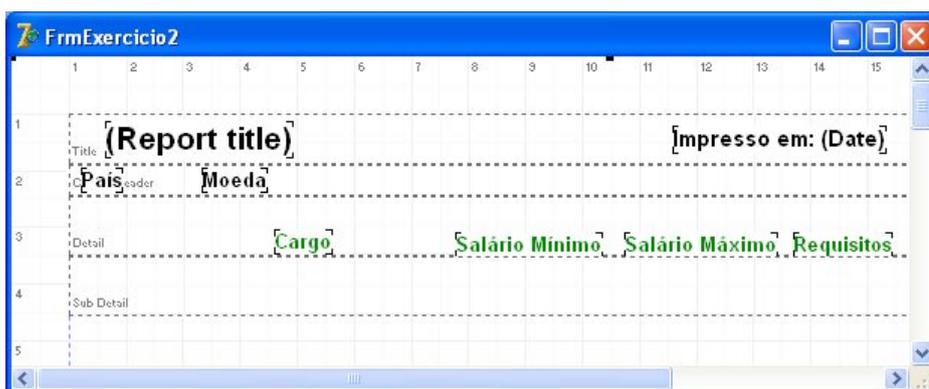


- Adicione um *QRSysData* dentro da banda título e altere a propriedade *Data* para *qrsReportTitle*;
- Altere o tamanho da fonte do *QRSysData* para "16" e defina-a como negrito através da propriedade *Font*;
- Adicione mais um *QRSysData* dentro da banda título e altere a propriedade *Data* para *qrsDate*. Altere também a propriedade *Text* para "Impresso em: " e defina a fonte para tamanho "11" e negrito através da propriedade *Font*;
- Adicione um *QRSubDetail* ao relatório.

Adicione a unit *untDMPrincipal* a cláusula *uses* do *FrmExercicio2*. Selecione o *QuickRep1* e aponte a propriedade *DataSet* para *DMPrincipal.cdsPaises*. Na banda de título das colunas (*rbColumnHeader*) adicione dois *QRLabel*. Altere a propriedade *Caption* dos componentes para "País" e "Moeda". Altere a fonte dos componentes para negrito através da propriedade *Font*.

Adicione quatro *QRLabel*, dentro da banda detalhe, altere o *Caption* para "Cargo", "Salário Mínimo", "Salário Máximo" e "Requisitos". Altere a cor da fonte dos componentes para verde e coloque negrito a partir da propriedade *Font*.

Veja como ficou o layout do relatório na **Figura 6.15**.



**Figura 6.15.** Layout do relatório Cargos por País

Adicione dois *QRDBText* dentro da banda detalhe. Configure-os de acordo com a **Tabela 6.9**.

Componente	DataSet	DataField
QRDBText1	<i>DMPrincipal.cdsPaises</i>	COUNTRY
QRDBText2	<i>DMPrincipal.cdsPaises</i>	CURRENCY

**Tabela 6.9.** Configurações da banda Detalhe

Selecione a banda *SubDetail* e aponte a propriedade *DataSet* para *DMPrincipal.cdsCargos*. Adicione três *QRDBText* dentro da banda *SubDetail* e configure-os de acordo com a **Tabela 6.10**.



Componente	DataSet	DataField	Alignment
QRDBText3	DMPPrincipal.cdsCargos	JOB_TITLE	taLeftJustify
QRDBText4	DMPPrincipal.cdsCargos	MIN_SALARY	taRightJustify
QRDBText5	DMPPrincipal.cdsCargos	MAX_SALARY	taRightJustify

**Tabela 6.10.** Configurações da banda SubDetail

Para o campo JOB\_REQUIREMENT adicione um *QRDBRichText*. Altere a propriedade *DataSet* para *DMPPrincipal.cdsCargos* e *DataField* para *JOB\_REQUIREMENT*.

Veja como ficou o relatório na **Figura 6.16**.



**Figura 6.16.** Layout do relatório Cargos por País em tempo de designer

No evento *OnCreate* do *FrmExercicio2* digite o seguinte código:

```
DMPPrincipal.cdsPaíses.Open;
DMPPrincipal.cdsCargos.Open;
```



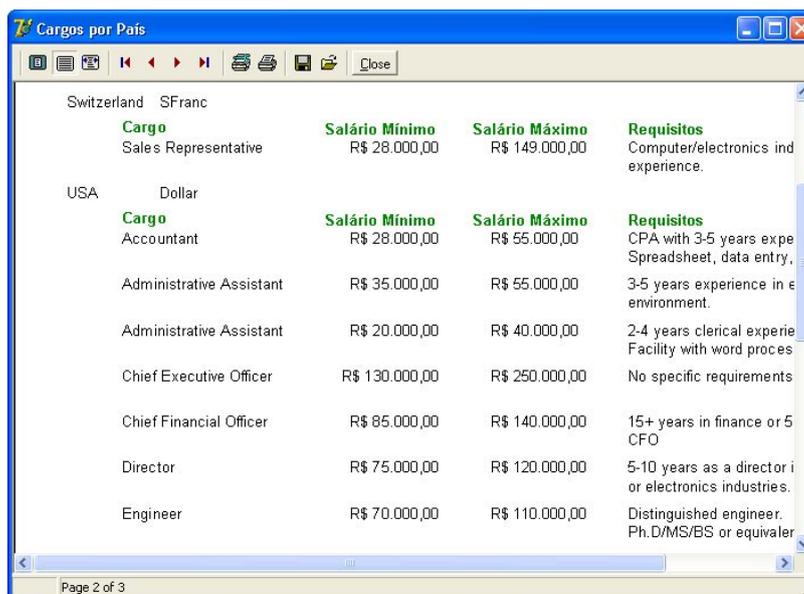
Poderíamos ativar normalmente os *ClientDataSets* em tempo de *design*, alterando a propriedade *Active* para *True*.

No *FrmPrincipal* adicione um botão alterando a propriedade *Caption* para "Exercicio2 – Cargos por País". Para o evento *OnClick* do botão digite o seguinte código:

```
FrmExercicio2.QuickRep1.Preview;
```

Adicione a unit *untFrmExercicio2* a cláusula *uses* do *FrmPrincipal*. Rode a aplicação, clique sobre o botão e visualize o relatório em execução (**Figura 6.17**).





Cargos	Salário Mínimo	Salário Máximo	Requisitos
Switzerland SF Franc			
Sales Representative	R\$ 28.000,00	R\$ 149.000,00	Computer/electronics ind experience.
USA Dollar			
Accountant	R\$ 28.000,00	R\$ 55.000,00	CPA with 3-5 years expe Spreadsheet, data entry,
Administrative Assistant	R\$ 35.000,00	R\$ 55.000,00	3-5 years experience in e environment.
Administrative Assistant	R\$ 20.000,00	R\$ 40.000,00	2-4 years clerical experie Facility with word proces
Chief Executive Officer	R\$ 130.000,00	R\$ 250.000,00	No specific requirements
Chief Financial Officer	R\$ 85.000,00	R\$ 140.000,00	15+ years in finance or 5 CFO
Director	R\$ 75.000,00	R\$ 120.000,00	5-10 years as a director i or electronics industries.
Engineer	R\$ 70.000,00	R\$ 110.000,00	Distinguished engineer. Ph.D/MS/BS or equivaler

