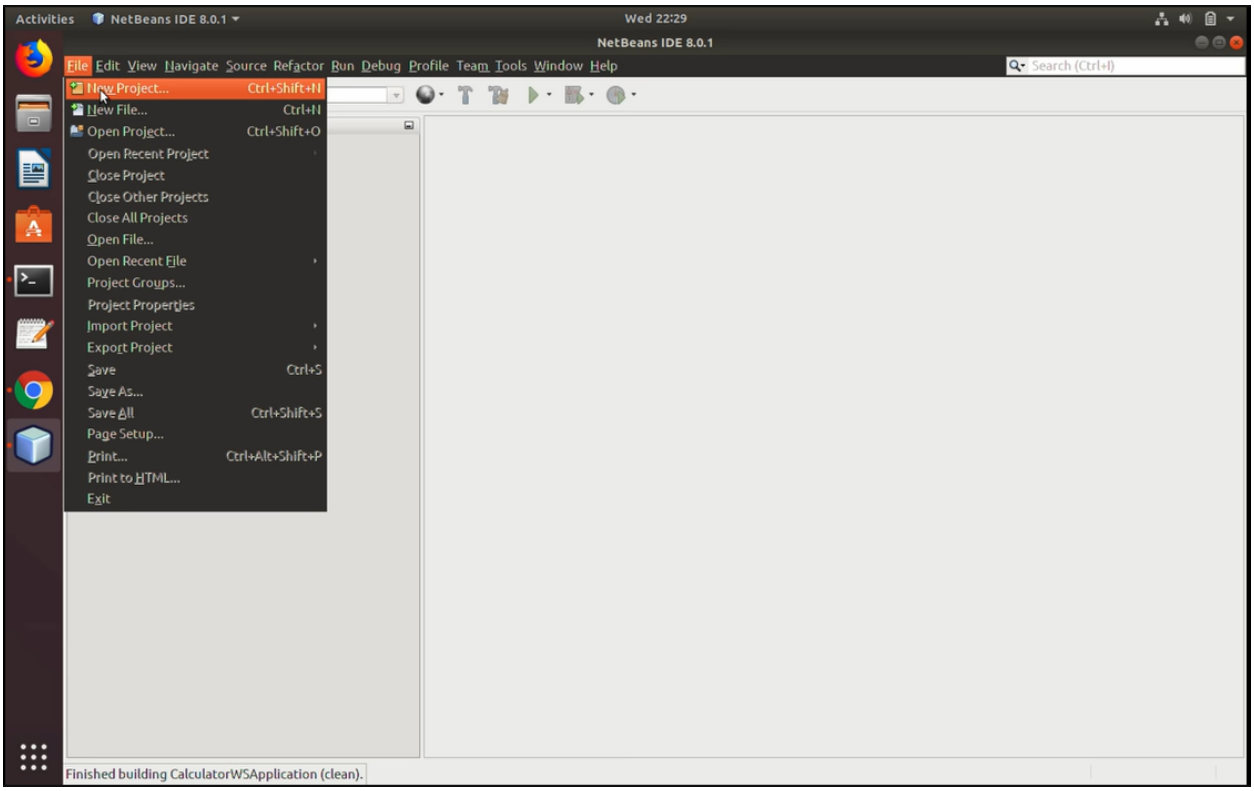
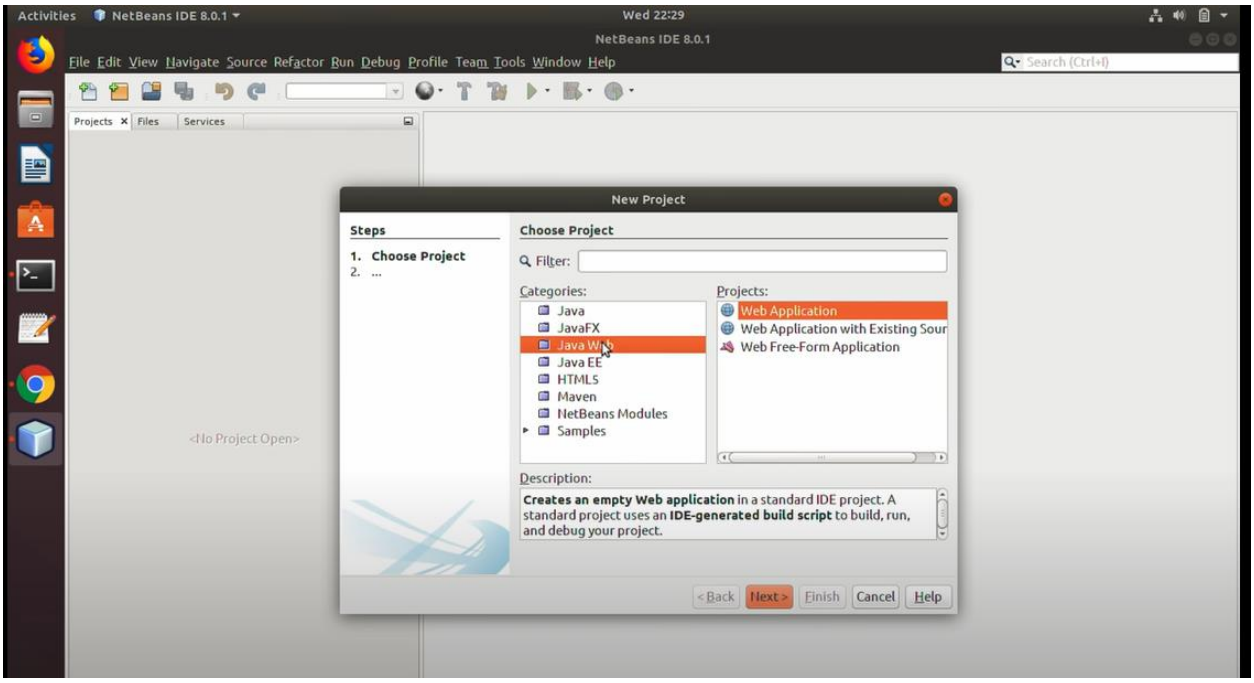
