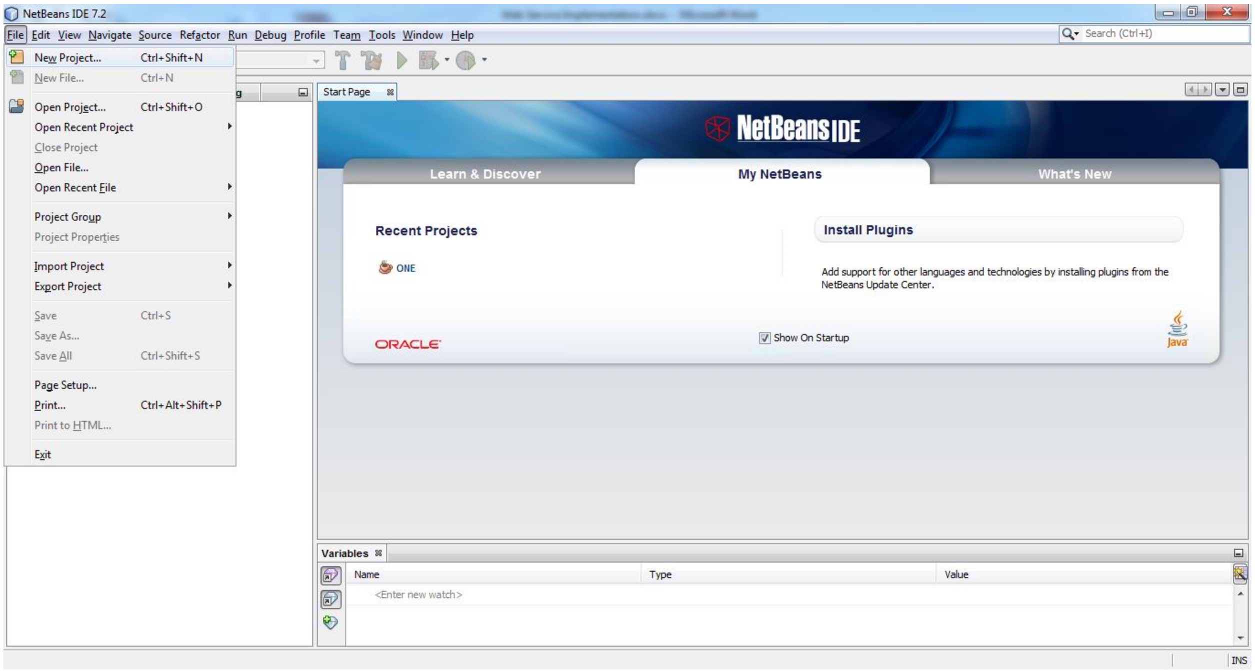
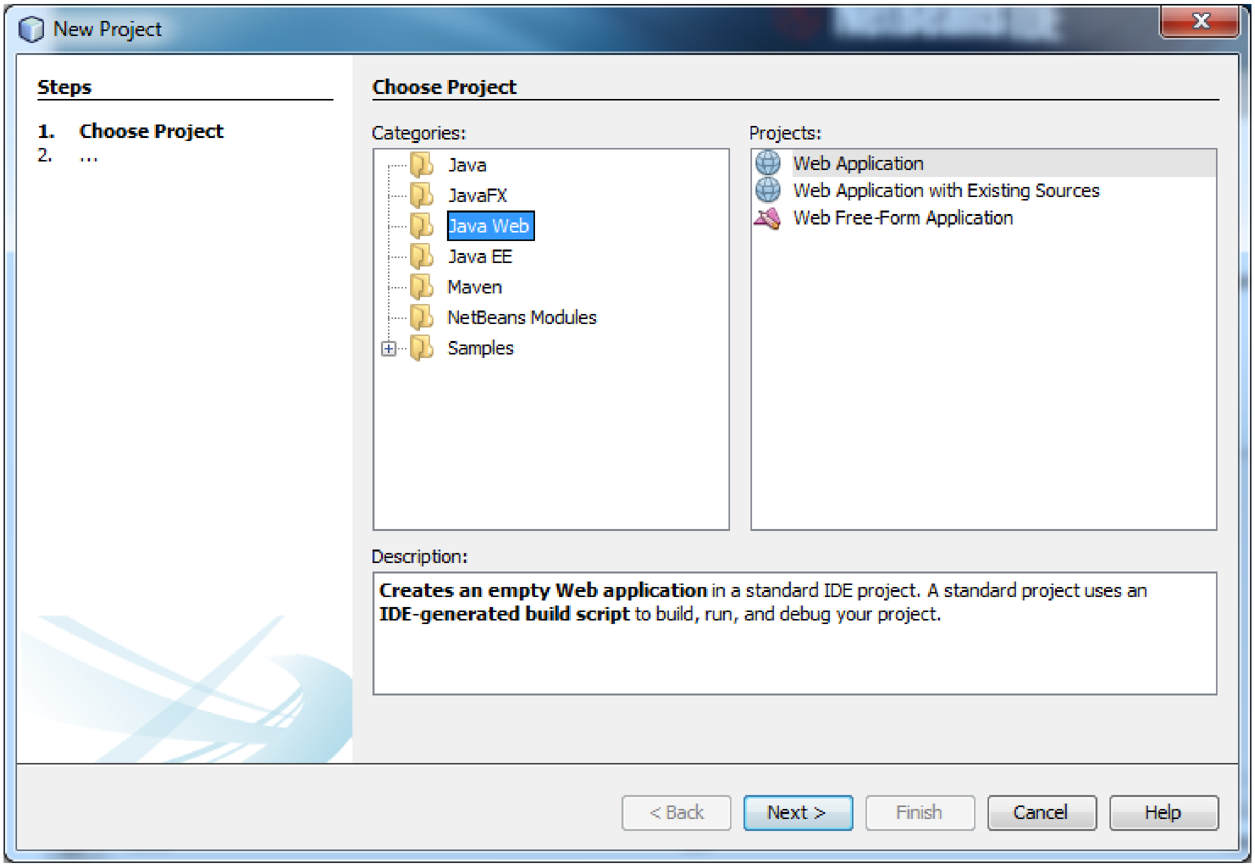
Practical 1 Web Service Implementation

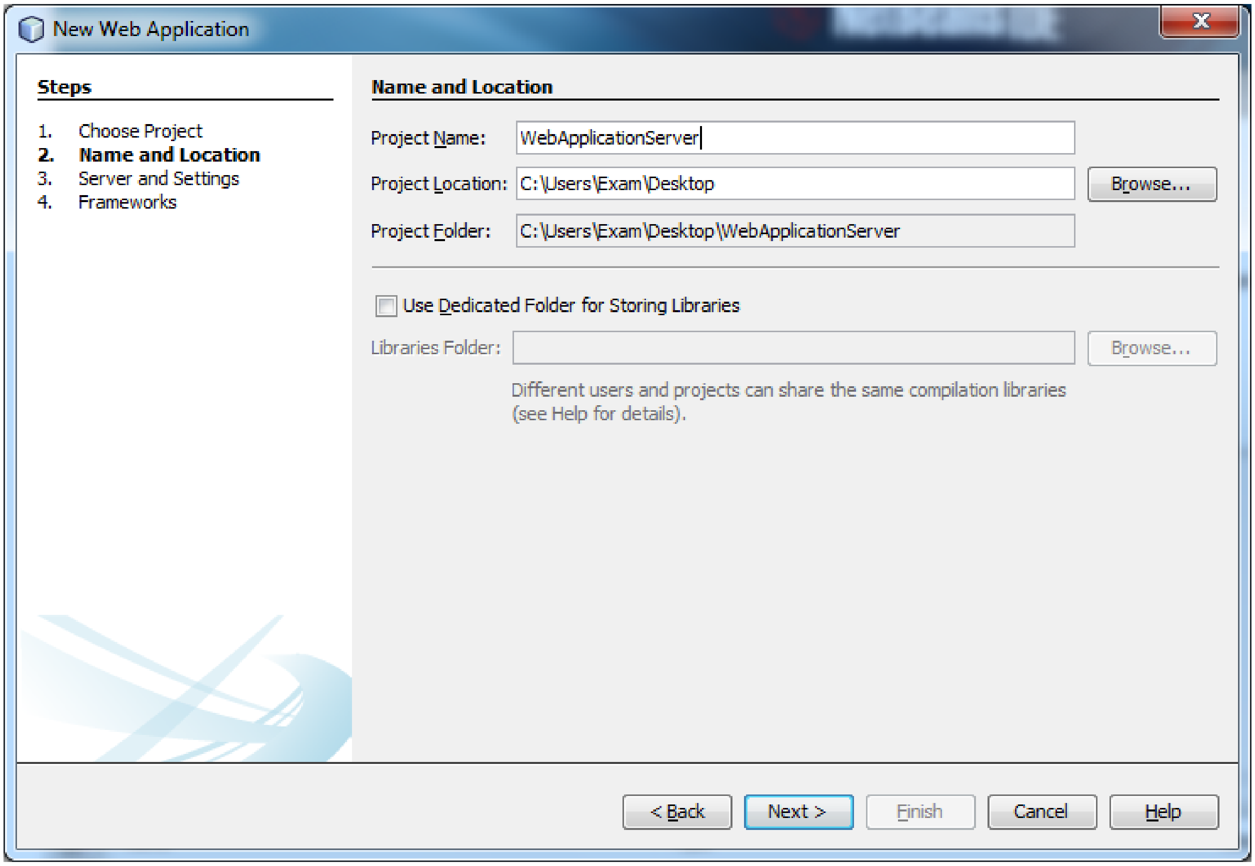
1) Create Web Application in NetBeans

File > New Project > Java Web > Web Application





Creates an empty Web application in a standard IDE project. A standard project uses an **IDE-generated build script** to build, run, and debug your project.