Figura 6.17. Relatório Cargos por País em tempo de execução

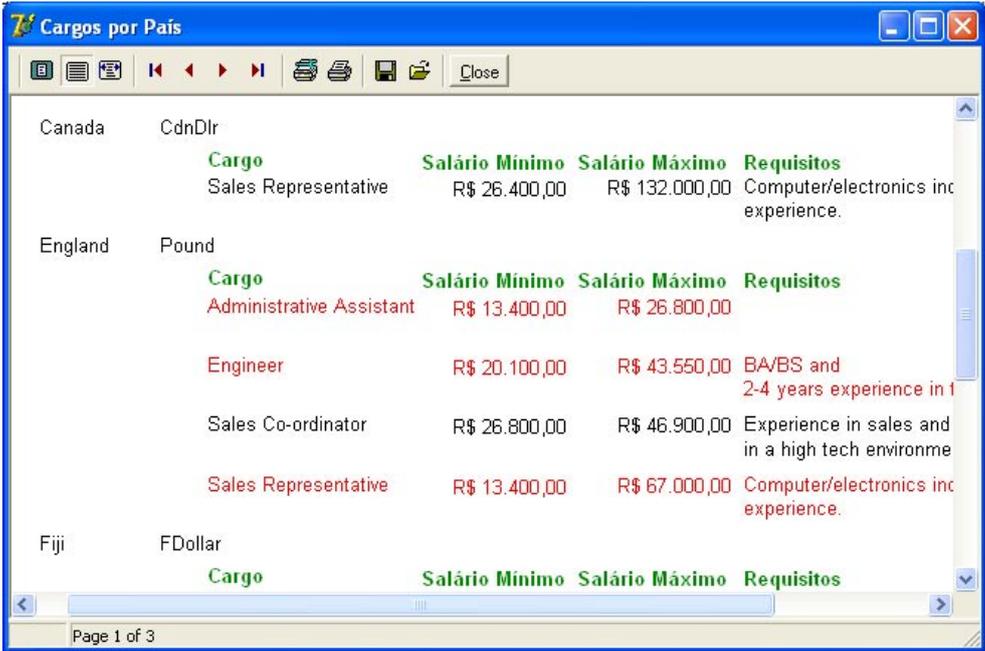
### Exercício 3

Com o *FrmExercicio2* selecionado, digite o seguinte código para o evento *BeforePrint* da banda sub-detalle:

```
procedure TFrmExercicio2.QRSubDetail1BeforePrint(
  Sender: TQRCustomBand; var PrintBand: Boolean);
begin
  if DMPrincipal.cdsCargosMIN_SALARY.AsCurrency <
    25000 then
  begin
    QRDBText3.Font.Color := clRed; { Cargo }
    QRDBText4.Font.Color := clRed; { Salário Mínimo }
    QRDBText5.Font.Color := clRed; { Salário Máximo }
    QRDBRichText1.Font.Color := clRed; { Requisitos }
  end
  else
  begin
    QRDBText3.Font.Color := clWindowText;
    QRDBText4.Font.Color := clWindowText;
    QRDBText5.Font.Color := clWindowText;
    QRDBRichText1.Font.Color := clWindowText;
  end;
end;
```

Rode novamente a aplicação, clique sobre o botão e visualize o relatório em execução (Figura 6.18).





País	Currency	Carga	Salário Mínimo	Salário Máximo	Requisitos
Canada	CdnDlr	Sales Representative	R\$ 26.400,00	R\$ 132.000,00	Computer/electronics inc experience.
England	Pound	Administrative Assistant	R\$ 13.400,00	R\$ 26.800,00	
England	Pound	Engineer	R\$ 20.100,00	R\$ 43.550,00	BA/BS and 2-4 years experience in t
England	Pound	Sales Co-ordinator	R\$ 26.800,00	R\$ 46.900,00	Experience in sales and in a high tech environme
England	Pound	Sales Representative	R\$ 13.400,00	R\$ 67.000,00	Computer/electronics inc experience.
Fiji	FDollar				

Figura 6.18. Relatório Cargos por País em tempo de execução destacando os salários inferiores a 25.000



# Capítulo 7

## Formatando Relatórios

Módulo I





Neste capítulo veremos algumas técnicas de formatação dos relatórios criados com QuickReport. Normalmente os relatórios de um sistema precisam seguir um padrão de layout, isso é, devem ser construídos sob um mesmo layout base.

Por exemplo, alguns relatórios precisam imprimir o logotipo da empresa na banda título, como também a data / hora da impressão, número de páginas etc. Você pode criar um relatório base, com todos os componentes e configurações que serão comuns a todos os relatórios do sistema, e criar os novos relatórios como descendentes do relatório base, o que chamamos no Delphi de *herança visual de formulários*.

### Exemplo 1 - Relatório base

Neste exemplo criaremos um relatório que possuirá todas as configurações e componentes básicos que todos os relatórios precisarão possuir. A vantagem de criar um relatório base para um sistema é que as alterações em sua estrutura afetam (atualiza) todos os relatórios que forem descendentes dele.

Uma outra vantagem, é que ao criarmos um novo relatório descendente de outro relatório, todos os componentes e configurações base já estarão presentes. Suponhamos que todos os nossos relatórios precisarão ter uma banda de título com o logotipo e nome da empresa, uma banda cabeçalho das colunas, uma banda detalhe, e uma banda rodapé da página, que fará a impressão da data / hora que o relatório foi impresso, como também o número da página atual do relatório.

Inicie uma nova aplicação no Delphi e altere o nome do formulário para "FrmPrincipal" e salve a unit como "untFrmPrincipal.pas". Para o arquivo de projeto dê o nome de "capitulo7.dpr". Crie um novo formulário (*File|New>Form*), altere o nome para "FrmRelatorioModelo" e salve-o como "untFrmRelatorioModelo.pas". Adicione um *QuickRep* ao formulário, juntamente com quatro bandas (*QRBand*) ao *QuickRep*.

➔ Altere a propriedade *BandType* da primeira banda para *rbTitle*, da segunda para *rbColumnHeader*, da terceira para *rbDetail* e finalmente da quarta para *rbPageFooter*.

Adicione um *QRImage* a banda título e selecione uma figura a partir da propriedade *Picture*. Ainda na banda título adicione dois *QRLabel* e digite na propriedade *Caption* "Nome da Empresa" e "Título do Relatório", respectivamente. Configure os componentes no estilo negrito, para o componente que mostrará o nome da empresa altere o tamanho da fonte para "22", e para o outro componente altere o tamanho da fonte para "14". Todas essas configurações através da propriedade *Font*.

Para a banda *Page Footer*, adicione dois *QRSysData* e digite na propriedade *Text* "Impresso em: " e "Pág.", e altere a propriedade *Data* para *qrsDateTime* e *qrsPageNumber*, respectivamente. Veja como ficou o layout do relatório modelo na **Figura 7.1**.



Figura 7.1. Layout do relatório modelo

Terminado as configurações do relatório base, basta agora criar os novos relatórios a partir dele. Para criar um novo relatório como descendente do relatório modelo, selecione o menu *File|New>Other*. Na janela *New Items* acesse a aba *capitulo7* (aba do projeto atual), e selecione o formulário *FrmRelatorioModelo*, como mostra a **Figura 7.2**.

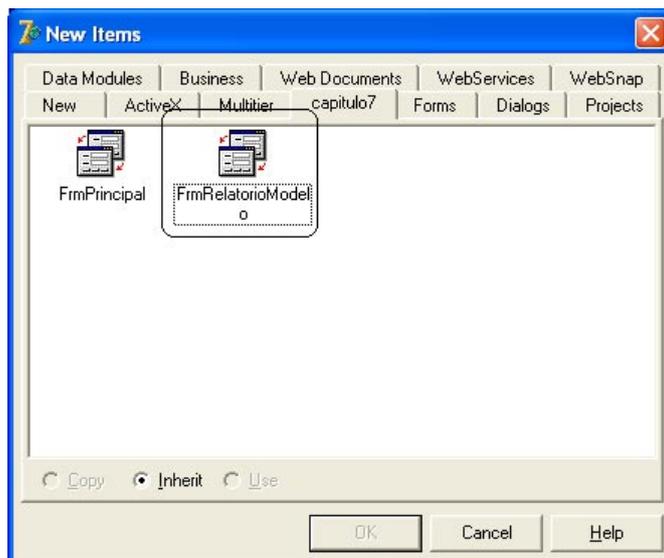


Figura 7.2. Criando novos relatórios a partir do relatório modelo

Clique no botão OK. Será criado um novo formulário com todos os componentes e configurações definidas no formulário modelo (*FrmRelatorioModelo*).



Lembre-se que alterações feitas no *FrmRelatorioModelo* afeta todos os relatórios herdados a partir dele. É importante lembrar também, que todos os componentes criados no novo relatório herdadado não podem ser apagados, pois esses pertencem ao formulário modelo (*FrmRelatorioModelo*).

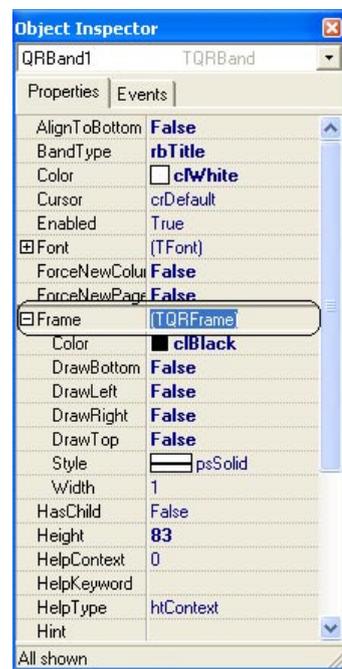


Caso você não queira mostrar um componente, altere para *False* a propriedade *Visible*. Se não deseja mostrar a banda, altere para *False* à propriedade *Enabled*.

