

# PROGRAMAÇÃO EM DELPHI 2009

## PROCEDIMENTOS E FUNÇÕES

Alterando o tipo de gráfico em tempo de execução – nível básico

Deve se ter os seguintes objetos:

- Tchar
- ComboBox

No evento OnChange da combo incluir o seguinte código

```
var i :integer;
    vSeries :TChartSeries;
begin
    if ComboBox1.Text = 'LINHAS' then
        BEGIN
            for i := 0 to Chart1.SeriesCount -1 do
                begin
                    vSeries := Chart1.Series[i];
                    ChangeSeriesType(vSeries, TPieSeries);
                end;
            END else if ComboBox1.Text = 'BARRAS' then
                begin
                    for i := 0 to Chart1.SeriesCount -1 do
                        begin
                            vSeries := Chart1.Series[i];
                            ChangeSeriesType(vSeries, TBarSeries);
                        end;
                    end;
                end;
            end;
```

Inclusa em 08/10/2010 – Marcos Calixto

Montando um Gráfico em Delphi

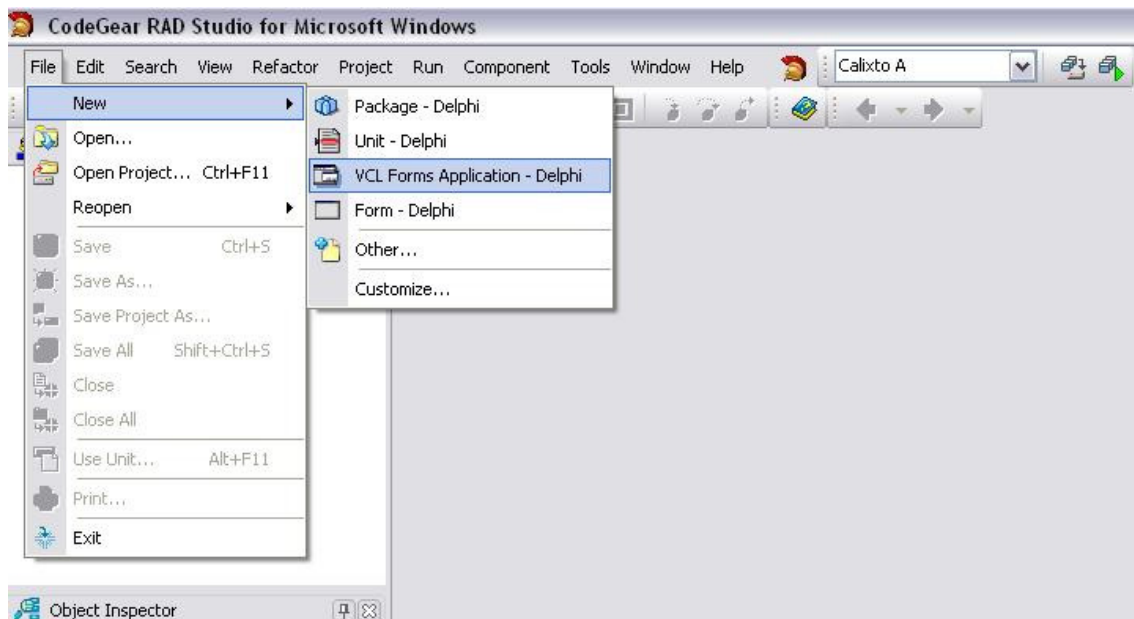
Esse artigo tem objetivo de iniciar a manipulação de informações de uma tabela para ser visualizada em um gráfico no Delphi. Estaremos utilizando o POSTGRES como banco de dados e os objetos ADO como acesso ao banco.

PRÉ-REQUISITOS:

- Instalado o Delphi 2009

- Instalado o Postgres 8.4.
- Drive ODBC ([http://www.postgresql.org/ftp/odbc/versions/msi/psqlodbc\\_08\\_04\\_0200.zip](http://www.postgresql.org/ftp/odbc/versions/msi/psqlodbc_08_04_0200.zip))
- Conhecimentos básicos em Banco de Dados Postgres.

