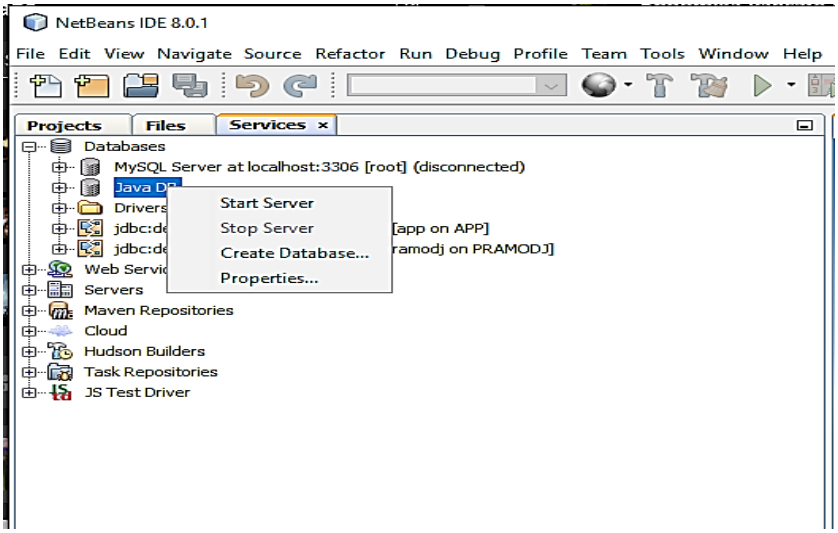
Practical:2

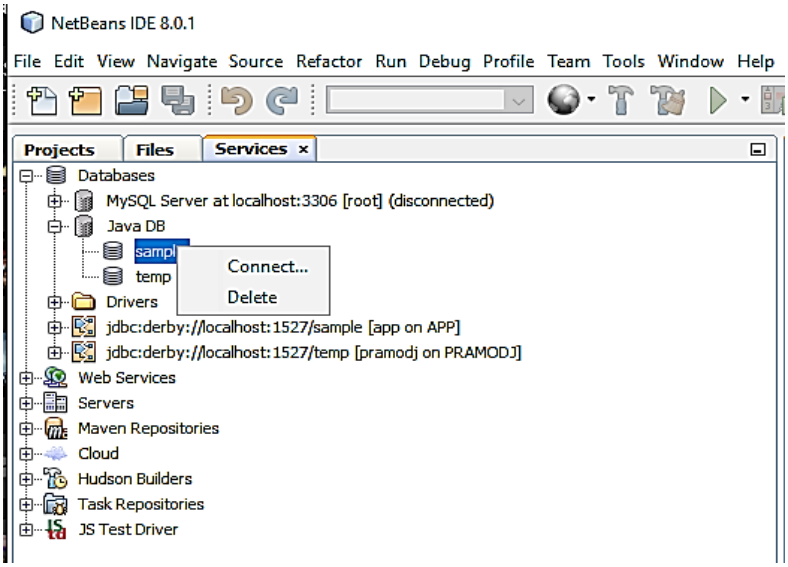
Aim: Write a program to implement the operation and can receive requests and will return a response in two ways. a) One - Way operation b) Request –Response.

Step 1. Click on Window menu and click on Projects, Files & Services to open it.

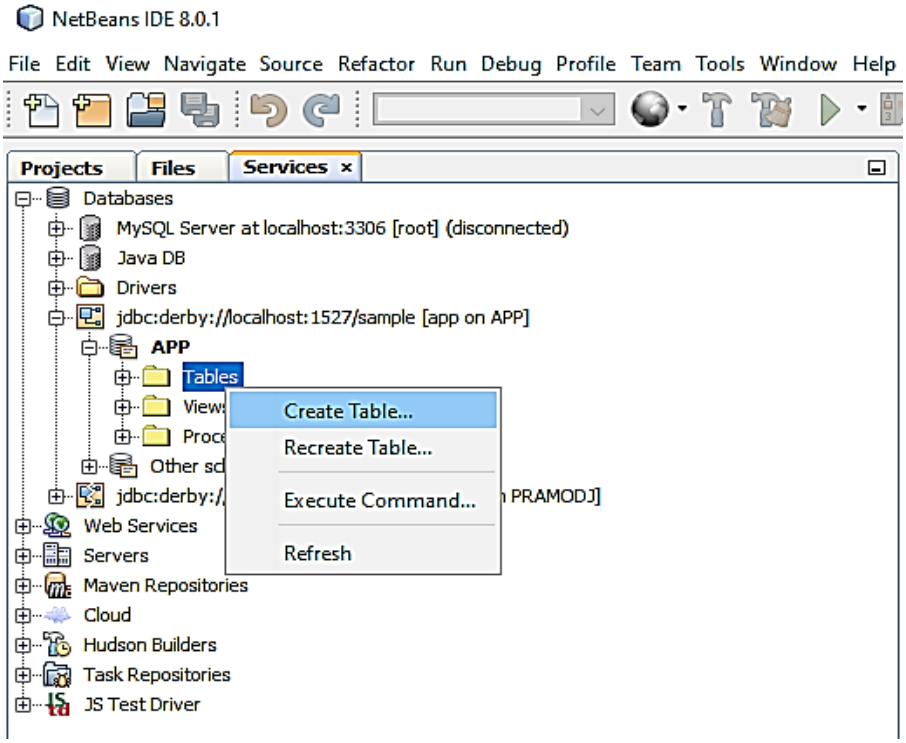
Step 2. Go to the services in palate. Right click on Java DB & then click on start server to start the server



Step 3. Now expand Java DB and right click on sample and then click on connect to connect the sample database with server. If there is any error to sample then create new database. To create a new database, right click on Java DB and then click on create database. Give the database name, username & password. Click on OK.



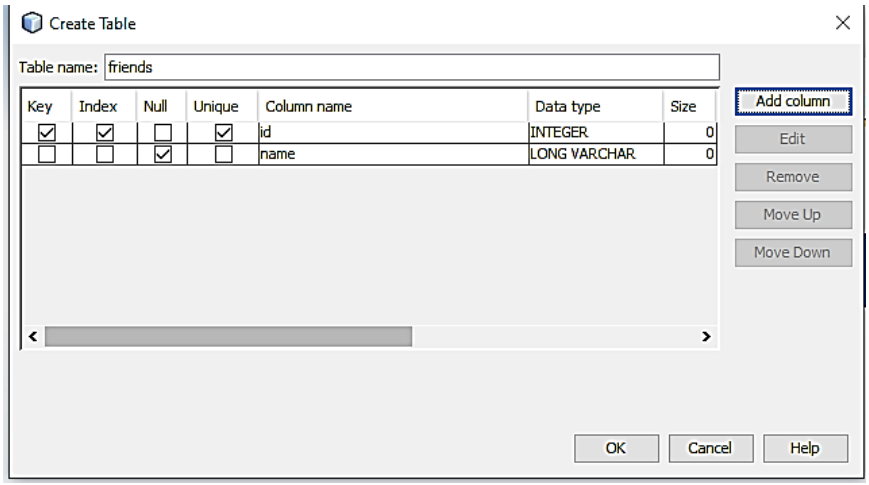
Step 4. Now we are going to create a table in default database sample. Right click on Table -> Create Table.



Step 5. Give table name as FRIENDS.

Step6. Now click on Add column button to add columns in table. Enter details as in below pic and select Primary key. After that click on OK button.

