

Nesta seção você encontra artigos para iniciantes na linguagem Delphi

## Dicas de QuickReport

Conheça algumas dicas para facilitar seus relatórios com QuickReport



**Maikel Marcelo Scheid**

([maikelscheid@gmail.com](mailto:maikelscheid@gmail.com))

é técnico em Informática com ênfase em Análise e Programação de Sistemas. Atua na área de Desenvolvimento de Softwares em Delphi para plataforma Win32 e .NET com banco de dados Firebird e MSSQL. É membro da Equipe Editorial ClubeDelphi.

As ferramentas de relatórios são um meio do programador proporcionar aos usuários formas mais amplas e práticas de visualizar informações armazenadas em uma base de dados. Informações estas que podem ser oriundas de telas de cadastros ou de ações geradas pelo próprio sistema, como cadastros de cidades, clientes, produtos ou até mesmo o registro de *log* criado por um usuário ao executar alguma ação no sistema.

O QuickReport acompanha o Delphi desde sua primeira versão e até a versão 6 vinha incorporado ao seu IDE. Na versão do Delphi 7 e posteriores, com o surgimento do Rave Reports, o QuickReport deixou de vir pré-instalado no IDE, ou seja, é necessário fazer a instalação manual do componente. Já no BDS 2006 e RAD Studio 2007 é necessário fazer o *download* e instalação a partir do site da CodeGear (somente usuários registrados).

Neste artigo veremos a instalação dos componentes do QuickReport no Delphi 7 e criaremos relatórios dinâmicos através de formulários e listagem de informações resgatadas do banco de dados. Passaremos a aplicar filtros de pesquisa em tempo de execução. Veremos também uma forma prática de customizar o layout dos relatórios criados com o QuickReport. Aplicaremos regras de formatação e também uma forma simples de agrupar registros.

---

**Nota:** Para consultar mais detalhes, novidades, informações técnicas, download e/ou suporte técnico do pacote QuickReport consulte o site do fabricante em [www.qusoft.com](http://www.qusoft.com).

---

### Instalando o pacote de componentes

Durante este artigo iremos utilizar o Delphi 7 para criação da aplicação e exemplos práticos para o uso de

relatórios com os componentes do QuickReport. Como abordado na introdução ao artigo, na versão do Delphi 7 e posteriores o QuickReport deixou de vir pré-instalado no IDE. Para tê-lo instalado será necessário adicionar sua BPL ao Delphi.

Feche todos os formulários abertos o menu *File|Close All* e inicie o processo de instalação no menu *Component>Install Packages*. Clique no botão *Add* e navegue até a pasta de *Bin* do Delphi que por *default* é *C:\Arquivos de Programas\Borland\Delphi7\Bin\*. Localize o arquivo *dclqrt70.bpl* e selecione-o. Feito isso o pacote de componente já é mostrado na lista como *QuickReport Components* conforme podemos ver na **Figura 1**.

### Criando a aplicação

Para demonstração dos relatórios criaremos uma aplicação com acesso ao banco de dados *Employee.fdb* que acompanha a instalação do Firebird e a partir deste iremos configurar nossas situações de relatório por meio de pesquisas e filtros realizadas nos componentes.

No menu *File|New>Application* crie uma nova aplicação e altere a propriedade *Name* do formulário inicial para "frmPrincipal" e a propriedade *Caption* para "Dicas QuickReport". Salve a *Unit* principal do projeto como "uPrincipal.pas" e o projeto salve como "prjQuickReport.dpr".

Para criar a conexão com banco de dados e estruturar os componentes, vamos utilizar um *Data Module* que deverá ser criado através do menu *File|New>Data Module*. Altere seu nome para "DM" e salve a *Unit* como "uDM.pas". Adicione um componente *SQLConnection*("sqlConexao") da paleta *dbExpress*.

