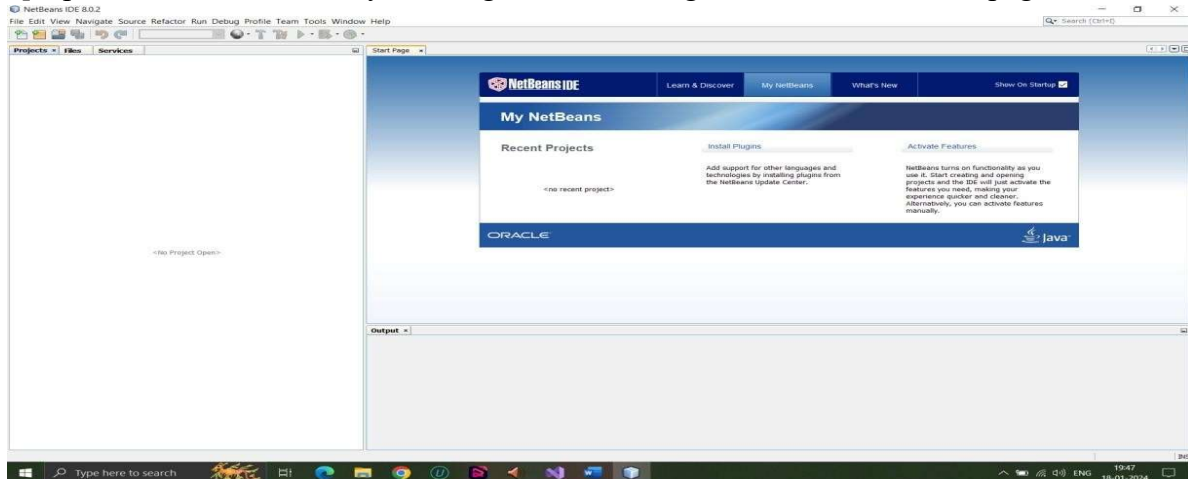


Practical - 2

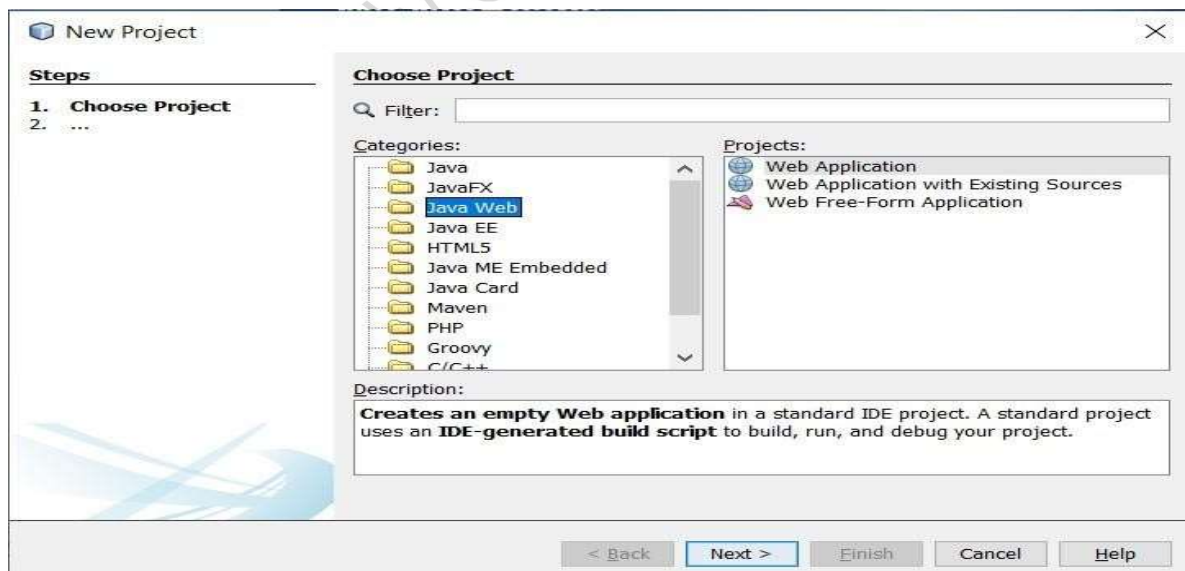
Create a Simple SOAP service.