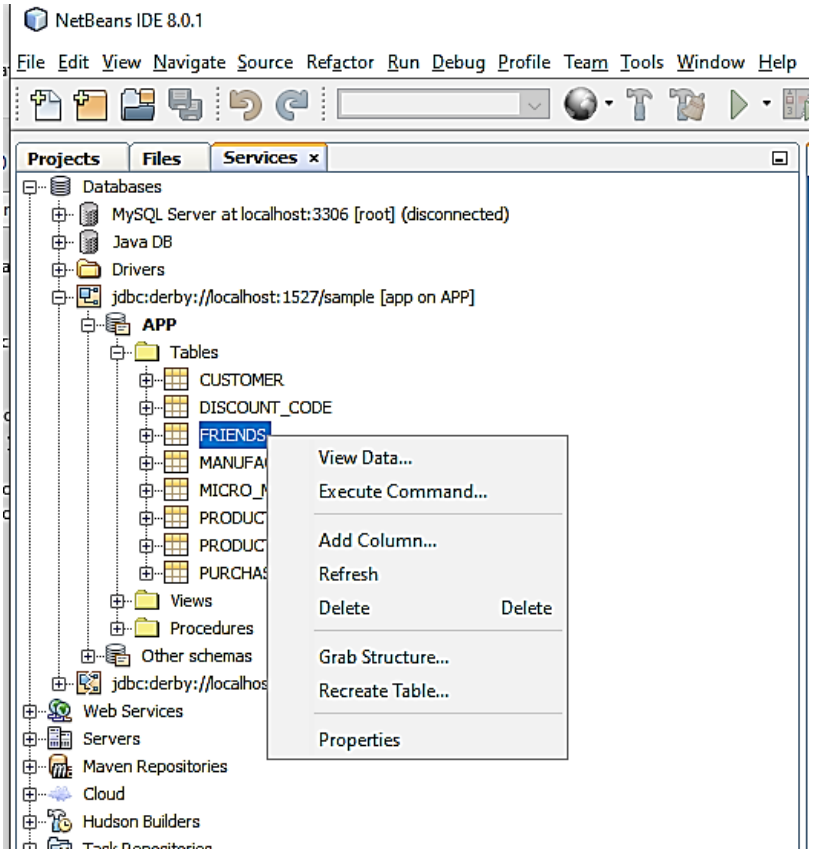
Step 7. Now add second column with following detail. But don't select primary & click on OK button.



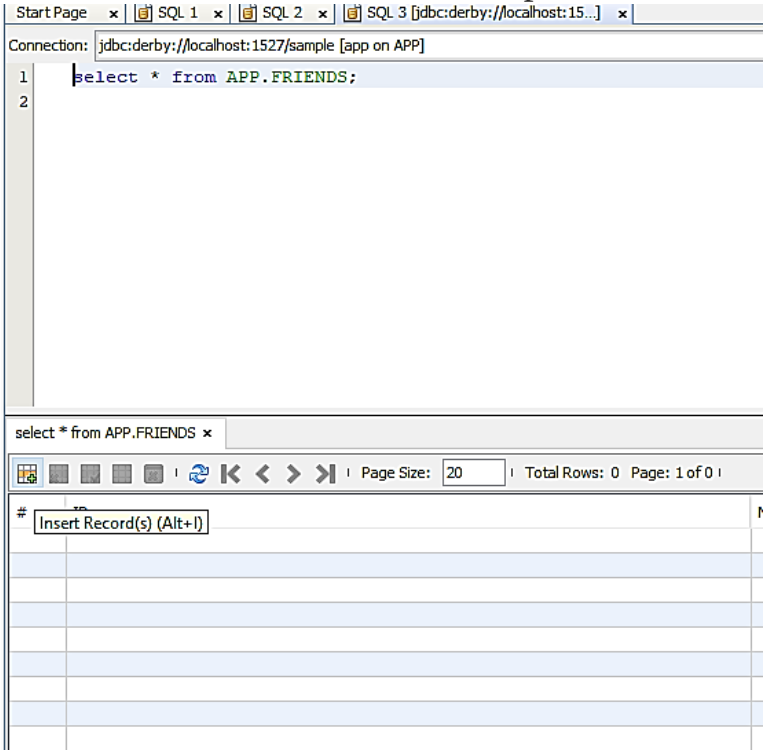
Step 8. Click on OK button.

Step 9. Now you can see a table with name FRIENDS in the table.

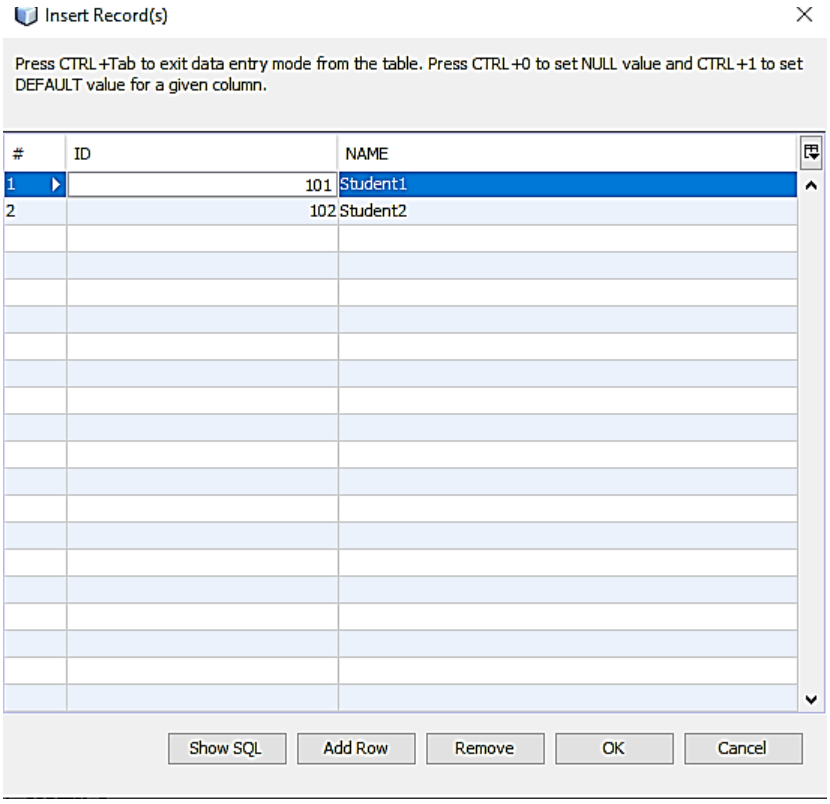
Step 10. Right click on FRIENDS to view & add records into it.



Step 11. Now click on the leftmost icon in second panel to insert some record.

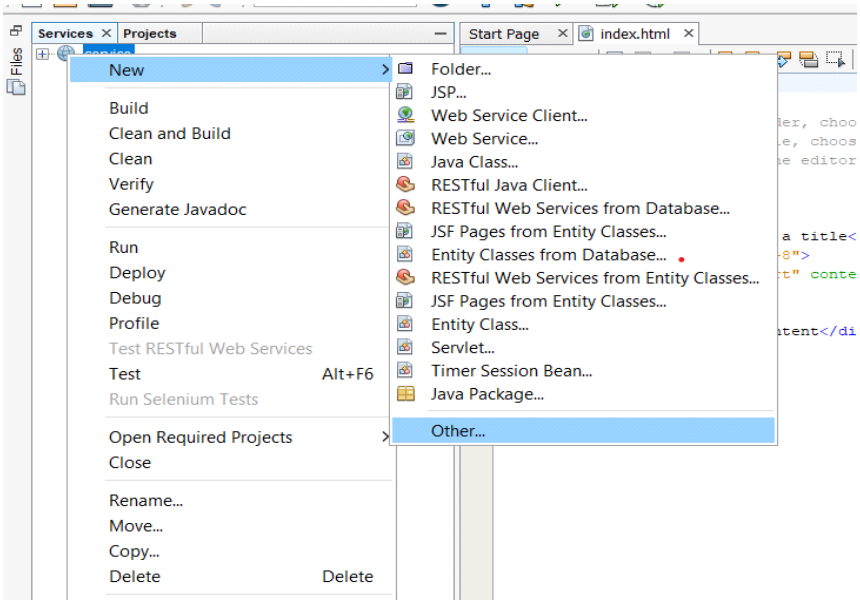


Step 12. Insert a record & then click on Add Row button to insert more record. After that click on OK button to finish.

