**SOAP WebService- Standard Operating procedure**

Version 0.1

**Document History:**

**Document Amendment History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Revision** | **Description** | **Date** | **By** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Document Review History**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Document Version** | **Date Reviewed** | **Reviewed By** | **Role** | **Review Status** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Document Approval**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Document Version** | **Date Approved** | **Approved By** | **Role** | **Document Status** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

[**1.** **Overview** 3](#_Toc452740252)

[**Introduction** 3](#_Toc452740253)

[**What is SOAP?** 3](#_Toc452740254)

[**SOAP Architecture** 3](#_Toc452740255)

[**What is WSDL?** 4](#_Toc452740256)

[**What is UDDI?** 4](#_Toc452740257)

[**2.** **RESTful Web Services - Environment Setup** 4](#_Toc452740258)

[**2.1** **Setup Java Development Kit (JDK)** 4](#_Toc452740259)

[**2.2** **Setup Eclipse IDE** 5](#_Toc452740260)

[**3.** **Creating a SOAP Webservice:** 5](#_Toc452740261)

[4. Reference and Sample. 18](#_Toc452740262)

1. **Overview**

**Introduction**

This document illustrates the steps that are followed to create a simple SOAP web service application.

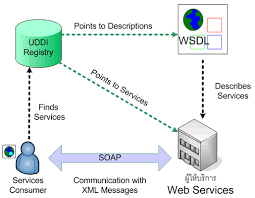
**What is SOAP?**

SOAP stands for Simple Object Access Protocol. It is a XML-based protocol for accessing web services.

SOAP is a W3C recommendation for communication between two applications.

SOAP is XML based protocol. It is platform independent and language independent. By using SOAP, you will be able to interact with other programming language applications.

**SOAP Architecture**



**What is WSDL?**

WSDL stands for Web Services Description Language. It is the standard format for describing a web service.

* WSDL is an XML-based protocol for information exchange in decentralized and distributed environments.
* WSDL definitions describe how to access a web service and what operations it will perform.
* WSDL is a language for describing how to interface with XML-based services.
* WSDL is an integral part of Universal Description, Discovery, and Integration (UDDI), an XML-based worldwide business registry.
* WSDL is the language that UDDI uses.
* WSDL is pronounced as 'wiz-dull' and spelled out as 'W-S-D-L'.

**What is UDDI?**

UDDI is an XML-based standard for describing, publishing, and finding web services.

* UDDI stands for Universal Description, Discovery, and Integration.
* UDDI is a specification for a distributed registry of web services.
* UDDI is a platform-independent, open framework.
* UDDI can communicate via SOAP, CORBA, Java RMI Protocol.
* UDDI uses Web Service Definition Language(WSDL) to describe interfaces to web services.
* UDDI is seen with SOAP and WSDL as one of the three foundation standards of web services.
* UDDI is an open industry initiative, enabling businesses to discover each other and define how they interact over the Internet.

1. **RESTful Web Services - Environment Setup**
   1. **Setup Java Development Kit (JDK)**

You can download the latest version of SDK from Oracle's Java site: [Java SE Downloads](http://www.oracle.com/technetwork/java/javase/downloads/index.html). You will find instructions for installing JDK in downloaded files, follow the given instructions to install and configure the setup. Finally set PATH and JAVA\_HOME environment variables to refer to the directory that contains java and javac, typically java\_install\_dir/bin and java\_install\_dir respectively.

If you are running Windows and installed the JDK in C:\jdk1.7.0\_75, you would have to put the following line in your C:\autoexec.bat file.

set PATH=C:\jdk1.7.0\_75\bin;%PATH%

set JAVA\_HOME=C:\jdk1.7.0\_75

**2.2** **Setup Eclipse IDE**

To install Eclipse IDE, download the latest Eclipse binaries from <http://www.eclipse.org/downloads/>. Once you downloaded the installation, unpack the binary distribution into a convenient location. For example in C:\eclipse on windows, or /usr/local/eclipse on Linux/Unix and finally set PATH variable appropriately.