New Web Application

Steps

1. Choose Project
2. **Name and Location**
3. Server and Settings
4. Frameworks

Name and Location

Project Name: WebApplicationServer

Project Location: C:\Users\Exam\Desktop

Project Folder: C:\Users\Exam\Desktop\WebApplicationServer

□ Use Dedicated Folder for Storing Libraries

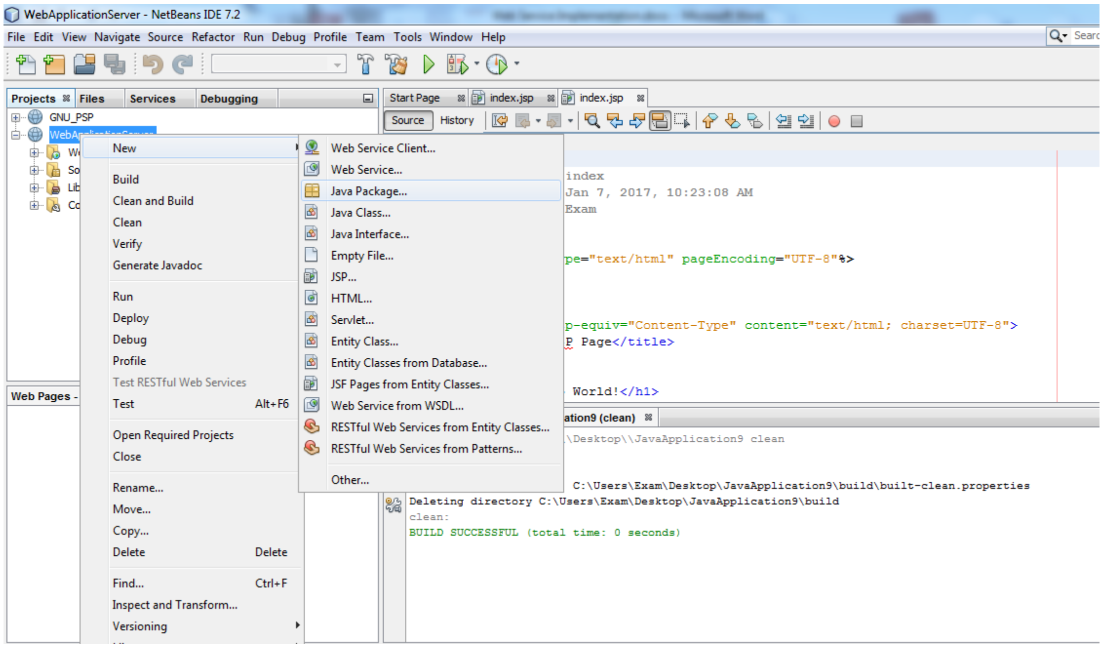
Libraries Folder:

Different users and projects can share the same compilation libraries (see Help for details).

Click Next and Finish.

2) **Create Java Packages from**

Web Application > Source Packages > New > Java Package

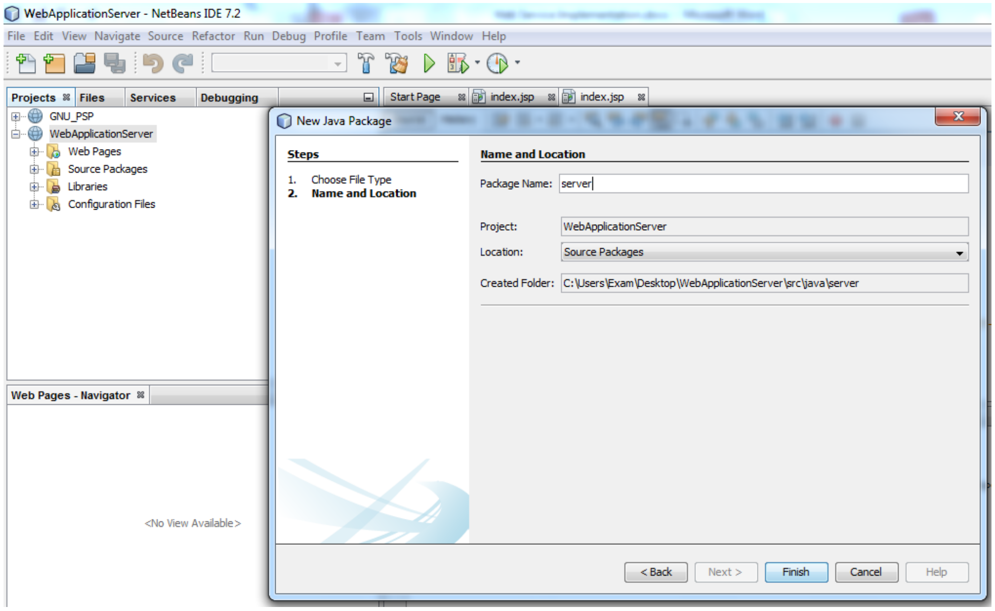


WebApplicationServer - NetBeans IDE 7.2

```
index
Jan 7, 2017, 10:23:08 AM
Exam

pe="text/html" pageEncoding="UTF-8"%>

p-equiv="Content-Type" content="text/html; charset=UTF-8">
P Page</title>

 World!</h1>
```

```
\Desktop\\JavaApplication9 clean

 C:\Users\Exam\Desktop\JavaApplication9\build\built-clean.properties
Deleting directory C:\Users\Exam\Desktop\JavaApplication9\build
clean:
BUILD SUCCESSFUL (total time: 0 seconds)
```