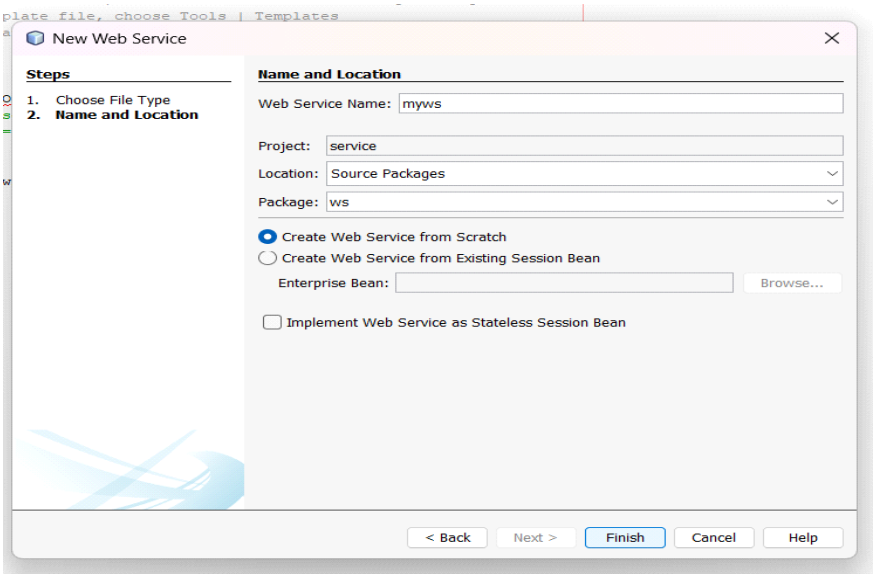


Step 13: create a new project, java web application with a name service.

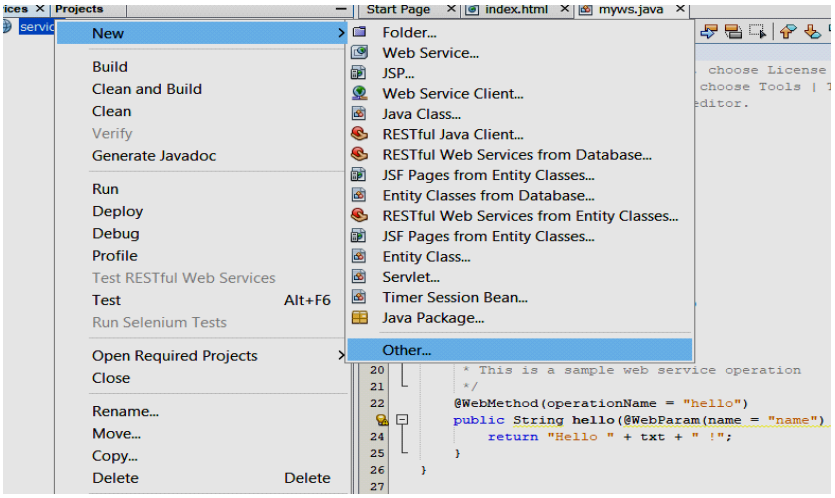
Step 14: right click on project, select web service option.



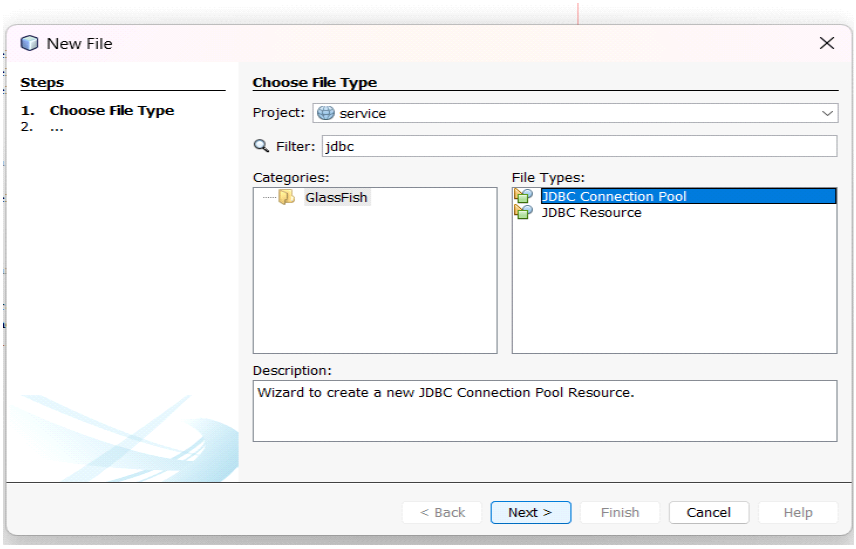
Step 15: write the web service name as "myws" and the package name as "ws"



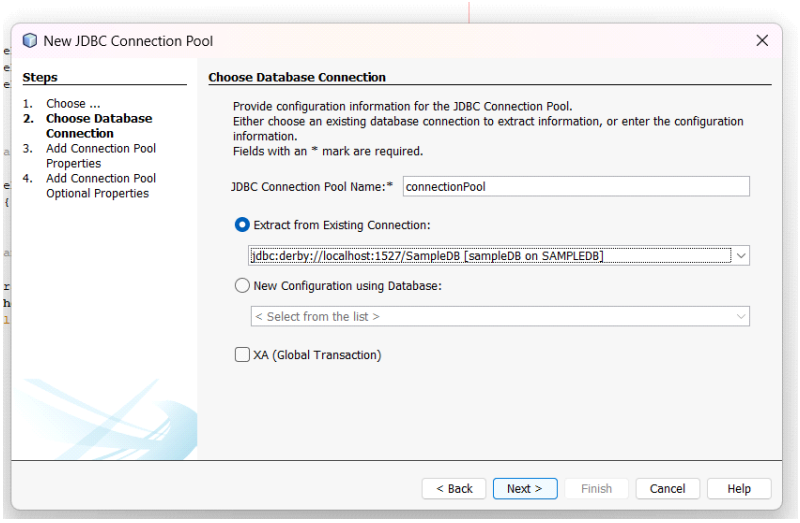
Step 16: Right click on the created project, select new and the select on others



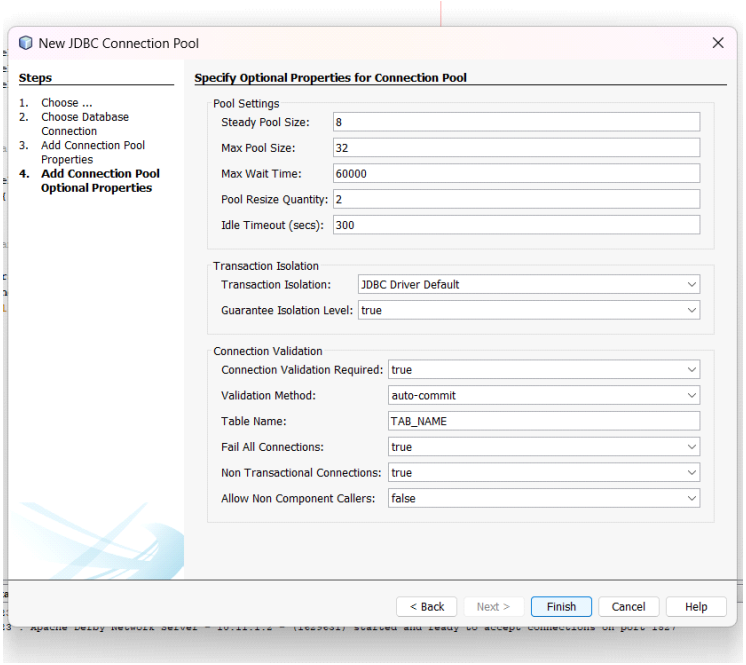
Step 17: Search "jdbc" in the filter search and select "JDBC connection pool" and the select next



