

Writing a library for .NET ?

Source: <http://coding.derkeiler.com/Archive/Delphi/alt.comp.lang.borland-delphi/2004-08/0675.html>

From: Skybuck Flying (*nospam_at_hotmail.com*)

Date: 08/26/04

Date: Thu, 26 Aug 2004 02:00:04 +0200

Ok one of the greatest things of .NET should be to ability to share libraries...

Not only procedures and functions but also classes/objects ?!

Though I am having a little bit of trouble with even this simple library:

```
library LibraryTest;
```

```
uses
```

```
  SysUtils,  
  Classes,  
  System.Reflection;
```

```
[assembly: AssemblyTitle("")]  
[assembly: AssemblyDescription("")]  
[assembly: AssemblyConfiguration("")]  
[assembly: AssemblyCompany("")]  
[assembly: AssemblyProduct("")]  
[assembly: AssemblyCopyright("")]  
[assembly: AssemblyTrademark("")]  
[assembly: AssemblyCulture("")]
```

```
//
```

```
// Version information for an assembly consists of the following four  
values:
```

```
//
```

```
// Major Version  
// Minor Version  
// Build Number  
// Revision  
//
```

```
// You can specify all the values or you can default the Revision and Build  
Numbers
```

```
// by using the '*' as shown below:
```

```
[assembly: AssemblyVersion('1.0.*')]
```

```
//  
// In order to sign your assembly you must specify a key to use. Refer to  
the  
// Microsoft .NET Framework documentation for more information on assembly  
signing.  
//  
// Use the attributes below to control which key is used for signing.  
//  
// Notes:  
// (*) If no key is specified, the assembly is not signed.  
// (*) KeyName refers to a key that has been installed in the Crypto  
Service  
// Provider (CSP) on your machine. KeyFile refers to a file which  
contains  
// a key.  
// (*) If the KeyFile and the KeyName values are both specified, the  
// following processing occurs:  
// (1) If the KeyName can be found in the CSP, that key is used.  
// (2) If the KeyName does not exist and the KeyFile does exist, the  
key  
// in the KeyFile is installed into the CSP and used.  
// (*) In order to create a KeyFile, you can use the sn.exe (Strong Name)  
utility.  
// When specifying the KeyFile, the location of the KeyFile should be  
// relative to the project output directory. For example, if your  
KeyFile is  
// located in the project directory, you would specify the  
AssemblyKeyFile  
// attribute as [assembly: AssemblyKeyFile('mykey.snk')], provided  
your output  
// directory is the project directory (the default).  
// (*) Delay Signing is an advanced option – see the Microsoft .NET  
Framework  
// documentation for more information on this.  
//  
[assembly: AssemblyDelaySign(false)]  
[assembly: AssemblyKeyFile(")]  
[assembly: AssemblyKeyName(")]  
  
procedure test_procedure_integer( var a : integer ); stdcall;  
begin  
  a := 123456;  
end;  
  
procedure test_procedure_string( var s : string ); stdcall;  
begin  
  s := 'HI YOU !';  
end;  
  
function test_function_integer : integer; stdcall;  
begin
```

```
result := 10101;  
end;
```

```
function test_function_string : string; stdcall;  
begin  
  result := 'IT'S ME !';  
end;
```

```
exports  
  test_procedure_integer,  
  test_procedure_string,  
  test_function_integer,  
  test_function_string;
```

```
begin  
end.
```

[Error] LibraryTest.dpr(66): Procedure definition must be ILCODE calling convention

No where in the help does it say how to do that ?

If I remove stdcall it says this:

[Error] LibraryTest.dpr(81): Unsafe procedure only allowed if compiling with {\$UNSAFECODE ON}

HUH ???

Bye,
Skybuck.