3) Specify Name of Java Package and click on Finish Button



Steps

1. Choose File Type
2. **Name and Location**

Name and Location

Package Name: server

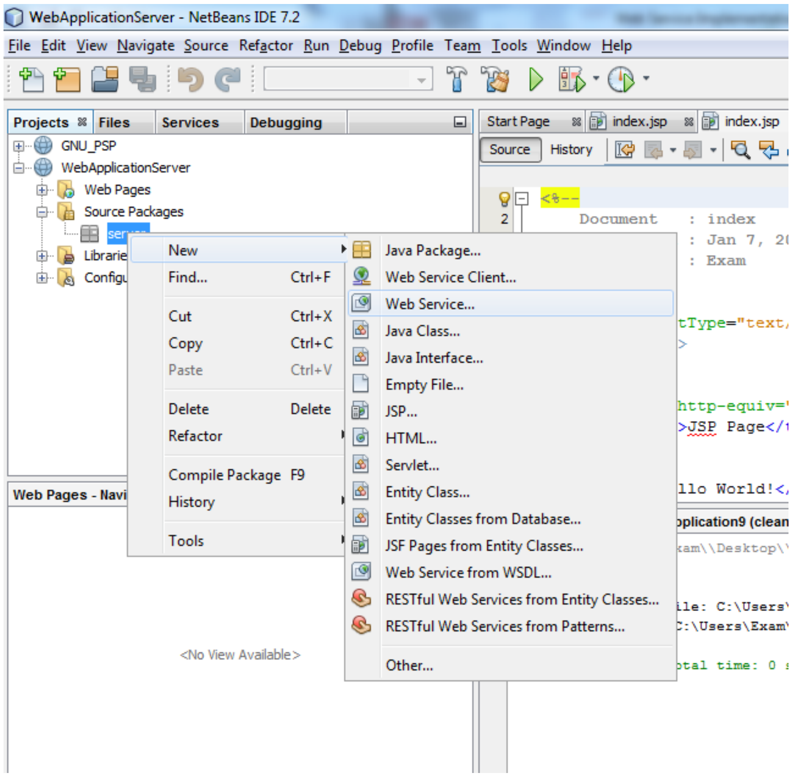
Project: WebApplicationServer

Location: Source Packages

Created Folder: C:\Users\Exam\Desktop\WebApplicationServer\src\java\server

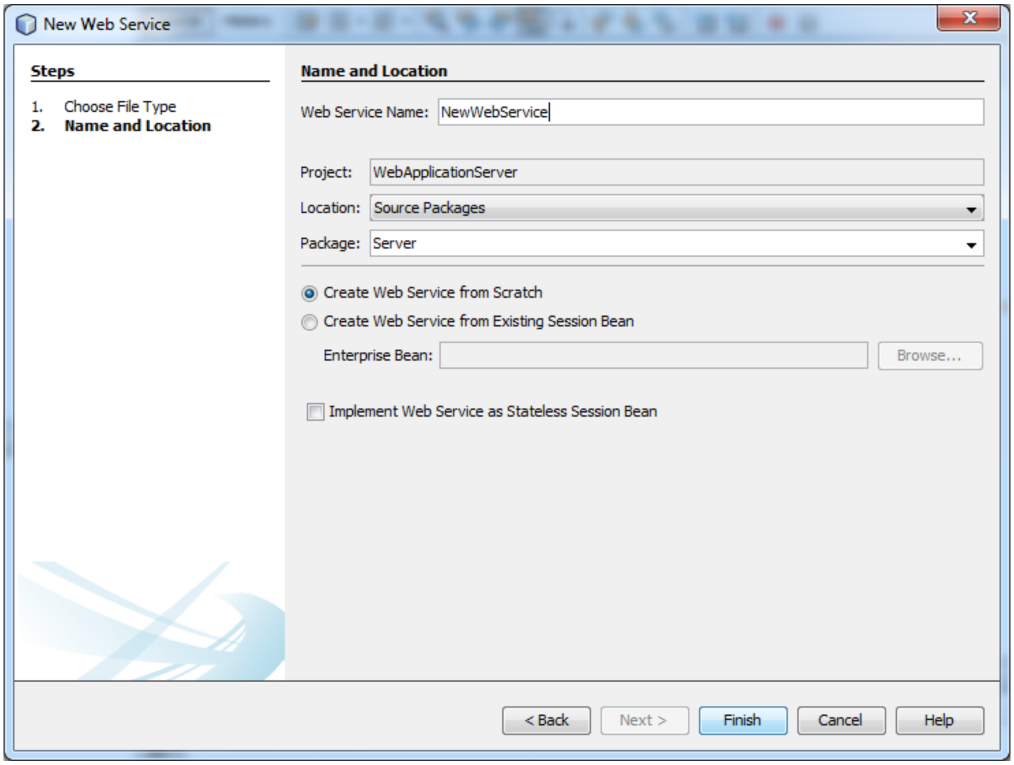
4) Right Click on Package and create web service

New > Web Service



```
<%--
    Document   : index
               : Jan 7, 2(
               : Exam
tType="text,
>
http-equiv='
>JSP Page</t
llo World!<,
```

5) Specify Name of Web service and select Java Package



Web Service Name: NewWebService

Project: WebApplicationServer

Location: Source Packages

Package: Server

● Create Web Service from Scratch

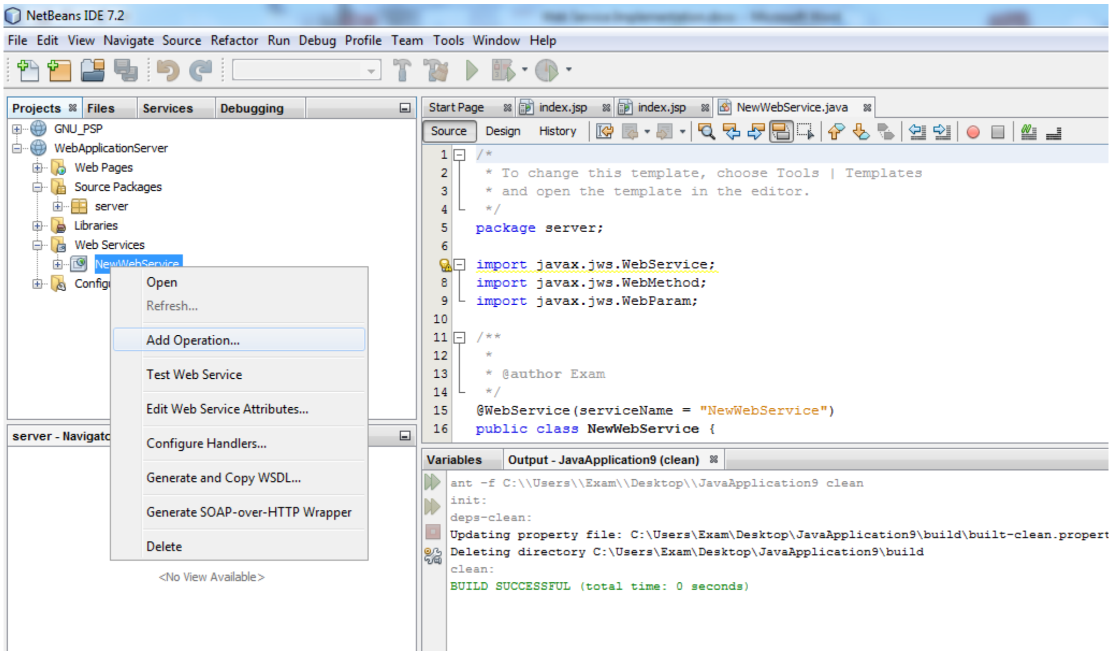
○ Create Web Service from Existing Session Bean

Enterprise Bean:

□ Implement Web Service as Stateless Session Bean

Web Services Folder is created

6) Under Web Services Folder Select Web service and Right click > Add Operation

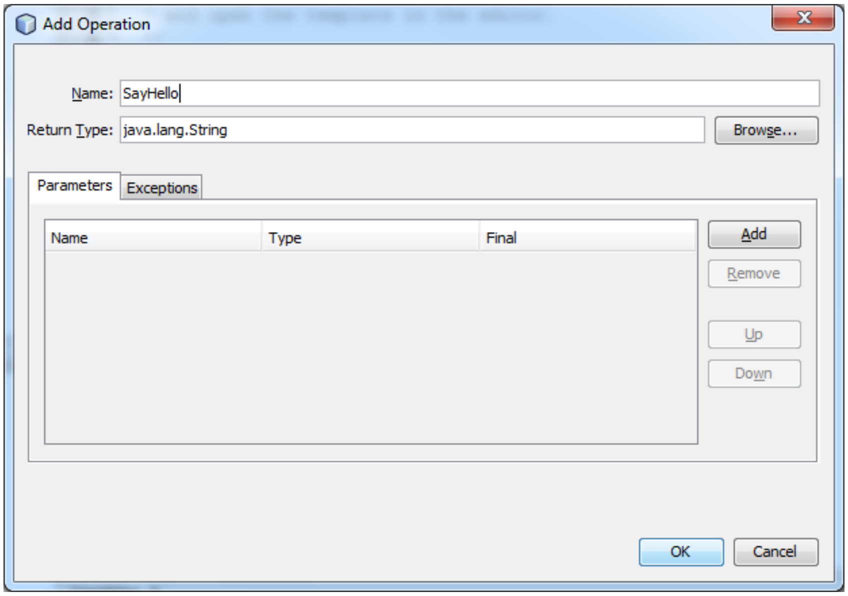


```
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
package server;

import javax.jws.WebService;
import javax.jws.WebMethod;
import javax.jws.WebParam;

/**
 *
 * @author Exam
 */
@WebService(serviceName = "NewWebService")
public class NewWebService {
```

```
ant -f C:\\Users\\Exam\\Desktop\\JavaApplication9 clean
init:
deps-clean:
Updating property file: C:\Users\Exam\Desktop\JavaApplication9\build\built-clean.propert
Deleting directory C:\Users\Exam\Desktop\JavaApplication9\build
clean:
BUILD SUCCESSFUL (total time: 0 seconds)
```