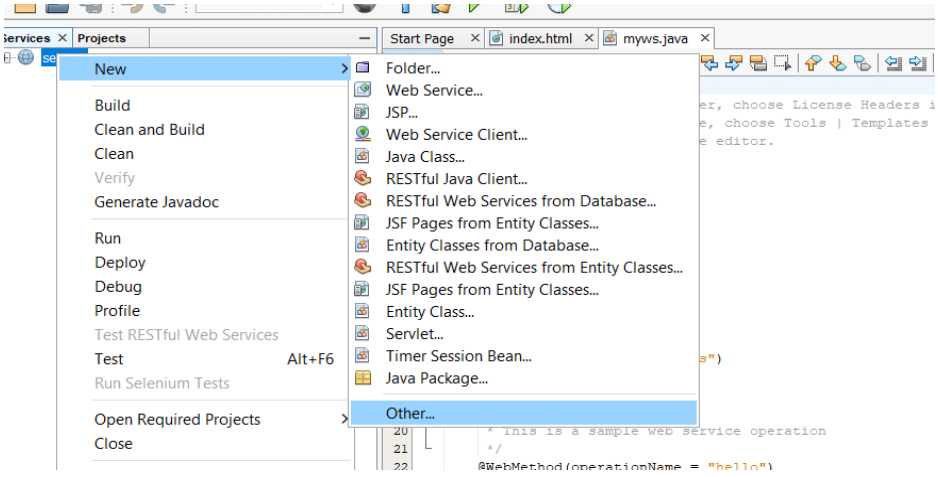
Step 18:click on next



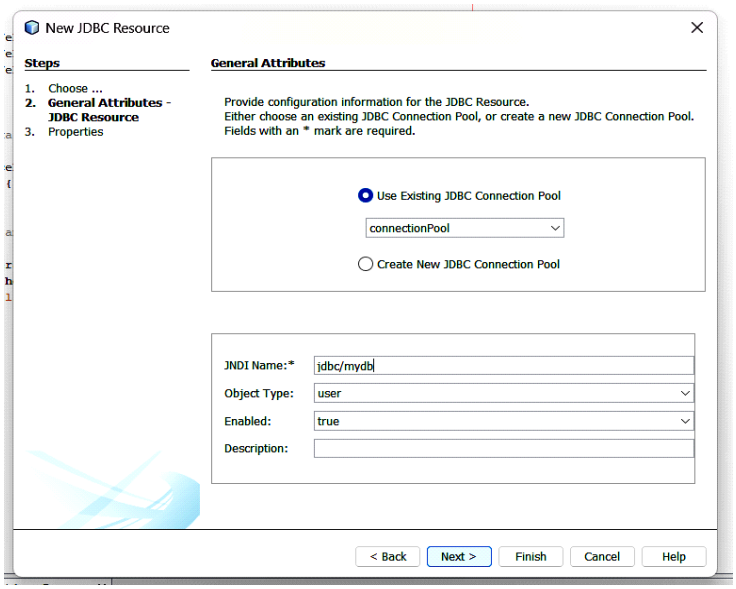
Step 19: Click on Finish.



Step 20: Right click on the project, select new and select others.



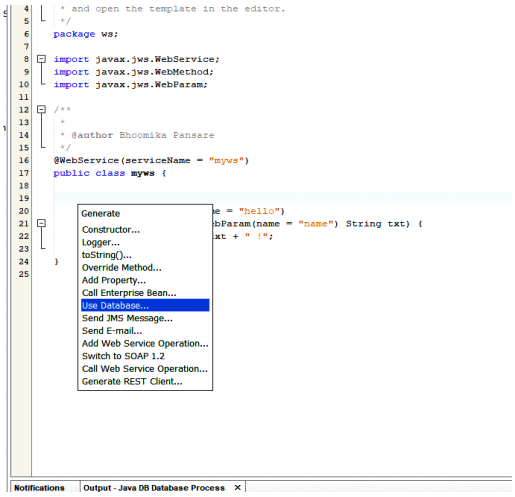
Step 22: select connectionPool from the dropdown menu and change the JNDI name to ‘ jdbc/mydb”



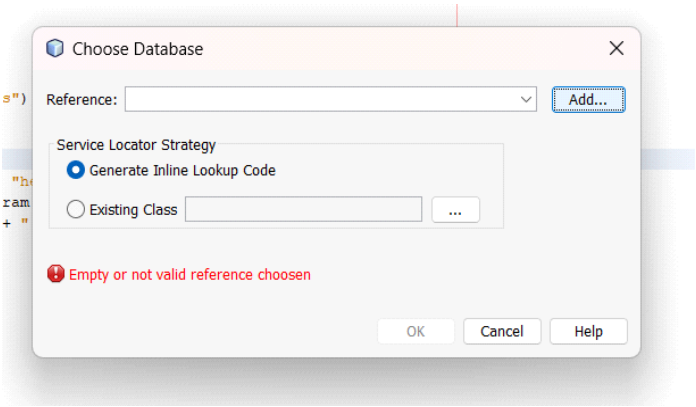
Step 23: click next then click finish.

Step 24: In the file, on line 19 right click, and then click on insert code.