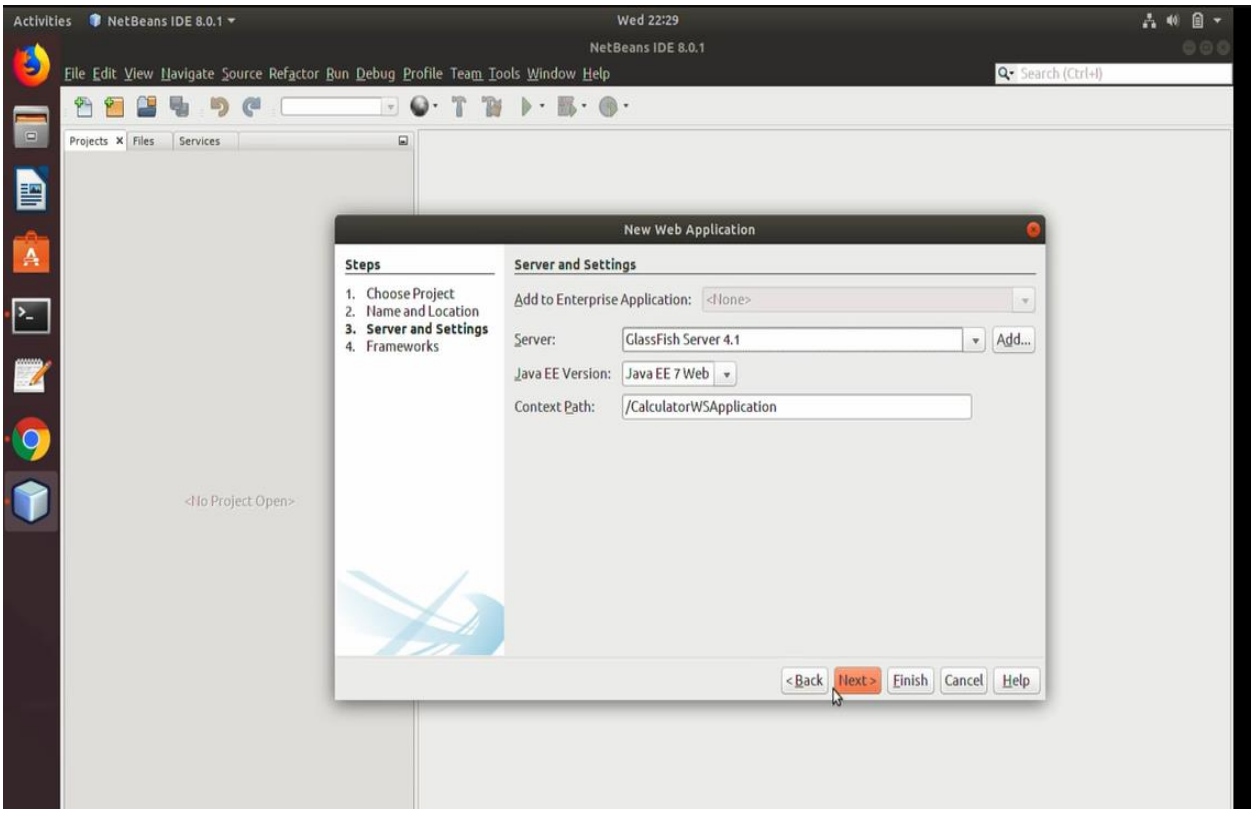
File -1(server side)