O *sqlConexao* é o responsável pela conexão com o arquivo do banco de dados, em nosso caso será o arquivo *Employee.fdb*, normalmente situado em *C:\Arquivos de Programas\Firebird\Firebird(versão)\examples*. Para configurar os parâmetros de acesso ao banco de dados, com um duplo clique sobre o componente *sqlConexao*, crie uma nova conexão utilizando o ícone *Add Connection* e defina o *Driver Name* para *Interbase* e *Connection Name* para "Employee" (**Figura 2**). Con-

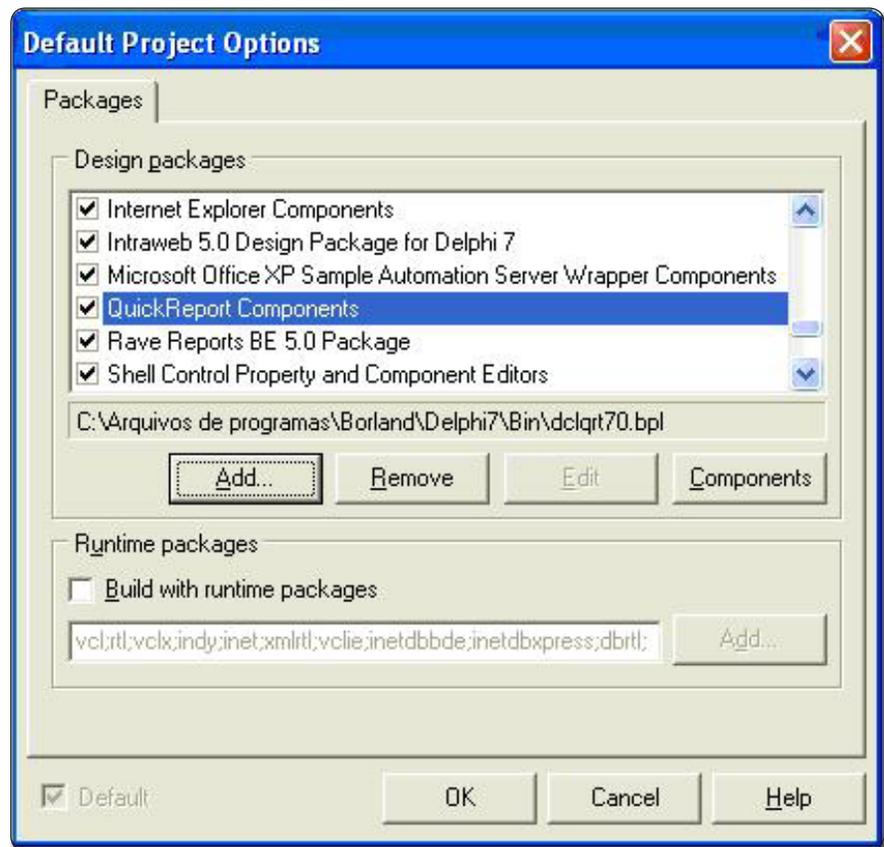


Figura 1. Janela de instalação de pacotes

firme e prossiga para a configuração dos demais parâmetros, começando pela propriedade *Database* onde deverá ser informado o caminho da base de dados *Employee.fdb*.

Defina o usuário do Firebird (SYSDBA) e a senha (*masterkey*) para acesso ao BD. Na propriedade *ServerCharSet* utilizaremos o padrão *Win1252* e em *SQLDialect* utilize o padrão *3* para que possamos utilizar comandos de *SQL* mais complexos (**Figura 3**).

Ainda no componente *sqlConexao*, defina sua propriedade *LoginPrompt* para *False* e altere a propriedade *Connected* para *True* testando assim o sucesso da conexão.

Adicione também no *Data Module* um componente *SQLDataSet*("sqlPesquisa"), relacione a propriedade *SQLConnection* ao componente *sqlConexao* e digite a instrução *SQL* da **Listagem 1** em sua propriedade *CommandText*.

O comando de *SQL* que adicionamos irá trazer como resultado todos os registros de empregados contidos na tabela *EMPLOYEE* e relacionando-os a



Figura 2. Criando uma nova conexão



seus respectivos departamentos através de um *Inner Join* com a tabela *DEPARTAMENT*. O objetivo de selecionar os empregados com seus respectivos departamentos é criar logo mais um relatório onde serão criados grupos de empregados de acordo com o departamento onde cada um atua.