7) Add Operation

Specify Name of Operation and click on OK



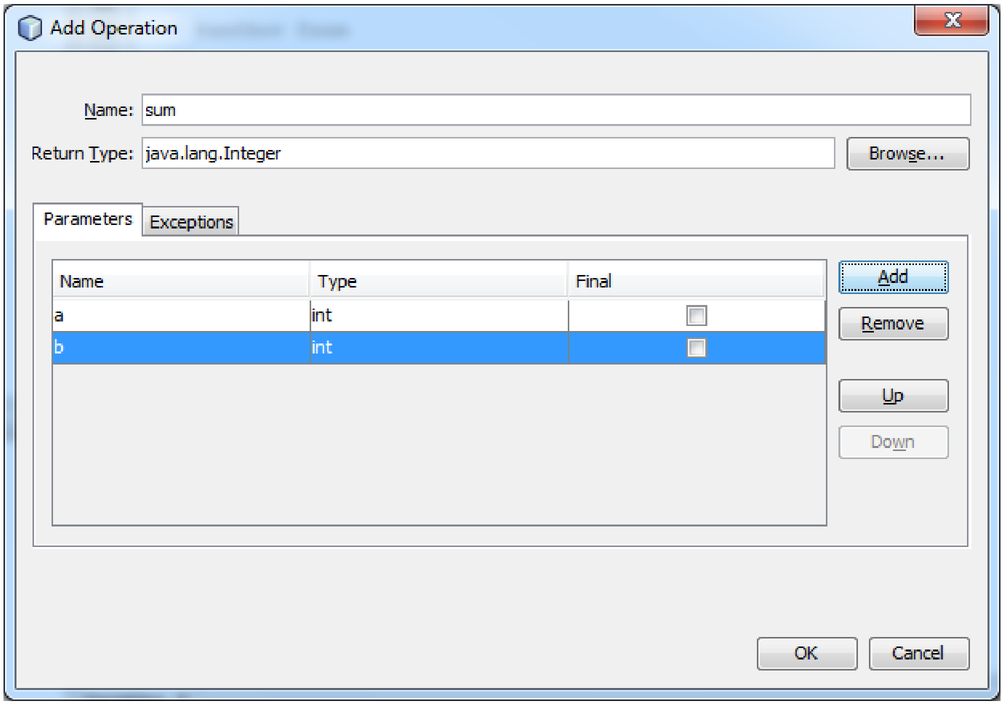
Add Operation

Name: SayHello

Return Type: java.lang.String

Name	Type	Final

8) You May also add operation with parameters



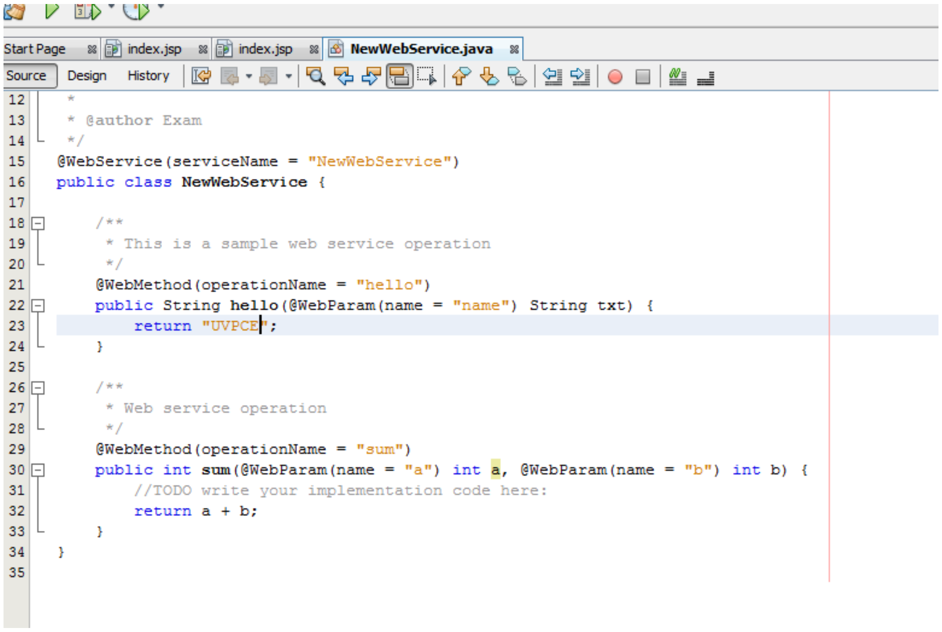
Add Operation

Name: sum

Return Type: java.lang.Integer

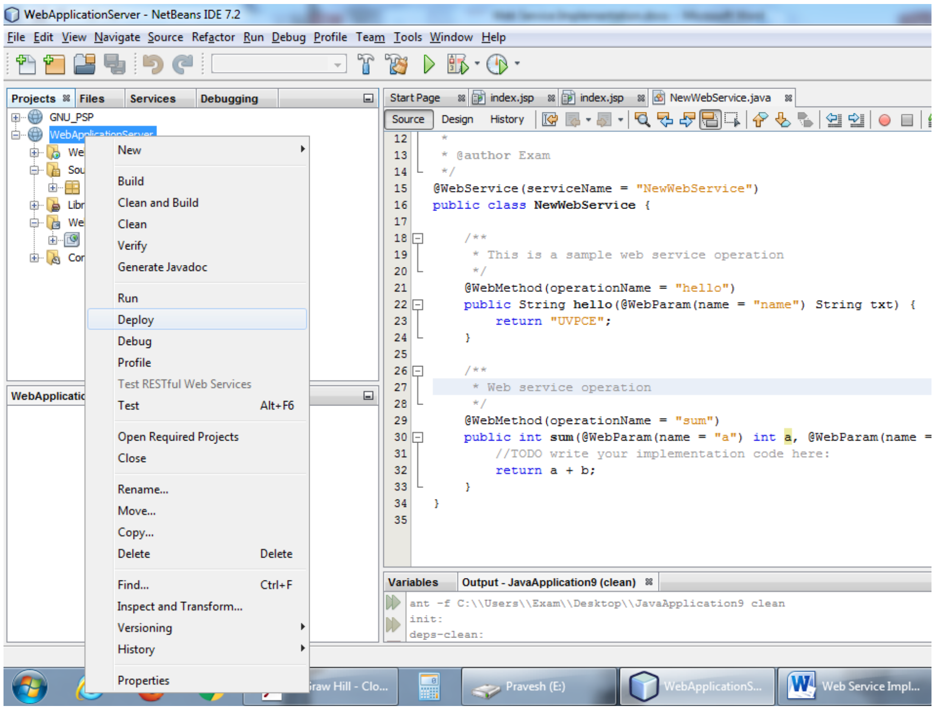
Name	Type	Final
a	int	□
b	int	□

9) Modify Method according to requirement.. like I want to return string is UVPCE and perform sum operation then code



```
 *
 * @author Exam
 */
@WebService(serviceName = "NewWebService")
public class NewWebService {

    /**
     * This is a sample web service operation
     */
    @WebMethod(operationName = "hello")
    public String hello(@WebParam(name = "name") String txt) {
        return "UVPCE";
    }

    /**
     * Web service operation
     */
    @WebMethod(operationName = "sum")
    public int sum(@WebParam(name = "a") int a, @WebParam(name = "b") int b) {
        //TODO write your implementation code here:
        return a + b;
    }
}
```

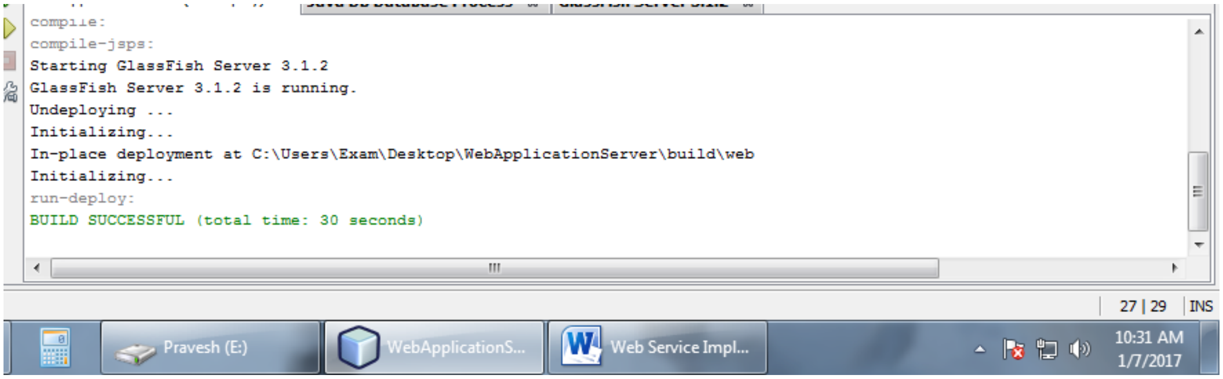
10) Deploy Project



```
 *
 * @author Exam
 */
@WebService(serviceName = "NewWebService")
public class NewWebService {

    /**
     * This is a sample web service operation
     */
    @WebMethod(operationName = "hello")
    public String hello(@WebParam(name = "name") String txt) {
        return "UVPCE";
    }

    /**
     * Web service operation
     */
    @WebMethod(operationName = "sum")
    public int sum(@WebParam(name = "a") int a, @WebParam(name =
        //TODO write your implementation code here:
        return a + b;
    }
}
```

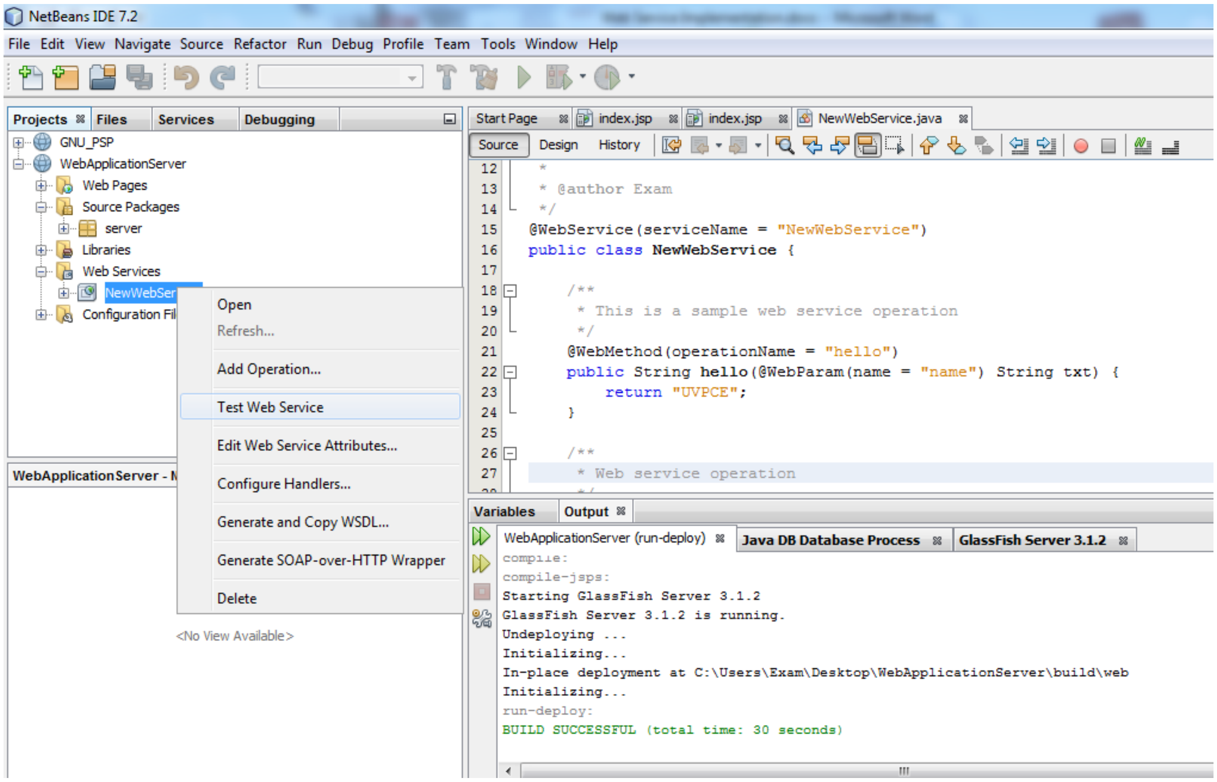
```
ant -f C:\\Users\\Exam\\Desktop\\JavaApplication9 clean
init:
deps-clean:
```