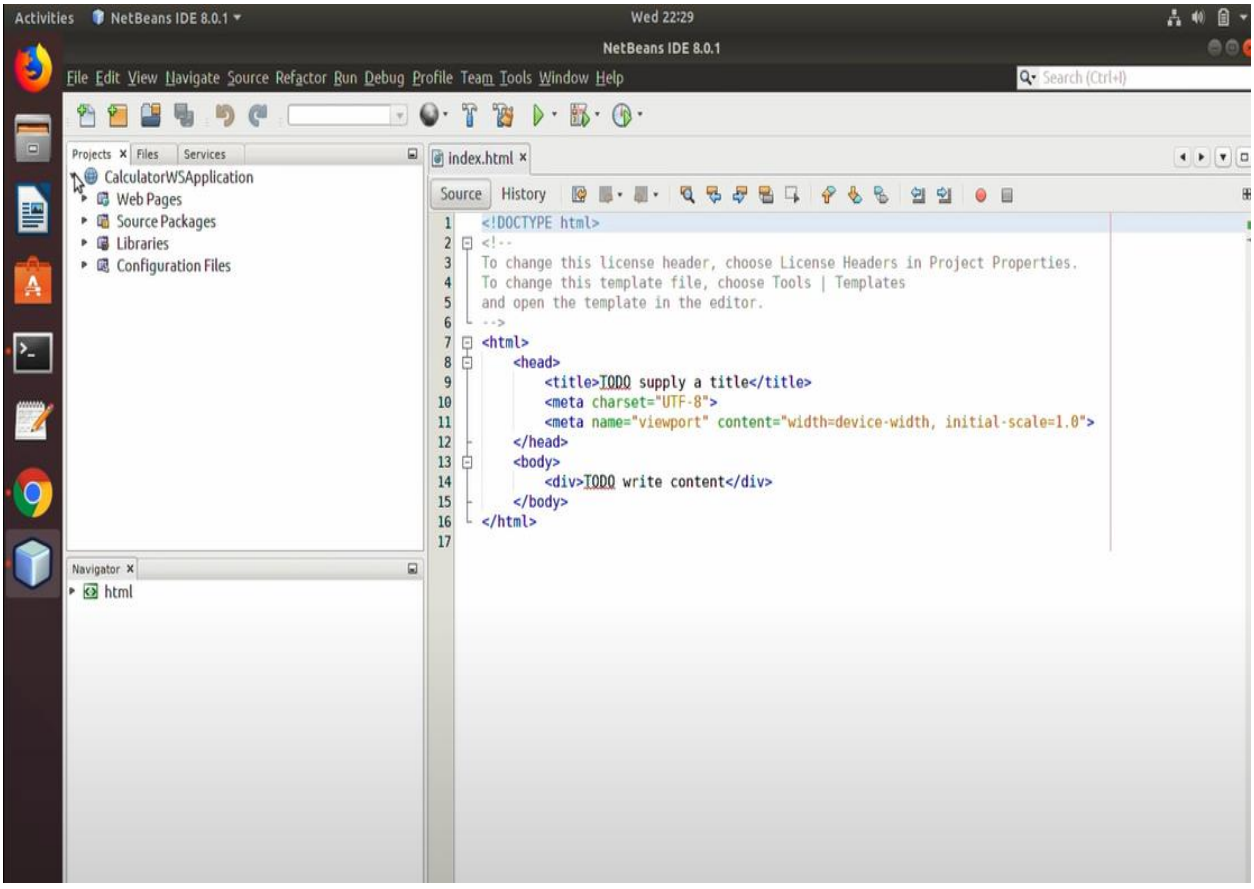
1) Download net-beans(version 13)