## Exemplo 2 - Opções diversas

Neste segundo exemplo vamos mostrar algumas opções de formatação que podem ser usados nos relatórios. Normalmente usuários do QuickReport utilizam o *QRShape* para criar linhas divisórias entre bandas do relatório, caixas retangulares para destacar valores no relatório etc.

Caso você queira criar um retângulo, ou uma linha superior, ou inferior por exemplo em torno de uma banda do relatório, você pode, ao invés de utilizar o *QRShape*, utilizar a propriedade *Frame* disponível na banda (**Figura 7.3**).



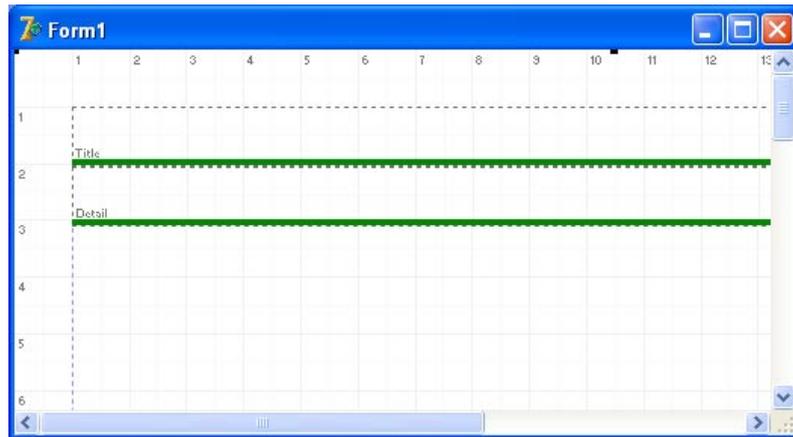
**Figura 7.3.** Propriedade *Frame* para definição de linhas em torno da banda

A propriedade *Frame* possui várias opções, como:

- *Color*: para seleção da cor da linha;
- *DrawBottom*: adiciona uma linha na parte inferior da banda;
- *DrawLeft*: adiciona uma linha na parte esquerda da banda;
- *DrawRight*: adiciona uma linha na parte direita da banda;
- *DrawTop*: adiciona uma linha na parte superior da banda;
- *Style*: propriedade para a definição do estilo da linha, isso é, se a linha será contínua, pontilhada, etc.
- *Width*: propriedade para a definição da largura da linha.

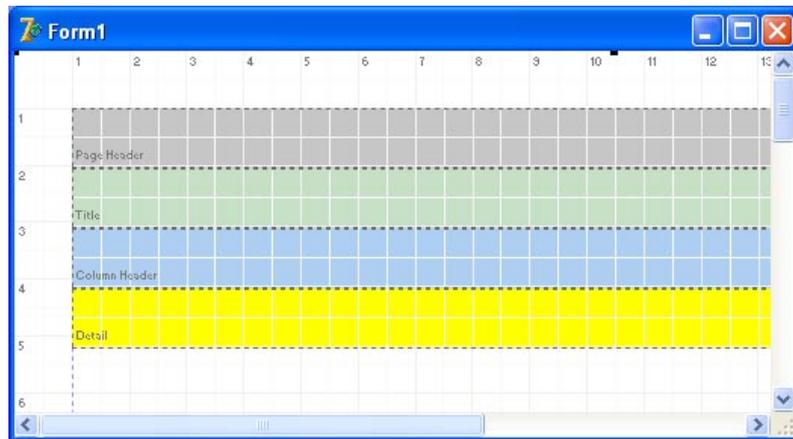
Veja na **Figura 7.4** a configuração da frame (na cor verde, *Width(3)* e *DrawBottom(True)*) das bandas *Title* e *Detail*.





**Figura 7.4.** Definindo a propriedade *frame* da banda

Toda banda possui a propriedade *Color*, onde a partir dela podemos alterar a cor de fundo de uma banda (Figura 7.5).



**Figura 7.5.** Alterando a cor das bandas de um relatório

Não podemos esquecer também da propriedade *Font*, onde é possível alterar a cor, tamanho, tipo e estilo do componente que será impresso (Figura 7.6).



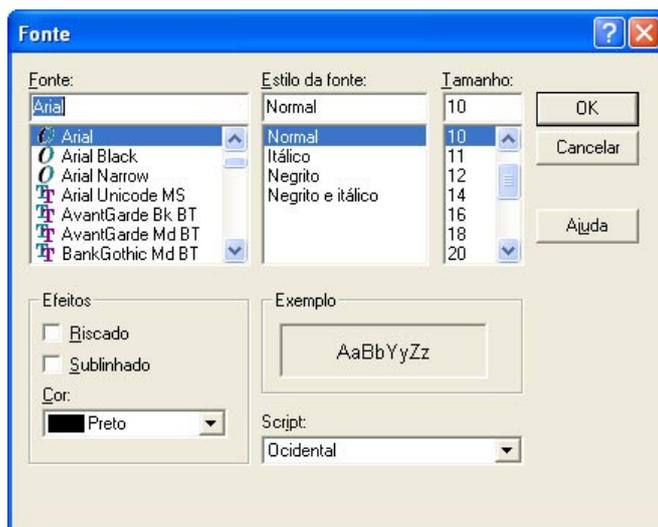


Figura 7.6. Configurando a fonte de um componente do relatório

### Exemplo 3 - Alternando a cor da banda detalhe

Neste exemplo mostraremos como alternar a cor de fundo de uma banda detalhe durante sua impressão. Essa opção de formatação é bastante utilizada, pois facilita a leitura do relatório.



Você pode aplicar esse exemplo em todos os relatórios que fizemos nos capítulos anteriores, desde que possuam uma banda detalhe.

Crie um novo formulário, altere o nome para "FrmRelatorio2" e salve-o como "untFrmRelatorio2.pas". Adicione um *QuickRep* ao formulário, e adicione duas bandas (*QRBand*) ao *QuickRep*.

➔ Altere a propriedade *BandType* da primeira banda para *rbColumnHeader* e da segunda para *rbDetail*.

Adicione quatro *QRLabel* a banda Título e configure-os de acordo com a Tabela 7.1.

Componente	Caption	Font
QRLabel1	Nome	<i>Negrito, tamanho 11</i>
QRLabel2	Sobrenome	<i>Negrito, tamanho 11</i>
QRLabel3	Telefone	<i>Negrito, tamanho 11</i>
QRLabel4	Salário	<i>Negrito, tamanho 11</i>

Tabela 7.1. Configurando os QRLabels



Adicione um *ClientDataSet* ao relatório e altere a propriedade *Name* para “*cdsEmpregados*”. Clique com o botão direito do mouse e escolha a opção *Load from MyBase table*. Na janela *Open* selecione o arquivo *employee.xml* (que por padrão, está em *C:\Arquivos de Programas\Arquivos Comuns\Borland Shared\Data\*). Selecione o *QuickRep* e aponte a propriedade *DataSet* para *cdsEmpregados*. Adicione quatro *QRDBText* a banda detalhe e configure-os de acordo com a **Tabela 7.2**.

Componente	DataSet	DataField
QRDBText1	<i>cdsEmpregados</i>	FIRSTNAME
QRDBText2	<i>cdsEmpregados</i>	LASTNAME
QRDBText3	<i>cdsEmpregados</i>	PHONEEXT
QRDBText4	<i>cdsEmpregados</i>	SALARY

**Tabela 7.2.** Configuração dos QRDBText do relatório

Veja como ficou o layout do relatório na **Figura 7.7**.



**Figura 7.** Layout do relatório de exemplo.

Todas as bandas de um relatório possuem os eventos *AfterPrint* e *BeforePrint* que são disparados antes e depois da banda ser impressa. Como o objetivo deste exemplo é alternar a cor da banda detalhe, devemos fazer a condição para a alteração da cor antes da banda detalhe ser impressa.

Selecione a banda detalhe e adicione ao evento *BeforePrint* o seguinte código:

```
{ Altere QRBand2 para o nome da banda Detail
do seu relatório }
if QRBand2.Color = clSilver then
  QRBand2.Color := clWhite
else
  QRBand2.Color := clSilver;
```

O código anterior verifica a cor atual da banda detalhe, e caso ela seja cinza, é impresso em branca, senão, é impressa em cinza.



