# Exercise 2 COM Fundamentals (Client)

In this exercise you will be working with a prewritten COM object. You will instantiate the object and then invoke its methods. You will write 3 different test harnesses to do this. Each test illustrates a different method of instantiating the object, as follows:

- Using direct creation (***CoCreateInstance***).
- Using a class factory (***CoGetClassObject***).
- Using Visual Basic (***CreateObject***).

The object has 4 methods to call:

Square
Multiply
Average
SquareRoot

### *Step 1 - Getting Started*

Open the workspace file:

**COM Programming\Exercises\COM Fundamentals (Client)\COM Fundamentals (Client).dsw**

This workspace has 4 projects defined. The ***Component*** project is a complete implementation of a COM object and does not require modification. The other three projects: ***Q1***, ***Q2*** and ***Q3*** are the test harnesses that you need to complete.

### *Step 2 - Building the COM Object*

Select the ***Component*** project and then build it. This will create and register the COM object. You can determine the GUID of this object by looking in the file ***Control.rgs***. Run up ***REGEDIT*** and track down the registry entries for the object.

### *Step 3 - Writing the test harness Q1*

Use the detailed instructions given in ***main.cpp***. You will create your COM object directly using ***CoCreateInstance***. Do not create a class factory.

### *Step 4 - Writing the test harness Q2*

Use the detailed instructions given in ***main.cpp***. You will create a class factory and then use the factory to create two identical COM objects.

### *Step 5 - Writing the test harness Q3*

Open the file ***dummy.cpp*** - its empty! That's right, there is no C++ code to write for this exercise. Instead you should write a test harness in Visual Basic. If you are not to sure of how to write VB then either peek at the solution or read the hints below.

The dummy C++ file is used to generate a dummy DLL that can be run under control of Visual Basic. Take a look at the ***Project/Settings/Debug*** tab. VB6 is the executable for the debug session and ***Test1.vbp*** is a dummy VB project for you to modify.

**Hints:**

1. You must make VB aware of our COM object. Select **Project/References** and check the ***Component 1.0 Type Library.***

2. Use a statement like

```
Public ptr As COMPONENTLib.Control
```

   to declare an interface pointer.

3. ***CoCreateInstance*** is called using the VB statement

```
Set ptr = CreateObject("Simple.Control.1")
```

   where ***Simple.Control.1*** is the ***progid*** of the control. Look in the COM object's registration file (***Control.rgs***) to confirm the ***progid***.

4. Any parameter declared in the ***Component***'s IDL file as a ***[retval]*** can be used as a return value. For example, call the square method with code like:

```
x = ptr.Square(y)
```