

**unit** UCadCliente;

**interface**

**uses**

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
Dialogs, StdCtrls, WideStrings, DBXMsSQL, DB, SqlExpr, FMTBcd, ComCtrls,  
ToolWin, ExtCtrls, DBCtrls, uConexaoBanco, uControle, uClienteControle, Grids,  
DBGrids;

**type**

```
TFormCadCliente = class(TForm)
  LbledtCodigo: TLabeledEdit;
  LbledtNome: TLabeledEdit;
  LbledtTelefone: TLabeledEdit;
  BtnIncluir: TButton;
  BtnAlterar: TButton;
  BtnConsultar: TButton;
  BtnExcluir: TButton;
  BtnEncerrar: TButton;
  StrGrdCliente: TStringGrid;
  stabarprincipal: TStatusBar;
procedure BtnIncluirClick(Sender: TObject);
procedure LbledtCodigoEnter(Sender: TObject);
procedure LbledtCodigoExit(Sender: TObject);
procedure LbledtNomeEnter(Sender: TObject);
procedure LbledtNomeExit(Sender: TObject);
procedure LbledtTelefoneEnter(Sender: TObject);
procedure LbledtTelefoneExit(Sender: TObject);
procedure BtnEncerrarClick(Sender: TObject);
procedure FormKeyUp(Sender: TObject; var Key: Word; Shift: TShiftState);
procedure FormShow(Sender: TObject);
procedure BtnExcluirClick(Sender: TObject);
procedure BtnConsultarClick(Sender: TObject);
procedure BtnAlterarClick(Sender: TObject);
private
  { Private declarations }
procedure ganha_foco(Sender : TObject);
procedure perde_foco(Sender : TObject);
procedure view_cliente;
procedure ativar_classes;
public
  { Public declarations }
  Conexao : TConexaoBanco;
  Controle : TControle;
  Cliente : TClienteControle;
```

//

**end;**

**var**

```
FrmCadCliente: TFormCadCliente;
Indexador      : Integer;
IndArray       : Integer;
```

**implementation**

{ \$R \*.dfm }

**procedure** limpa\_campo;  
**begin**

```
  Indexador := 0;
while Indexador < FrmCadCliente.ComponentCount do begin
  if (FrmCadCliente.Components[Indexador] is TLabeledEdit) then begin
    (FrmCadCliente.Components[Indexador] as TLabeledEdit).Clear;
  end;
  Indexador := Indexador + 1;
end;
```

**end;**

**procedure** TFormCadCliente.ativar\_classes;  
**begin**

```

Conexao := TConexaoBanco.Create;
Controle := TControle.Create;
Cliente := TClienteControle.Create(controle);
end;

procedure TFrmCadCliente.BtnAlterarClick(Sender: TObject);
begin
    if Conexao.ConexaoBanco.Connected = False then begin
        stabarprincipal.Panels[0].Text := 'Banco de Dados NÃO CONECTADO';
        exit;
    end;

    cliente.Codigo := LbledtCodigo.Text;
    cliente.Nome := LbledtNome.Text;
    cliente.Telefone := LbledtTelefone.Text;
    if cliente.AlterarCliente = True then begin
        limpa_campo;
        showmessage('Dados do Cliente Alterado com Sucesso');
    end else
    begin
        showmessage('Problema na Alteração dos dados do Cliente');
    end;

    view_cliente;

    // Conexao.Destroy;
    // Controle.Destroy;
    // Cliente.Destroy;

end;

procedure TFrmCadCliente.BtnConsultarClick(Sender: TObject);
begin
    if Conexao.ConexaoBanco.Connected = False then begin
        stabarprincipal.Panels[0].Text := 'Banco de Dados NÃO CONECTADO';
        exit;
    end;

    if Conexao.ConexaoBanco.Connected = False then begin
        stabarprincipal.Panels[0].Text := 'Banco de Dados NÃO CONECTADO';
        exit;
    end;

    if Trim(LbledtCodigo.Text) = '' then begin
        application.MessageBox('Informe o Código do Cliente', 'Consistência');
        exit;
    end;

    cliente.Codigo := LbledtCodigo.Text;
    if cliente.PesquisaCliente = True then begin
        LbledtNome.Text := cliente.Nome;
        LbledtTelefone.Text := cliente.Telefone;
    end else
    begin
        showmessage('Cliente Não Localizado');
    end;

    // Conexao.Destroy;
    // Controle.Destroy;
    // Cliente.Destroy;

end;

procedure TFrmCadCliente.BtnEncerrarClick(Sender: TObject);
begin
    Conexao.Destroy;
    Controle.Destroy;
    Cliente.Destroy;