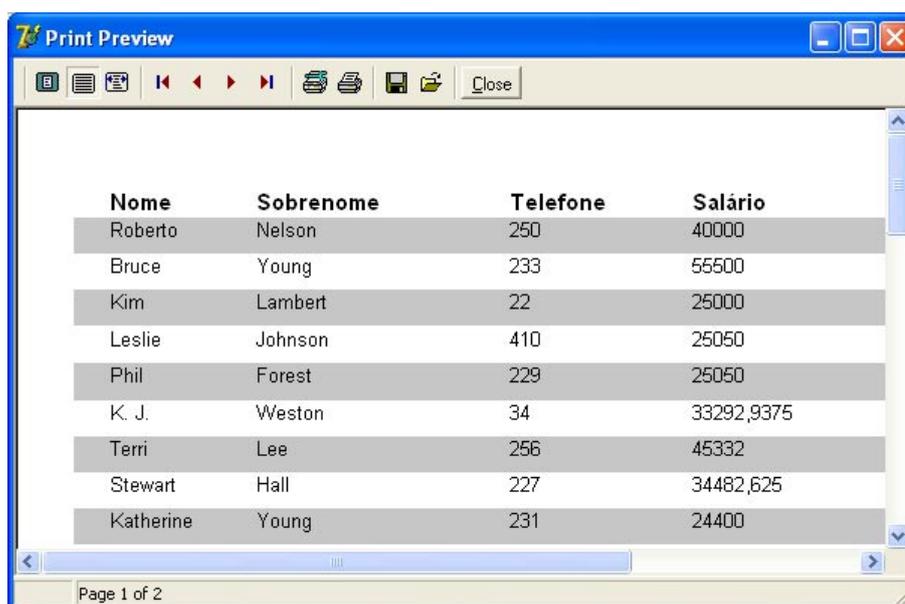
Para que os *QRDBText* que estão dentro da banda detalhe não sejam impressos com o fundo branco, independente da cor da banda detalhe, altere a propriedade *Transparent* para *True*. Dessa forma, os componentes *QRDBText* ficarão transparentes, isso é, sem cor de fundo.



Para executar o relatório, primeiramente, verifique se a propriedade *Active* do *cdsEmpregados* está *True*, e adicione um botão ao *FrmPrincipal* alterando a propriedade *Caption* para "Relatório 2 - Alternando cores". Para o evento *OnClick* do botão digite o seguinte código:

```
FrmRelatorio2.QuickRep1.Preview;
```

Adicione a unit *untFrmRelatorio2* a cláusula *uses* do *FrmPrincipal*. Rode a aplicação, clique sobre o botão e visualize o relatório em execução na **Figura 7.8**.



Nome	Sobrenome	Telefone	Salário
Roberto	Nelson	250	40000
Bruce	Young	233	55500
Kim	Lambert	22	25000
Leslie	Johnson	410	25050
Phil	Forest	229	25050
K. J.	Weston	34	33292,9375
Terri	Lee	256	45332
Stewart	Hall	227	34482,625
Katherine	Young	231	24400

**Figura 7.8.** Relatório "zebrado" em tempo de execução



# Capítulo 8

Trabalhando com expressões

Módulo I



Neste capítulo veremos algumas das técnicas disponíveis no componente *QRExpr* para a criação de expressões em relatórios QuickReport. Além do *QRExpr*, o QuickReport possibilita o uso de expressões nos componentes *QRExprMemo* e *QRGroup*, a partir da propriedade *Expression*.

As expressões podem ser usadas para manipular campos de banco de dados, como também para formatações mais avançadas. A sintaxe utilizada nas expressões são muito parecidas com o *Object Pascal*. Uma expressão suporta os tipos *boolean*, *integer*, *float* ou *string*. Campos data e hora são convertidos para *string* dentro da expressão.



Campos do tipo Memo e BLOB não são suportados em expressões no QuickReport.

Veja na **Tabela 8.1** os operadores suportados pelas expressões.

Operador	Descrição
+	Adiciona e concatena <i>strings</i>
-	Subtração de valores
*	Multiplificação de valores
/	Divisão de valores
And, Or, Not	Operadores lógicos
=, < >, >=, <= e <>	Operadores de comparação

**Tabela 8.1.** Operadores suportados pelas expressões

Na **Tabela 8.2** veja as funções que as expressões suportam.



Função	Descrição
UPPER( <i>string</i> )	Retorna a string em maiúsculo
LOWER( <i>string</i> )	Retorna a string em minúsculo
TIME	Retorna a hora corrente como uma string
DATE	Retorna a data corrente como uma string
SUM( <i>campo</i> )	Retorna a soma dos valores do campo
MAX( <i>campo</i> )	Retorna o maior valor do campo
MIN( <i>campo</i> )	Retorna o menor valor do campo
AVERAGE( <i>campo</i> )	Retorna a média dos valores do campo
DIV(X, Y)	Retorna a divisão inteira de X e Y
COPY( <i>str</i> , I, F)	Retorna a Substring de <i>str</i> começando no caracter I, com tamanho F
FRAC( <i>numero</i> )	Retorna a parte fracionário do numero
FORMATNUMERIC(F, Num)	Formata o número Num usando a máscara F
INT( <i>numero</i> )	Retorna a parte inteira de um número
IF( <i>expressão</i> , A1, A2)	Retorna A1 ou A2 dependendo da expressão lógica
STR( <i>numero</i> )	Converte um número para uma string
PRETTY( <i>string</i> )	Retorna a primeira letra em maiúsculo, e as demais em minúsculo

Tabela 8.2. Funções suportadas pelas expressões

O QuickReport possui um editor exclusivo para a criação de expressões (Figura 8.1).

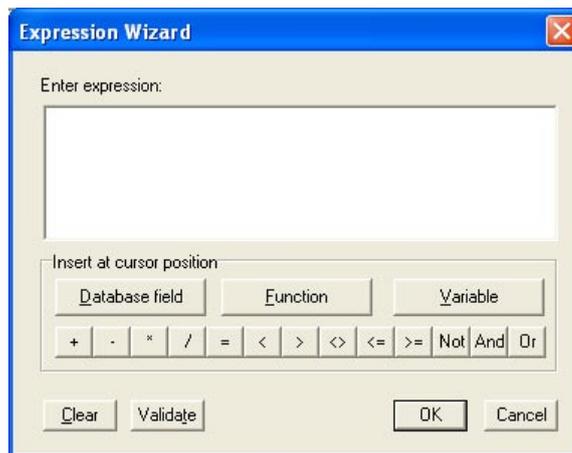


Figura 8.1. Editor para a criação de expressões

Utilizando o botão *Database field* é possível selecionar os campos que você deseja utilizar na expressão (Figura 8.2).

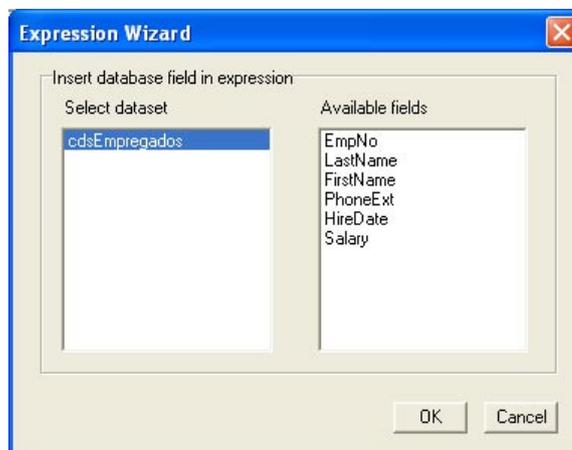


Figura 8.2. Selecionando os campos a serem usados na expressão





A janela para seleção dos campos do banco de dados só traz os DataSets que estiverem dentro do relatório, isso é, DataSets em DataModules não são mostrados nessa janela.

Utilizando o botão *Function* é possível selecionar as funções que serão usadas na expressão (Figura 8.3).

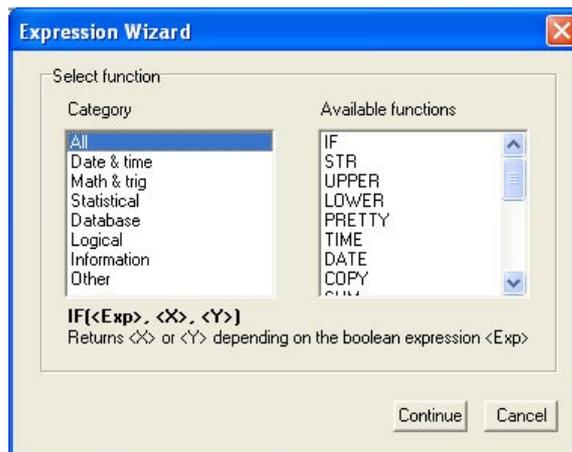


Figura 8.3. Selecionando a função a ser usada na expressão

Através do botão *Variable* é possível selecionar algumas variáveis de sistema, como número da página, título do relatório etc. Terminando a expressão você pode validá-la e testá-la através da opção *Validate*. Caso sua expressão esteja correta, é mostrada uma mensagem trazendo o valor de retorno da expressão, senão uma mensagem de erro é gerada.