Steps:

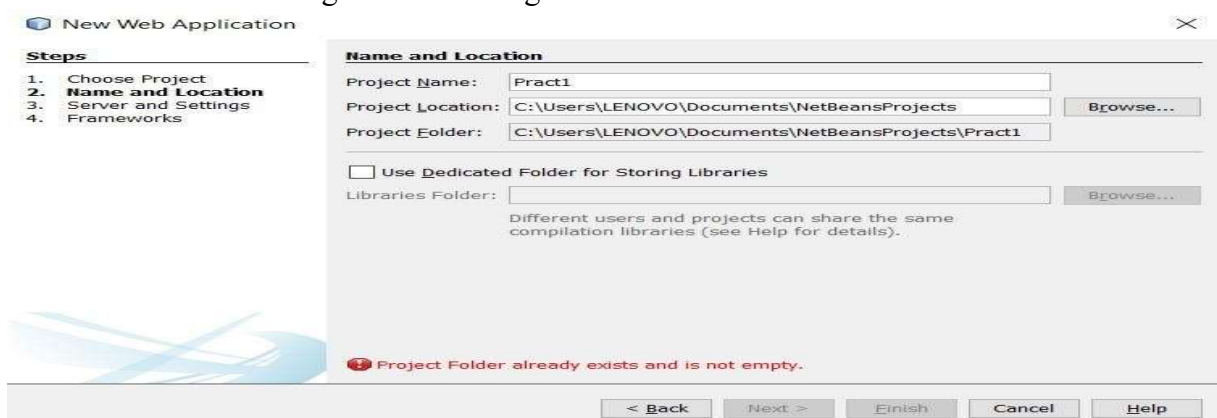
1] Open the NetBeans , and you will get the following screen. Close the startpage.



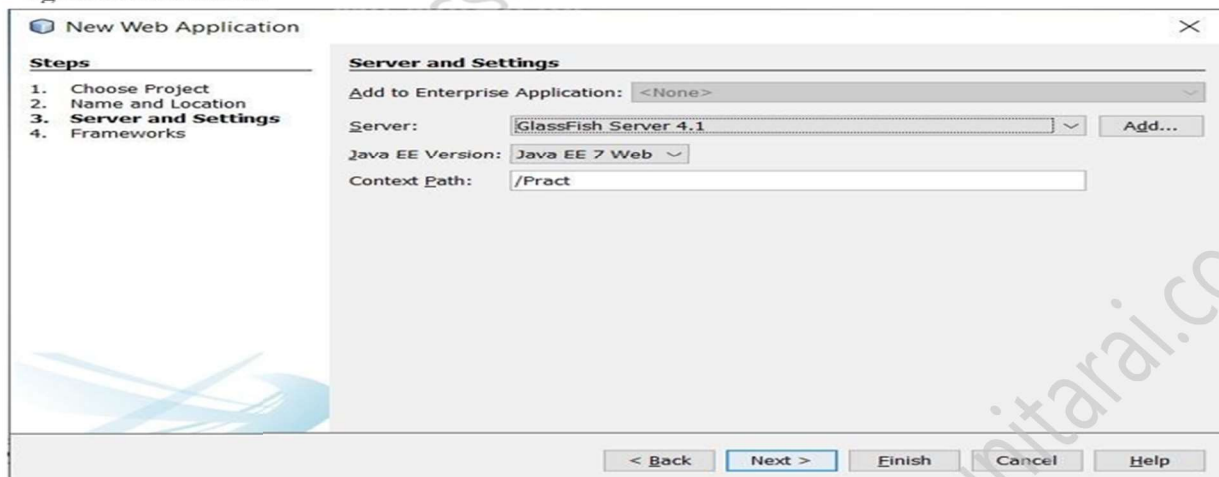
2] Now clickon the file tab and clickon new project you will get the following screen :