1 – Devemos iniciar um novo projeto.



Agora vamos salvar.

Unit: UExemplo\_Char

Project: Exemplo\_Char

2 – Vamos formatar nosso form1 (principal)

Propriedade	Valor
Height	800
width	1024
Caption	Exemplo Char

3 – Vamos incluir nossos objetos de ação:

- Não Visuais
  - ADOConnection
  - ADOQuery

- DataSource
- Visuais
  - Panel
  - Chart

## 4 – Vamos Configurar nossa Estrutura de Banco de Dados

### 4.1 Criação e Inserção de dados em nossa tabela:

Tabela: grafico\_modelo

Estrutura:

```
CREATE TABLE "public"."grafico_modelo" (
  "grafmoid" SERIAL,
  "grafmvalor" INTEGER,
  "grafmtipo" VARCHAR(2),
  "grafmdata" DATE,
  "grafmpendente" NUMERIC,
  "grafmfaturado" NUMERIC,
  "grafmtotal" NUMERIC,
  CONSTRAINT "grafico_modelo_pkey" PRIMARY KEY("grafmoid")
) WITH OIDS;
```

Inclusão:

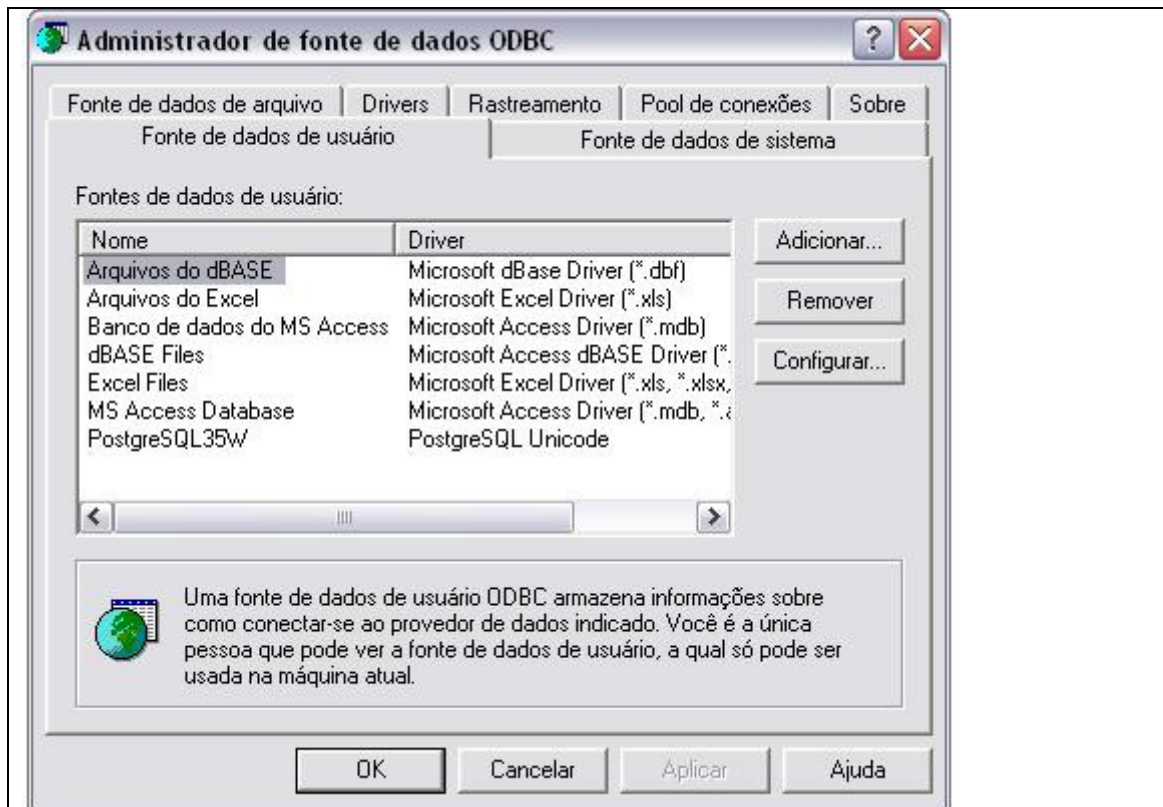
```
INSERT INTO
public.grafico_modelo(
grafmdata,
grafmpendente,
grafmfaturado,
grafmtotal) VALUES...
```