## Exemplo 1 - Expressões

Neste exemplo vamos criar um relatório simples para imprimir uma lista a partir de uma tabela do banco de dados. Utilizaremos algumas das funções disponíveis no *QRExpr* para criar expressões em cima dos campos da tabela.

Inicie uma nova aplicação Delphi (*File|New>Application*), altere o nome do formulário para "FrmPrincipal" e salve a unit como "untFrmPrincipal.pas". Para o arquivo de projeto dê o nome de "capitulo8.dpr". Crie um novo formulário, altere o nome para "FrmRelatorio1" e salve-o como "untFrmRelatorio1.pas". Adicione um *QuickRep* ao formulário, juntamente com duas bandas (*QRBand*).

➔ Altere a propriedade *BandType* da primeira banda para *rbColumnHeader* e da segunda para *rbDetail*.

Adicione cinco *QRLabel* a banda Título das colunas e configure-os de acordo com a Tabela 8.3.



Componente	Caption	Font
QRLabel1	Nome	Negrito, tamanho 11
QRLabel2	Sobrenome	Negrito, tamanho 11
QRLabel3	Nome completo	Negrito, tamanho 11
QRLabel4	Data	Negrito, tamanho 11
QRLabel5	Salário	Negrito, tamanho 11

**Tabela 8.3.** Configurações dos QRLabel

Adicione um *ClientDataSet* ao relatório e altere a propriedade *Name* para "cdeEmpregados". Clique com o botão direito do mouse e escolha a opção *Load from MyBase table*. Na janela *Open* selecione o arquivo *employee.xml* (que por padrão, está em *C:\Arquivos de Programas\Arquivos Comuns\Borland SharedData*).

Abra o editor de campos (*Fields Editor*) e adicione os campos, clicando com o botão direito e escolhendo a opção *Add all fields*. Aponte a propriedade *DataSet* do *QuickRep1* para *cdeEmpregados*. Veja como ficou o layout do relatório na **Figura 8.4**.



**Figura 8.4.** Layout do relatório de exemplo

Criaremos cinco expressões simples utilizando os campos do *cdeEmpregados*. Adicione cinco *QRExpr* a banda detalhe e configure-os de acordo com a **Tabela 8.4**:

Componente	Expressão	Explicação
QRExpr1	UPPER(FirstName)	Altera o campo <i>FirstName</i> para maiúsculo
QRExpr2	LOWER(LastName)	Altera o campo <i>LastName</i> para minúsculo
QRExpr3	FirstName + ' ' + LastName	Concatena os campos <i>FirstName</i> e <i>LastName</i>
QRExpr4	COPY(HireDate,1,2) + '-' + COPY(HireDate,4,2) + '-' + COPY(HireDate,7,4)	Separa o dia, o mês, e o ano com hífen.
QRExpr5	FORMATNUMERIC('R\$ ###,##0.00', Salary)	Formata o campo <i>Salário</i>

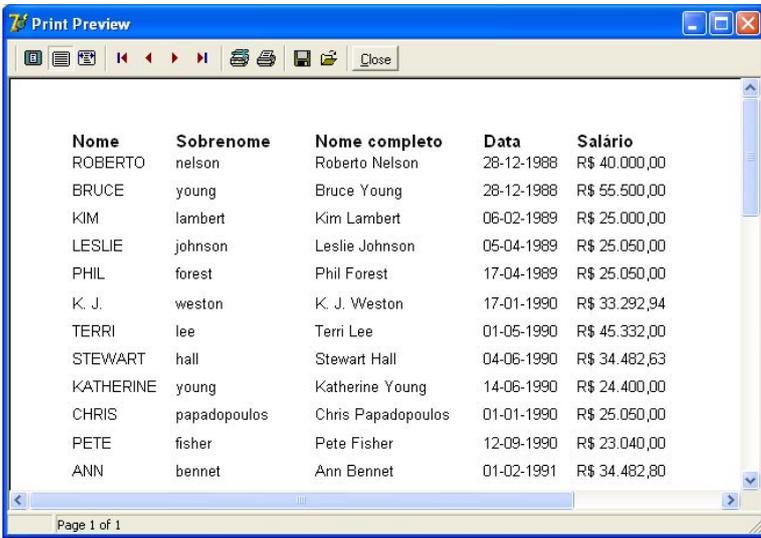
**Tabela 8.4.** Configuração das expressões com o QRExpr



Para executar o relatório, primeiramente, verifique se a propriedade *Active* do *cdsEmpregados* está *True* e adicione um botão ao *FrmPrincipal* alterando a propriedade *Caption* para "Relatório 1 - Expressões". Para o evento *OnClick* do botão digite o seguinte código:

```
FrmRelatorio1.QuickRep1.Preview;
```

Adicione a unit *untFrmRelatorio1* a cláusula *uses* do *FrmPrincipal*. Rode a aplicação, clique sobre o botão e visualize o relatório em execução na **Figura 8.5**.



The screenshot shows a 'Print Preview' window with a table of employee data. The table has five columns: 'Nome', 'Sobrenome', 'Nome completo', 'Data', and 'Salário'. The data is as follows:

Nome	Sobrenome	Nome completo	Data	Salário
ROBERTO	nelson	Roberto Nelson	28-12-1988	R\$ 40.000,00
BRUCE	young	Bruce Young	28-12-1988	R\$ 55.500,00
KIM	lambert	Kim Lambert	06-02-1989	R\$ 25.000,00
LESLIE	johnson	Leslie Johnson	05-04-1989	R\$ 25.050,00
PHIL	forest	Phil Forest	17-04-1989	R\$ 25.050,00
K. J.	weston	K. J. Weston	17-01-1990	R\$ 33.292,94
TERRI	lee	Terri Lee	01-05-1990	R\$ 45.332,00
STEWART	hall	Stewart Hall	04-06-1990	R\$ 34.482,63
KATHERINE	young	Katherine Young	14-06-1990	R\$ 24.400,00
CHRIS	papadopoulos	Chris Papadopoulos	01-01-1990	R\$ 25.050,00
PETE	fisher	Pete Fisher	12-09-1990	R\$ 23.040,00
ANN	bennet	Ann Bennet	01-02-1991	R\$ 34.482,80

Page 1 of 1

**Figura 8.5.** Relatório de exemplo em tempo de execução



# Capítulo 9

Unindo relatórios

Módulo I



Neste capítulo veremos como fazer a impressão de dois relatórios desenvolvidos com QuickReport ao mesmo tempo, isso é, imprimir um relatório atrás do outro em uma única operação.

Por exemplo, digamos que você precise imprimir um relatório analítico de vendas e precise analisar os valores totais gerados por essas vendas a partir de um relatório sintético. Você pode fazer a impressão do relatório sintético logo após o término do relatório analítico das vendas. Para que isso seja possível, devemos utilizar o *QRCompositeReport*.

### Exemplo 1 - Unindo dois relatórios

Neste exemplo vamos criar dois relatórios bastante simples para mostrar o funcionamento do *QRCompositeReport*. Inicie uma nova aplicação Delphi (*File|New>Application*), altere o nome do para "FrmPrincipal" e salve a unit como "untFrmPrincipal.pas". Para o arquivo de projeto dê o nome de "capitulo9.dpr". Crie um novo formulário (*File/New/Form*), altere o nome para "FrmRelatoriosSimples" e salve-o como "untFrmRelatoriosSimples.pas". Adicione dois *QuickRep* ao formulário. Adicione uma banda detalhe para cada *QuickRep* (**Figura 9.1**).

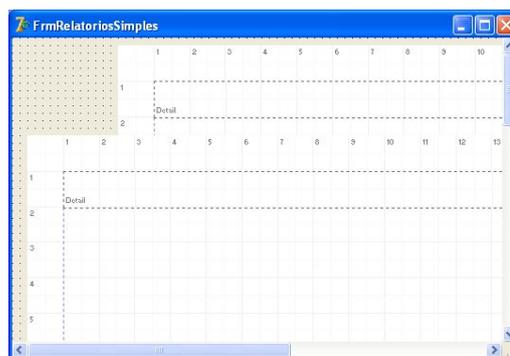


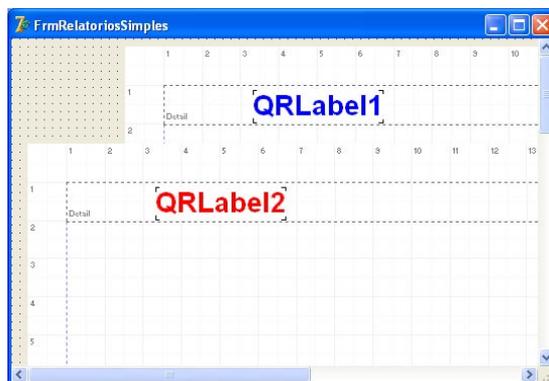
Figura 9.1. Componentes QuickRep

Adicione dois *QLabel*, um em cada banda detalhe. Configure-os de acordo com a Tabela 9.1.