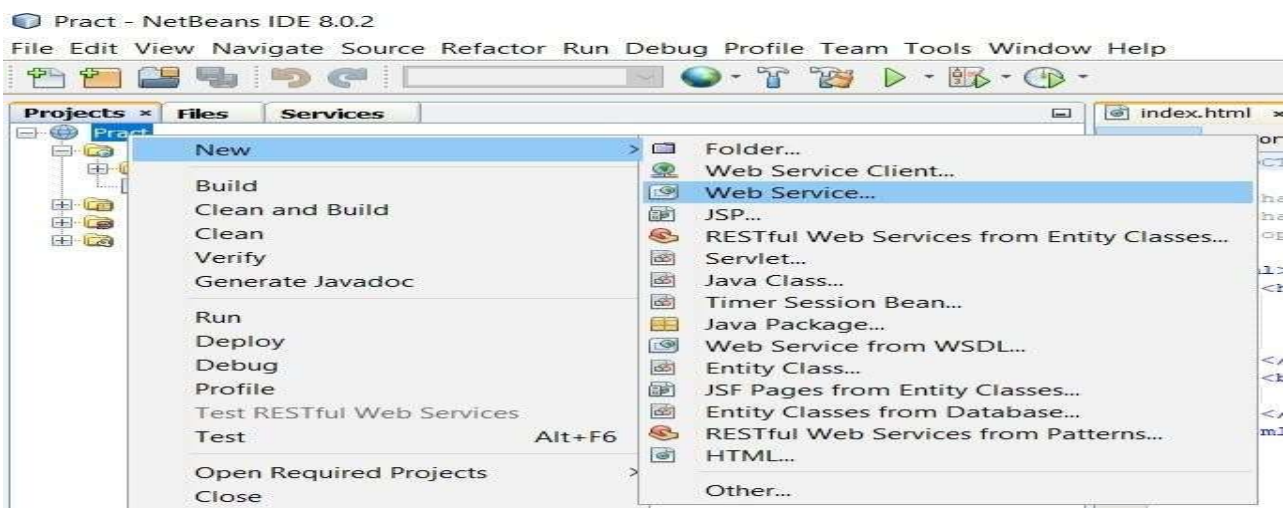
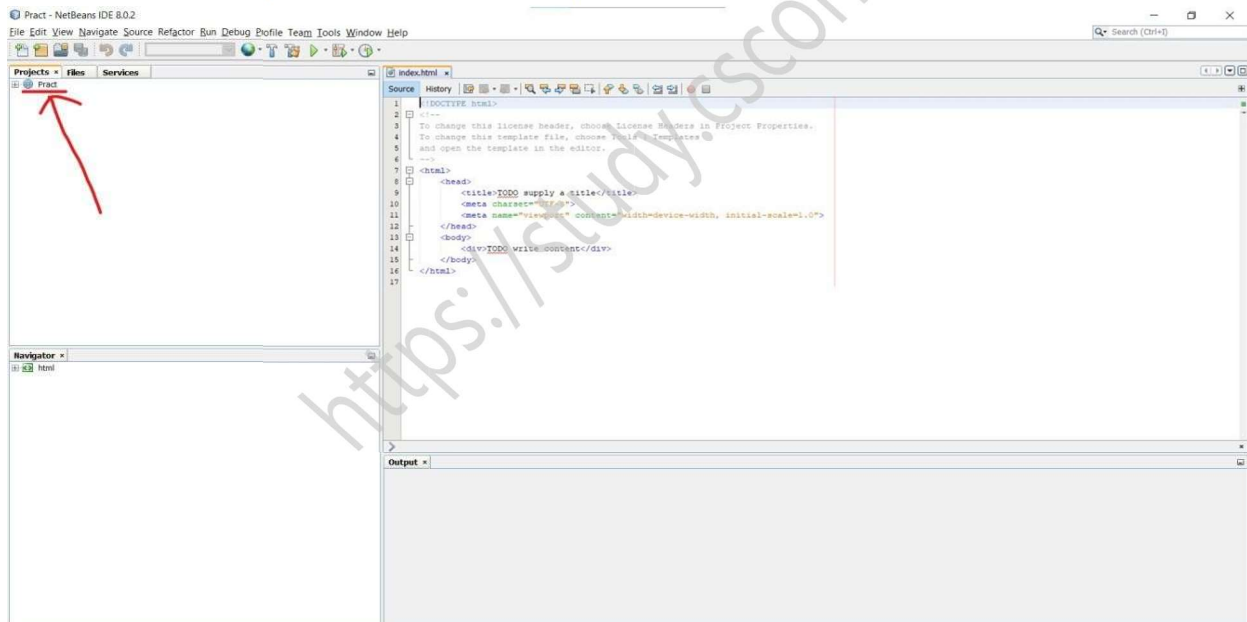
3] In Categories select Java Web and in Projects , Select Web Application .After selecting click onnext . You will get the following window:



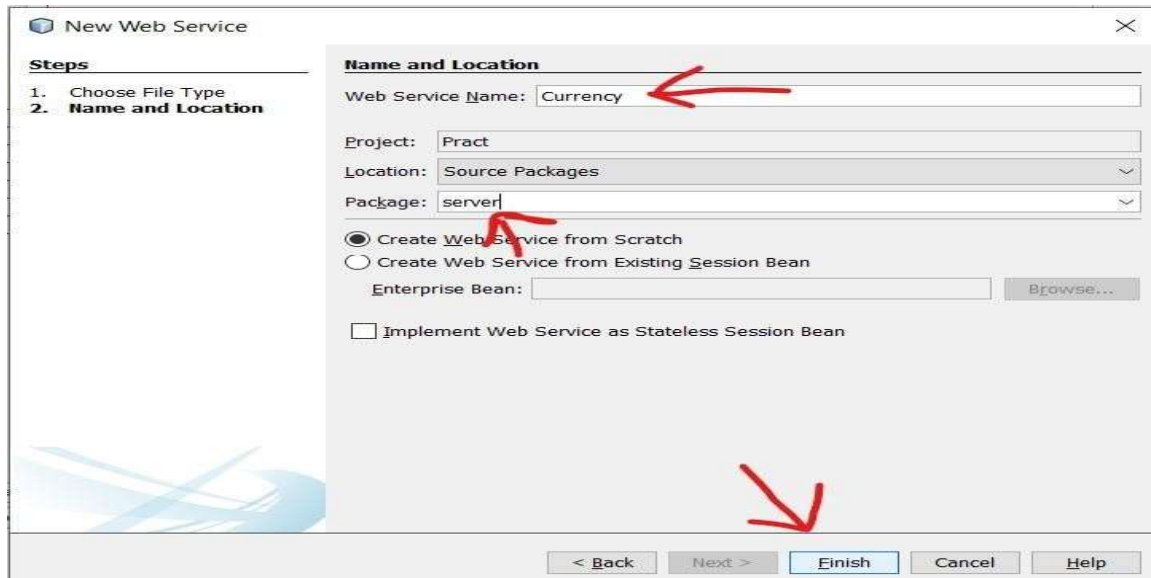
4] Now Give name to the ProjectName, and click on next . you will get the following window then again click on finish:



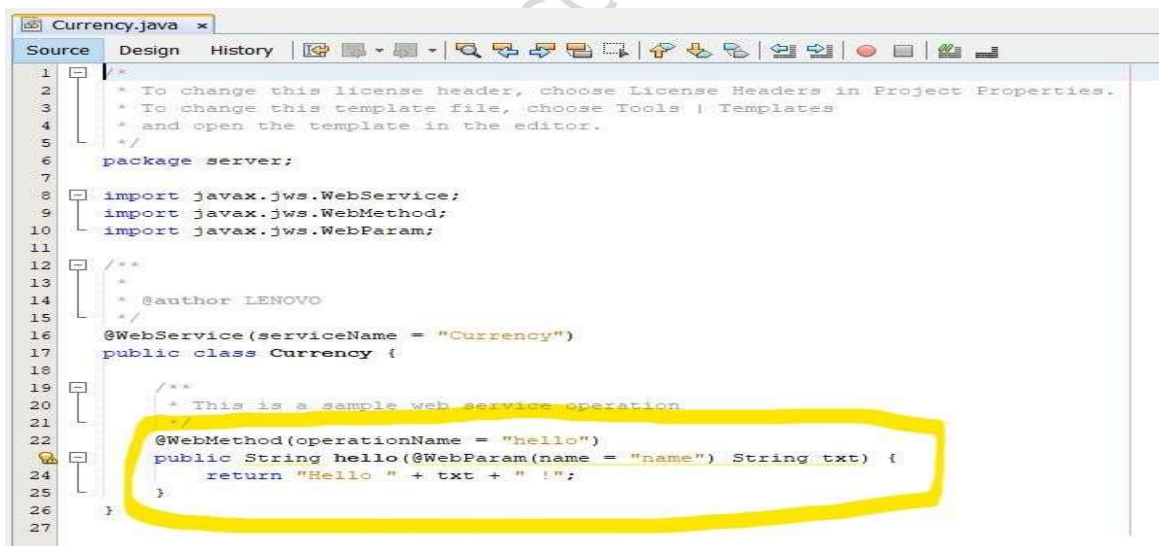
5] You will get the following screen now. Right click on the project and select new and then select Web Service, As shown Below created project appears double click to expand it:



- 7] after clicking Web Service following window should appear , now give name to web service and give package as “server”. As shown in the image:



After clicking finish you should get the following window erase the mentioned code :