Adicione ainda ao *Data Module* um componente *DataSetProvider*("dspPesquisa") da paleta *Data Access* relacionando sua

propriedade *DataSet* para *sqlPesquisa*. Adicione ainda da paleta *Data Access* um componente *ClientDataSet*("cdsPesquisa") e configure a ligação da propriedade *ProviderName* ao *dspPesquisa*. Com um duplo clique sobre o *cdsPesquisa* abra o *Fields Editor* e adicione todos os campos da tabela através do clique de direita e *Add all fields*.

Retornando ao formulário principal, adicione três componentes *Button* da

paleta *Standard* alterando sua propriedade *Caption* para "Geral", "Agrupado" e "Filtrar", respectivamente. Adicione ainda um componente *Edit*("edtFiltro") e exclua o valor em sua propriedade *Text*. Adicione também um *Label* e altere seu *Caption* para "Pesquisa".

Adicione um componente *DataSource* ("dsEmpregados") relacionando sua propriedade *DataSet* ao *cdsPesquisa* configurado no *DM*. Adicione um componente *DBGrid*("gridEmpregados") e ligue a propriedade *DataSource* ao *dsEmpregados*. Organizando os componentes no formulário de forma semelhante à **Figura 4**.

Adicione ao evento *OnShow* do *frmPrincipal* a seguinte linha de comando que fará a abertura do *ClientDataSet* de forma a exibir todos os registros contidos na base de dados:

```
dsEmpregados.DataSet.Open;
```

Adicione também ao evento *OnClose* do *frmPrincipal* o seguinte código que irá fechar a conexão de pesquisa com a tabela da base de dados:

```
dsEmpregados.DataSet.Close;
```

### Criando os relatórios

A criação de relatórios através do *QuickReport* poderá ser feita por meio de duas seqüências dos processos, uma onde criamos o formulário e adicionamos todos os componentes do *QR* manualmente e outra onde criamos um novo *Report* que traz uma estrutura de relatório previamente configurada. Veremos as duas maneiras começando pela criação de um relatório geral de amostragem dos registros. O desenvolvimento será totalmente manual.

Crie, através do menu *File\New>Form*, um novo formulário("frmReportGeral") alterando o *Caption* para "Relatório Geral de Empregados" e sua propriedade *KeyPreview* para *True* e salve sua *Unit* como "uReportGeral.pas". Adicione ao evento *OnKeyDown* do *frmReportGeral* o seguinte código com a função de fechar a tela de exibição ao pressionarmos a tecla *ESC*.

```
if Key = VK_ESCAPE then  
Close;
```

Insira um componente *QuickRep* ("QRelatorio") da paleta *QReport*. Note que na área em branco do componente

**Listagem 1.** Instrução SQL do *sqlPesquisa*

```
SELECT  
EMPLOYEE.FULL_NAME, EMPLOYEE.PHONE_EXT,  
EMPLOYEE.JOB_COUNTRY, EMPLOYEE.SALARY, DEPARTMENT.DEPARTMENT  
FROM  
EMPLOYEE  
INNER JOIN DEPARTMENT  
ON(EMPLOYEE.DEPT_NO = DEPARTMENT.DEPT_NO)
```