Componente	Caption	Font
QLabel1	Relatório 1	<i>Negrito, tamanho 20, cor azul</i>
QLabel2	Relatório 2	<i>Negrito, tamanho 20, cor vermelha</i>

Tabela 9.1. Configuração dos QLabel para cada relatório

Veja como ficou o layout dos relatórios na **Figura 9.2**.



**Figura 9.2.** Layout dos relatórios de exemplo

Adicione um *QRCompositeReport* ao *FrmPrincipal*. O *QRCompositeReport* possui o evento *OnAddReports*, no qual precisamos definir os relatórios que serão impressos. Para o evento *OnAddReports* digite o seguinte código:

```
QRCompositeReport1.Reports.Add(  
    FrmRelatoriosSimples.QuickRep1);  
QRCompositeReport1.Reports.Add(  
    FrmRelatoriosSimples.QuickRep2);
```

Através da propriedade *Reports* (do tipo *TList*) podemos armazenar os relatórios que serão impressos. Você pode adicionar quantos relatórios achar necessário. Adicione a unit *FrmRelatoriosSimples* a cláusula *uses* do *FrmRelatorio1*.

Para executar o relatório, adicione um botão ao formulário alterando a propriedade *Caption* para "Relatório 1 - Unindo relatórios". Para o evento *OnClick* do botão digite o seguinte código:

```
QRCompositeReport1.Preview;
```



Caso você queira enviar o relatório direto para a impressora, basta trocar o método *Preview* pelo método *Print* do *QRCompositeReport*.

Rode a aplicação, clique sobre o botão e visualize o relatório em execução na **Figura 9.3**.



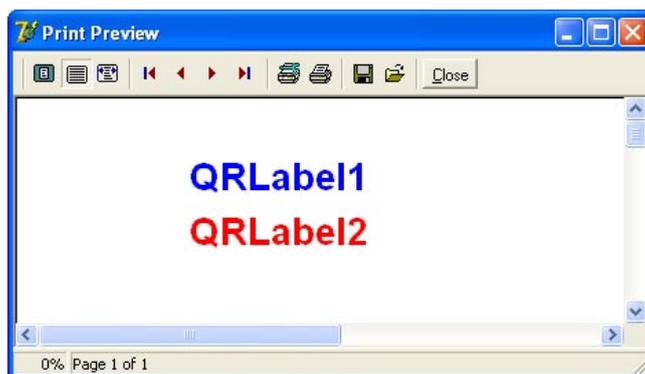


Figura 9.3. Relatório de exemplo em tempo de execução

## Exemplo 2 - Relatório de Vendas

Neste exemplo vamos criar dois relatórios: um trazendo todas as vendas efetuadas e outro trazendo as melhores vendas. Faremos a impressão desses relatórios utilizando o *QRCompositeReport*.

Crie um novo formulário, altere o nome para "FrmRelatorioVendas" e salve-o como "untFrmRelatorioVendas.pas". Adicione um *QuickRep* ao formulário, juntamente com duas bandas (*QRBand*).

➔ Altere a propriedade *BandType* da primeira banda para *rbColumnHeader* e da segunda para *rbDetail*.

Adicione um *QRLabel* a banda título das colunas e altere a propriedade *Caption* para "Vendas efetuadas". Altere também o tamanho da fonte para "14", negrito e de cor azul. Adicione quatro *QRLabel* a banda Título das colunas e configure-os de acordo com a Tabela 9.2.

Componente	Caption	Font
QRLabel2	Nº Venda	Negrito, tamanho 10
QRLabel3	Cód. Cliente	Negrito, tamanho 10
QRLabel4	Data da Venda	Negrito, tamanho 10
QRLabel5	Valor da Venda	Negrito, tamanho 10

Tabela 9.2. Configurações dos QRLabel

Adicione um *ClientDataSet* ao relatório e altere a propriedade *Name* para "cdsVendas". Clique com o botão direito do mouse e escolha a opção *Load from MyBase table*. Na janela *Open* selecione o arquivo *orders.xml* (que por padrão, está em *C:\Arquivos de Programas\Arquivos Comuns\Borland Shared\Data\*). Selecione o *QuickRep* e aponte a propriedade *DataSet* para *cdsVendas*. Adicione quatro *QRDBText* a banda detalhe e configure-os de acordo com a Tabela 9.3.

Componente	DataSet	DataField	Alignment
QRDBText1	cdsVendas	ORDERNO	taRightJustify
QRDBText2	cdsVendas	CUSTNO	taRightJustify
QRDBText3	cdsVendas	SALEDATE	taLeftJustify
QRDBText4	cdsVendas	ITEMSTOTAL	taRightJustify

Tabela 9.3. Configuração dos QRDBText



Veja como ficou o layout do relatório na **Figura 9.4**.



**Figura 9.4.** Layout do relatório de vendas

Verifique se a propriedade *Active* do *cdsVendas* está *True*. Crie um novo formulário, altere o nome para "FrmRelatorioMelhoresVendas" e salve-o como "untFrmRelatorioMelhoresVendas.pas". Adicione um *QuickRep* ao formulário, juntamente com duas bandas (*QRBand*).

→ Altere a propriedade *BandType* da primeira banda para *rbColumnHeader* e da segunda para *rbDetail*;

Adicione um *QRLabel* a banda título das colunas e altere a propriedade *Caption* para "Melhores Vendas". Altere também o tamanho da fonte para "14", negrito e de cor vermelha. Adicione quatro *QRLabel* a banda Título das colunas e configure-os de acordo com a **Tabela 9.4**.

Componente	Caption	Font
QRLabel2	Nº Venda	<i>Negrito, tamanho 10</i>
QRLabel3	Cód. Cliente	<i>Negrito, tamanho 10</i>
QRLabel4	Data da Venda	<i>Negrito, tamanho 10</i>
QRLabel5	Valor da Venda	<i>Negrito, tamanho 10</i>

**Tabela 9.4.** Configurações dos rótulos

Adicione um *ClientDataSet* ao relatório e altere a propriedade *Name* para "cdsMelhoresVendas". Clique com o botão direito do mouse e escolha a opção *Load from MyBase table*. Na janela *Open* selecione o arquivo *orders.xml* (que por padrão, está em *C:\Arquivos de Programas\Arquivos Comuns\Borland Shared\Data\*). Selecione o *QuickRep* e aponte a propriedade *DataSet* para *cdsMelhoresVendas*. Adicione quatro *QRDBText* a banda detalhe e configure-os de acordo com a **Tabela 9.5**.

Componente	DataSet	DataField	Alignment
QRDBText1	<i>cdsMelhoresVendas</i>	ORDERNO	<i>taRightJustify</i>
QRDBText2	<i>cdsMelhoresVendas</i>	CUSTNO	<i>taRightJustify</i>
QRDBText3	<i>cdsMelhoresVendas</i>	SALEDATE	<i>taLeftJustify</i>
QRDBText4	<i>cdsMelhoresVendas</i>	ITEMSTOTAL	<i>taRightJustify</i>

**Tabela 9.5.** Configurações dos campos do relatório



Veja como ficou o layout do relatório na **Figura 9.5**.



**Figura 9.5.** Layout do relatório de Melhores vendas

Selecione o *QuickRep1*, e no evento *BeforePrint* digite o seguinte código:

```
if cdsMelhoresVendas.Active then
  cdsMelhoresVendas.Close;
cdsMelhoresVendas.Filter := 'ItemsTotal > 30000';
cdsMelhoresVendas.Filtered := True;
cdsMelhoresVendas.Open;
```

No código anterior aplicamos um filtro no *cdsMelhoresVendas* para que seja impresso apenas as vendas com o valor maior que trinta mil.



Usamos filtro apenas para fins didáticos, em aplicações utilize instrução SQL parametrizada.