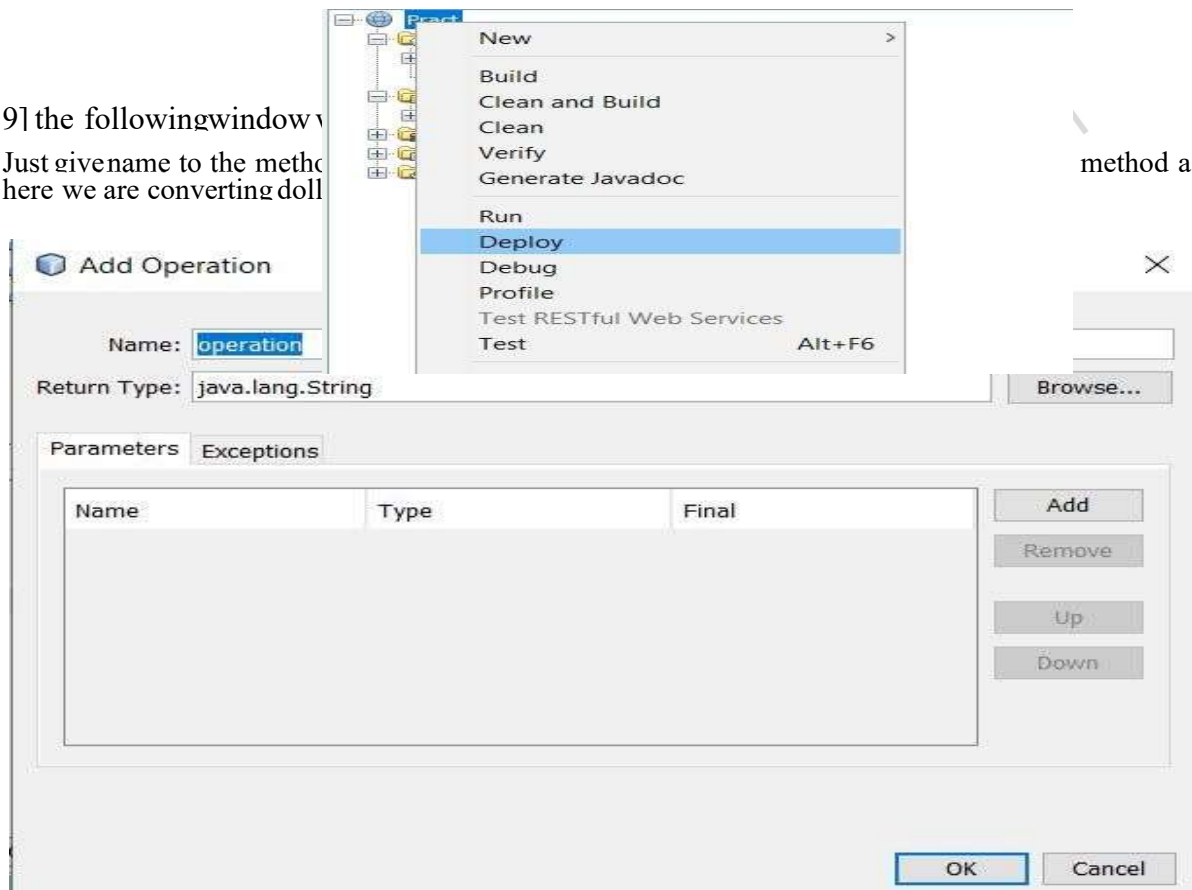


- 8] Now right click any where and click on insert code and select Add Web Service Operation :

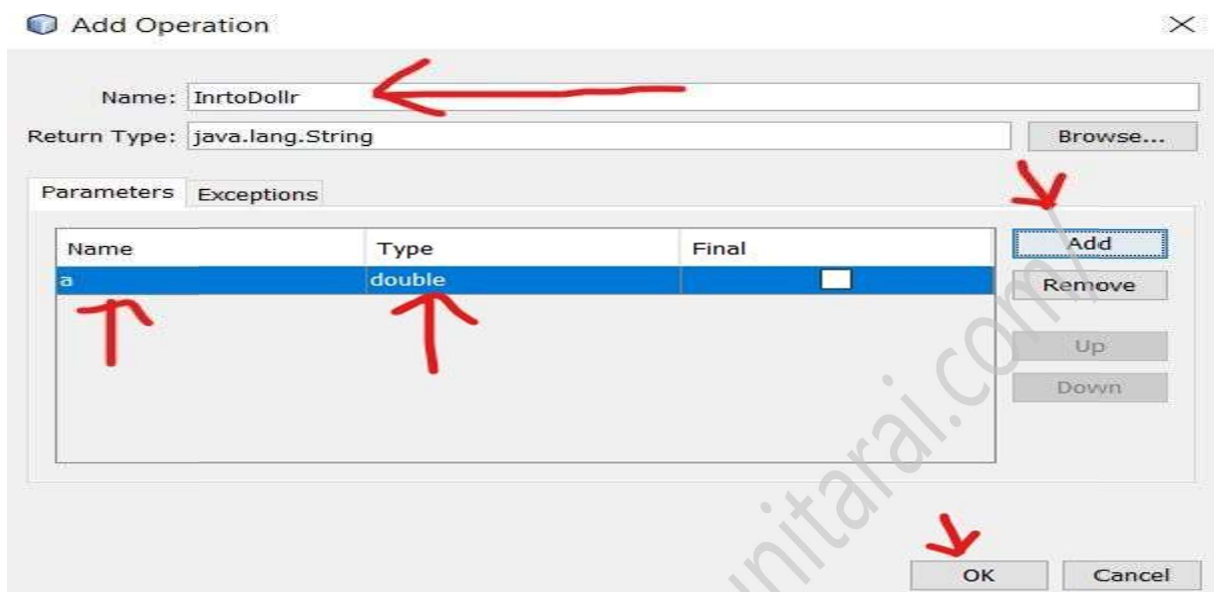


9] the following window
Just give name to the method
here we are converting doll

method as



10] give the name to the variable select data type as double and click on ok as shown below:



11] After clicking ok code will autogenerate, make changes in that code as mentioned below:

```
@WebMethod(operationName = "InrtoDollar")
public String InrtoDollar(@WebParam(name = "a") double a) {
    //TODO write your implementation code here:
    return "The Indian rupees "+a+" in Dollars is "+(a/83.17);
}
```

12] now our web service is ready now right click on project and click on deploy:



13] Now right click on webservice and click on test web service you will get the following output:

CurrencyConvertor 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 java.lang.String server.CurrencyConvertor.inrtoDollar(double)

inrtoDollar

inrtoDollar Method invocation

Method parameter(s)

| Type | Value |
|--------|-------|
| double | 1233 |

Method returned

java.lang.String : "The Indian rupees 1233.0 in Dollars is 14.825057111939401"

