

**Michel F. Marques - ME**

# **Tutorial**

## **CloseMessages**

**Santos**

**Novembro/2009**

## **SUMÁRIO**

|   |          |
|---|----------|
| <b>INTRODUÇÃO .....</b>                 | <b>1</b> |
| <b>1.   INSTALAÇÃO.....</b>             | <b>2</b> |
| <b>2.   APLICATIVO DE TESTE 1 .....</b> | <b>3</b> |
| <b>3.   APLICATIVO DE TESTE 2 .....</b> | <b>8</b> |

## INTRODUÇÃO

Nos aplicativos as vezes precisamos que sejam fechadas janelas de erro sem a intervenção do usuário, como em um aplicativo de monitoração que fica em execução diretamente sem um usuário a sua frente. Pois um aplicativo sem um que necessita ficar em execução diretamente sem um usuário a sua frente o tempo todo pode ficar travado durante uma mensagem de *exceção* como perda de conexão com o banco, um cálculo que teve erro de divisão por zero, etc.

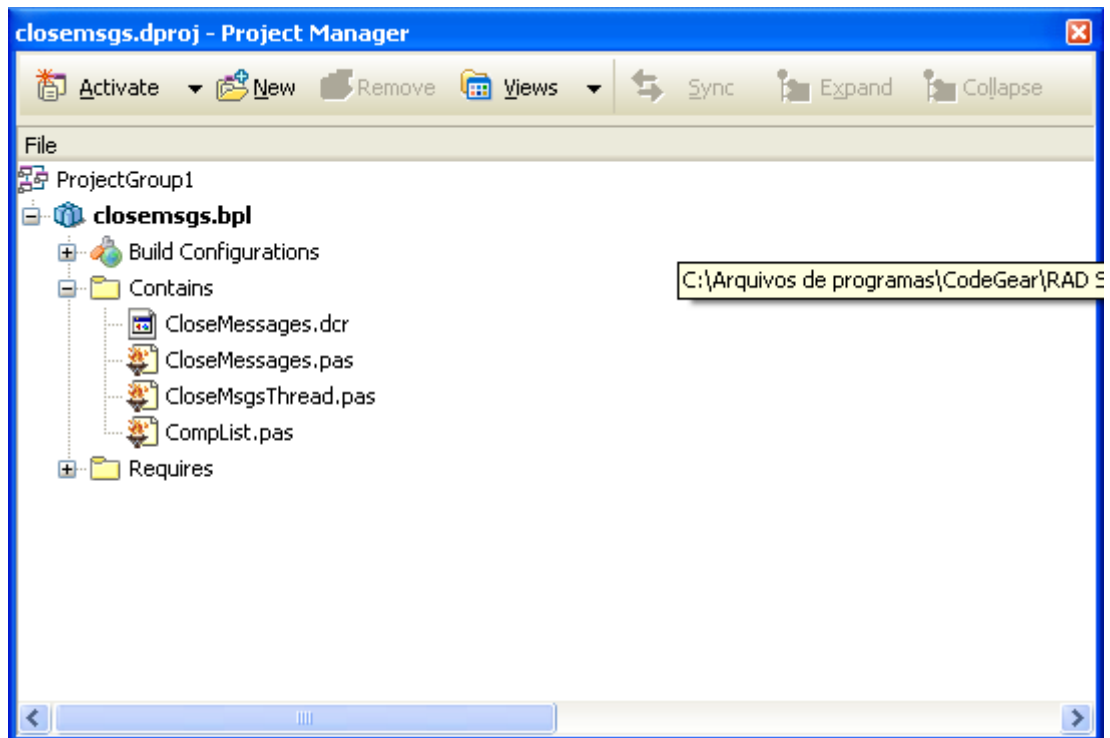
É para estes tipos de caso que CloseMessages foi desenvolvido. Para que possa continuar a execução do seu aplicativo após o fechamento da janela de mensagem e podendo-se acessar as mensagens fechadas.

## 1. INSTALAÇÃO

Para separar bem as coisas, crie uma pasta *Components* na pasta do *Delphi* caso não exista e copie a pasta *CloseMessages* para ela.