Adicione um *QRCompositeReport* ao *FrmPrincipal* e no evento *OnAddReports* do componente, digite o seguinte código:

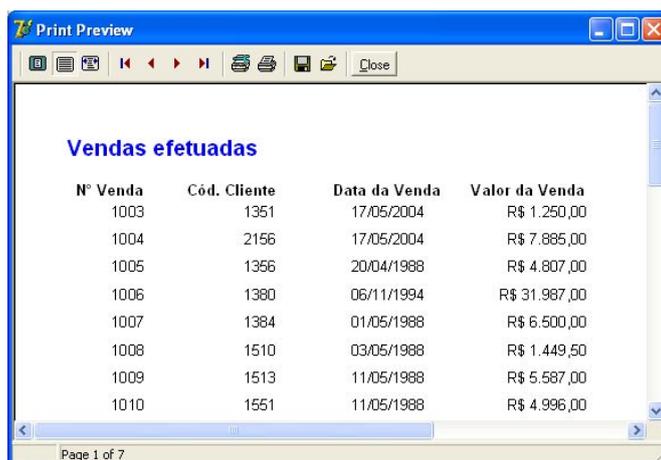
```
QRCompositeReport2.Reports.Add(
  FrmRelatorioVendas.QuickRep1);
QRCompositeReport2.Reports.Add(
  FrmRelatorioMelhoresVendas.QuickRep1);
```

Adicione as units *untFrmRelatorioVendas* e *untFrmRelatorioMelhoresVendas* a cláusula *uses* do *FrmPrincipal*. Para executar o relatório, adicione um botão ao *FrmPrincipal* alterando a propriedade *Caption* para "Relatório 2 - Relatório de Vendas". Para o evento *OnClick* do botão digite o seguinte código:

```
QRCompositeReport2.Preview;
```

Rode a aplicação, clique sobre o botão adicionado anteriormente e visualize o relatório em execução nas **Figuras 9.6 e 9.7**.





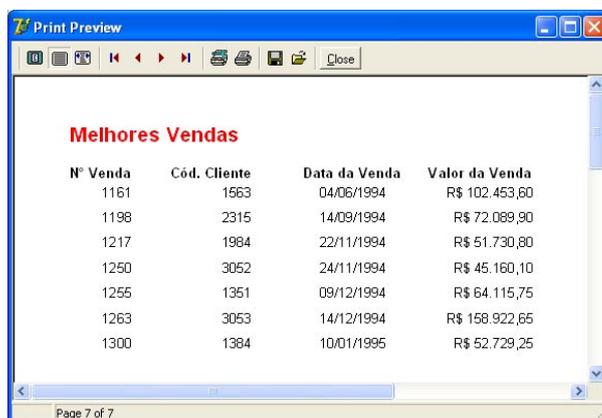
**Vendas efetuadas**

Nº Venda	Cód. Cliente	Data da Venda	Valor da Venda
1003	1351	17/05/2004	R\$ 1.250,00
1004	2156	17/05/2004	R\$ 7.885,00
1005	1356	20/04/1988	R\$ 4.807,00
1006	1380	06/11/1994	R\$ 31.987,00
1007	1384	01/05/1988	R\$ 6.500,00
1008	1510	03/05/1988	R\$ 1.449,50
1009	1513	11/05/1988	R\$ 5.587,00
1010	1551	11/05/1988	R\$ 4.996,00

Page 1 of 7

Figura 9.6. Relatório de Vendas efetuadas

Movendo o relatório para a última página, você verá o relatório das melhores vendas.



**Melhores Vendas**

Nº Venda	Cód. Cliente	Data da Venda	Valor da Venda
1161	1563	04/06/1994	R\$ 102.453,80
1198	2315	14/09/1994	R\$ 72.089,90
1217	1984	22/11/1994	R\$ 51.730,80
1250	3052	24/11/1994	R\$ 45.160,10
1255	1351	09/12/1994	R\$ 64.115,75
1263	3053	14/12/1994	R\$ 158.922,65
1300	1384	10/01/1995	R\$ 52.729,25

Page 7 of 7

Figura 9.7. Relatório de Melhores Vendas



# Capítulo 10

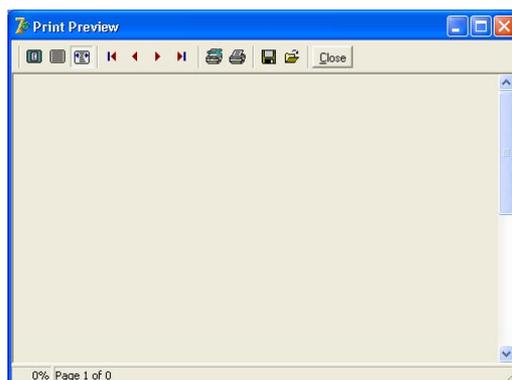
Criando um Preview personalizado

Módulo I



O QuickReport trás um visualizador de relatórios por padrão (**Figura 10.1**), quando invocamos o método *Preview* do *QuickRep*. Neste capítulo veremos como criar nosso próprio visualizador de relatórios.

Ganharemos várias vantagens ao criar um visualizador personalizado para os relatórios, entre elas: adição de novas funcionalidades, como *zoom*, tradução das opções disponíveis no visualizador para o português, etc.



**Figura 10.1.** Preview padrão para visualização de relatórios do QuickReport

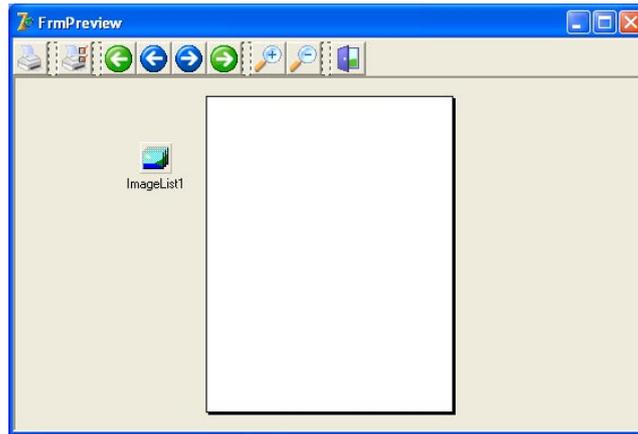
Inicie uma nova aplicação Delphi (*File|New>Application*), altere o nome para "FrmPrincipal" e salve a unit como "untFrmPrincipal.pas". Para o arquivo de projeto dê o nome de "capitulo10.dpr".

Crie um novo formulário, altere o nome para "FrmPreview" e salve-o como "untFrmPreview.pas". Adicione um *QRPreview (QReport)* ao formulário.



O *QRPreview* é utilizado para mostrar relatórios desenvolvidos em QuickReport.

Adicione um *Toolbar*, um *ImageList (Win32)* e adicione nove botões (clique com o botão direito e escolha *New Button*) no *FrmPreview*. Ligue o *ImageList* ao *Toolbar* através da propriedade *Images*. Adicione alguns ícones ao *ImageList* e tome por base o formulário na **Figura 10.2**.



**Figura 10.2.** Formulário para visualização dos relatórios

A próxima etapa é criar uma classe para o preview customizado derivada da classe *TQRPreviewInterface*. Na seção *type* do *FrmPreview* digite o seguinte código:

```
type
  TPreviewCustomizadoInterface = class(
    TQRPreviewInterface)
  public
    function Show(AQRPrinter: TQRPrinter):
      TWinControl; override;
  end;
  ...
```

Definimos uma função (*Show*) na classe *TPreviewCustomizadoInterface*, para mostrar o *FrmPreview*. Na seção *public* do *FrmPreview* digite o seguinte código:

```
public
  constructor Create(AOwner: TComponent;
    aQRPrinter: TQRPrinter);
end;
```

Implemente a função com o código a seguir:

```
function TPreviewCustomizadoInterface.Show(
  AQRPrinter: TQRPrinter): TWinControl;
var
  FrmPreview: TFrmPreview;
begin
  FrmPreview := TFrmPreview.Create(
    Application, AQRPrinter);
  FrmPreview.Show;
  Result := FrmPreview;
end;
```

Implemente o construtor do *FrmPreview* com o seguinte código:

```
constructor TFrmPreview.Create(
  AOwner: TComponent; aQRPrinter: TQRPrinter);
begin
  inherited Create(AOwner);
  QRPreview1.QRPrinter := aQRPrinter;
end;
```

Terminado a definição dos métodos, vamos implementar as funcionalidades dos botões do *FrmPreview*, conforme a **Tabela 10.1**.