For Successful deployment



```
compile:
compile-jsps:
Starting GlassFish Server 3.1.2
GlassFish Server 3.1.2 is running.
Undeploying ...
Initializing...
In-place deployment at C:\Users\Exam\Desktop\WebApplicationServer\build\web
Initializing...
run-deploy:
BUILD SUCCESSFUL (total time: 30 seconds)
```

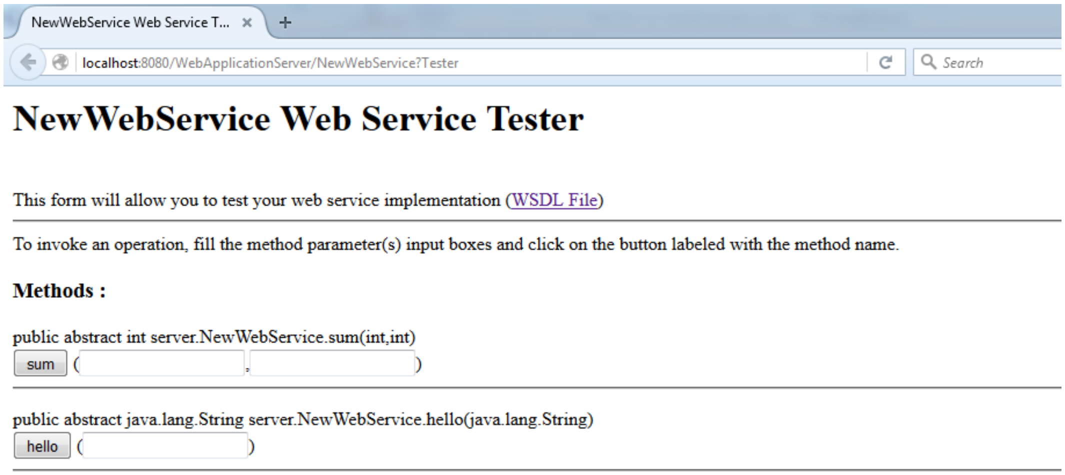
11) Test Web Service



```
 *
 * @author Exam
 */
@WebService(serviceName = "NewWebService")
public class NewWebService {

    /**
     * This is a sample web service operation
     */
    @WebMethod(operationName = "hello")
    public String hello(@WebParam(name = "name") String txt) {
        return "UVPCE";
    }

    /**
     * Web service operation
```

```
compile:
compile-jsps:
Starting GlassFish Server 3.1.2
GlassFish Server 3.1.2 is running.
Undeploying ...
Initializing...
In-place deployment at C:\Users\Exam\Desktop\WebApplicationServer\build\web
Initializing...
run-deploy:
BUILD SUCCESSFUL (total time: 30 seconds)
```

Output of Test Web service in web Browser


NewWebService Web Service T...
localhost:8080/WebApplicationServer/NewWebService?Tester
Search

NewWebService Web Service Tester

This form will allow you to test your web service implementation (WSDL File)

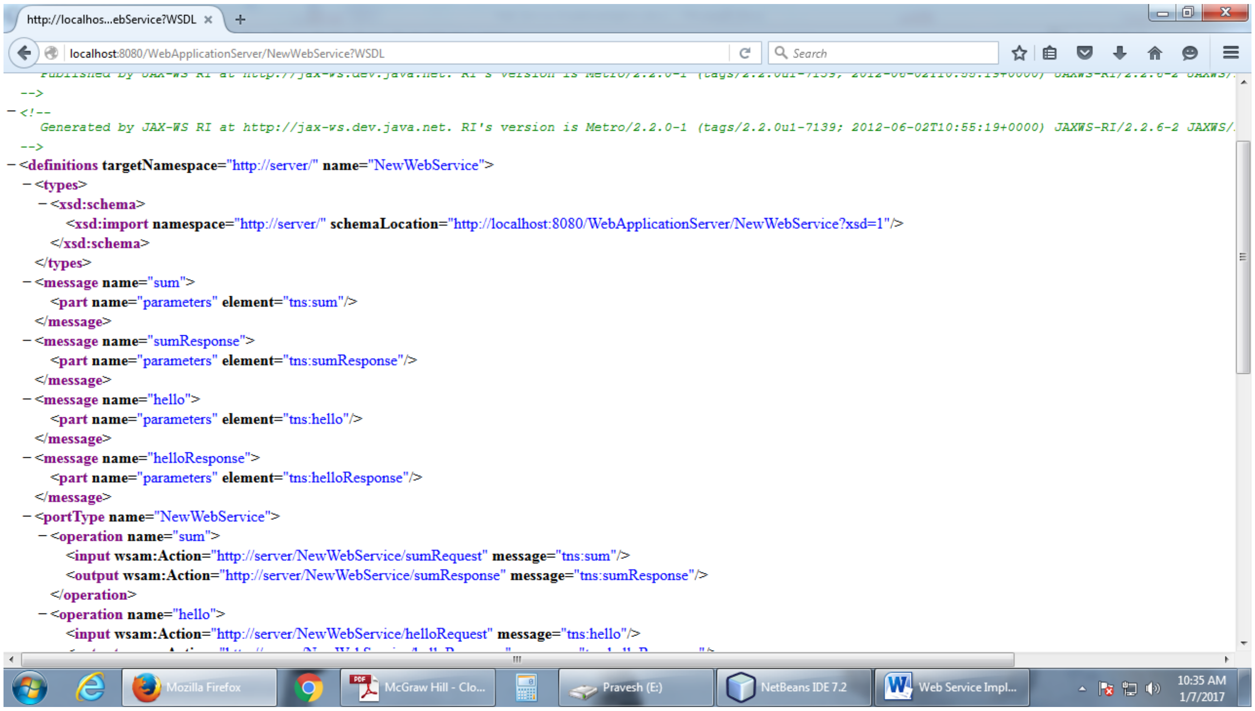
To invoke an operation, fill the method parameter(s) input boxes and click on the button labeled with the method name.

Methods :

public abstract int server.NewWebService.sum(int,int)
sum (,)

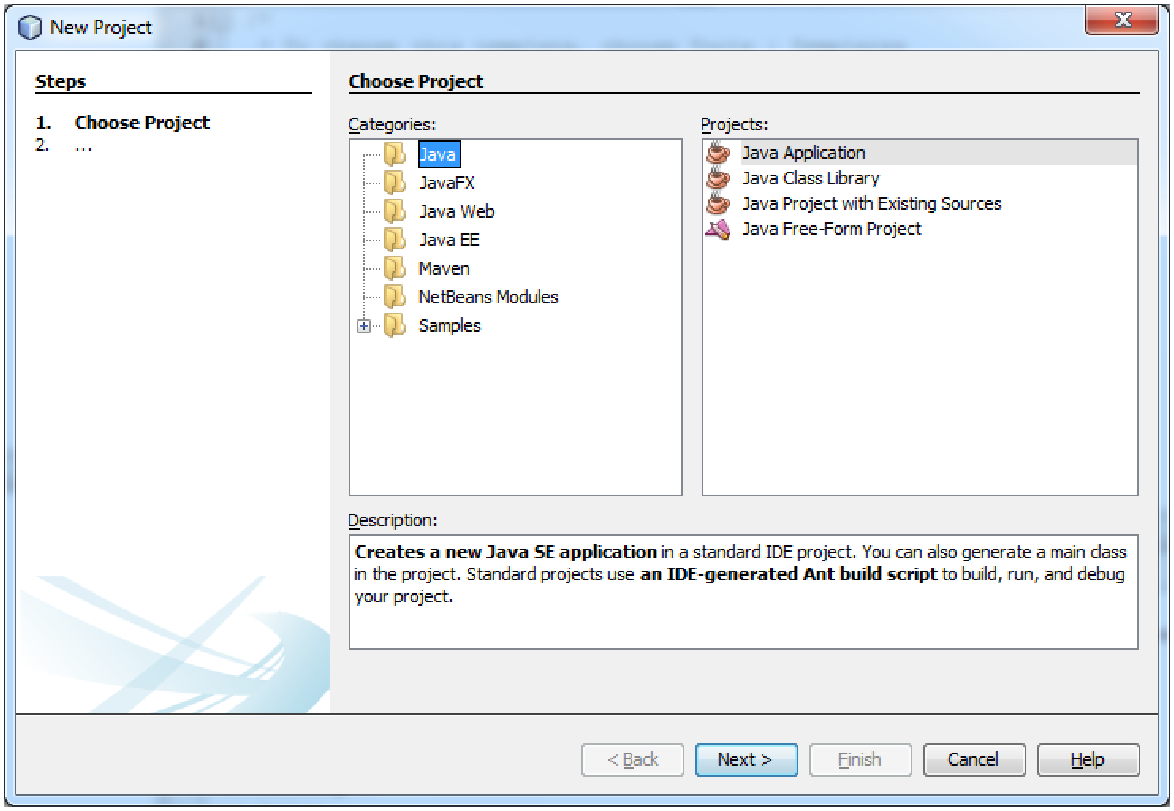
public abstract java.lang.String server.NewWebService.hello(java.lang.String)
hello ( )