2) Go to file ->new project->java web->web application->fill details(proj name) ->finish.

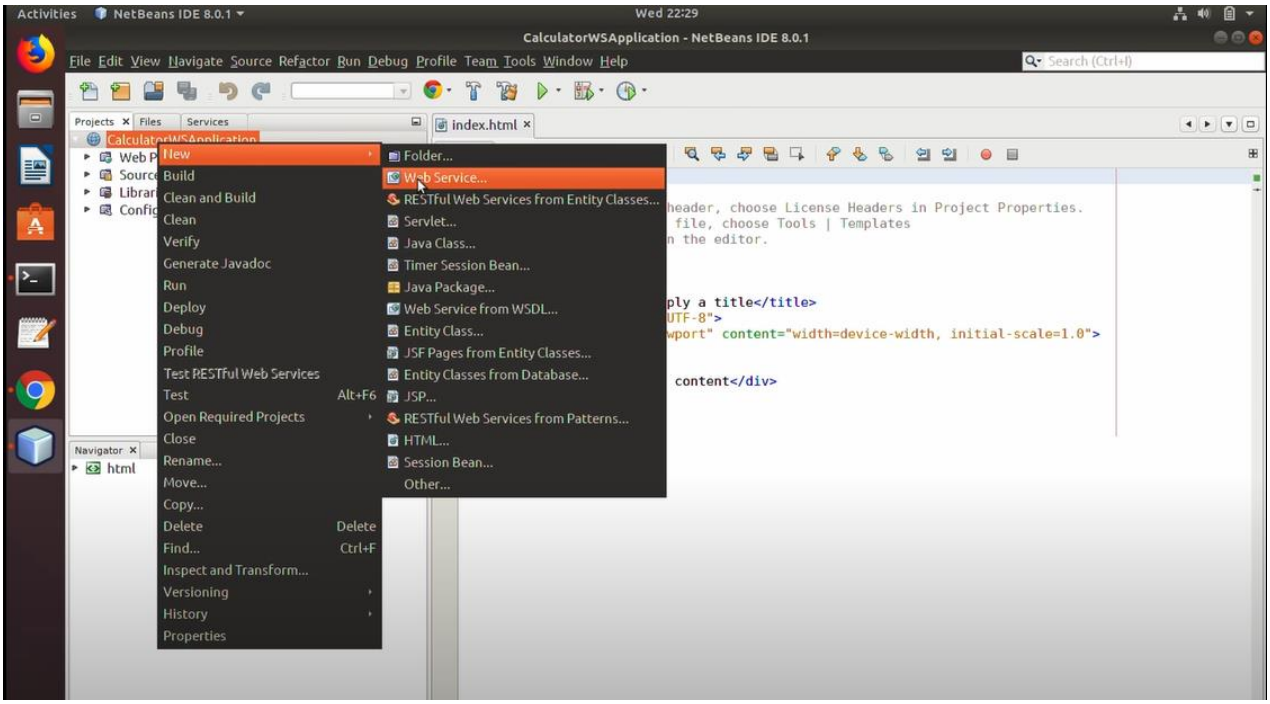


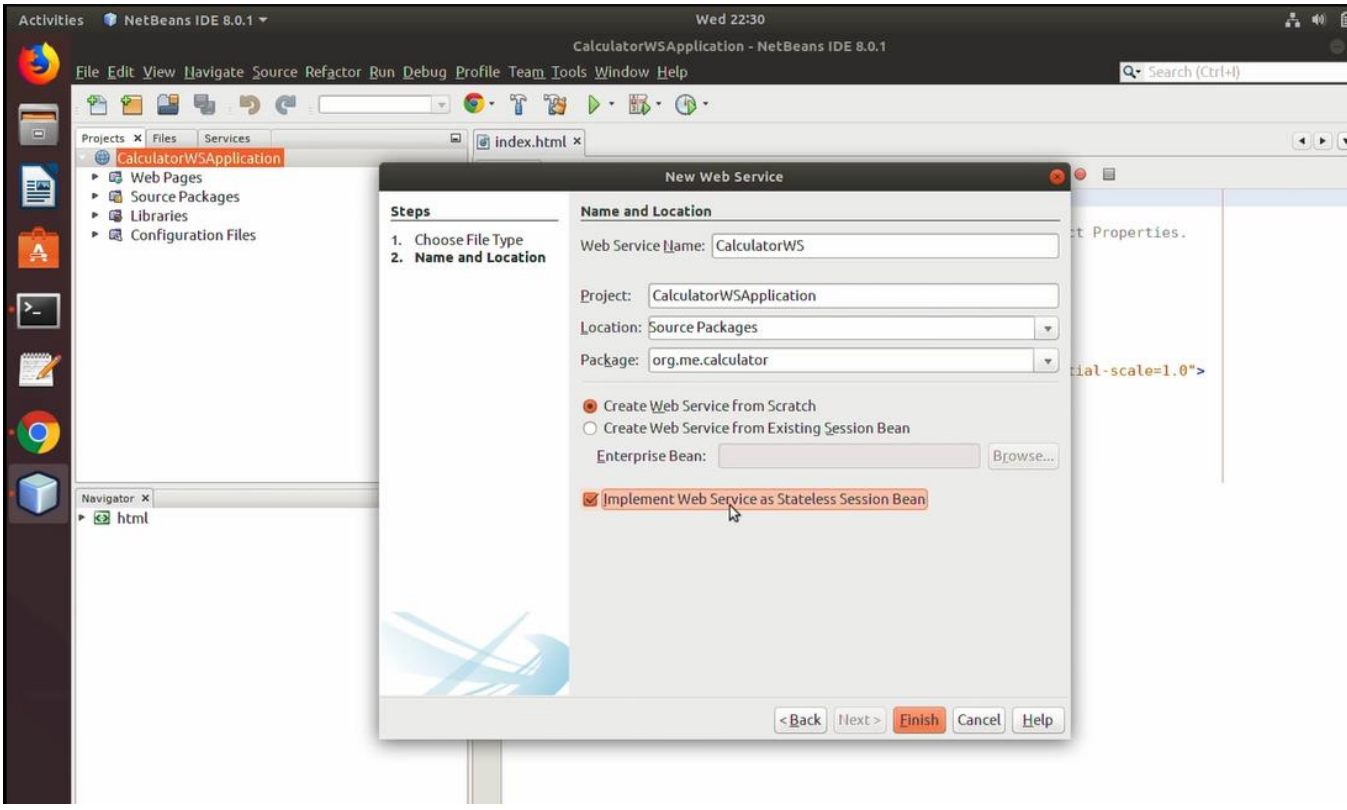



Activities
NetBeans IDE 8.0.1
Wed 22:29
NetBeans IDE 8.0.1
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Search (Ctrl+I)
Projects
Files
Services
<No Project Open>
New Web Application
Steps
1. Choose Project
2. Name and Location
3. Server and Settings
4. Frameworks
Server and Settings
Add to Enterprise Application:
<None>
Server:
GlassFish Server 4.1
Add...
Java EE Version:
Java EE 7 Web
Context Path:
/CalculatorWSApplication
< Back
Next >
Finish
Cancel
Help


Activities
NetBeans IDE 8.0.1
Wed 22:29
NetBeans IDE 8.0.1
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Search (Ctrl+I)
Projects
Files
Services
CalculatorWSApplication
Web Pages
Source Packages
Libraries
Configuration Files
Navigator
html
index.html
Source
History
<!DOCTYPE html>
<!--
To change this license header, choose License Headers in Project Properties.
To change this template file, choose Tools | Templates
and open the template in the editor.
-->
<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>
TODO write content
</body>
</html>