Deixe com os seguintes valores

grafmdata	grafmpendente	grafmfaturado	grafmtotal
1/10/2010	59	0	59
2/10/2010	57	3	60
3/10/2010	53	7	63
4/10/2010	44	24	68
5/10/2010	56	33	89
6/10/2010	89	35	125

### 4.2 Criando uma fonte de Dados ODBC

No Windows criaremos em: PAINEL DE CONTROLE – FERRAMENTAS ADMINISTRATIVOS – FONTES DE DADOS ODBC, você deverá chegar em:



Click em Adicionar e configurar

Selecione o Drive: Postgres Unicode

Data Source: Exemplo Chart

Data base: O nome do banco de dados onde você criou a tabela  
[gráfico\_modelo]

Server: localhost

User Name: nome do usuário do Banco de Dados.

Password: Senha do usuário do banco de Dados.

OK, agora já temos objeto ODBC criado e já podemos conectar o Delphi com o Banco.

#### 4.3 Conectando o Delphi ao Postgres com ADO.

Para isso iremos nos utilizar dos componentes ADO.

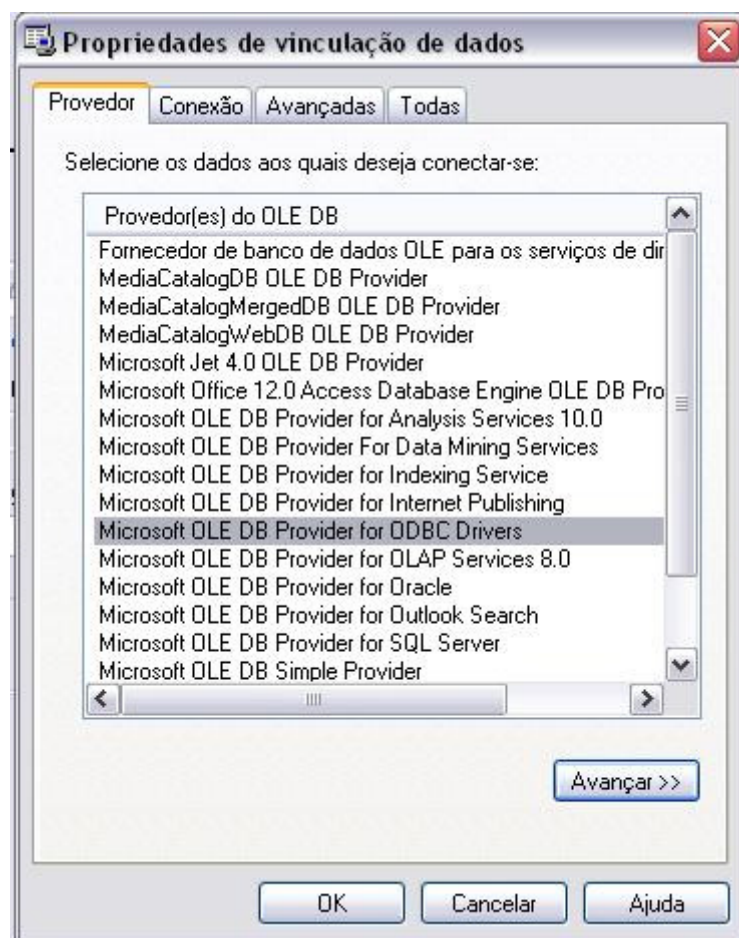
ADOConnection: Objeto responsável pela abstração da camada Modelo para integração da IDE com BD. Vou dividir em passo aqui:

Passo 1: Acesse a propriedade [ConnectionString [...]]