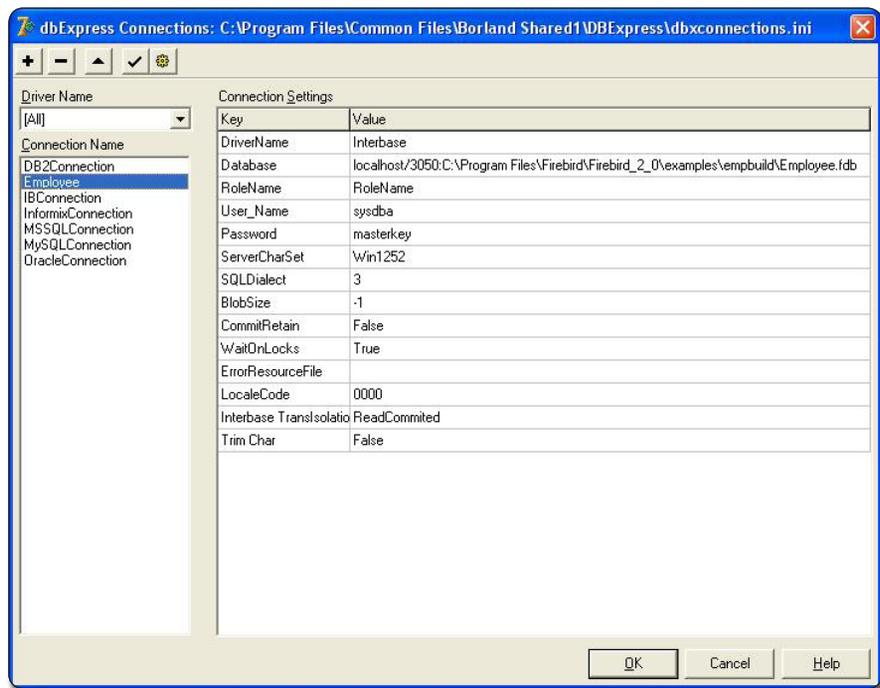


Figura 3. Configurando a conexão

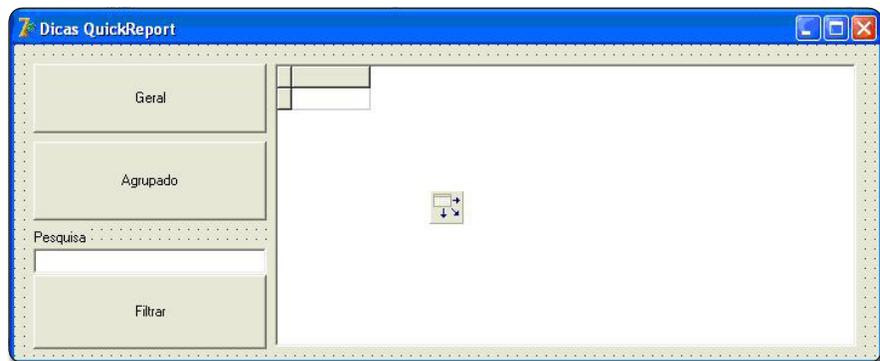


Figura 4. Layout do formulário principal

que acabamos de adicionar existe uma margem em torno de todo o documento, margem esta que delimita o espaço útil do relatório, ou seja, é a área de impressão que podemos utilizar. Por *default* o relatório vem sem nenhuma *banda*. As bandas são áreas no relatório onde incluímos os componentes de visualização dos dados.

Para adicionar bandas no relatório, ou seja, espaços divididos para a exibição de cabeçalho, descrição, resultados, entre outros precisamos adicionar ao nosso formulário componentes *QRBand* e configurá-los de acordo com a função que os mesmos deverão assumir.

Adicionando um componente *QRBand* ("bdCabeçalho"), que por *Default* é adicionado como *rbTitle* em *BandType* observe que o mesmo já tomou posição ao topo do relatório, onde deverá permanecer e será configurado para que este seja o cabeçalho do relatório. Altere o *BandType* para *rbPageHeader*. Arraste para dentro do *bdCabeçalho* um componente *QRLabel* alterando seu *Caption* para "Relatório Geral de Empregados" e definindo um tamanho de fonte diferenciado para que assuma as características de título do relatório.

Para configurar os itens de descrição dos campos que serão exibidos no relatório adicione novamente ao formulário um componente *QRBand* ("bdTitulos"), onde deverão ser adicionados quatro componentes *QRLabel* com as seguintes descrições na propriedade *Caption*: "Empregado", "Telefone", "Salário" e "Departamento". Configure a altura da *bdTitulos* para que o tamanho da linha fique ajustado de forma correta em relação ao tamanho dos títulos de colunas e altere sua propriedade *BandType* para *rbTitle*.

Para dar mais destaque à linha de descrição de cada coluna, selecione a *bdTitulos* e altere sua propriedade *Color* para *clSilver*. Da mesma forma altere também a cor de fundo dos componentes *QRLabel* também para *clSilver*. O próximo passo na criação do relatório será adicionar uma nova *QRBand* ("bdCampos") onde deverão ser exibidos os registros através do componente de texto *QRDBText* da paleta *QReport*.