* 1. **Setup Apache Tomcat**

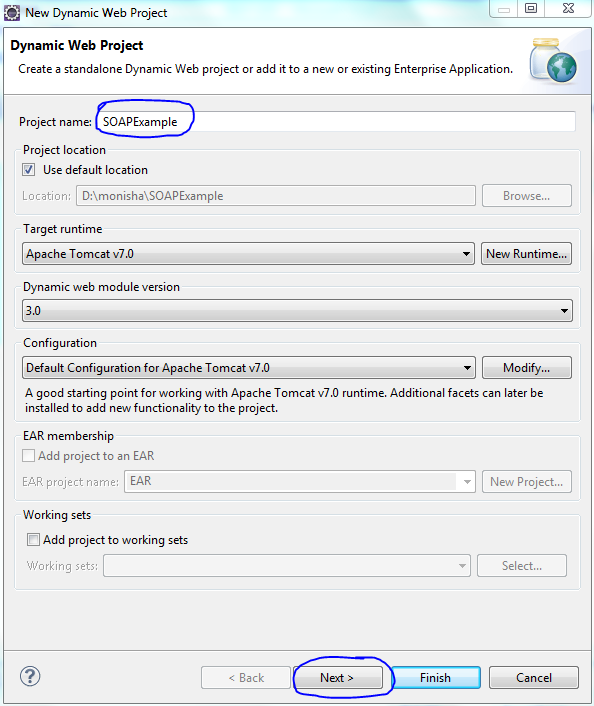
You can download the latest version of Tomcat from <http://tomcat.apache.org/>. Once you downloaded the installation, unpack the binary distribution into a convenient location. For example in C:\apache-tomcat-7.0.59 on windows, or /usr/local/apache-tomcat-7.0.59 on Linux/Unix and set CATALINA\_HOME environment variable pointing to the installation locations.

1. **Creating a SOAP Webservice:**

**3.1 Create a new Dynamic web project**

Lets create a simple SOAPExample example for SOAP web Services

File -> New -> Project and finally select Dynamic Web Project



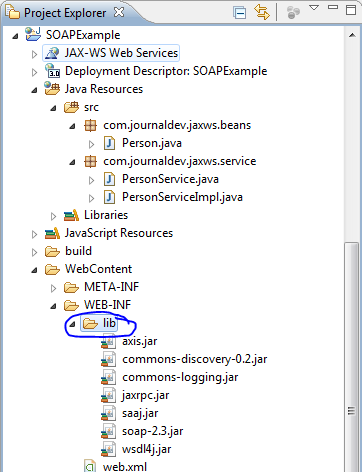
Give a name to your project and give click on finish button.

**3.2 Add required library**

For example I am naming my project as SOAPExample

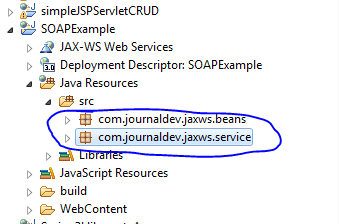
For installing jars do the following steps

SOAPExample🡪WebContent🡪Lib🡪Paste the Jar Files Which you have downloaded.

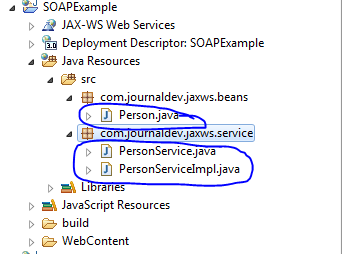


**3.3 Creating Source File**

Now let us create actual source files under the SOAPExample project. First we need to create a package called com.journaldev.jaxws.beans and com.journaldev.jaxws.service . To do this, right click on src in package explorer section and follow the option: New -> Package.



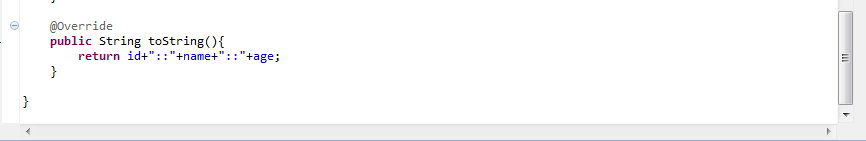
Now we will create Person.java under that Package com.journaldev.jaxws.beans and PersonService.java and PersonServiceImpl.java under com.journaldev.jaxws.service.



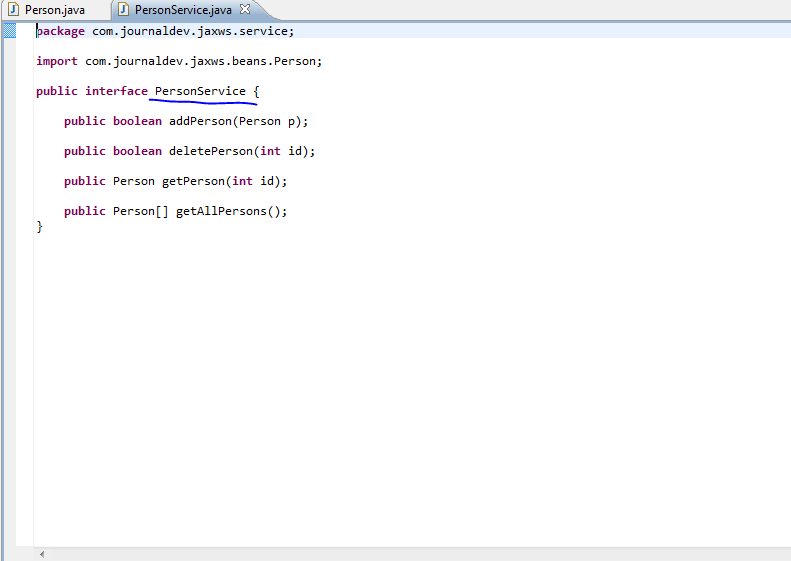
Type the data shown below to that newly created class