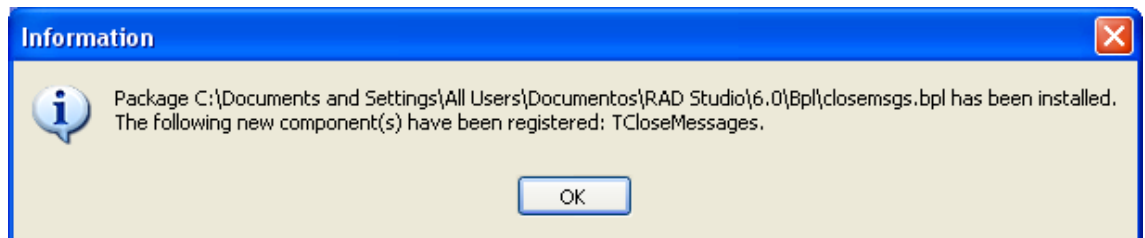
1 - Abra o arquivo *closemsgs.dproj*.


2 – Clique no menu *View >> Project Manager* para exibir a janela abaixo.



3 – Clique com o botão direito do mouse sobre *closemsgs.bpl* e em seguida clique em *Install*.

Se nada der errado deverá aparecer a janela abaixo.



Se você até a paleta de componentes em *Samples* deverá haver um componente *TCloseMessages* com o seguinte ícone .

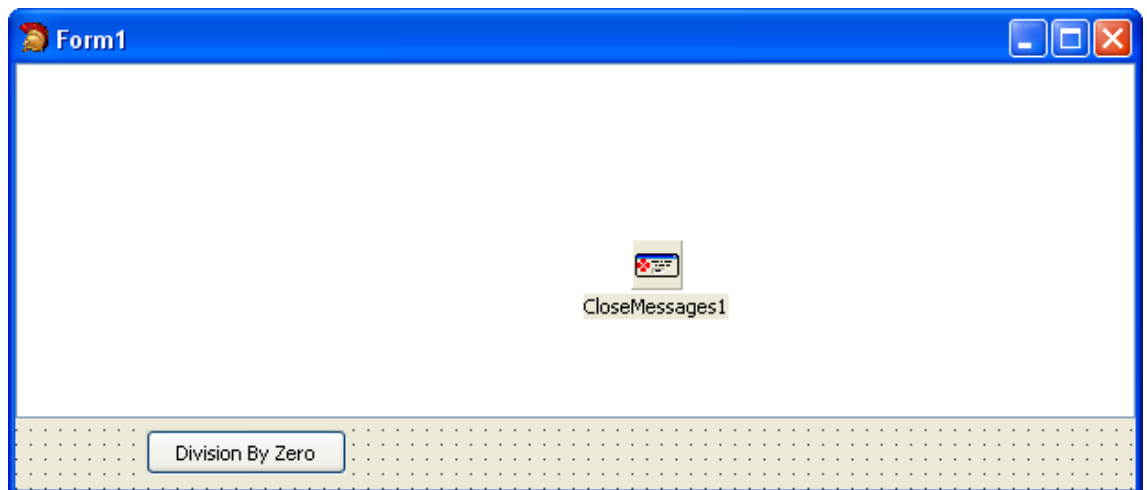
## 2. APLICATIVO DE TESTE 1

Inicie o Delphi e crie um novo aplicativo.

Agora insira os componentes abaixo com suas respectivas configurações:

```
btnDivisaoZero: TButton
    Left = 71
    Top = 200
    Width = 110
    Height = 25
    Caption = 'Division By Zero'
ListBox1: TListBox
    Left = 0
    Top = 0
    Height = 194
    Align = alTop
CloseMessages1: TCloseMessages
    Enabled = True
```

A janela deverá ficar da seguinte forma



Agora selecione o componente *CloseMessages1* e altere sua propriedade *Enabled* para *True*.

Neste mesmo componente alterne para a guia *Events* e no evento *OnAfterCloseMessage* inclua o seguinte código:

```
Listbox1.Items.Add(Msg.AsText);
```

Agora selecione o component *btnDivisaoZero* e no evento *OnClick* inclua o seguinte código:

```
ShowMessage(FloatToStr(1 / Random(0))) ;
```

*Obs: A mensagem acima não será exibida porque será gerada uma exceção de divisão por zero.*

Agora execute a aplicação e clique no botão para gerar o erro e veja que a janela é fechada e a mensagem é listada no ListBox.