Passo 2: Será exibida uma caixa de dialogo[Contrutora], click em [build].



Passo 3: será exibida uma caixa de dialogo[Propriedade], na Aba[provedor]: selecionar [Microsoft OLE DB provider for ODBC Driver]



Após selecionado click em avançar.

Passo 4: você será direcionado para Aba[Conexão], no campo [Especifique a fonte de dados], selecione a que você criou.

Nos campos Nome do Usuário e Senha, inclua as mesma que foram inclusas na criação do ODBC. Ficará conforme a figura abaixo:



Após tudo feito Teste sua conexão, caso não obtenha sucesso, refaça os passos do item 4.

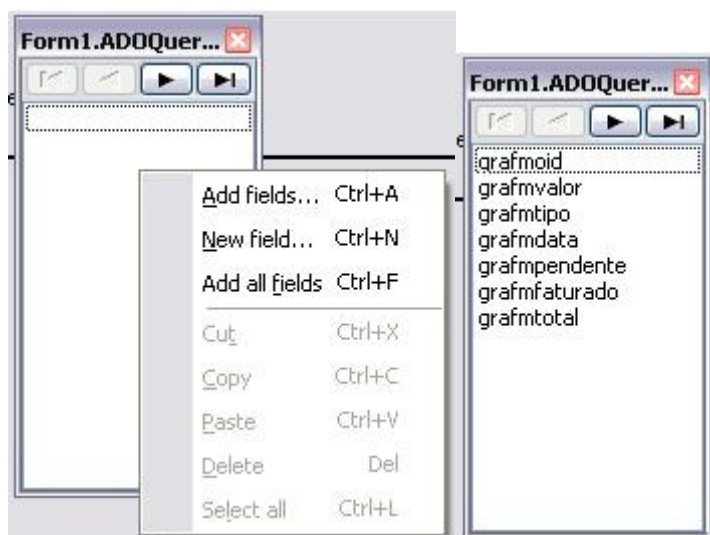
#### 4.4 – Configurando o Objeto ADOQuery

Esse carinha é muito simples de configura

PROPRIEDADE	VALOR
Connection	ADOConnection
SQL	SELECT * FROM

	gráfico_modelo
Active	True

De um duplo Click no objeto, será exibido o Editor de Campos, Com o botão direito do Mouse selecione o item [Add All fields] ou Ctrl+F.



#### 4.5 Configurando o Objeto DataSource

Esse objeto é o responsável em fazer a “Ponte” entre objetos dentro da programação, assim evita-se ter interação direta dos objetos com Banco de Dados respeitando cada um sua camada de aplicação.

PROPRIEDADE	VALOR
DataSet	ADOQuery

Ok aqui fechamos configuração dos objetos com o banco de dados.

#### 5. Fazendo o Tchar trabalhar.

Iremos colocar nossos códigos no Evento OnShow do Form principal, isso quer dizer que quando o sistema abrir já irá exibir o gráfico na sua interface inicial.