```

```

close;
end;

procedure TFrmCadCliente.BtnExcluirClick(Sender: TObject);
begin

    if MessageDlg('Tem Certeza que Deseja Excluir o Cliente',mtconfirmation,[mbyes,mbno],0) = mrNo then
        exit;
    end;

    if Conexao.ConexaoBanco.Connected = False then begin
        stabarprincipal.Panels[0].Text := 'Banco de Dados NÃO CONECTADO';
        exit;
    end;

    cliente.Codigo := LbledtCodigo.Text;
    if cliente.ExcluiCliente = True then begin
        limpa_campo;
        showmessage('Cliente Excluído com Sucesso');
    end else
    begin
        showmessage('Problema na Exclusão do Cliente');
    end;

    view_cliente;

    // Conexao.Destroy;
    // Controle.Destroy;
    // Cliente.Destroy;

end;

procedure TFrmCadCliente.BtnIncluirClick(Sender: TObject);
begin

    if Conexao.ConexaoBanco.Connected = False then begin
        stabarprincipal.Panels[0].Text := 'Banco de Dados NÃO CONECTADO';
        exit;
    end;

    cliente.Codigo := LbledtCodigo.Text;
    cliente.Nome := LbledtNome.Text;
    cliente.Telefone := LbledtTelefone.Text;
    if cliente.InsereCliente = True then begin
        limpa_campo;
        showmessage('Cliente Incluído com Sucesso');
    end else
    begin
        showmessage('Código do Cliente já Cadastrado ou Problema na Inclusão do Cliente');
    end;

    view_cliente;

end;

procedure TFrmCadCliente.FormKeyUp(Sender: TObject; var Key: Word; Shift: TShiftState);
begin
    if key = 13 then begin
        perform(cm_dialogkey,vk_tab,0);
    end;
end;

procedure TFrmCadCliente.FormShow(Sender: TObject);
begin

    ativar_classes;

    view_cliente;

    LbledtCodigo.SetFocus;

end;

```

```

procedure TFrmCadCliente.ganha_foco(Sender: TObject);
begin
    if (Sender is TLabelledEdit) then begin
        (Sender as TLabelledEdit).Color := clYellow;
    end;
end;

procedure TFrmCadCliente.LbledtCodigoEnter(Sender: TObject);
begin
    ganha_foco(Sender);
end;

procedure TFrmCadCliente.LbledtCodigoExit(Sender: TObject);
begin
    perde_foco(Sender);
end;

procedure TFrmCadCliente.LbledtNomeEnter(Sender: TObject);
begin
    ganha_foco(Sender);
end;

procedure TFrmCadCliente.LbledtNomeExit(Sender: TObject);
begin
    perde_foco(Sender);
end;

procedure TFrmCadCliente.LbledtTelefoneEnter(Sender: TObject);
begin
    ganha_foco(Sender);
end;

procedure TFrmCadCliente.LbledtTelefoneExit(Sender: TObject);
begin
    perde_foco(Sender);
end;

procedure TFrmCadCliente.perde_foco(Sender: TObject);
begin
    if (Sender is TLabelledEdit) then begin
        (Sender as TLabelledEdit).Color := clWhite;
    end;
end;

procedure TFrmCadCliente.view_cliente;
begin

// limpa a stringgrid da listagem do cliente
with StrGrdCliente do
    for Indexador := 0 to RowCount -1 do
        Rows[Indexador].Clear;

StrGrdCliente.Cells[0,0] := 'Código';
StrGrdCliente.Cells[1,0] := 'Nome';
StrGrdCliente.Cells[2,0] := 'Telefone';

// Conexao := TConexaoBanco.Create;
// Controle := TControle.Create;
// Cliente := TClienteControle.Create(controls);

Indexador := 0;
if cliente.ConsultaCliente = True then begin
    IndArray := 0;
    while Indarray < Length(cliente.aCodigo) do begin
        Indexador := Indexador + 1;
        StrGrdCliente.Cells[0,Indexador] := cliente.aCodigo[IndArray];
        StrGrdCliente.Cells[1,Indexador] := cliente.aNome[IndArray];
        StrGrdCliente.Cells[2,Indexador] := cliente.aTelefone[IndArray];
        IndArray := IndArray + 1;
    end;
end;
end;

```

end.