### **Código do form:**

```
object Form1: TForm1
    Left = 413
    Top = 257
    Caption = 'Form1'
    ClientHeight = 233
    ClientWidth = 611
    Color = clBtnFace
    Font.Charset = DEFAULT_CHARSET
    Font.Color = clWindowText
    Font.Height = -11
    Font.Name = 'Tahoma'
    Font.Style = []
    OldCreateOrder = False
    PixelsPerInch = 96
    TextHeight = 13

object btnDivisaoZero: TButton
    Left = 71
    Top = 200
    Width = 110
    Height = 25
    Caption = 'Division By Zero'
```

```

        TabOrder = 0
        OnClick = btnDivisaoZeroClick
    end
    object ListBox1: TListBox
        Left = 0
        Top = 0
        Width = 611
        Height = 194
        Align = alTop
        ItemHeight = 13
        TabOrder = 1
    end
    object CloseMessages1: TCloseMessages
        Enabled = True
        OnAfterCloseMessage =
CloseMessages1AfterCloseMessage
        Left = 336
        Top = 96
    end
end
end

```

### Código do Fonte:

```

unit Unit1;

interface

uses

    Windows, Messages, SysUtils, Variants, Classes, Graphics,
Controls, Forms,

    Dialogs, StdCtrls, CloseMessages;

```

## **type**

```
TForm1 = class (TForm)

    CloseMessages1: TCloseMessages;

    btnDivisaoZero: TButton;

    ListBox1: TListBox;

    procedure btnDivisaoZeroClick(Sender: TObject);

    procedure CloseMessages1AfterCloseMessage(Sender:
TObject; FormHandle: HWND;
    Msg: TErrorMessage);

private

    { Private declarations }

public

    { Public declarations }

end;
```

## **var**

```
Form1: TForm1;
```

## **implementation**

```
{ $R *.dfm }
```

```
procedure TForm1.btnDivisaoZeroClick(Sender: TObject);

begin

    ShowMessage(FloatToStr(1 / Random(0)));

end;
```



```
procedure TForm1.CloseMessages1AfterCloseMessage(Sender:
TObject;

    FormHandle: HWND; Msg: TErrorMessage);

begin

    ListBox1.Items.Add(Msg.AsText);

end;

end.
```

### 3. APLICATIVO DE TESTE 2

Utilizando o exemplo anterior, vamos fazer umas alterações.

Inclua os seguintes componentes com suas respectivas propriedades:

Label1: TLabel

Left = 176

Top = 200

Width = 92

Height = 13

Caption = 'Exception Message'

edtException: TEdit

Left = 176

Top = 216

Width = 121

Height = 21

Text = 'My Exception'

btnException: TButton

Left = 303

Top = 215

Width = 75

Height = 25

Caption = 'Exception'

btnEnable: TButton

Left = 440

Top = 215

Width = 75

Height = 25

Caption = 'Enable'

btnDisable: TButton

Left = 521

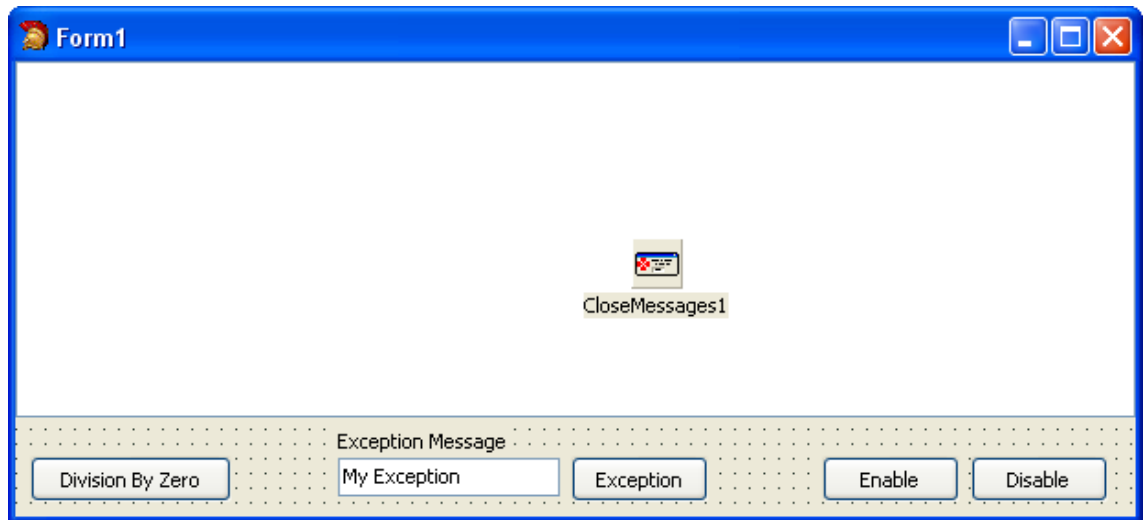
Top = 215

Width = 75

Height = 25

Caption = 'Disable'