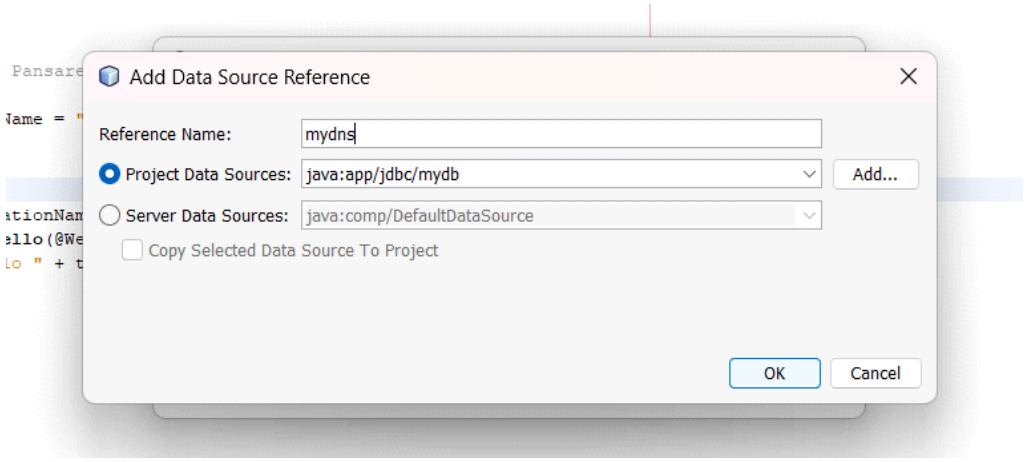
Step 25: Select Use Database



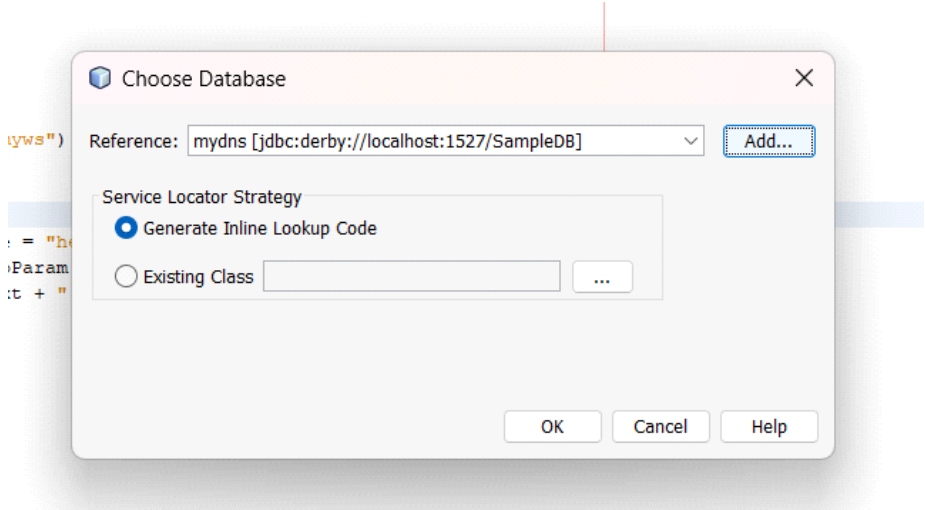
Step 26: click on add



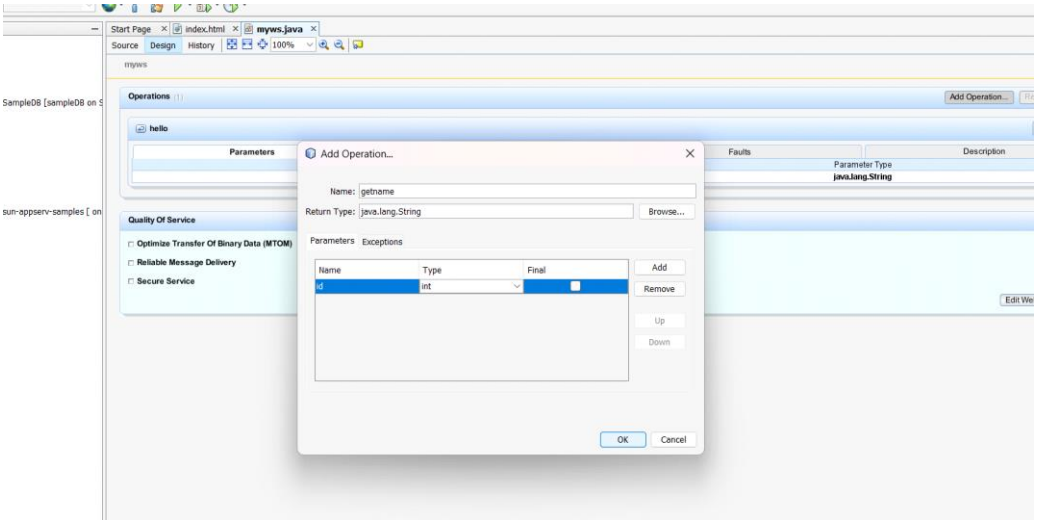
Step 27: write mydns as the reference name and then click ok.



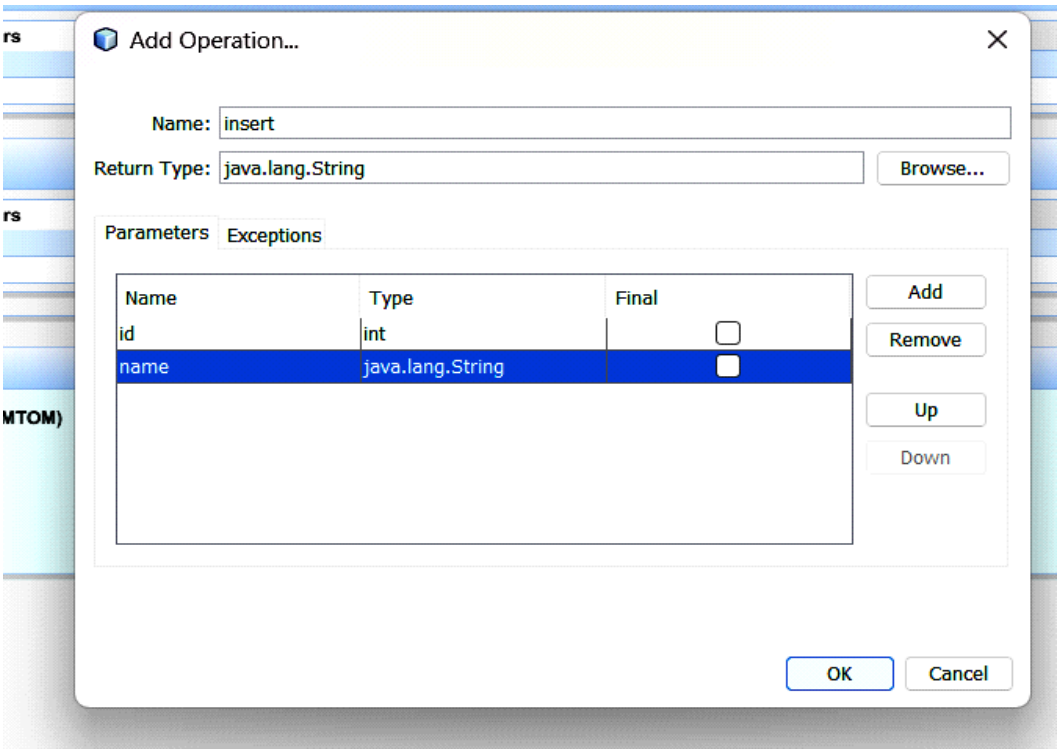
Step 28: The Database reference is added, now click on ok



Step 29: Go to design, click on add operation. Then add the operation name as ‘ getname ‘ and then click on add, and then write "id" in the name and select "int" as the type.



Step 30: Now add another operation the same with the opeartiom name "insert" and the parametr's name as follows.



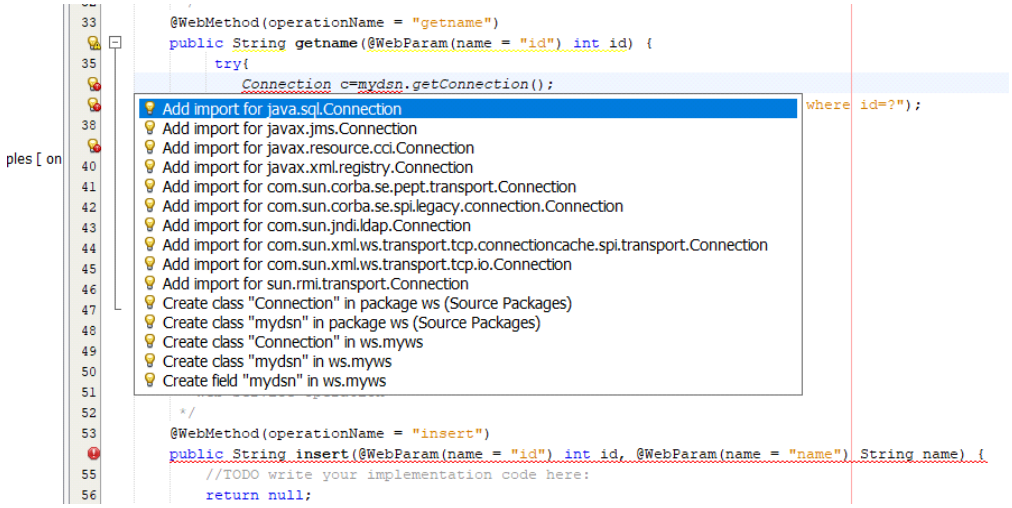
Step 32 : Write this particular code in there

```
try{
        Connection c=mydns.getConnection();
        PreparedStatement ps=c.prepareStatement("select * from friends where id=?");
        ps.setInt(1, id);
        ResultSet r=ps.executeQuery();
        if(r.next())
           return r.getString(2);
        else
           return "No name found";
        }catch(Exception e)
           {return "error";}
   }
```

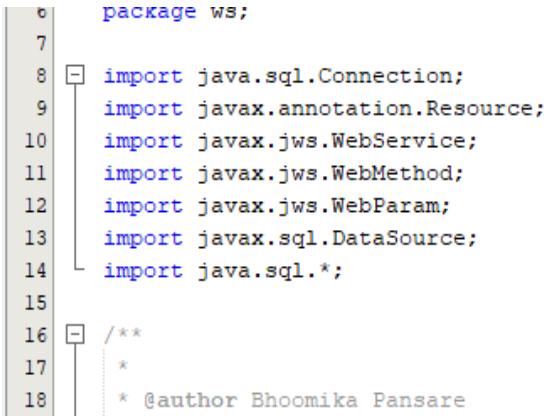
```
    * Web service operation
    */
   @WebMethod(operationName = "getname")
   public String getname(@WebParam(name = "id") int id) {
        try{
           Connection c=mydns.getConnection();
           PreparedStatement ps=c.prepareStatement("select * from friend where id=?");
           ps.setInt(1, id);
           ResultSet r=ps.executeQuery();
           if(r.next())
               return r.getString(2);
           else
               return "No name found";
           }catch(Exception e)
               {return "error";}

           }
   }
```