Seguindo o mesmo esquema do cabeçalho das colunas, adicione na *bdCampos*, imediatamente abaixo os campos do ca-

beçalho, quatro componentes *QRDBText*, observando uma configuração individual para cada um deles que deverá ser de atribuir sua propriedade *DataSet* ao *cdsPesquisa* configurado no *Data Module* e também de configurar qual será o campo ao qual o *QRDBText* irá se relacionar, alterando sua propriedade *DataField* ao campo respectivo com a descrição do texto no cabeçalho.

---

**Nota:** Não se esqueça de acrescentar a *Unit* do *Data Module* ao *Uses* do formulário de relatório. Use a opção *File>Use unit* ou *Alt + F11*

---

Ainda antes de executar seu relatório, selecione o componente *QuickRep* ("QRelatorio") e ligue a propriedade *DataSet* ao componentes *cdsPesquisa* também configurado no *Data Module* da nossa aplicação.

Depois da organização dos componentes, seu relatório deverá ficar semelhante ao relatório ilustrado na **Figura 5**. Para executar o relatório, clique com o botão direito do mouse sobre o relatório e selecione a opção *Preview*, que irá exibir as informações contidas no *ClientDataSet* configurado. Tome o cuidado de ativar o *cdsPesquisa* antes da visualização dos registros.

Para executar a tela do relatório em *Runtime*, retorne ao *frmPrincipal* onde deverá atribuir a seguinte linha de código ao evento *OnClick* do botão *Geral*.

```
Application.CreateForm(TfrmReportGeral, frmReportGeral);
DM.cdsPesquisa.Open;
frmReportGeral.QRelatorio.Preview;
```

O código cria o formulário, abre o *ClientDataSet* que contém os dados e por fim exibe o relatório. Vá também até o menu *Project|Options* do Delphi e no configurador de opções que irá aparecer localize a aba *Forms*. Veja que existem, relacionados na seção *Auto-Create forms*, o *Data Module* e também os dois formulários que criamos. Arraste o *frmReportGeral* para a seção *Available forms* e também arraste o *DM* para que seja criado antes que o formulário principal do sistema (**Figura 6**).

## Aplicando filtros ao relatório geral de empregados

A opção de aplicar filtros ao relatório nada mais é do que uma simples configuração que faremos da propriedade *Filter* do *cdsPesquisa*, mas preferi abordar este detalhe aqui no artigo para demonstrar diferentes situações possíveis de customização e exibição do seu relatório.

Posicionado no *frmPrincipal*, adicione ao evento *OnClick* do botão *Filtrar* o código da **Listagem 2** que fará uma varredura em todos os registros do campo *FULL\_NAME* procurando pelo valor digitado no espaço da pesquisa.

Ao executar sua aplicação, digite um valor no campo de pesquisa e aplique o botão *Filtrar* para visualizar os resultados.



**Figura 5.** Layout do relatório geral



**Listagem 2.** Aplicando filtros aos registros exibidos no relatório

```

procedure TfrmPrincipal.Button3Click(Sender: TObject);
begin
  Application.CreateForm(TfrmReportGeral, frmReportGeral);
  DM.cdsPesquisa.Open;
  DM.cdsPesquisa.Filtered := False;
  DM.cdsPesquisa.Filter := 'UPPER(FULL_NAME) LIKE '+
    QuotedStr('%'+UpperCase(edtFiltro.Text)+'%');
  DM.cdsPesquisa.Filtered := True;
  frmReportGeral.QRelatorio.Preview;
end;

```

**Listagem 3.** Instrução SQL para impressão de relatórios por grupo

```

SELECT
  EMPLOYEE.FULL_NAME, EMPLOYEE.PHONE_EXT,
  EMPLOYEE.JOB_COUNTRY, EMPLOYEE.SALARY,
  DEPARTMENT.DEPARTMENT
FROM
  EMPLOYEE
INNER JOIN DEPARTMENT
ON (EMPLOYEE.DEPT_NO = DEPARTMENT.DEPT_NO)

```

Ao pesquisar, por exemplo, pelo valor “be” serão listados somente alguns registros que contenham estas siglas no nome. Uma opção interessante que podemos usar na hora de exibir o relatório, já que a estrutura de componentes utilizados para ambos os exemplos anteriores é a mesma, será utilizar um componente *RadioGroup* e definir as opções para que o usuário escolha entre um relatório completo (“sem filtros”) ou um relatório customizado, onde poderá antes filtrar pelos registros que realmente deseja que sejam impressos.