Botão	Código
Imprimir	QRPreview1.QRPrinter.Print;
Setup	QRPreview1.QRPrinter.PrintSetup;
Primeira	QRPreview1.PageNumber := 1;
Anterior	QRPreview1.PageNumber := QRPreview1.PageNumber - 1;
Próxima	QRPreview1.PageNumber := QRPreview1.PageNumber + 1;
Última	QRPreview1.PageNumber := QRPreview1.QRPrinter.PageCount;
Zoom (-)	QRPreview1.Zoom := QRPreview1.Zoom - 10;
Zoom (+)	QRPreview1.Zoom := QRPreview1.Zoom + 10;
Fechar	Close;

**Tabela 10.1.** Código dos botões da barra de ferramentas

O método *Print* da propriedade *QRPrinter* do *QRPreview* envia o relatório corrente para a impressora. O *PrintSetup* mostra a janela para seleção e configuração da impressora que será usada para a impressão do relatório. Os botões de navegação, possuem o código para navegarem entre as páginas do relatório. Os botões de *zoom* incrementam ou decrementam o *zoom* do relatório em 10%, e finalmente o botão *Fechar*, fecha o formulário.

O formulário personalizado para visualização de relatórios QuickReport está pronto. Vamos criar um relatório simples para que possamos testar o preview personalizado. Crie um novo formulário, altere o nome para "FrmRelatorio1", salve-o como "untFrmRelatorio1.pas" e adicione um *QuickRep*, juntamente com duas bandas (*QRBand*) ao *QuickRep*.

- ➔ Altere a propriedade *BandType* da primeira banda para *rbColumnHeader* e da segunda para *rbDetail*.

Adicione quatro *QRLabel* a banda Título das colunas e configure-os de acordo com a **Tabela 10.2.**

Componente	Caption	Font
QRLabel1	Nome	<i>Negrito, tamanho 10</i>
QRLabel2	Categoria	<i>Negrito, tamanho 10</i>
QRLabel3	Espécie	<i>Negrito, tamanho 10</i>
QRLabel4	Figura	<i>Negrito, tamanho 10</i>

**Tabela 10.2.** Configuração dos rótulos do relatório

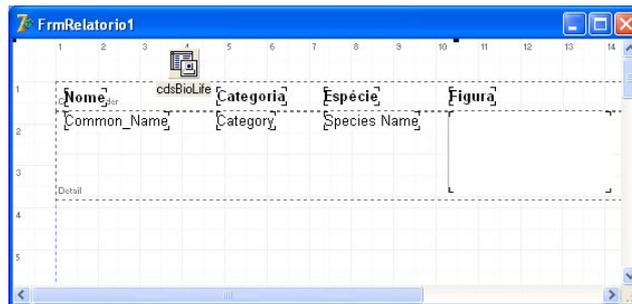
Adicione um *ClientDataSet* ao relatório e altere a propriedade *Name* para "cdsBiolife". Clique com o botão direito do mouse e escolha a opção *Load from MyBase table*. Na janela *Open* selecione o arquivo *biolife.xml* (que por padrão, está em *C:\Arquivos de Programas\Arquivos Comuns\Borland Shared\Data\*). Aponte a propriedade *DataSet* do *QuickRep1* para *cdsBiolife*. Adicione três *QRDBText* a banda detalhe e configure-os de acordo com a **Tabela 10.3.**



Componente	DataSet	DataField
QRDBText1	<i>cdsBiolife</i>	COMMON_NAME
QRDBText2	<i>cdsBiolife</i>	CATEGORY
QRDBText3	<i>cdsBiolife</i>	SPECIES_NAME

**Tabela 10.3.** Configuração dos componentes para mostrar os registros no relatório

Adicione um *QRDBImage* a banda detalhe e aponte a propriedade *DataSet* para *cdsBiolife* e *DataField* para o campo GRAPHIC. Veja como ficou o layout do relatório na **Figura 10.3**.

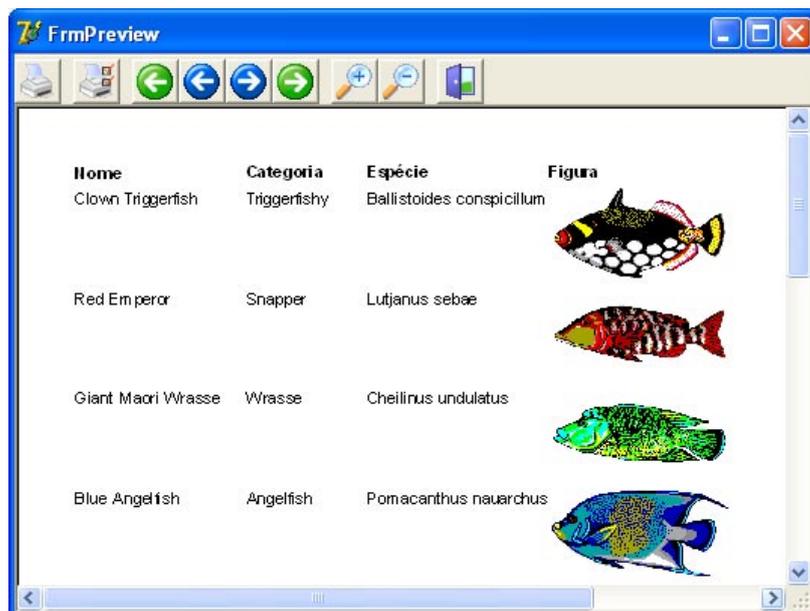


**Figura 10.3.** Layout do relatório de exemplo

Verifique se a propriedade *Active* do *cdsBiolife* está *True* e adicione um botão ao *FrmPrincipal* alterando a propriedade *Caption* para "Relatório 1 - Preview Customizado". Para o evento *OnClick* do botão digite o seguinte código:

```
RegisterPreviewClass(TPreviewCustomizadoInterface);
FrmRelatorio1 := TFrmRelatorio1.Create(Self);
FrmRelatorio1.QuickRep1.Preview;
```

Precisamos utilizar a função *RegisterPreviewClass* para registrar o preview personalizado antes de usá-lo. Declare as units *QRPrntr*, *untFrmRelatorio1* e *untFrmPreview* na cláusula *uses* do *FrmPrincipal*. Rode a aplicação, clique sobre o botão e visualize o relatório em execução



**Figura 10.4.** Relatório utilizando o Preview customizado



# Capítulo 11

Dicas

Módulo I





Neste capítulo mostraremos algumas dicas do QuickReport.

## Alterar Caption e Hint do Preview

Mostramos no capítulo 10 como criar um preview personalizado, mas mostraremos aqui como “traduzir” os botões e *Hint* do preview padrão do QuickReport. Para isso, abra o arquivo *Qrprev.dfm*, que esta no diretório *Lib* do Delphi, localize os botões (*TToolButton*) e altere a propriedade *Hint*, como por exemplo no código a seguir, referente ao botão que mostra a primeira página do relatório:

```
object FirstPage: TToolButton  
  Left = 85  
  Top = 0  
  Hint = 'Primeira Página'  
  Caption = 'FirstPage'  
  ImageIndex = 3  
  OnClick = FirstPageClick  
end
```

Após isso, recompile a sua aplicação e o preview padrão do QuickReport estará “traduzido” (Figura 11.1).



Figura 11.1. Traduzindo o preview padrão do QuickReport

## Criando rapidamente um relatório

Você pode criar um formulário com um *QuickRep*, através do menu *File|New>Other>Report* (Figura 11.2). Assim é criado um formulário e um *QuickRep* “incorporado” ao formulário. Vale lembrar que você não pode acessar os eventos e propriedades do formulário.

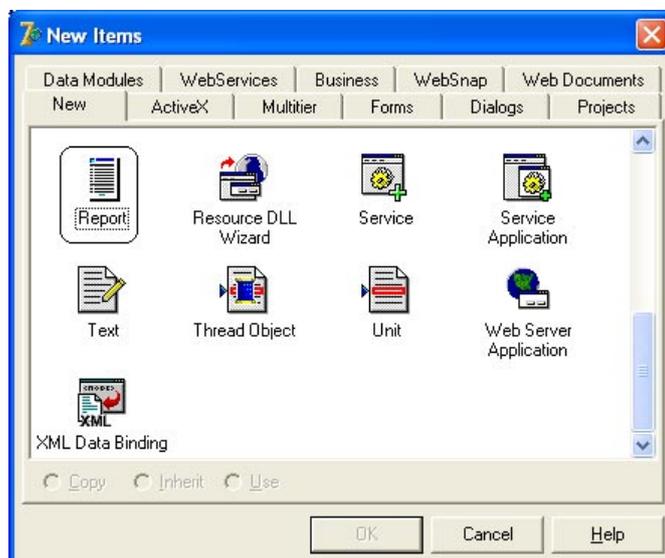


Tabela 11.2. Criando rapidamente um formulário com o QuickRep incorporado