Objeto: form

## Evento: OnShow

### Codificação:

```
procedure TForm1.FormShow(Sender: TObject);
var
    sql : string;
    i : integer;
begin
    i := 1;
    {Bloco de Inicialização do componente ADOQuery1}
    ADOQuery1.Close;
    ADOQuery1.SQL.Clear;
    sql := 'SELECT * FROM grafico_modelo';
    ADOQuery1.SQL.Add(sql);
    ADOQuery1.Open;

    {Bloco onde será retornados os valores de interação do grafico}
    while not (ADOQuery1.Eof) do
    {O que esse While faz? Percorre todos os registro do objeto enquanto ele não
    ele não encontrar o ponteiro EOF que indica "Fim do Arquivo" de todos os retornos}
    begin

        Chart1.Series[1].AddXY(i,ADOQuery1.grafmfaturado.AsFloat,FormatDateTime('dd/mm/yyyy',
        ADOQuery1.grafmdata.AsDateTime),clYellow);

        Chart1.Series[0].AddXY(i,ADOQuery1.grafmpendente.AsFloat,FormatDateTime('dd/mm/yyyy'
        ,ADOQuery1.grafmdata.AsDateTime),clRed);

        Chart1.Series[2].AddXY(i,ADOQuery1.grafmtotal.AsFloat,FormatDateTime('dd/mm/yyyy',ADO
        Query1.grafmdata.AsDateTime),clGreen);
        Inc(i,1);
        ADOQuery1.Next; // aponta para o proximo registro
    end;
    Chart1.Series[3].Clear;
    Chart1.Series[3].AddXY(1,0,'31/10/2010',clBlue);
    Chart1.Series[3].AddXY(23,240,'31/10/2010',clBlue);
end;
```

## 5.1 AJUSTANDO O TCHAR PARA ENTENDER O CÓDIGO

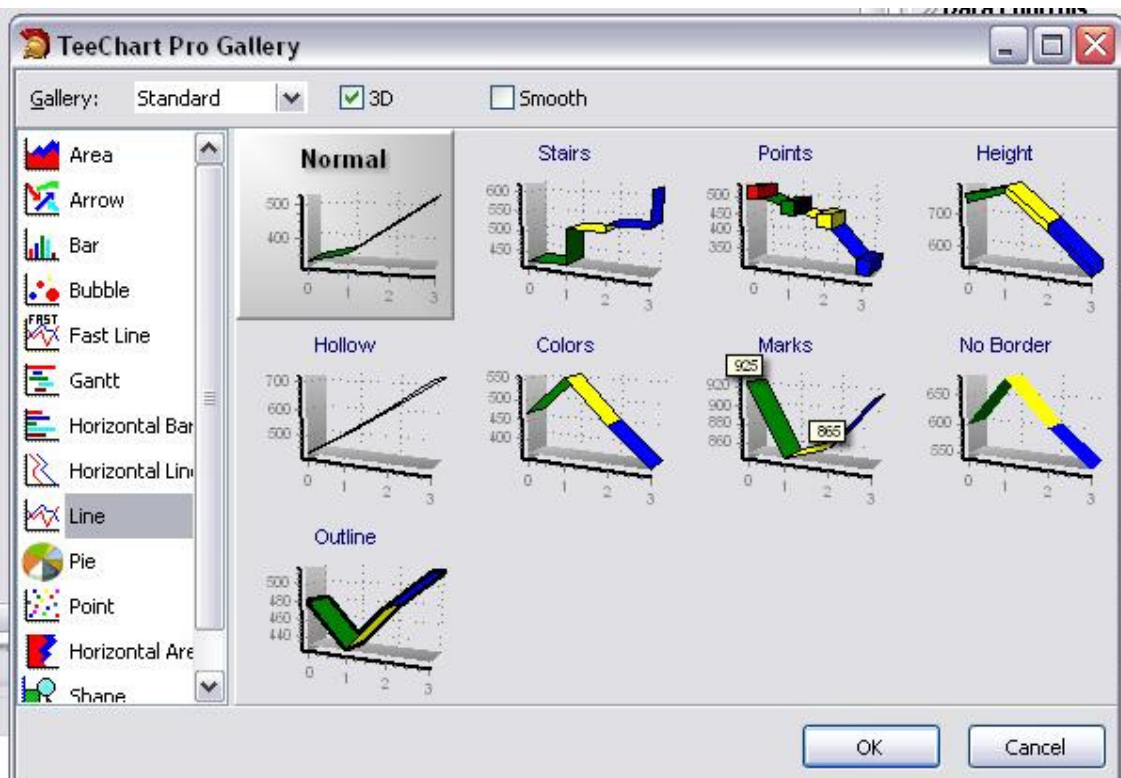
Se nesse exato momento você tentar executar irá ter o erros de acesso em memória, pois a propriedade Tseries, é uma alocação de memória física para programação, então devemos incluir as serie na Tchar para que possamos processar o código. Utilizei aqui engenharia reversa para que todos possa entender e visualizar alguns erros e saber o que são.