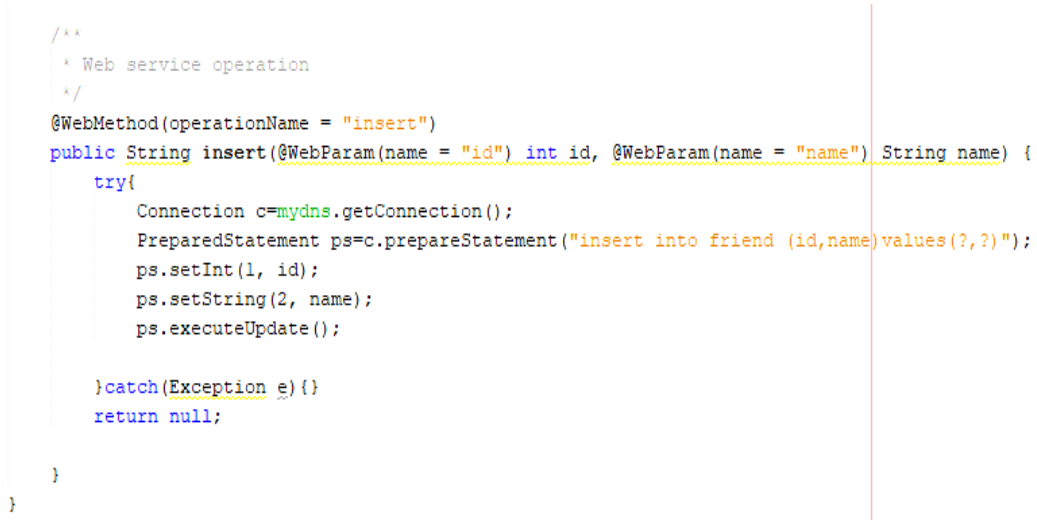
Step 33: for the errors you find click on them and select the first option.



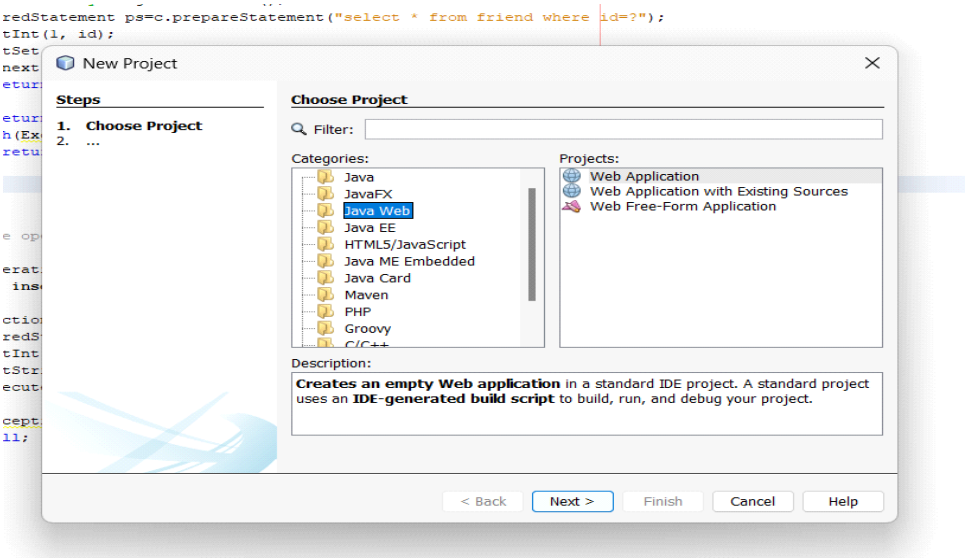
Step 34: Also add import java.sql.*; to avoid errors.



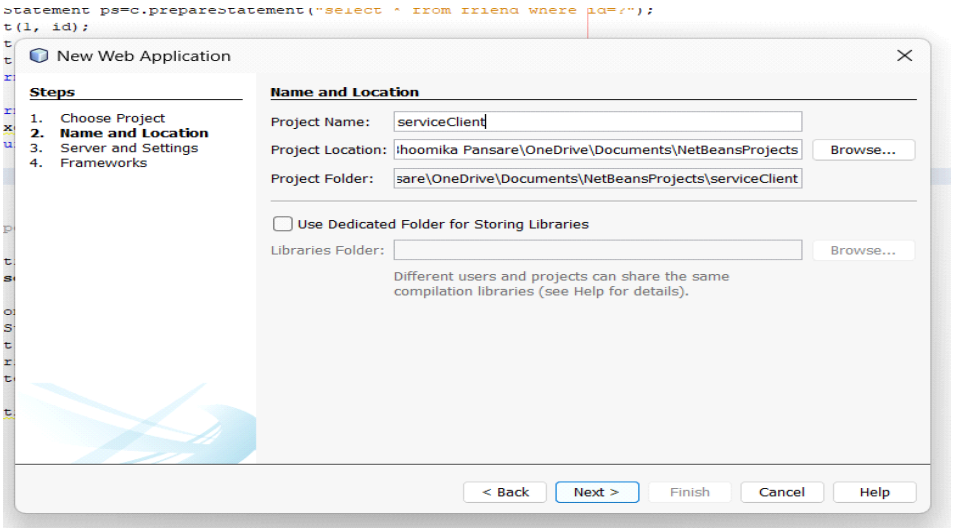
Step 35 : Write the following code for insert operation



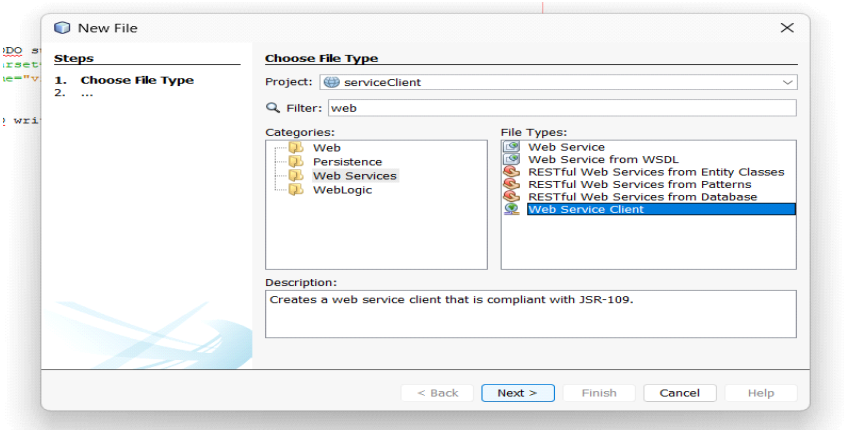
Step 36: Again, create a new web application project.