12) Click on WSDL File Link and copy newly open window URL



```
   Published by JAX-WS RI at http://jax-ws.dev.java.net. RI's version is Metro/2.2.0-1 (tags/2.2.0u1-7139; 2012-06-02T10:55:19+0000) JAXWS-RI/2.2.6-2 JAXWS/
  -->
– <!--
   Generated by JAX-WS RI at http://jax-ws.dev.java.net. RI's version is Metro/2.2.0-1 (tags/2.2.0u1-7139; 2012-06-02T10:55:19+0000) JAXWS-RI/2.2.6-2 JAXWS/
  -->
– <definitions targetNamespace="http://server/" name="NewWebService">
  – <types>
    – <xsd:schema>
        <xsd:import namespace="http://server/" schemaLocation="http://localhost:8080/WebApplicationServer/NewWebService?xsd=1"/>
      </xsd:schema>
    </types>
  – <message name="sum">
      <part name="parameters" element="tns:sum"/>
    </message>
  – <message name="sumResponse">
      <part name="parameters" element="tns:sumResponse"/>
    </message>
  – <message name="hello">
      <part name="parameters" element="tns:hello"/>
    </message>
  – <message name="helloResponse">
      <part name="parameters" element="tns:helloResponse"/>
    </message>
  – <portType name="NewWebService">
    – <operation name="sum">
        <input wsam:Action="http://server/NewWebService/sumRequest" message="tns:sum"/>
        <output wsam:Action="http://server/NewWebService/sumResponse" message="tns:sumResponse"/>
      </operation>
    – <operation name="hello">
        <input wsam:Action="http://server/NewWebService/helloRequest" message="tns:hello"/>
```

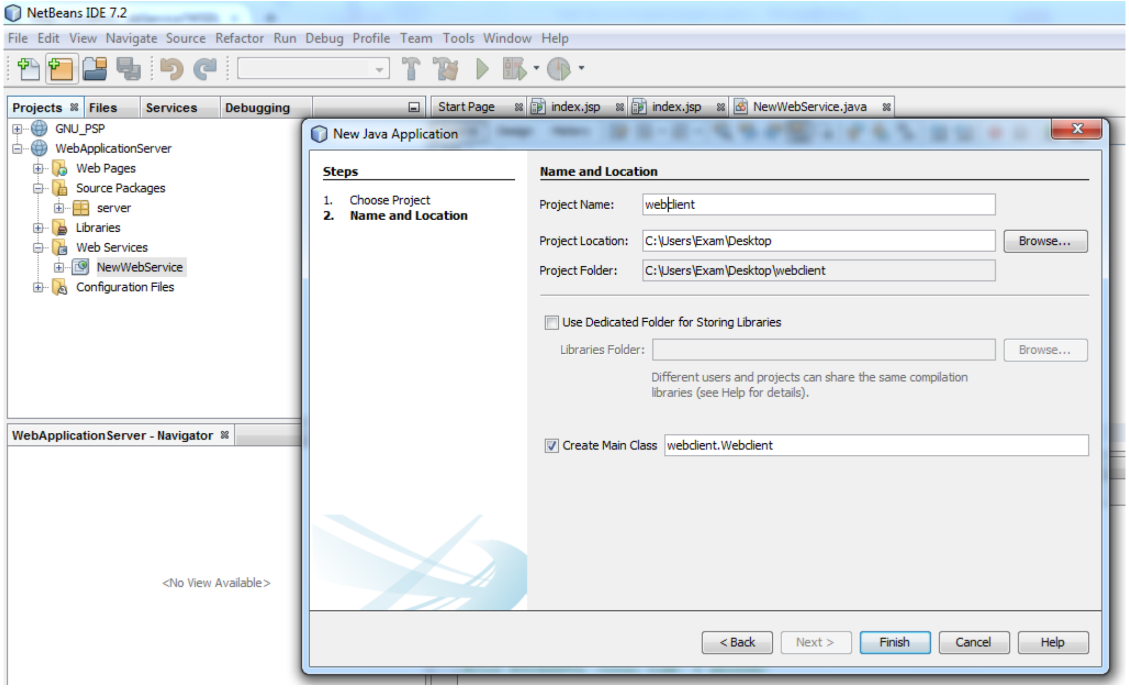
13) Create Simple Java Application from File > New > Java > java Application



Description:

Creates a new Java SE application in a standard IDE project. You can also generate a main class in the project. Standard projects use **an IDE-generated Ant build script** to build, run, and debug your project.

14) Specify Name and Finish



Project Name: webclient

Project Location: C:\Users\Exam\Desktop

Project Folder: C:\Users\Exam\Desktop\webclient

☐ Use Dedicated Folder for Storing Libraries

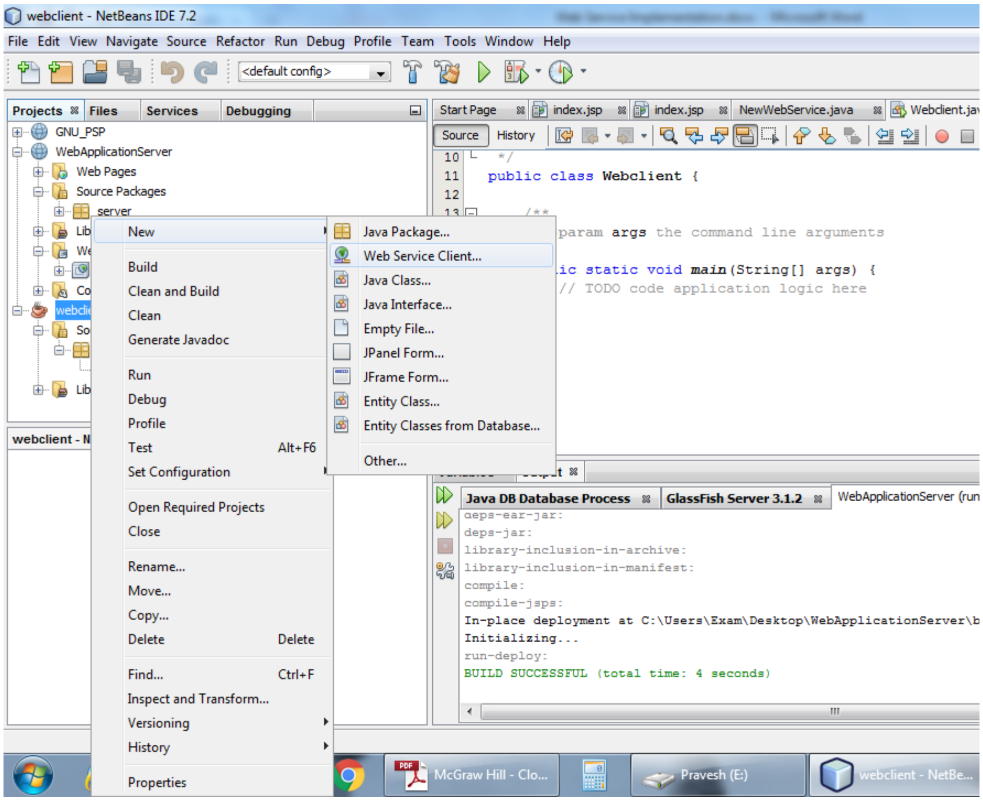
Libraries Folder:

Different users and projects can share the same compilation libraries (see Help for details).