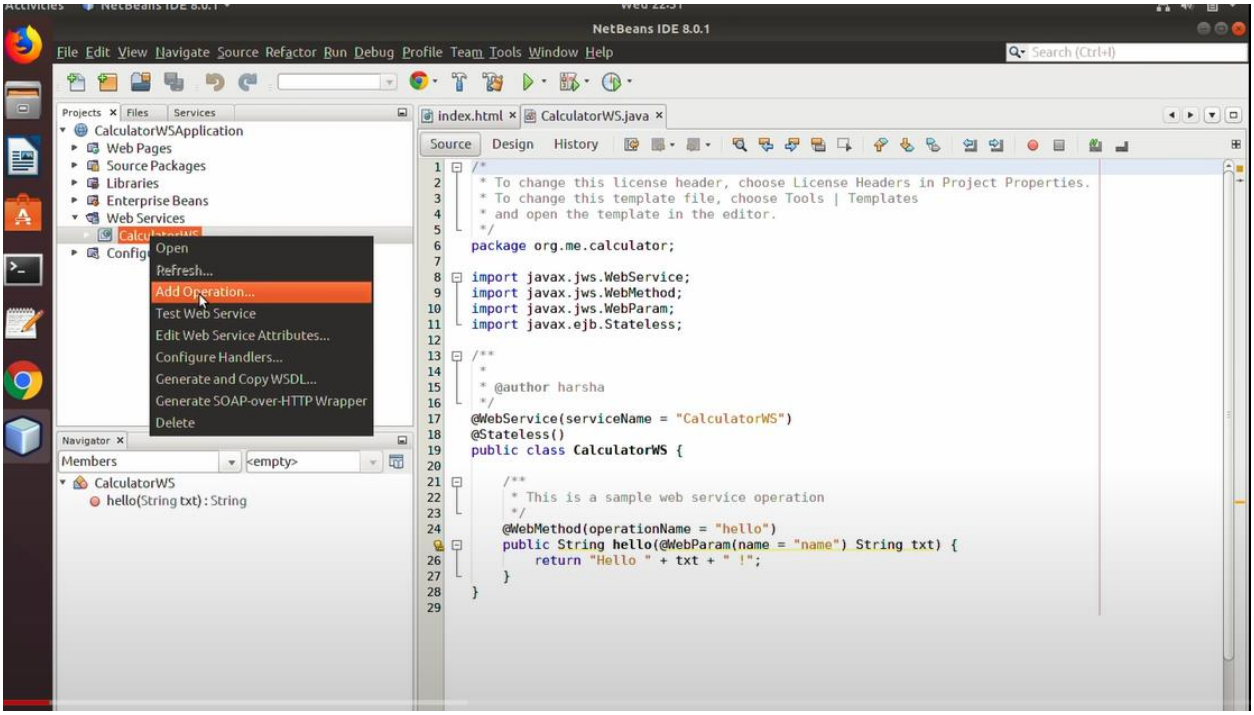
3) Right click on created project -> NEW->Web Service->add web service name->add package as(org.me.string)->click checkbox ->finish.





4) We can add operations under webservice folder(created)->add name of the operation(stringops) ->add parameters (Add variable, set their data type and return type) as per requirements.

(Likewise we can add one by one operation/function as per requirement in this way)



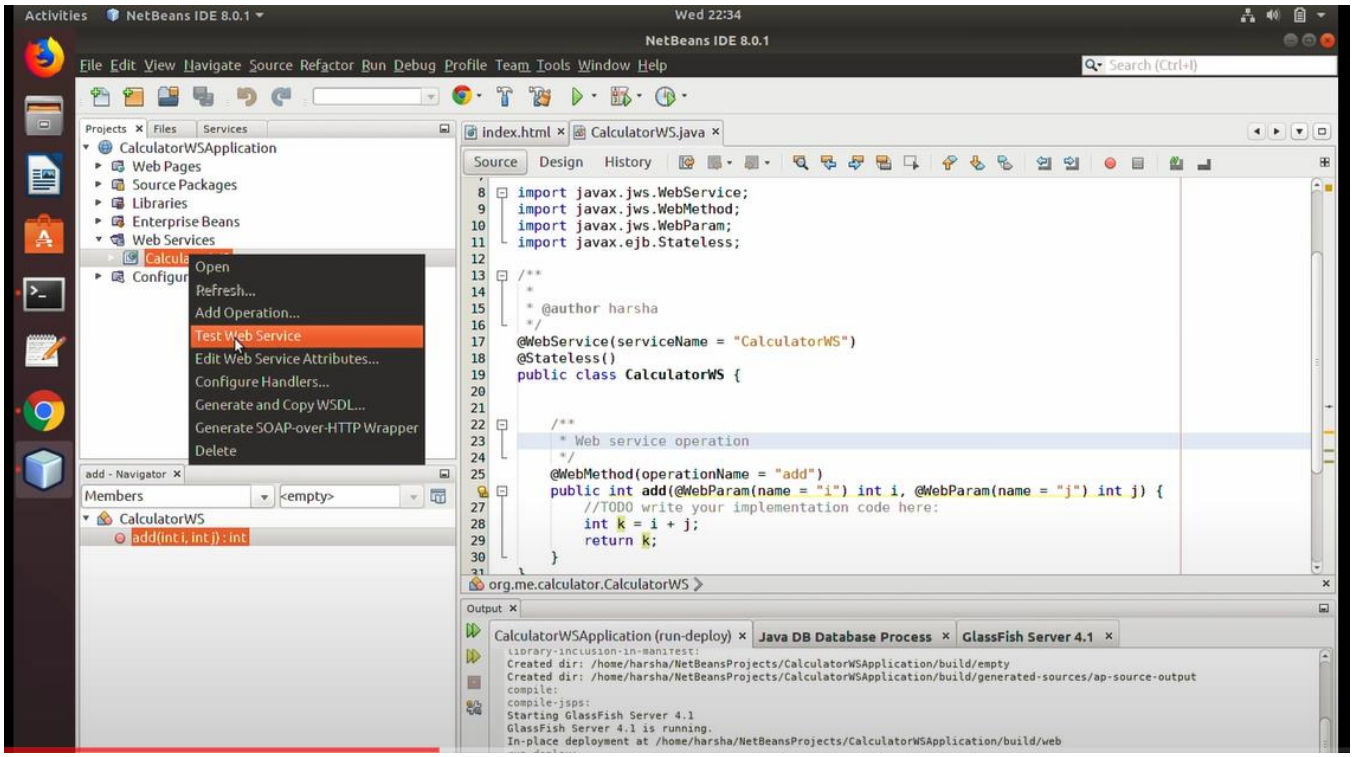
5) Add our logic to newly created stringops file.