Person.java

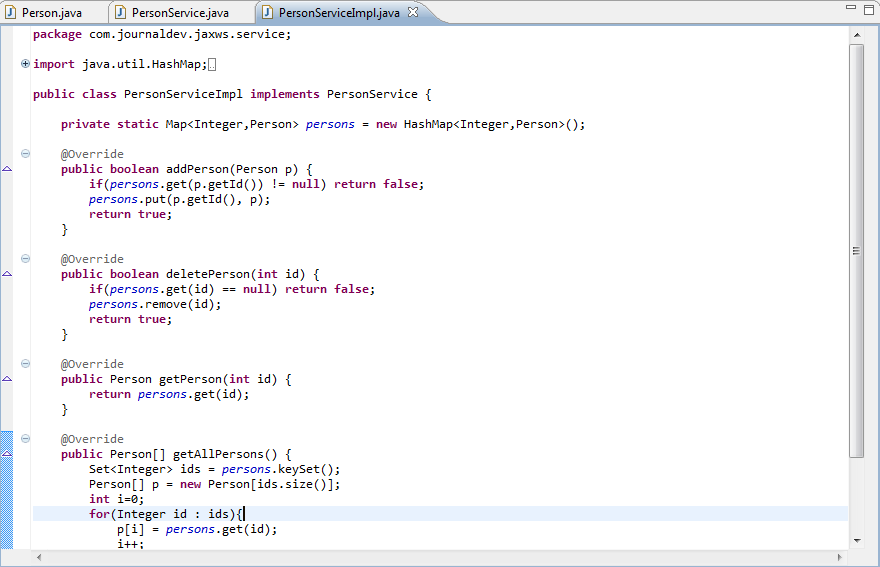


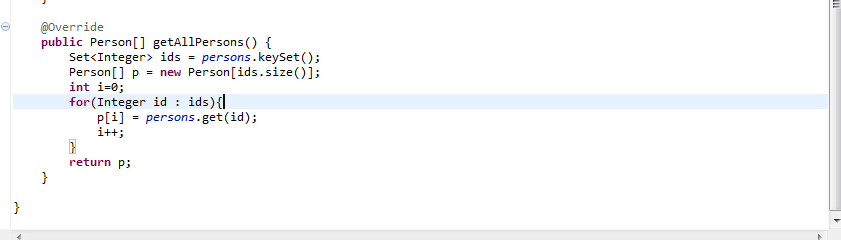


PersonService.java



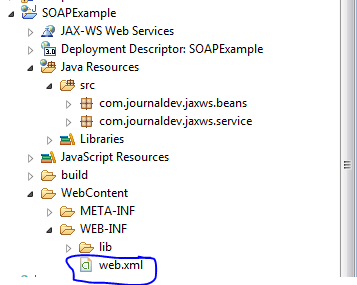
PersonServiceImpl.java



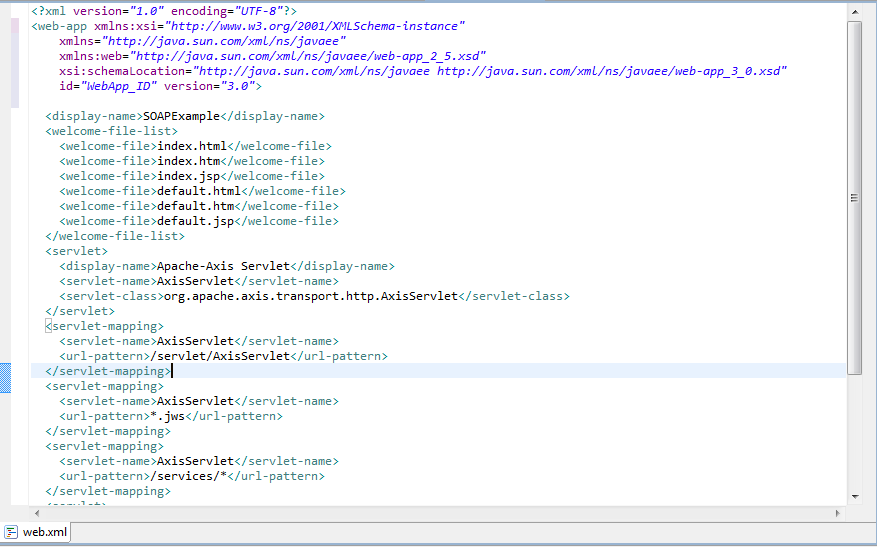


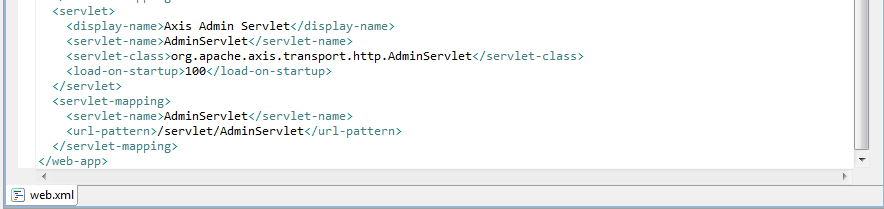
**3.4 Creating Web.Xml**

You need to create a Web xml Configuration file which is an XML file.



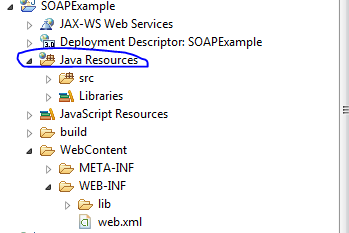
Now type the below code to it.