SOAP Request

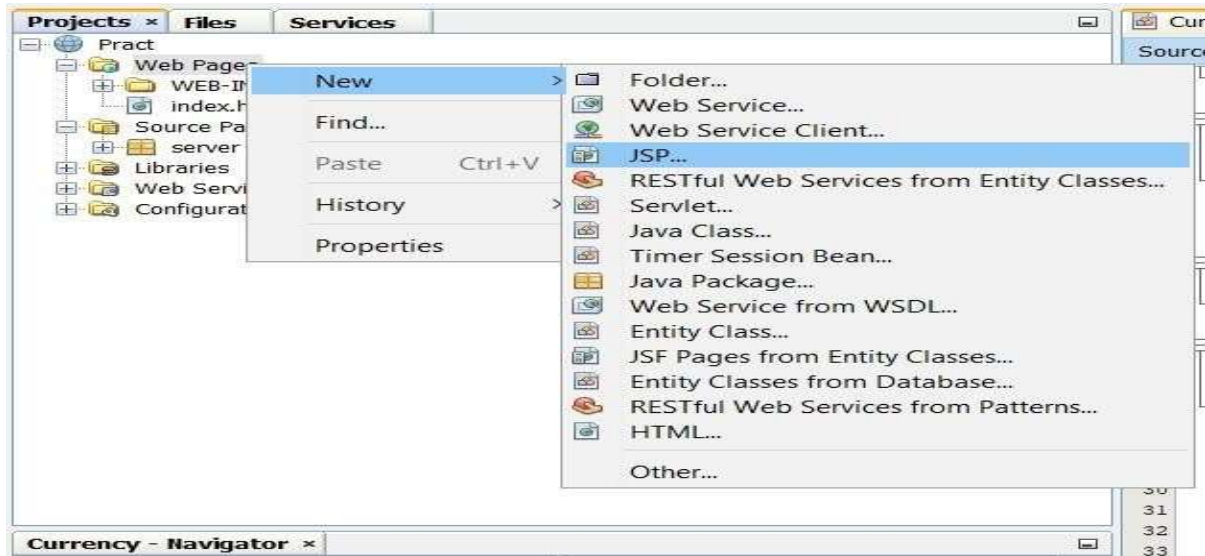
```
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <S:Body>
    <ns2:InrtoDollar xmlns:ns2="http://server/">
      <a>1233.0</a>
    </ns2:InrtoDollar>
  </S:Body>
</S:Envelope>
```

SOAP Response

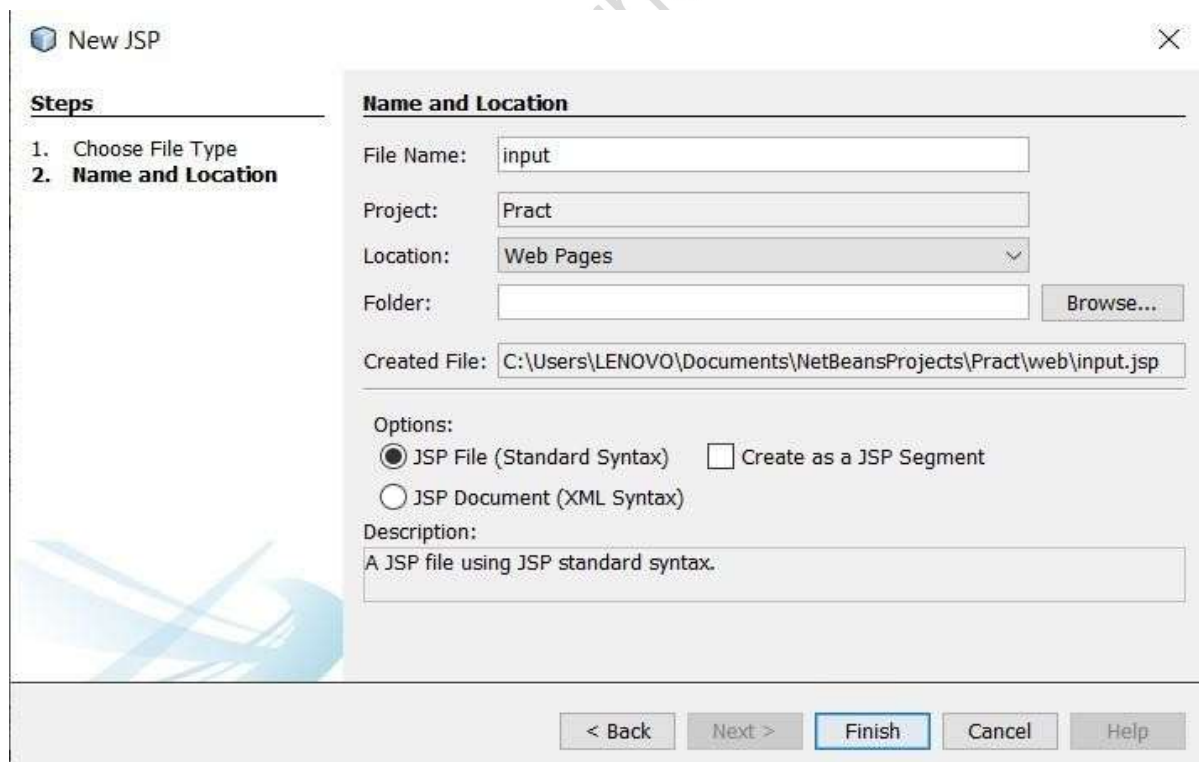
```
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <S:Body>
    <ns2:InrtoDollarResponse xmlns:ns2="http://server/">
      <return>The Indian rupees 1233.0 in Dollars is 14.825057111939401</return>
    </ns2:InrtoDollarResponse>
  </S:Body>
</S:Envelope>
```

So this is how we created our web service and deployed it.