### Criando relatório com grupos de registros

Nesta seção do artigo veremos uma opção interessante para criação de um relatório, utilizaremos basicamente as informações semelhantes ao exemplo anterior na qual agora iremos organizar os registros por grupos, ou seja, iremos criar uma lista de departamentos e logo abaixo de cada departamento uma listagem de todos os empregados.

Selecione *File|New|Other>New>Report*. Esta é uma outra forma de se criar relatórios no Delphi utilizando o Quick Report. Salve a *Unit* do formulário como “uGroupReport.pas”. Adicione, da paleta *QReport*, um componente *QRGroup* ao relatório e adicione também um componente *QRBand* e altere seu *BandType* para *rbDetail*. Para trabalhar com os resultados no relatório, adicione ao formulário os seguintes componentes realizando as devidas configurações a seguir:

- *SQLDataSet* (“sqlGroup”) da paleta *dbExpress*, utilize a *Unit* do *Data Module* e ligue sua propriedade *SQLConnection* ao *sqlConexao* previamente configurado. Adicione em *CommandText* a instrução de SQL da **Listagem 3**;
- *DataSetProvider* (“dspGroup”) da paleta *Data Access* e ligue-o através da propriedade *DataSet* ao *sqlGroup*;
- *ClientDataSet* (“cdsGroup”) da paleta *Data Access* configurando seu *ProviderName* para *dspGroup*. Com um duplo clique sobre o componente abra o *Fields Editor* e adicione todos os campos resultantes da pesquisa SQL.

Arraste agora para dentro do componente *QRGroup* um componente *QRDBText* e ligue sua propriedade *DataSet* ao *cdsGroup*

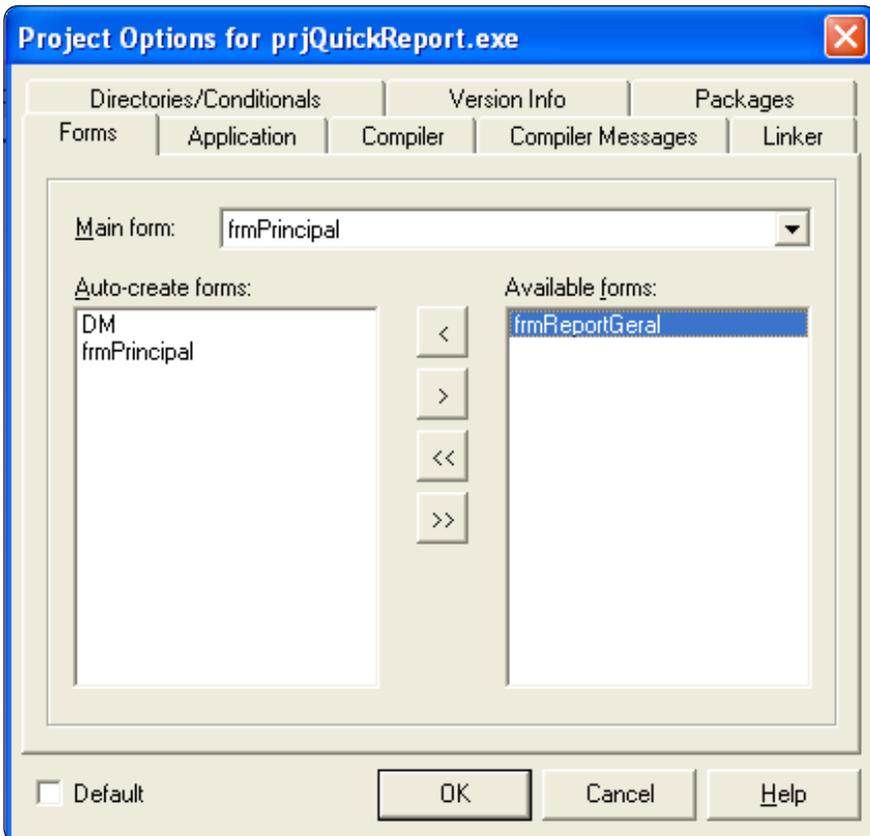


Figura 6. Opções do projeto

e *DataField* para DEPARTMENT. Selecione o componente *cdsGroup* para que possamos fazer uma configuração indispensável antes de continuar com a criação do relatório agrupado. Precisamos definir o campo pelo qual iremos criar um índice para formação dos grupos, onde na propriedade *IndexFieldNames* você deverá digitar o valor DEPARTMENT, que será o campo que utilizaremos para a formação dos grupos.