****

****

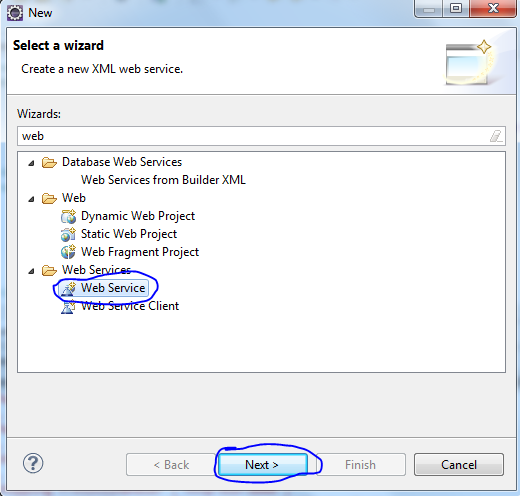
**3.5 Start the Service**

To create service follow the below steps.

****

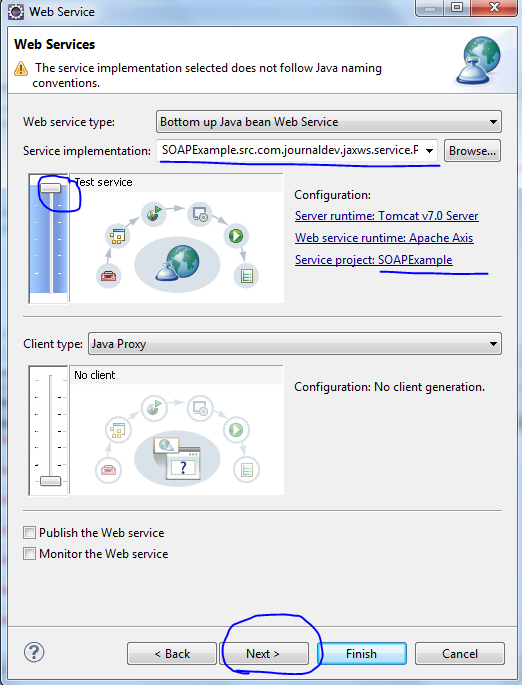
Right click java Resources🡪New🡪Other you will see the below window.

**Step 1:**



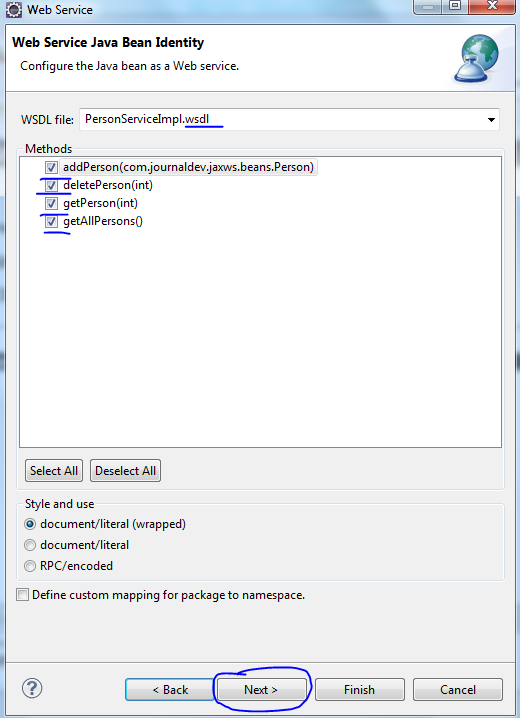
**Step 2:**

Select the service implementation and set the service project as SOAPExample and set level to Test Service and click next.