14] now our web service is successfully deployed. Right click on web pages and select new and select jsp as shown below:



15] give name and click on finish as shown below do it 2 times on for input and one for output:



16] In input.jsp create a form for taking user input as shown below:

```

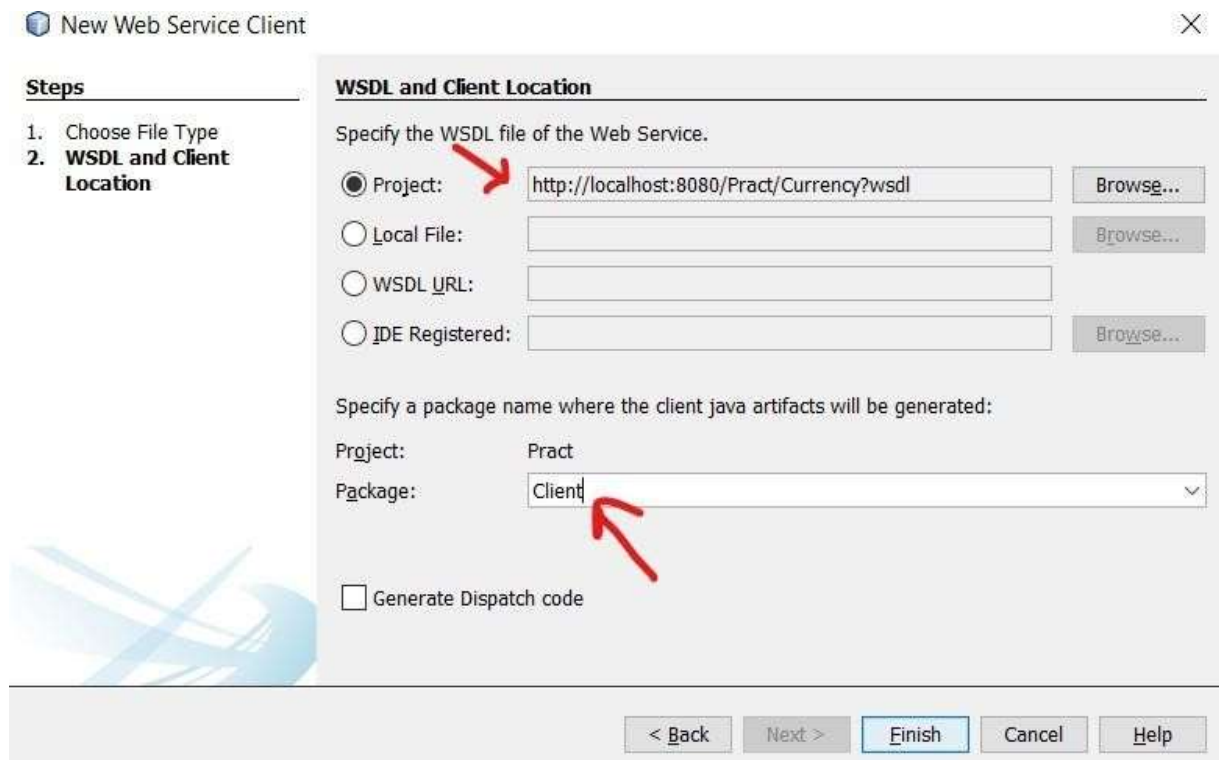
<!DOCTYPE html>
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  <title>JSP Page</title>
</head>
<body>
  <form action="output.jsp">
    <pre>
      Enter the currency in rupees : <input type="text" name="t1">
      <input type="submit"> <input type="reset">
    </pre>
  </form>
</body>
</html>

```

In this code set action = the jsp file where u want output and give name to textbox input.

17] now we have to create web service client, for same right click on project and select new then select web service client you will get the following screen:

At this step click on browse and select your project and click ok after clicking ok you will get wsdl url as shown below:



New Web Service Client

Steps

1. Choose File Type
2. **WSDL and Client Location**

WSDL and Client Location

Specify the WSDL file of the Web Service.

☒ **Project:**

☐ **Local File:**

☐ **WSDL URL:**

☐ **IDE Registered:**

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

Project: Pract

Package:

☐ Generate Dispatch code

< Back Next > **Finish** Cancel Help

At this stage copy the url and paste it in notepad for future.

Give package name as "Client".

Now click on finish.