E o form deverá ficar igual a imagem abaixo:



No evento *OnClick* do componente *btnException* inclua o código abaixo:

```
raise Exception.Create (edtException.Text) ;
```

No evento *OnClick* do component *btnEnable* inclua o código abaixo:

```
CloseMessages1.Enable;
```

No evento *OnClick* do component *btnDisable* inclua o código abaixo:

```
CloseMessages1.Disable;
```

Agora execute a aplicação e teste os botões. Veja que ao desabilitar o componente as mensagens permanecem na tela e não são gerados históricos das mensagens. Mas quando habilitado ele fecha automaticamente a janela e cria um histórico com a mensagem da janela.

Altere o texto do componente *edtException* em tempo de execução e clique no botão *btnException* e veja que a mensagem é listada.

### Código do Form:

```
object Form1: TForm1
```

```
Left = 413
Top = 257
Caption = 'Form1'
ClientHeight = 248
ClientWidth = 611
Color = clBtnFace
Font.Charset = DEFAULT_CHARSET
Font.Color = clWindowText
Font.Height = -11
Font.Name = 'Tahoma'
Font.Style = []
OldCreateOrder = False
PixelsPerInch = 96
TextHeight = 13

object Label1: TLabel
    Left = 176
    Top = 200
    Width = 92
    Height = 13
    Caption = 'Exception Message'
end

object btnDivisaoZero: TButton
    Left = 8
    Top = 215
    Width = 110
    Height = 25
    Caption = 'Division By Zero'
```

```

    TabOrder = 0

    OnClick = btnDivisaoZeroClick

end

object ListBox1: TListBox

    Left = 0

    Top = 0

    Width = 611

    Height = 194

    Align = alTop

    ItemHeight = 13

    TabOrder = 1

end

object btnEnable: TButton

    Left = 440

    Top = 215

    Width = 75

    Height = 25

    Caption = 'Enable'

    TabOrder = 2

    OnClick = btnEnableClick

end

object btnDisable: TButton

    Left = 521

    Top = 215

    Width = 75

    Height = 25

    Caption = 'Disable'

```

```
    TabOrder = 3
end
object edtException: TEdit
    Left = 176
    Top = 216
    Width = 121
    Height = 21
    TabOrder = 4
    Text = 'My Exception'
end
object btnException: TButton
    Left = 303
    Top = 215
    Width = 75
    Height = 25
    Caption = 'Exception'
    TabOrder = 5
    OnClick = btnExceptionClick
end
object CloseMessages1: TCloseMessages
    Enabled = True
    OnAfterCloseMessage = CloseMessages1 AfterCloseMessage
    Left = 336
    Top = 96
end
end
```

## **Código do Fonte:**

**unit** Unit1;

**interface**

**uses**

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
Dialogs, StdCtrls, CloseMessages;

**type**

TForm1 = class(TForm)

CloseMessages1: TCloseMessages;

btnDivisaoZero: TButton;

ListBox1: TListBox;

btnEnable: TButton;

btnDisable: TButton;

Label1: TLabel;

edtException: TEdit;

btnException: TButton;

**procedure** btnDivisaoZeroClick(Sender: TObject);

**procedure** CloseMessages1 AfterCloseMessage(Sender: TObject; FormHandle:  
HWND;

Msg: TErrorMessage);

**procedure** btnExceptionClick(Sender: TObject);

**procedure** btnEnableClick(Sender: TObject);

**private**

```

    { Private declarations }

public
    { Public declarations }

end;

var
    Form1: TForm1;

implementation

{$R *.dfm}

procedure TForm1.btnDivisaoZeroClick(Sender: TObject);
begin
    ShowMessage(FloatToStr(1 / Random(0)));
end;

procedure TForm1.btnEnableClick(Sender: TObject);
begin
    CloseMessages1.Enable;
end;

procedure TForm1.btnExceptionClick(Sender: TObject);
begin
    raise Exception.Create(edtException.Text);
end;

```



```
procedure TForm1.CloseMessages1AfterCloseMessage(Sender: TObject;  
    FormHandle: HWND; Msg: TErrorMessage);  
  
begin  
    ListBox1.Items.Add(Msg.AsText);  
  
end;  
  
end.
```