****

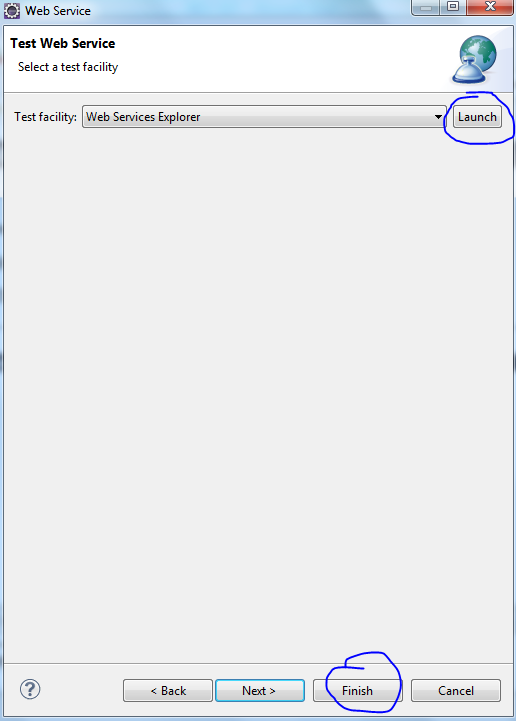
**Step 3:**

Check as below and click next.

****

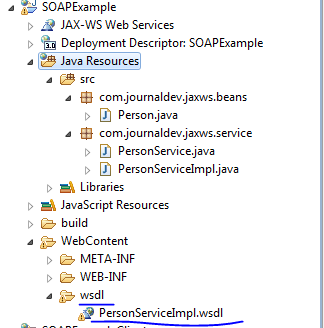
**Step 4:**

Launch the Service and Finish.



**Step 5:**

Now the wsdl File will be created automatically



**3.6 Running the Program**

**Step 1:**

To Run the Project Right Click your Project🡪Run As🡪Run On Server

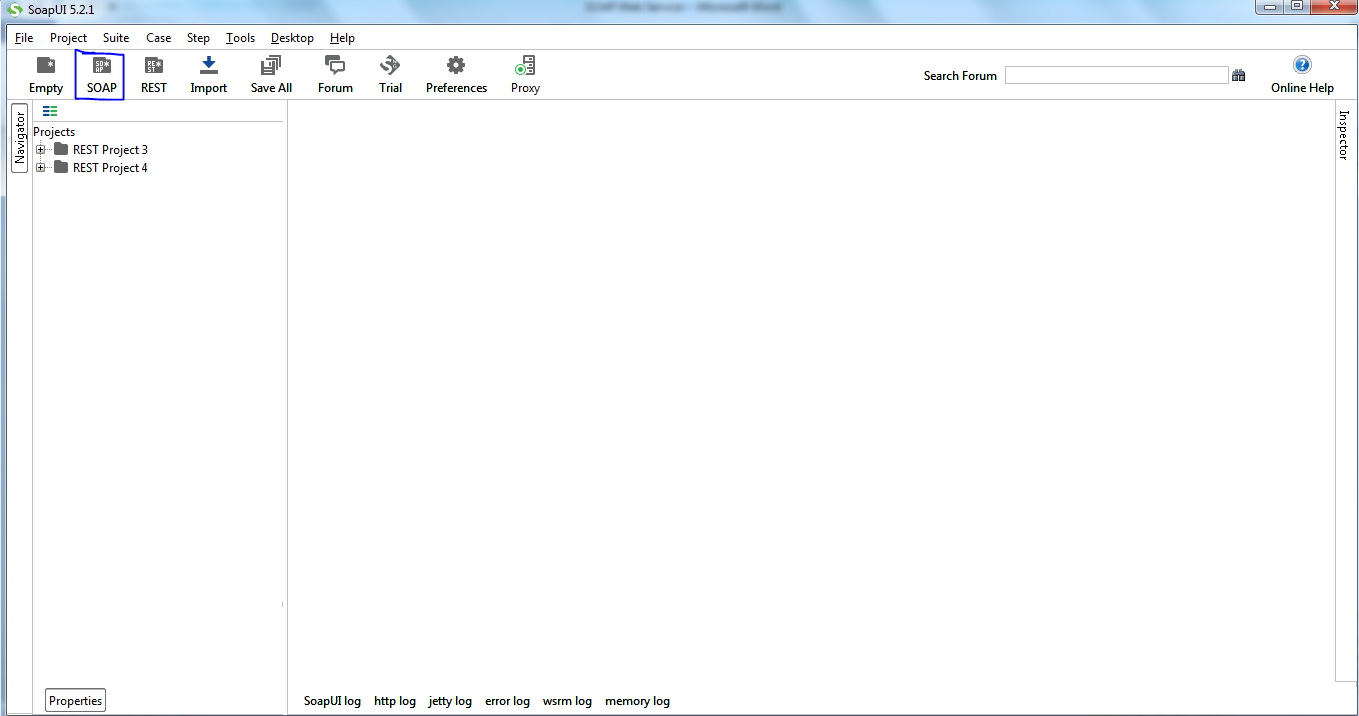
You will see the below window after the project is executed