6)Right click on NewWebApp(created file) ->Deploy

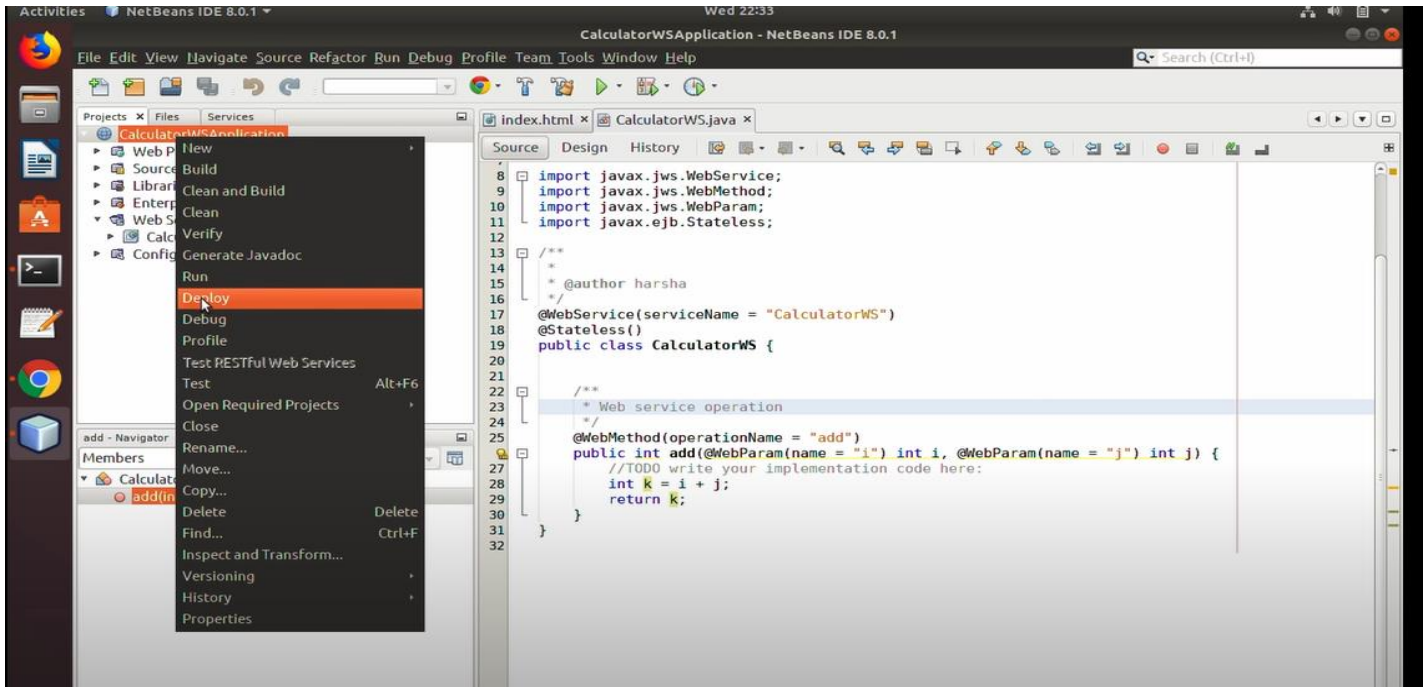
7) After deploy we can check output tab and under that server we can see where the server is listening.

8) Very important - After every change do deploy for reflecting the changes.