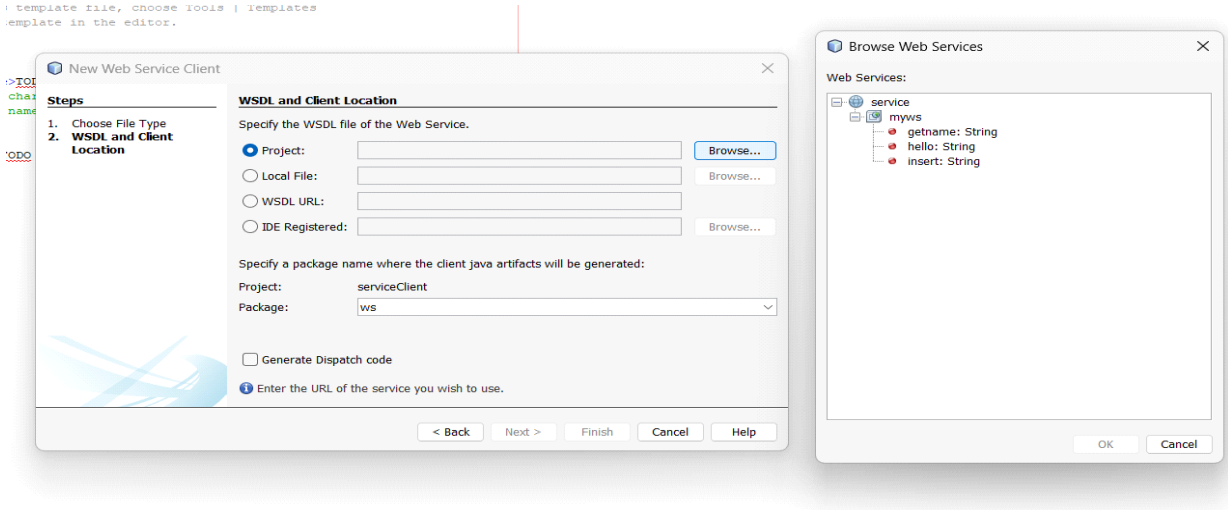
Step 37: wite the project name as "serviceClient".



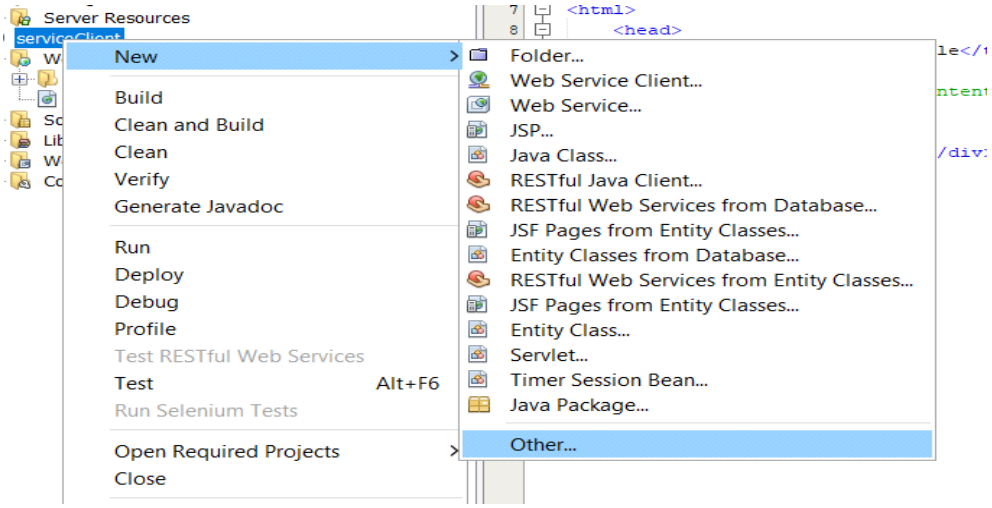
Step 38: Right click on created project, click on new and others, and select web service client. Then click on next.



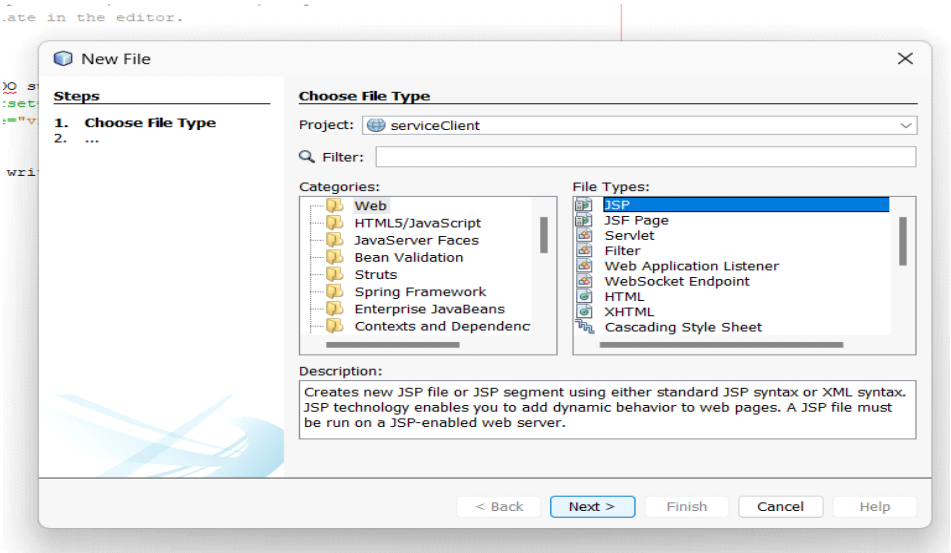
Step 39 : Click on Browse, open the service and select "getname :String " and then click ok.



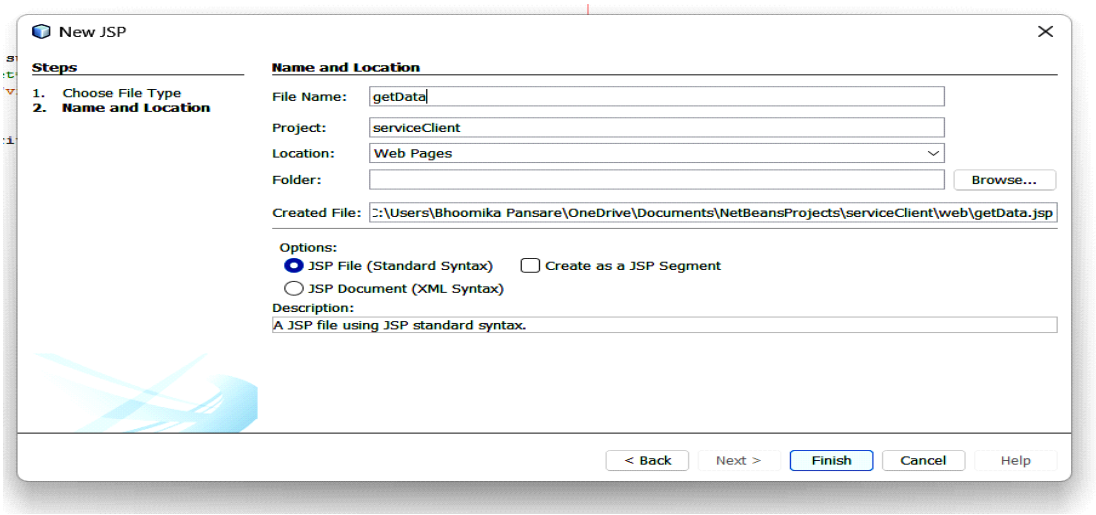
Step 40 : Right click on project serviceClient, select new, click on other.



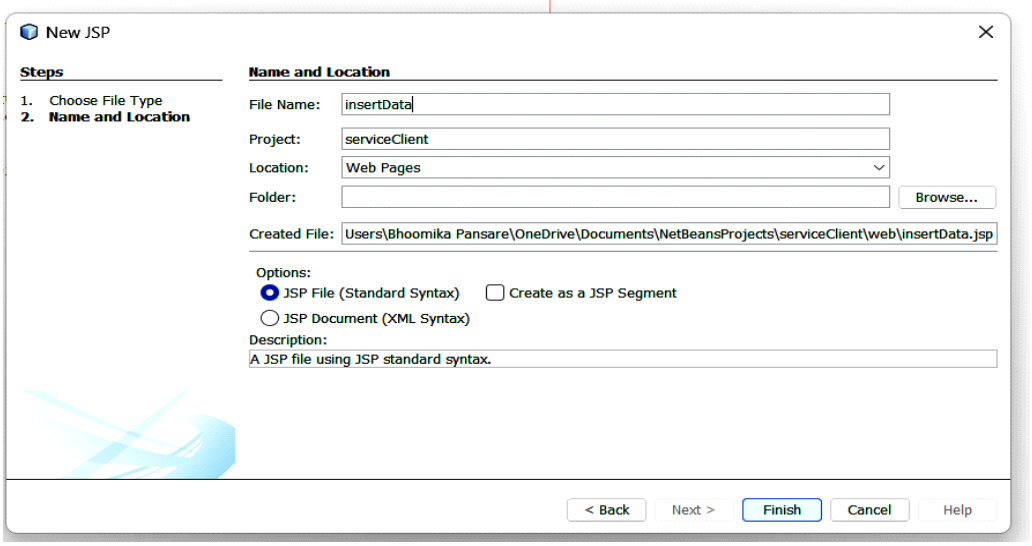
Step 41: Select JSP and then click on next.