18] now in project section you can see a new folder is been created named "web servicereference". double click to expand it you should get the following:



Hold the operation or your operation name and drag it into the output.jsp in body tag as shown

: You will get the following auto generated code:

```
<body>
    <!-- start web service invocation --%><hr/>
    <%
    try {
        Client.Currency_Service service = new Client.Currency_Service();
        Client.Currency port = service.getCurrencyPort();
        // TODO initialize WS operation arguments here
        double a = 0.0d;
        // TODO process result here
        java.lang.String result = port.operation(a);
        out.println("Result = "+result);
    } catch (Exception ex) {
        // TODO handle custom exceptions here
    }
    %>
    <!-- end web service invocation --%><hr/>
</body>
```

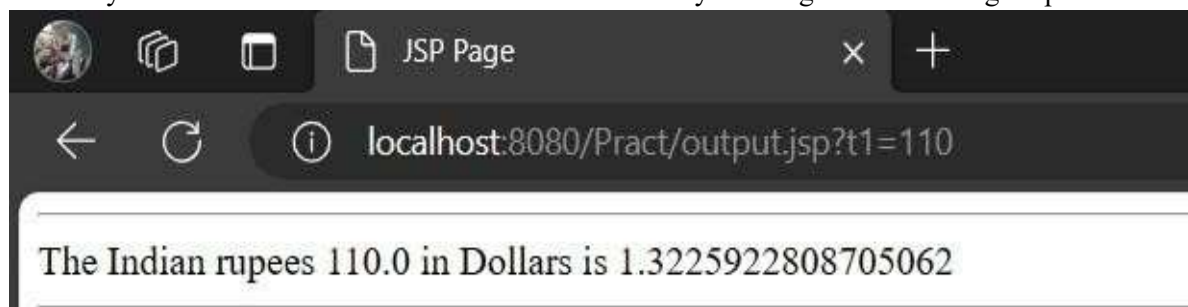
19] now make changes as shown below in the code:

```
<!-- start web service invocation --%><hr/>
<%
try {
    Client.Currency_Service service = new Client.Currency_Service();
    Client.Currency port = service.getCurrencyPort();
    // TODO initialize WS operation arguments here
    double a = Double.parseDouble(request.getParameter("t1"));
    // TODO process result here
    java.lang.String result = port.operation(a);
    out.println(result);
} catch (Exception ex) {
    // TODO handle custom exceptions here
}
%>
<!-- end web service invocation --%><hr/>
```

20] now Deploy our project again and right click on input.jsp and click run file you will redirect to a browser page as shown :



Enter any numerical value in text box and click on submit you will get the following output:



Python client:

1] for consuming java web services in python we should repeat the above steps till step no.19 .

Note: for getting output the project or web service should be deployed .

After doing all of the steps write the following code in python idle:

Code:

```
from zeep import Client

wsdl_url='http://localhost:8080/pract/Currency?wsdl'

client=Client(wsdl_url)

#call the method on the webservice

d=float(input("Enter Currency in Indian Rupees:"))

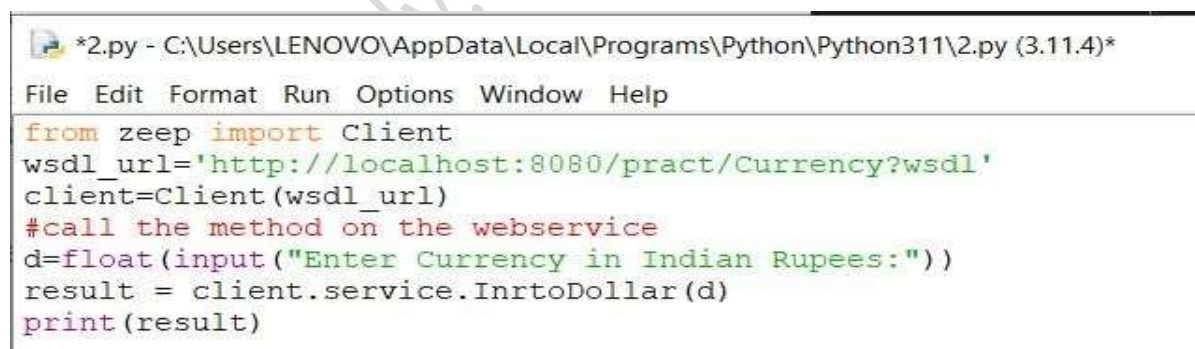
result = client.service.InrtoDollar(d)

print(result)
```

2] replace wsdl_url with the url which was copied at the time of web service client creation .

In step no.17 . and replace the InrtoDollar with your method or operation name. and pass parameter to it

3] final code should look like following:



```
*2.py - C:\Users\LENOVO\AppData\Local\Programs\Python\Python311\2.py (3.11.4)*
File Edit Format Run Options Window Help
from zeep import Client
wsdl_url='http://localhost:8080/pract/Currency?wsdl'
client=Client(wsdl_url)
#call the method on the webservice
d=float(input("Enter Currency in Indian Rupees:"))
result = client.service.InrtoDollar(d)
print(result)
```

4] after running the above code you should get the following output:

```
= RESTART: C:\Users\LENOVO\AppData\Local\Programs\Python\Python311\2.py
Enter Currency in Indian Rupees:12000
The Indian rupees 12000.0 in Dollars is 144.28279427678248
|
.Net Client
```