### 5.1.1 INCLUINDO AS SERIES NO TCHAR

Vou dividir aqui em passos:



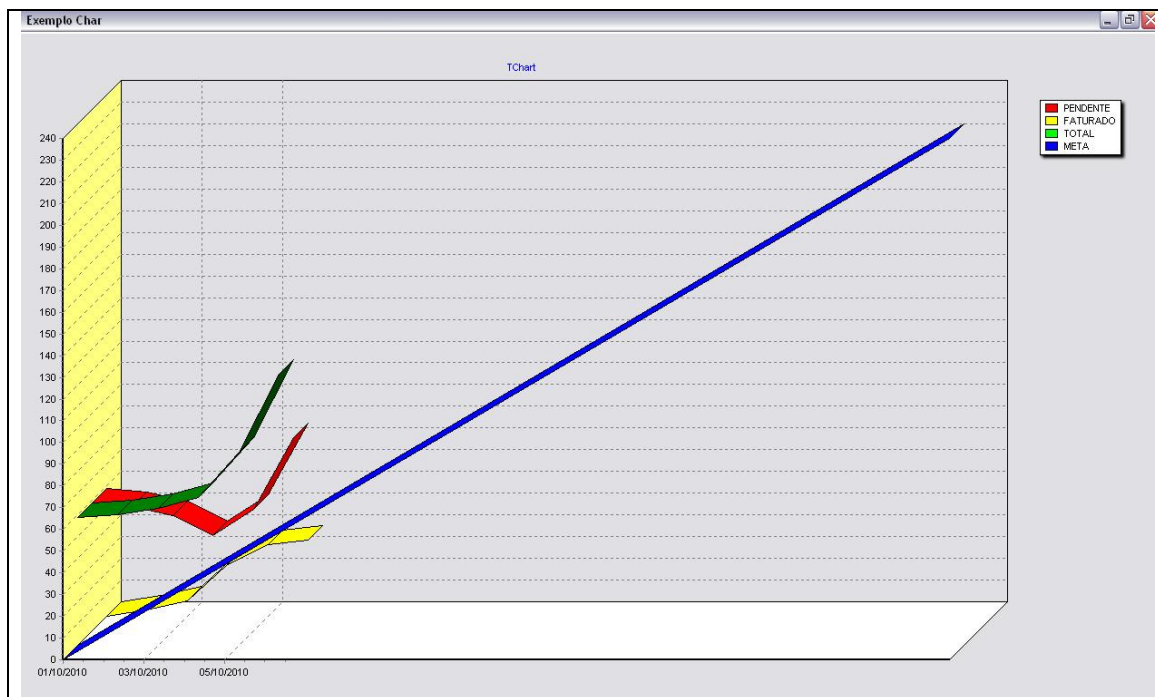
- 1) De um duplo click no objeto TChar.
- 2) Será exibido o editor, no Item[Series] click em [Add...]



Seleciono o Gráfico [Line -> Normal]

- 3) Após feito isso iremos altera o titulo dessa serie, Click em Title e inclua [PENDENTE], no icone de cor selecione a cor Vermelha.
- 4) Agora irmos clonar essa serie, alterando somente o titulo e a cor:
  - a. [FATURADO] [AMARELO]
  - b. [TOTAL] [VERDE]
  - c. [META] [AZUL]

Agora é só executar que você terá a seguinte resultante:



Agora é só brincar com o Tchart.

Abraço.