Step 42: Write the file name as "getData" and then click on finish.



Step 43: follow the same step for another JSP file creation for the name "insertData".

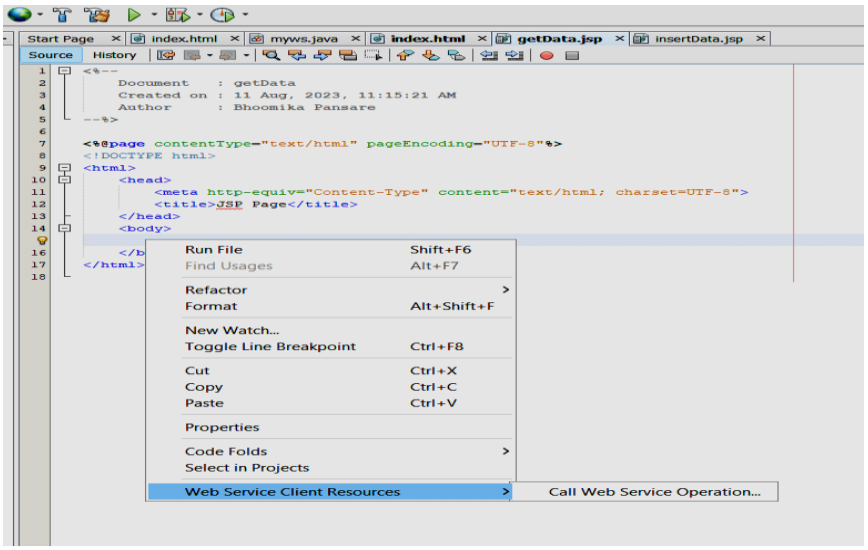


Step 44: Type the following code in index.html page under serviceClient project.

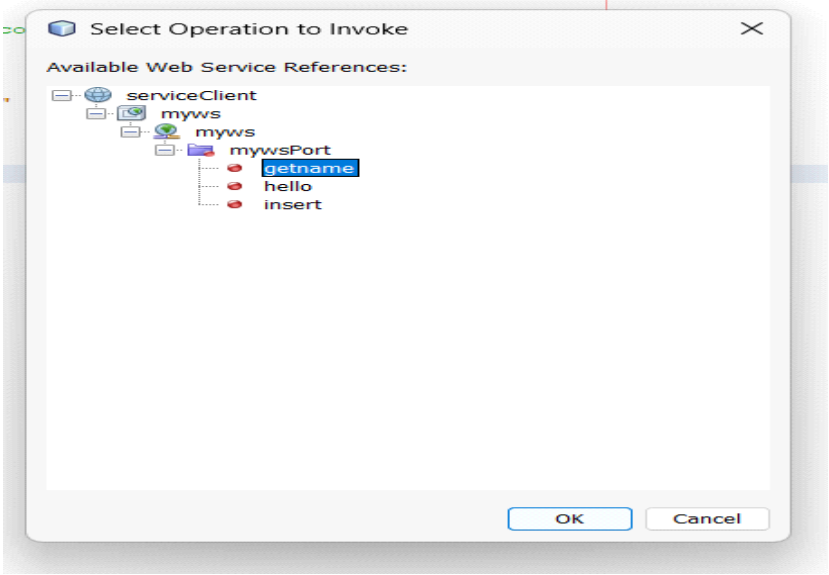
```
<form>
        <h2>One way operation</h2>
        <input type="text" name="txt1" placeholder="EnterId"><br>
        <input type="text" name="txt2" placeholder="EnterName"><br>
        <input type="submit" formaction="insertData.jsp" value="insert"><br>
        <h1>---------------------------------</h1>
        <h2> Request -Response operation </h2>
        <input type="text" name="txt3" placeholder="EnterId"><br>
        <input type="submit" formaction="getData.jsp" value="GET DATA"><br>
    </form>
```

```
-->
<html>
    <head>
        <title>TODO supply a title</title>
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
    </head>
    <body>
        <form>
            <h2>One way operation</h2>
            <input type="text" name="txt1" placeholder="EnterId"><br>
            <input type="text" name="txt2" placeholder="EnterName"><br>
            <input type="submit" formaction="insertData.jsp" value="insert"><br>
            <h1>-----------------------------------</h1>
            <h2> Request -Response operation </h2>
            <input type="text" name="txt3" placeholder="EnterId"><br>
            <input type="submit" formaction="getData.jsp" value="GET DATA"><br>
        </form>
    </body>
</html>
```

Step 45: Go to getData.jsp page in serviceClient project. Right click in the body tag and then select web service client resources and click on call web service operation.



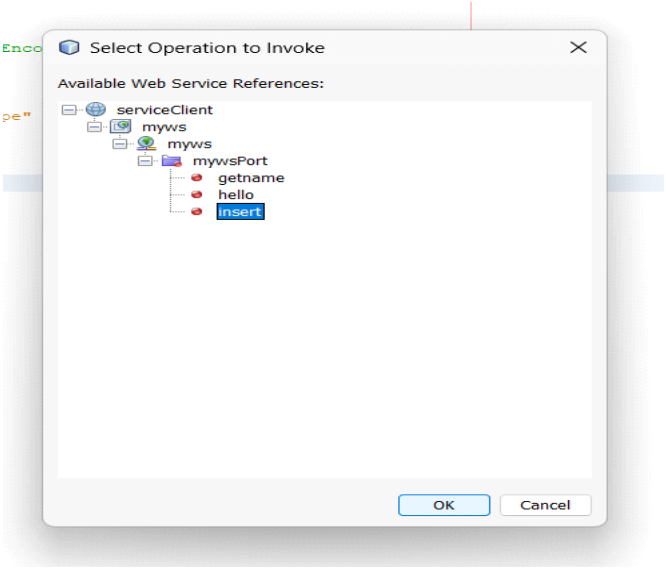
Step 46: A window like this would open up. select "getname" form the mywsport. click on ok.



Step 47 : Make the following changes in the code.

```
11          <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
12          <title>JSP Page</title>
13      </head>
14      <body>
16      <%
17          int id=Integer.parseInt(request.getParameter("txt3"));
18      try {
19          ws.Myws_Service service = new ws.Myws_Service();
20          ws.Myws port = service.getMywsPort();
21           // TODO initialize WS operation arguments here
22          // TODO process result here
23          java.lang.String result = port.getname(id);
24          out.println("NAME = "+result);
25      } catch (Exception ex) {
26          // TODO handle custom exceptions here
27      }
28      %>
29      <%-- end web service invocation --%><hr/>
30      </body>
31  </html>
32
```

Step 48 : Follow the same steps for insertData.jsp page in serviceClient. here select insert and then click on ok.



STEP 49 : Do the following changes in the code.

```
16      <%-- start web service invocation --%><hr/>
17      <%
18          int id=Integer.parseInt(request.getParameter("txt1"));
19          String name=request.getParameter("txt2");
20      try {
21          ws.Myws_Service service = new ws.Myws_Service();
22          ws.Myws port = service.getMywsPort();
23           // TODO initialize WS operation arguments here
24
25          // TODO process result here
26          port.insert(id, name);
27          out.println("DATA INSERTED");
28      } catch (Exception ex) {
29          // TODO handle custom exceptions here
30      }
31      %>
32      <%-- end web service invocation --%><hr/>
33      </body>
34  </html>
35
```

Step 50 :Deploy and Run the **serviceClient** project.