(Optional)- 9)after deploy go to string application which is available in folder Web Services

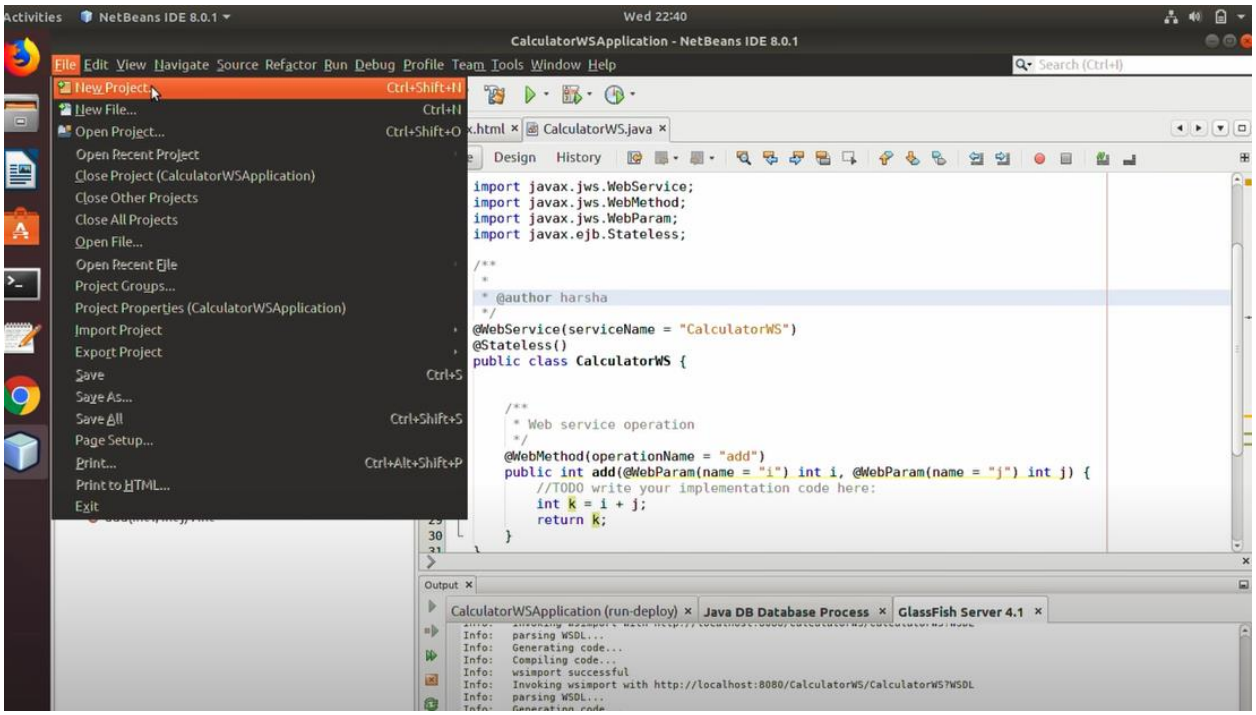


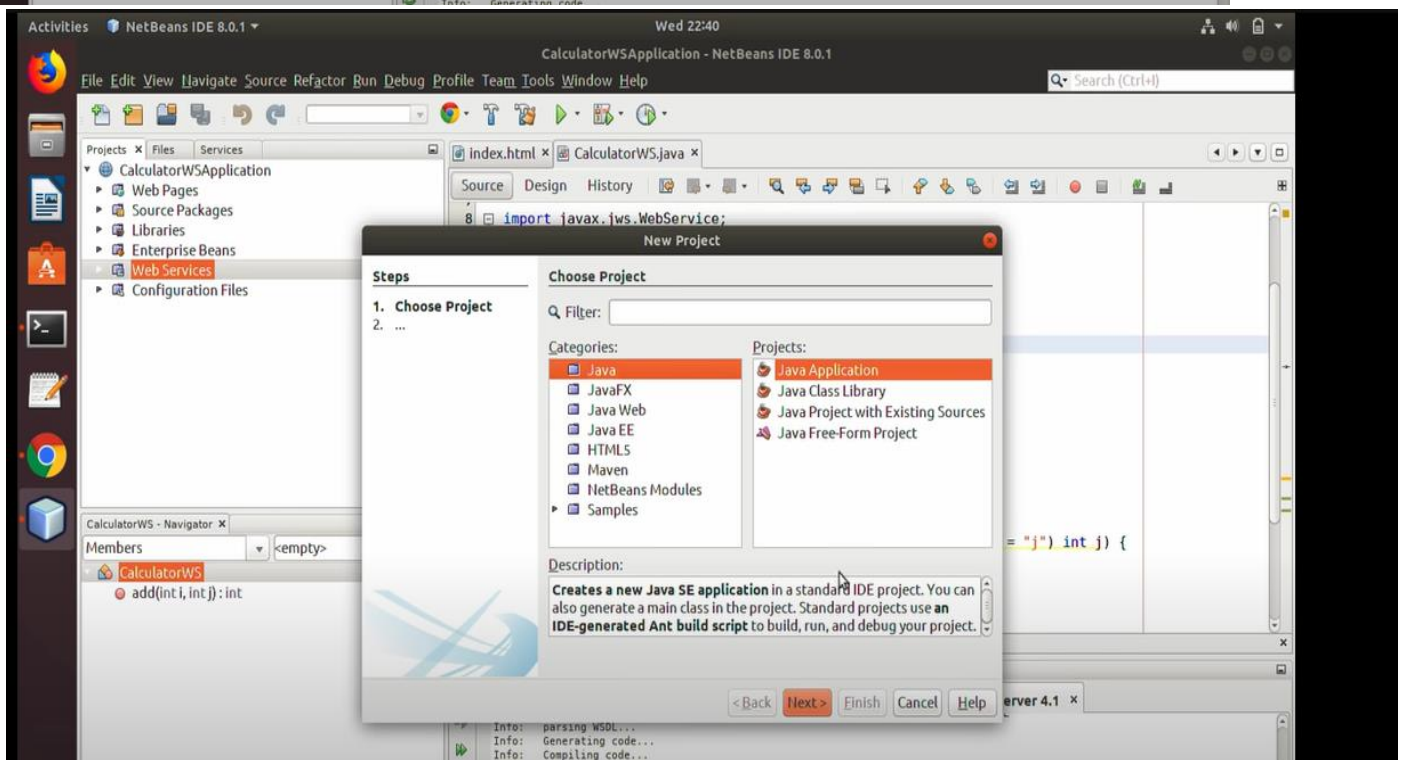
.->right click on string application and click on test services to test the web service is deployed or not.