☑ Create Main Class webclient.Webclient

15) Add Web Service Client from

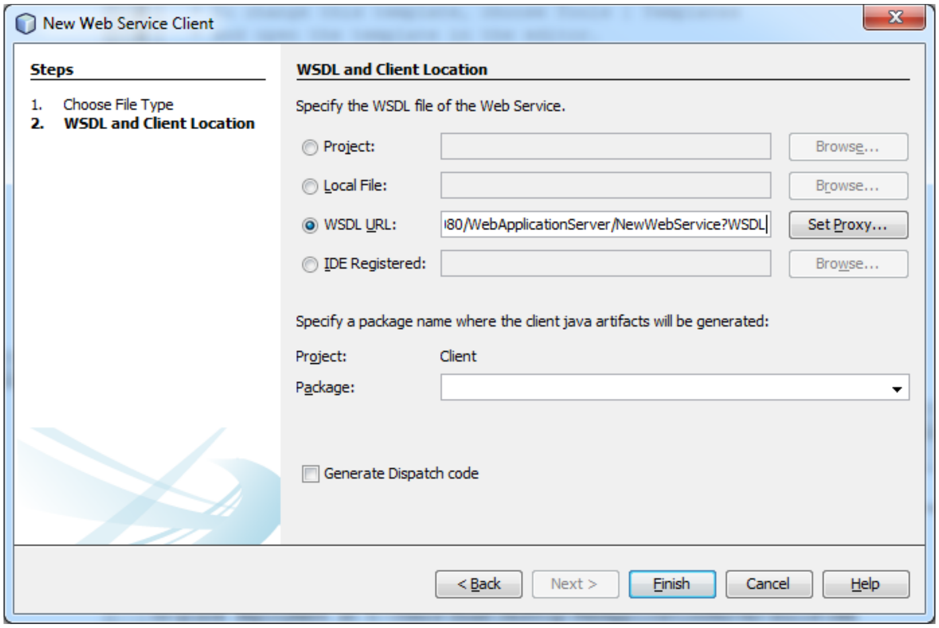
Client Project > Source Package > New > Web Service Client



```
     */
public class Webclient {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here
```

```
deps-ear-jar:
deps-jar:
library-inclusion-in-archive:
library-inclusion-in-manifest:
compile:
compile-jsps:
In-place deployment at C:\Users\Exam\Desktop\WebApplicationServer\b
Initializing...
run-deploy:
BUILD SUCCESSFUL (total time: 4 seconds)
```

16) Add WSDL and click on Finish



Specify the WSDL file of the Web Service.

○ Project:

○ Local File:

● WSDL URL: 80/WebApplicationServer/NewWebService?WSDL

○ IDE Registered:

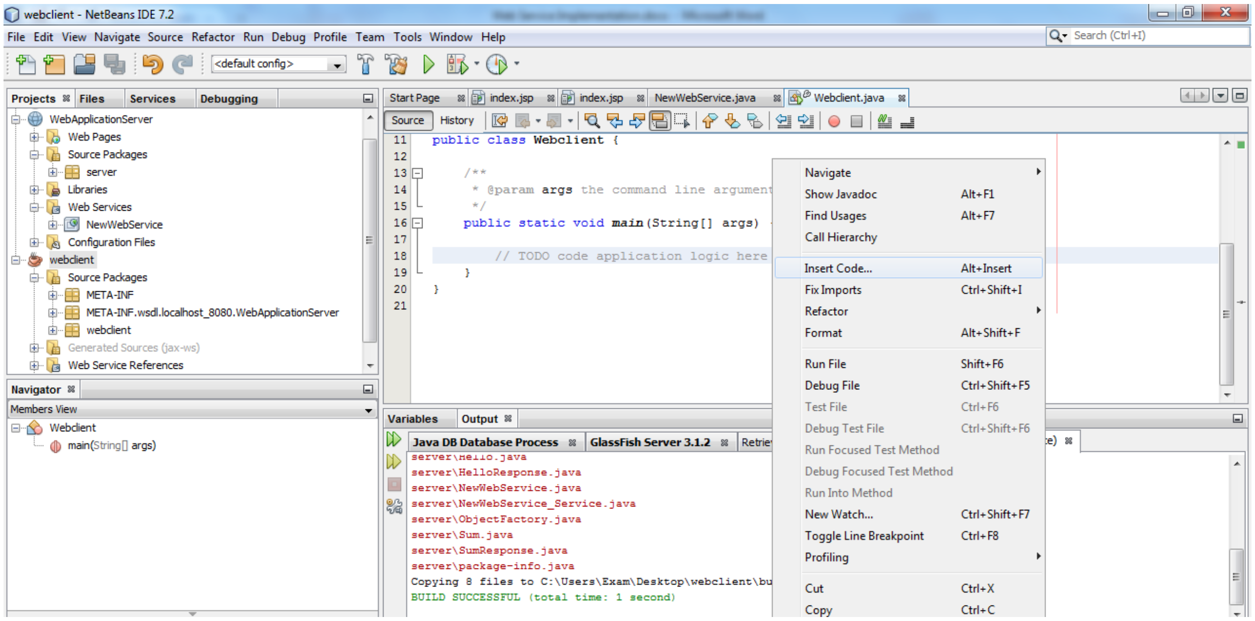
Specify a package name where the client java artifacts will be generated:

Project: Client

Package:

□ Generate Dispatch code

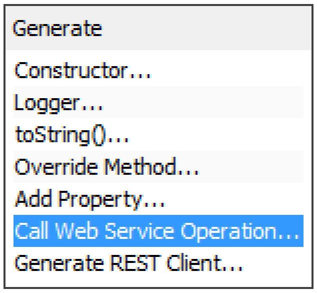
17) Open code of java file and Right click > Insert Code



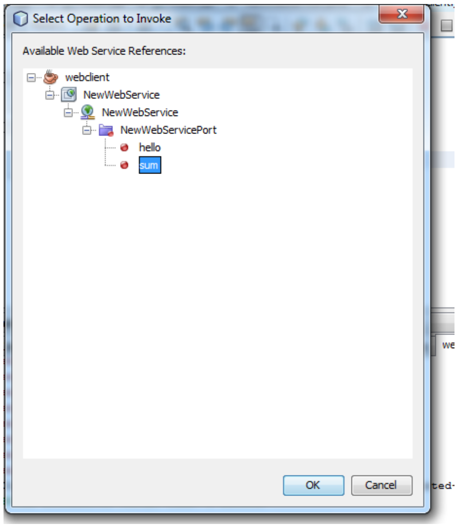
```
public class Webclient {

    /**
     * @param args the command line argument
     */
    public static void main(String[] args)

        // TODO code application logic here
    }
}
```

```
server\Hello.java
server\HelloResponse.java
server\NewWebService.java
server\NewWebService_Service.java
server\ObjectFactory.java
server\Sum.java
server\SumResponse.java
server\package-info.java
Copying 8 files to C:\Users\Exam\Desktop\webclient\bu
BUILD SUCCESSFUL (total time: 1 second)
```