****

**Step 2:**

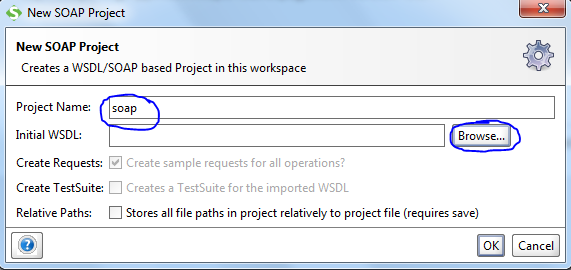
To view the page you have to use the UI. Here I have used the SoapUI.

Download the SoapUI from the browser and install it.



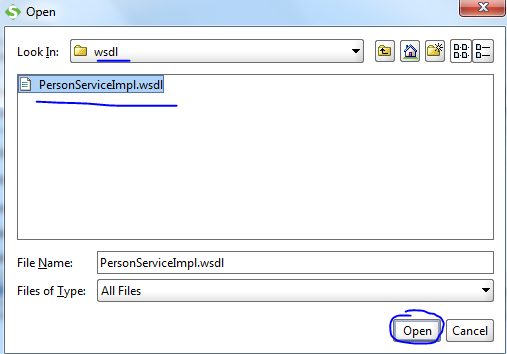
**Step 3:**

Give the Project name and open the WSDL for SOAPExample by browse.



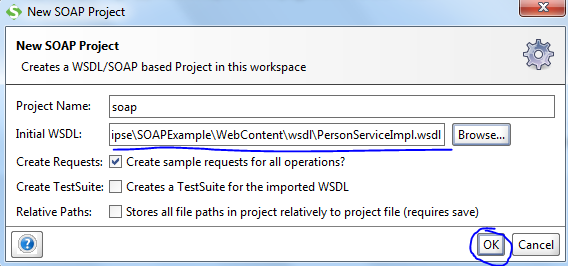
**Step 4:**

Select the required WSDL file and open.



