Lab 13: Java - .NET Interoperability

In this lab, you will create a Java service and invoke it via a .NET client.

Objectives:

- To experiment with Web services interoperability

Part 1: Creating the Service

We will use the service that you wrote in Lab02.

Steps:

_1. In the *lab02Web* project, redploy HelloWorld.java as a *document/literal* service.

Test the redeployed service with the *lab02Client* program.

_2. Use your Web browser to ensure that the service's WSDL is available at:

```
http://localhost:9080/lab02Web/HelloWorldService?wsdl
```

Part 2: Writing a .NET Client

In this part, you will write a simple C#.NET client to access the Lab02 service.

Steps:

_1. Open a command-prompt window from the the Windows Start menu by choosing *Run* and entering **cmd** then pressing Enter. Change to the proper directory in the command prompt. On Windows XP:

```
c:
cd \websrvclass
```

On Windows Vista or later:

```
c:
cd \Users\username\websrvclass
```

Substitute your actual user name for the "username" above.

_2. Execute a batch file to establish the PATH so you can invoke the .NET tools:

```
setcp
```

_3. Create a directory for your C# client, then make it the current directory:

Copyright © Descriptor Systems 2014. All Rights Reserved. Materials may not be reproduced in any form without prior written consent of Joel Barnum.

```
mkdir dotnetclient
cd dotnetclient
```

_4. Use the .NET wsdl.exe tool to generate a C# stub for the service:

```
wsdl /language:cs /protocol:SOAP http://localhost:9080/lab02Web/
    HelloWorldService?wsdl
```

Note: The above should be on a single line. We have split it here so it fits on the printed page.

This step should complete with no errors and generate a file named *HelloWorldService.cs* that contains the C# stub. Examine the stub in Notepad and see if you can make any sense of it.

_5. Next, write the client program itself. Use Notepad or other editor to create a file named Client.cs, and in it, enter:

```
using System;

namespace client
{
  class Client
  {
    [STAThread]static void Main()
    {
       HelloWorldService stub = new HelloWorldService();
       String result = stub.hello("Sam Smith");
       Console.WriteLine(result);
    }
  }
}
```

Note the similarity of C# to Java (namespace == package, etc.).

_6. After saving the client file, compile it:

```
csc Client.cs HelloWorldService.cs
```

Look in the directory -- you should see a file named Client.exe that contains your program.

_7. Run the client:

```
client
```

You should see the output of the Lab01 Web service.

Copyright © Descriptor Systems 2014. All Rights Reserved. Materials may not be reproduced in any form without prior written consent of Joel Barnum.