

Practical 3: Creating a Web Service with ASP.NET

Objective:

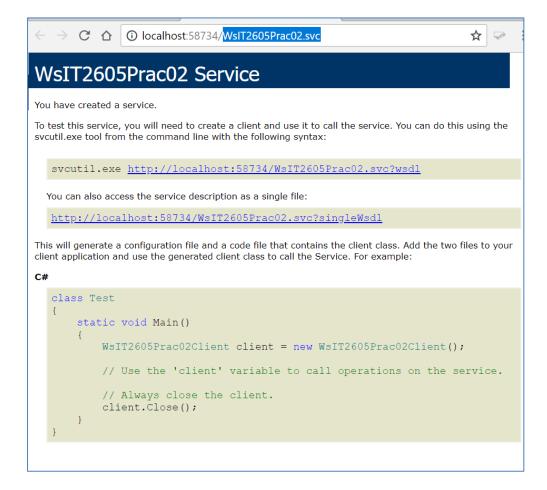
☐ Create a web application to consume the web service

In the previous practical, we learnt how to create a Web Service. In this practical, we will learn how to consume a Web Service using ASP.NET.

Launch the Web Service

Before a new web application is created, it is necessary to launch the web service.

- 1. Open the previous practical IT2605Prac02.
 - a. Right click Ws2605Prac02.svc from Solution Explorer → Select View in Browser
 - b. Do not close the browser, ensure web service is hosted.

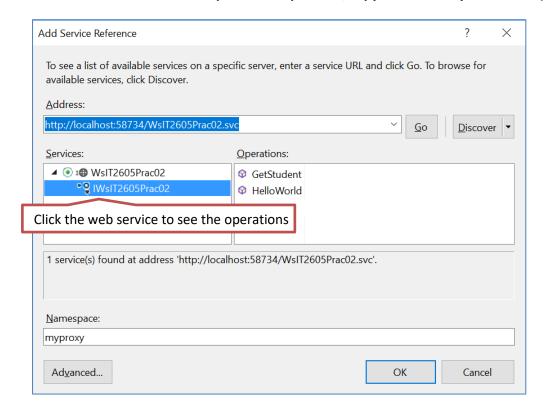




Create web application to consume web service

- 2. Start Visual Studio in another window. Select File → New Project.
- 3. From the *Add New Project* dialog, create an *ASP.NET Application* named it as *IT2605Prac03Client*.
- 4. Select Empty web site template.
- 5. In the solution explorer →Add new item → Add Service Reference.
- 6. Entering a URL of the web service launched in step 1 and click Go. You would see a list of available operation if connection is successful.

(Note do not enter URL from the screen print of this practical, copy the url from your browser)



7. Change the *Namespace* to *myproxy* and click the *OK* button.

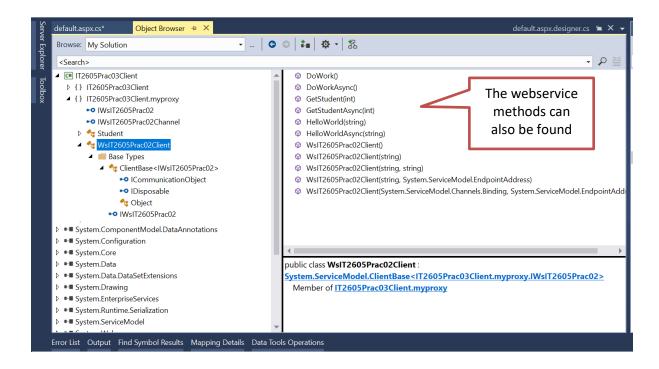


8. Double click **myproxy** in solution explorer to show Object browser.

Visual Studio generates the proxy class using WSDL of the web service. Now your project contains

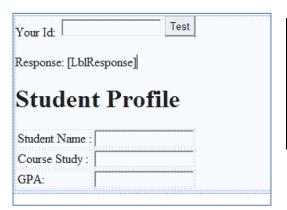
- All the methods that are exposed by the web service
- The parameters and their types
- The return types of the methods

Observed that a new class *WsIT2605Prac02Client* is generated. Visual Studio auto generates the class name by appending "Client" behind the web service name.





- 9. Next, add a web form to invoke the service.
 - a. Right click IT2605Prac03Client in the solution folder → add web form → named default.aspx.
 - b. Layout default.aspx as shown below. Use the IDs of the controls as described in the table.



UI Controls	Id of control
Student Id's text box	TbStudentId
Test button	BtnTest
Response label	LblResponse
Student name's text box	TbName
Course study's text box	TbCourse
GPA's text box	TbGPA

10. Double click the Test button to raise the button click event handler. Make sure you include using IT2605Prac03Client.myproxy at the top of the default.aspx.cs



- 11. Build the code to obtain the returned value from web service method to the web form.
 - Obtain the returned student property from GetStudent. The input parameter is the input value of TbStudentId
 - Obtain the return string from HelloWorld. The input parameter is StudentName property from Student object



```
Oreferences
protected void BtnTest_Click(object sender, EventArgs e)
{
    WsIT2605Prac02Client ws = new WsIT2605Prac02Client();

    int id = Convert.ToInt32(TbStudentId.Text);
    Student me = new Student();
    TbName.Text = me.StudentName;
    TbCourse.Text = me.CourseStudy;
    TbGPA.Text = me.StudentGPA.ToString();
    LblResponse.Text = ws.HelloWorld(me.StudentName);

    ws.Close();
}
```

- 12. Save your work and build the solution. Right-click default.aspx in your browser. Enter the id and click the *Test* button.
- 13. You should get results similar to figure below



Common Error

If you encountered the white page error as below, this mean that the process that is hosting the web service is not listening on that port. This may be because it is not running at all or because it is listening on a different port. You probably have made changes of you web services after adding the service reference in your client side. Sometime, the port number may be changed for updated web service.

If it is due to port number, you can either resolve it by

• Deleting the service reference and re add the service reference, such that it links to the correct port number **or**.

If it is because the web service is not running, then you could

• Run the web service at the back end by right clicking the web service file and select view in browser.



No connection could be made because the target machine actively refused it 127.0.0.1:11035 Description: An unhandled exception occurred during the execution of the current web request. Please review the stack trace for more information about the error and where it originated in the code. Exception Details: System.Net.Sockets.SocketException: No connection could be made because the target machine actively refused it 127.0.0.1:11035 Source Error: Line 160: Line 161: Line 162: Line 163: Line 163: Line 163: Line 164:

Other error

After adding the service reference in client side, you may change your web service with new variable, new classes, or new method, thus the proxy class in web form could not find this variable, method or object. Hence you have to refresh your service reference in the client side (i.e. the project hold the web form) by right click on the reference name and select update service reference.

~ End ~

Checkpoint

By the end of this practical, you should be able to:

✓ Able to consume web service