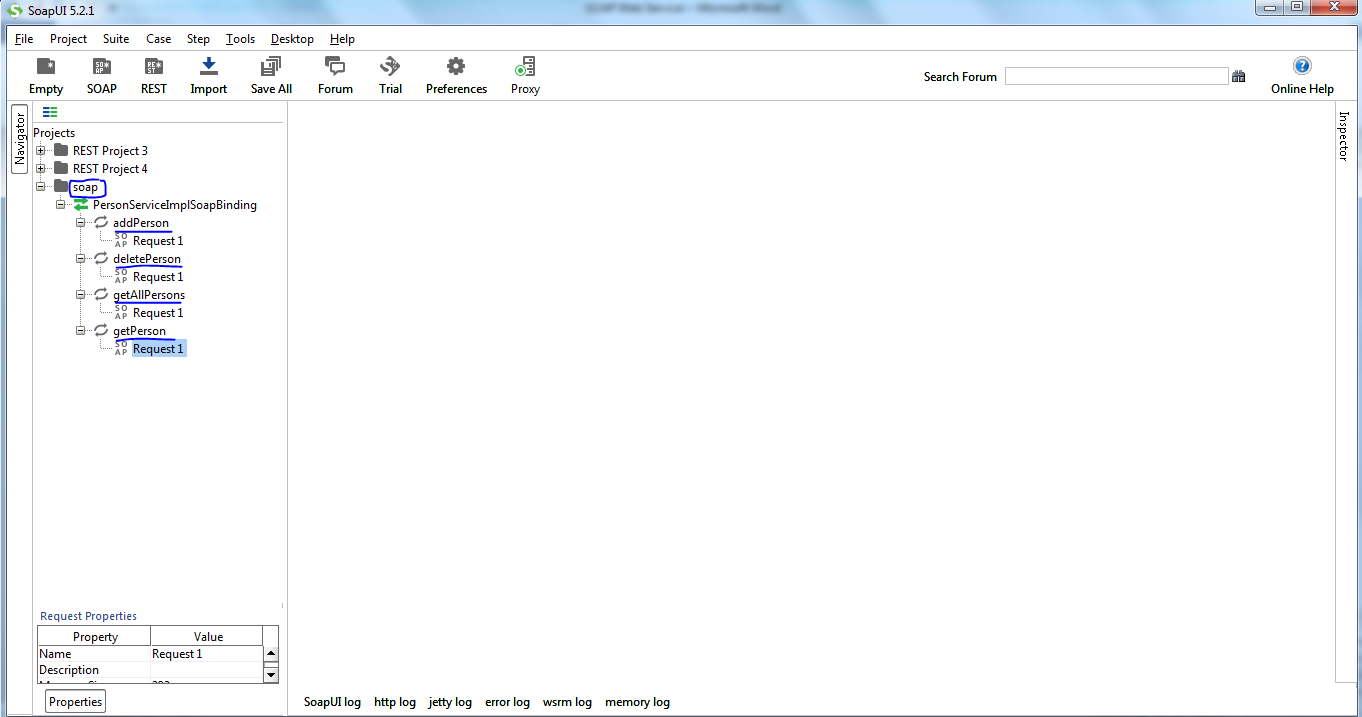
**Step 5:**

Click on OK.



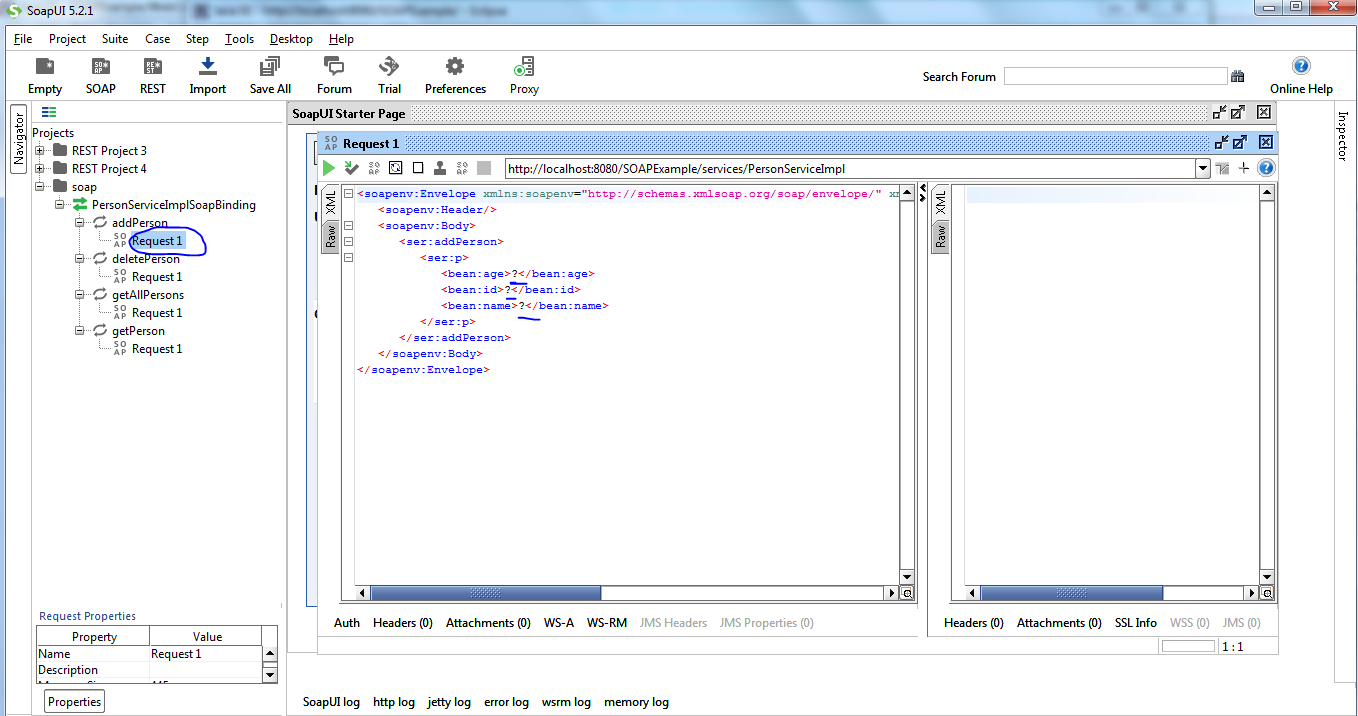
**Step 6:**

Now the Soap project is created with the methods.