Retornando ao formulário de criação do relatório, arraste agora para dentro do *QRBand* mais três componentes *QRDBText*, relacionando a propriedade *DataSet* de ambos ao *cdsGroup* e o *DataField* respectivamente para *FULL\_NAME*, *PHONE\_EXT* e *SALARY*. Selecione novamente o *QRGroup* e em sua propriedade *Expression* digite o campo pelo qual nosso relatório será agrupado, informando-lhe o seguinte valor:

```
ClientDataSet1.DEPARTMENT
```

Antes do *Preview* selecione ainda o objeto *QuickReport1* de modo que sua propriedade *DataSet* seja configurada ao *cdsGroup*. Realizadas todas as configurações acima descritas, utilize o botão direito do mouse sobre o relatório e selecione a opção *Preview* que deverá originar um relatório semelhante ao exibido na **Figura 7**.

Retornando ao formulário principal para a codificação do *Button* "Agrupado", adicione ao evento *OnClick* do mesmo o seguinte bloco de códigos:

```
Application.CreateForm(TQuickReport1,QuickReport1);
QuickReport1.cdsGroup.Open;
QuickReport1.Preview;
```

**Nota:** Acesse o menu *Project|Options* e na guia *Forms* remova o objeto *Quick Report1* da seção *Auto Create Forms* enviando-o para *Available Forms*.

## Conclusão

Durante este artigo vimos algumas dicas interessantes de como criar rapidamente um relatório utilizando compo-

nentes de acesso a dados e *QuickReport* de forma simples e também com a aplicação de filtros ao relatório.

Criamos também um relatório com a formação de grupos de registros, que facilita muito na análise das informações de um relatório. Dentre estas opções, descubra várias outras funcionalidades do *QuickReport*, como personalização de campos em *runtime* ou até mesmo a criação de um relatório *master/detail*. Grande abraço e até a próxima. ●

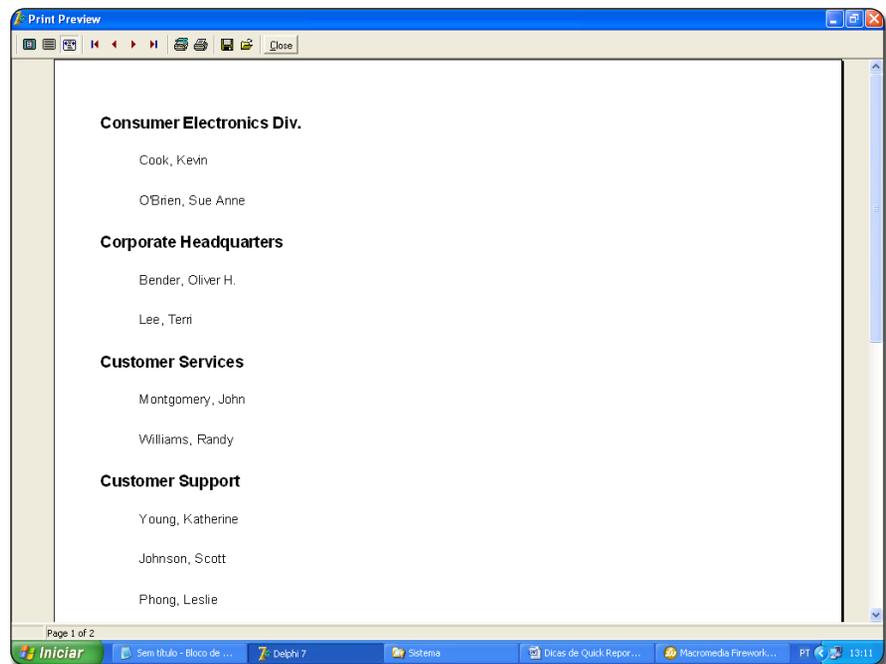


Figura 7. Preview do relatório agrupado por departamentos