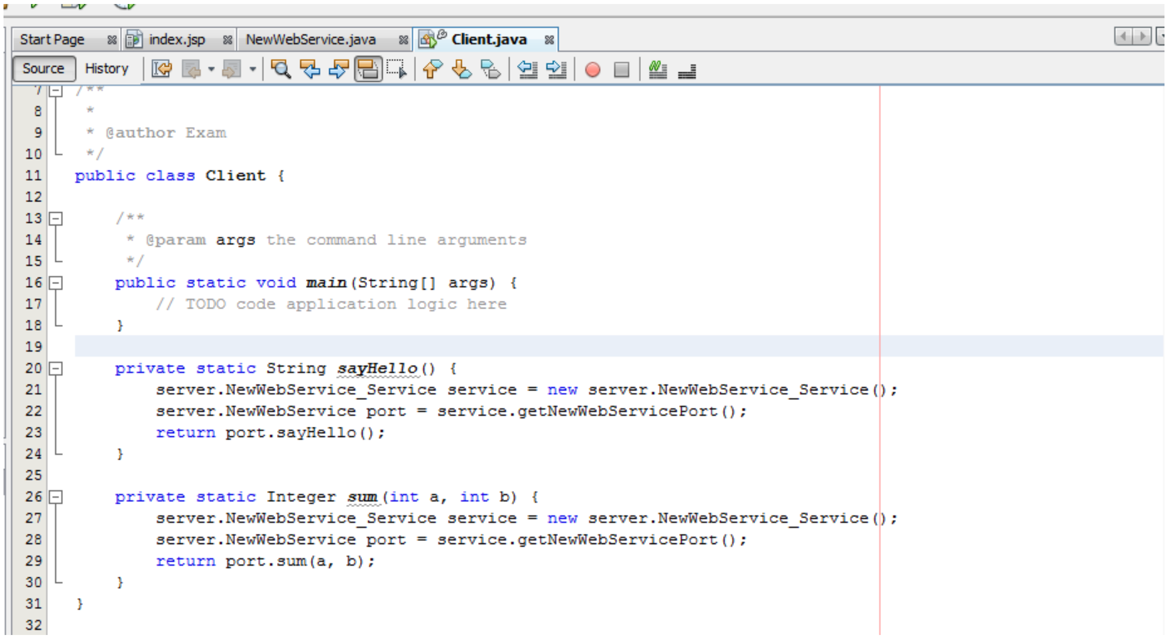
Then Select Call Web Service Operation



18) Select Operation from Web Service

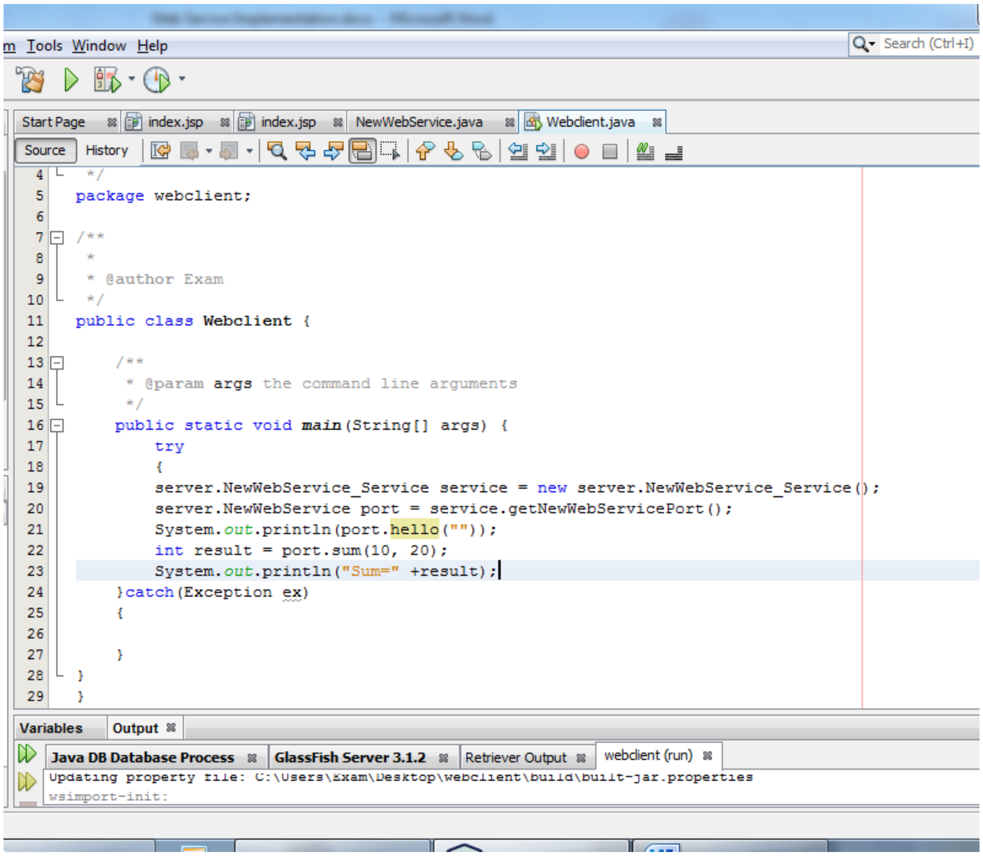


19) You may Invoke multiple operation one by one



```
/**
 *
 * @author Exam
 */
public class Client {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here
    }

    private static String sayHello() {
        server.NewWebService_Service service = new server.NewWebService_Service();
        server.NewWebService port = service.getNewWebServicePort();
        return port.sayHello();
    }

    private static Integer sum(int a, int b) {
        server.NewWebService_Service service = new server.NewWebService_Service();
        server.NewWebService port = service.getNewWebServicePort();
        return port.sum(a, b);
    }
}
```

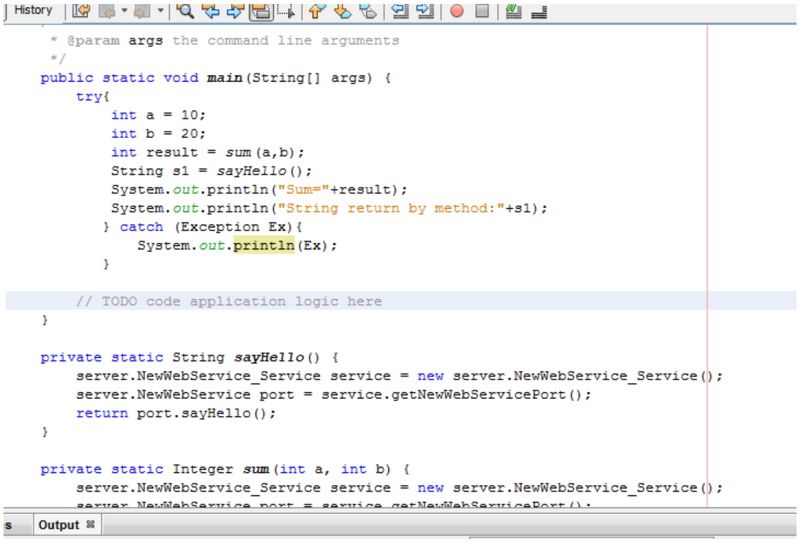
20) Modify code of Main Method in webClient.java



```
 */
package webclient;

/**
 *
 * @author Exam
 */
public class Webclient {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        try
        {
        server.NewWebService_Service service = new server.NewWebService_Service();
        server.NewWebService port = service.getNewWebServicePort();
        System.out.println(port.hello(""));
        int result = port.sum(10, 20);
        System.out.println("Sum=" +result);
    }catch(Exception ex)
    {

    }
}
}
```

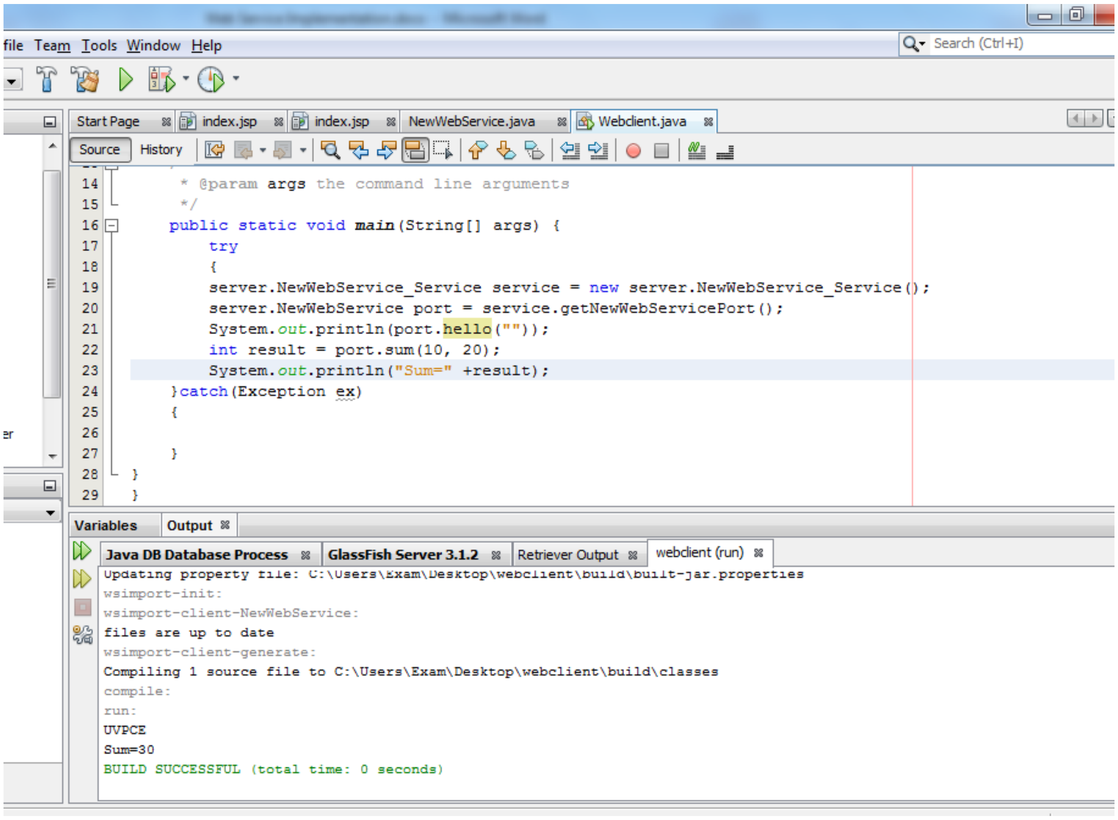
```
Updating property file: C:\Users\Exam\Desktop\webclient\build\built-jar.properties
wsimport-init:
```

OR You Many modify in this way (optional)



```
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        try{
            int a = 10;
            int b = 20;
            int result = sum(a,b);
            String s1 = sayHello();
            System.out.println("Sum="+result);
            System.out.println("String return by method:"+s1);
           } catch (Exception Ex){
               System.out.println(Ex);
           }

        // TODO code application logic here
    }

    private static String sayHello() {
        server.NewWebService_Service service = new server.NewWebService_Service();
        server.NewWebService port = service.getNewWebServicePort();
        return port.sayHello();
    }

    private static Integer sum(int a, int b) {
        server.NewWebService_Service service = new server.NewWebService_Service();
        server.NewWebService port = service.getNewWebServicePort();
```

21) Run Client Java File



```
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        try
        {
        server.NewWebService_Service service = new server.NewWebService_Service();
        server.NewWebService port = service.getNewWebServicePort();
        System.out.println(port.hello(""));
        int result = port.sum(10, 20);
        System.out.println("Sum=" +result);
    }catch(Exception ex)
    {

    }
  }
}
```

```
Updating property file: C:\Users\Exam\Desktop\webclient\build\built-jar.properties
wsimport-init:
wsimport-client-NewWebService:
files are up to date
wsimport-client-generate:
Compiling 1 source file to C:\Users\Exam\Desktop\webclient\build\classes
compile:
run:
UVPCE
Sum=30
BUILD SUCCESSFUL (total time: 0 seconds)
```