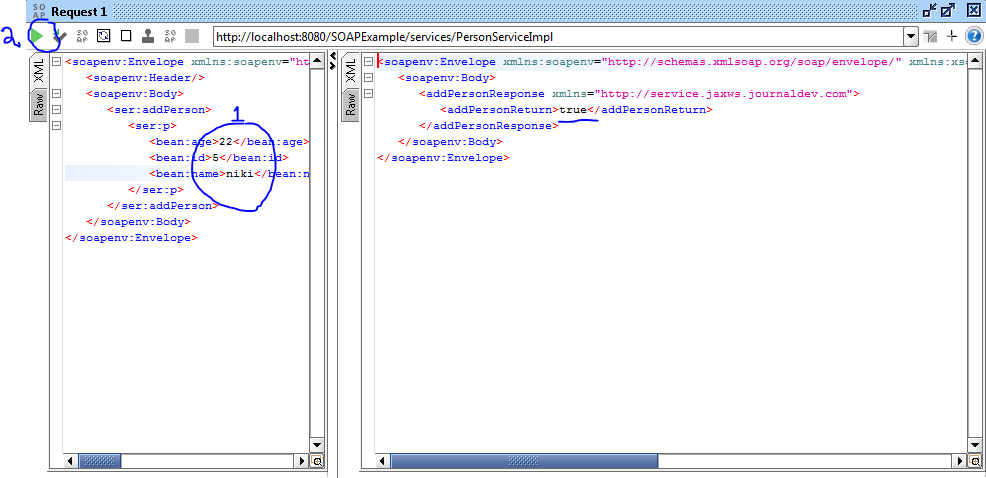


**Step 7:**

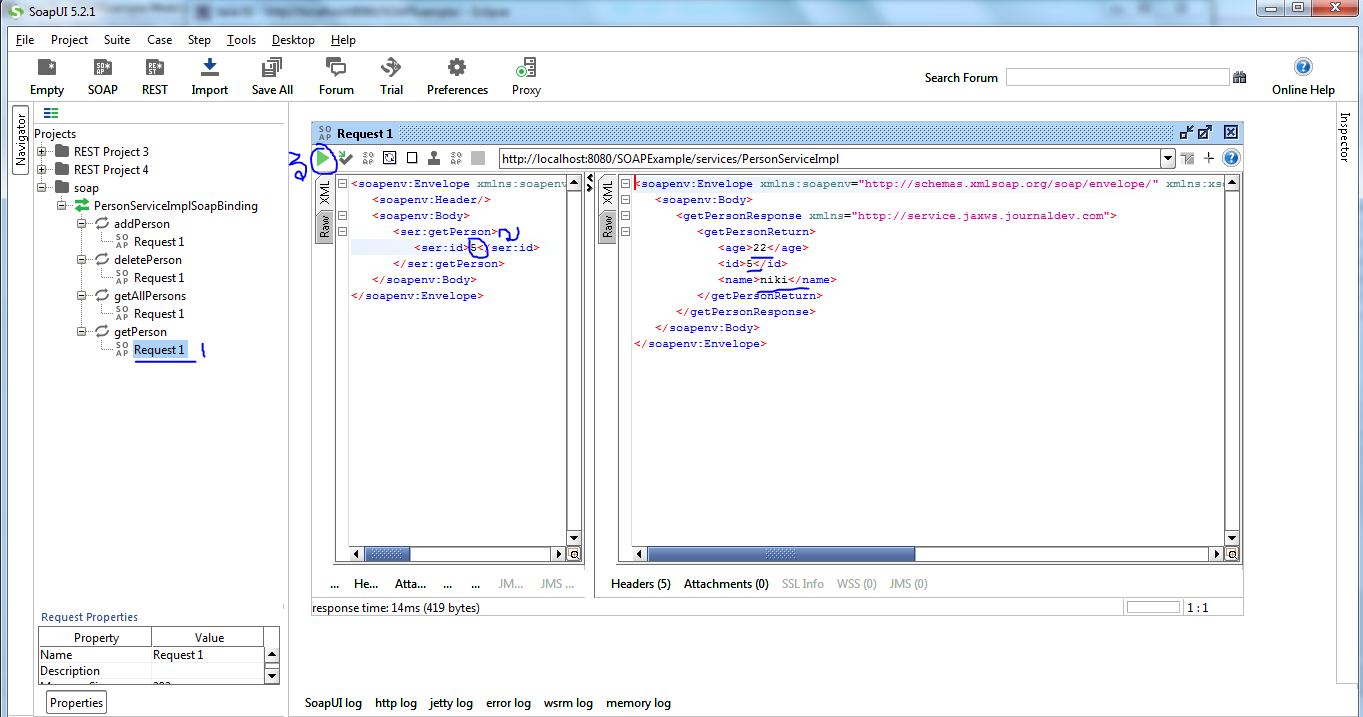
Click on Request1 from addPerson and you will see the below window.



Enter the data and execute now the data is added.so the <addPersonReturns> returns True.



Now to Check whether the data is been added. Click on request1 fron getPerson as shown below and type the id.



So the person is added.

This is the way to execute the Simple SOAP program.

# 4. Reference and Sample.

For Client – server example refer the below link

<http://www.journaldev.com/9131/java-soap-web-service-example-tutorial-using-eclipse>