

HOW TO WRITE AND CALL DLL'S WITHIN DELPHI

http://www.tutorialspoint.com/dll/dll_delphi_example.htm

Copyright © tutorialspoint.com

The following code creates a DLL containing two functions, Min and Max, with the objective of returning the larger of two integers.

- Start a new DLL project in Delphi *ClickFile – > New, selectDLL*.
- Save the project as delhpdll.
- Fill in the code in the library as given below.

```
// Uffe wrote: This is a toy dll to demonstrate the
// use of dll's
//
// The library export two functions Max and Min
//
// The dll matches the MainProject which is a Delphi
// project which calls the DLL.

//{
//  DELPHI WROTE THIS:
//  Important note about DLL memory management: ShareMem
//  must be the first unit in your library's USES clause
//  AND your project's (select Project-View Source) USES
//  clause if your DLL exports any procedures or functions
//  that pass strings as parameters or function results.
//  This applies to all strings passed to and from your
//  DLL--even those that are nested in records and classes.
//  ShareMem is the interface unit to the BORLNDMM.DLL
//  shared memory manager, which must be deployed along
//  with your DLL. To avoid using BORLNDMM.DLL, pass
//  string information using PChar or ShortString
//  parameters.
//}

uses
  SysUtils,
  Classes;

// Declare stdcall to make interface to other languages

function Min(X, Y: Integer): Integer; stdcall;
begin
  if X < Y then Min := X else Min := Y;
end;

function Max(X, Y: Integer): Integer; stdcall;
begin
  if X > Y then Max := X else Max := Y;
end;

exports // Make available to calling applications
  Min index 1,
  Max index 2;

begin
end.
```

- Build and Save the DLL project.

If you are a command line kind of guy, then you can just execute " dcc32delhpdll. dpr " from the command line. . . that will give you the same DLL but without the IDE stuff. . .

Then you need an application, to call the DLL:

- Start a new "main" application project.
- Make whatever GUI controls you need to test the DLL.
- Fill in the source code for interfacing the DLL as given below.

```
//Main Applet to demonstrate how to call a dll.
//
// SEARCH PATHS:
// The code makes no special effort to search for the DLL.
// Easiest if everything (including DLL) is in the same
// directory.
//
//DLL CALLING:
// This applet demonstrates both Win API calling and
// external calling (see below)
```

```

//
//

// NOTICE: I wasted a lot of time not declaring the
// functions "stdcall" throughout. If you
// declare it "stdcall" in the DLL and not
// in the calling application, then you get hard-to-find
// errors - "external" will crash on you,
// the Win API does not type-check so the function
// just gives "weird" results. (OK this is
// obvious when you think about it .. problem is,
// I didn't)

// Jump to "implementation" section,
// no DLL specs before.
interface

uses
    Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs, StdCtrls;

type
    TForm1 = class(TForm)

        Edit1: TEdit;
        Edit2: TEdit;

        Button1: TButton;
        Label1: TLabel;
        Label2: TLabel;

        Button2: TButton;
        Label3: TLabel;
        Label4: TLabel;
        Label5: TLabel;

        procedure Button1Click(Sender: TObject);
        procedure Button2Click(Sender: TObject);

    private
        { Private declarations }

    public
        { Public declarations }
    end;

    // TMaxFun becomes a function variable (think of
    // a "pointer to function" without the pointer)

    TMaxFun = function(i,j: integer):integer; stdcall;

var
    Form1: TForm1;

implementation

{$R *.DFM}

// This declares Max as a function which should be found
// externally, in the dll
function Max(i,j:integer): integer; stdcall; external 'delhpdll.dll';

// This procedure uses the "external" call to the DLL
procedure TForm1.Button1Click(Sender: TObject);
var
    i,j: integer;
begin
    i := strtoint(Edit1.Text);
    j := strtoint(Edit2.Text);

    Label1.Caption := inttostr(Max(i,j));
    // Easy, eh?
end;

// This calls the DLL directly through the Win API.
// More code, more control.

procedure TForm1.Button2Click(Sender: TObject);
var
    i,j,k: integer;
    Handle: THandle;

    // mmx is a function variable; see type declaration
    // of TMaxFun above.
    mmx : TMaxFun;
begin

```

```

i := strtoint(Edit1.Text);
j := strtoint(Edit2.Text);

// Load the library
Handle := LoadLibrary('DELHPDLL.DLL');

// If succesful ...

if Handle <> 0 then
begin

    // Assign function Max from the DLL to the
    // function variable mmax

    @mmax := GetProcAddress(Handle, 'Max');

    // If successful

    if @mmax <> nil then
    begin
        k := mmax(i,j);
        Label1.Caption := inttostr(k);
    end;

    // Unload library
    FreeLibrary(Handle);
end;
end;
end.

```

• Go fly run your program

Loading [Mathjax]/jax/output/HTML-CSS/jax.js