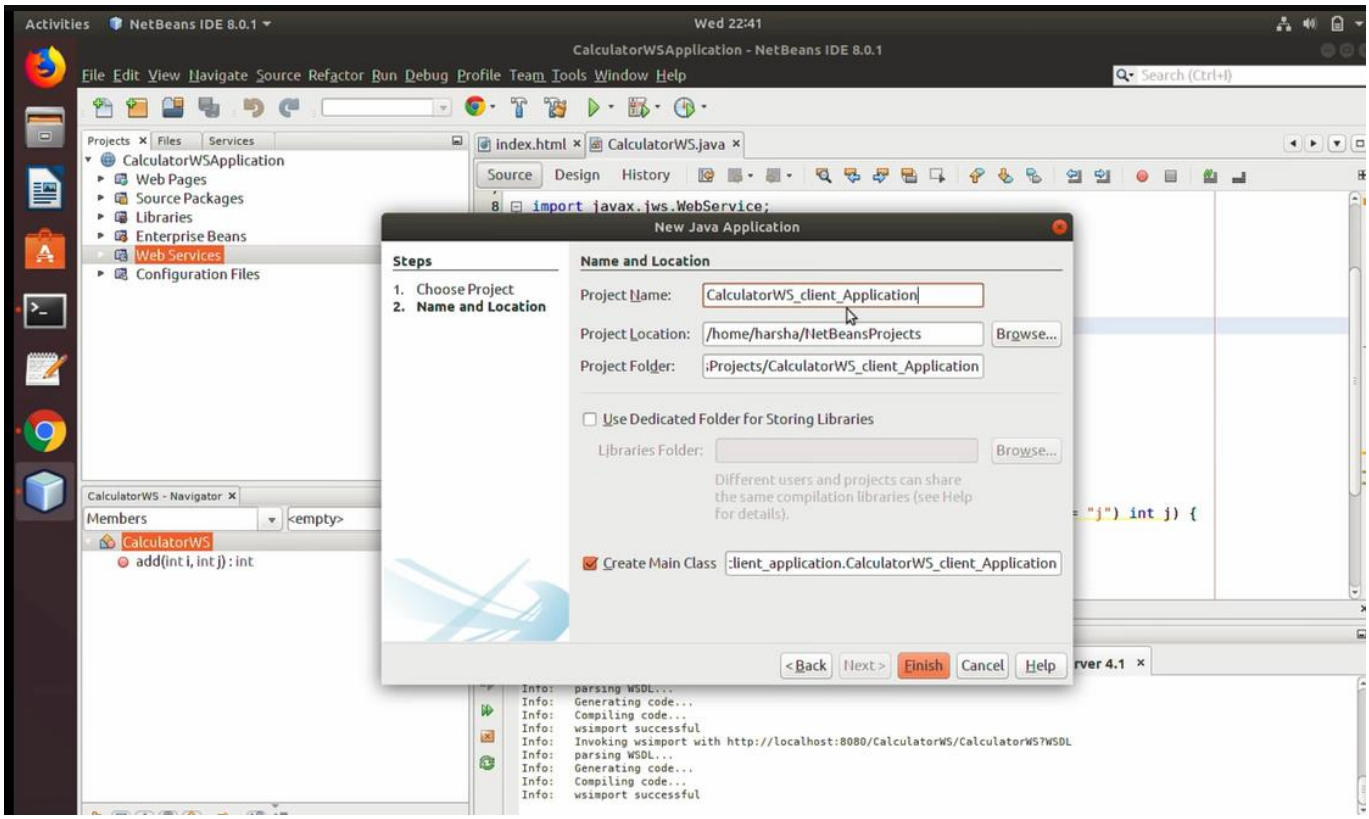
File-2(Client side):

1) Ggo to file ->new project->java ->web application->fill details(give project name- new_webApp clientApplin) ->finish.

2) 

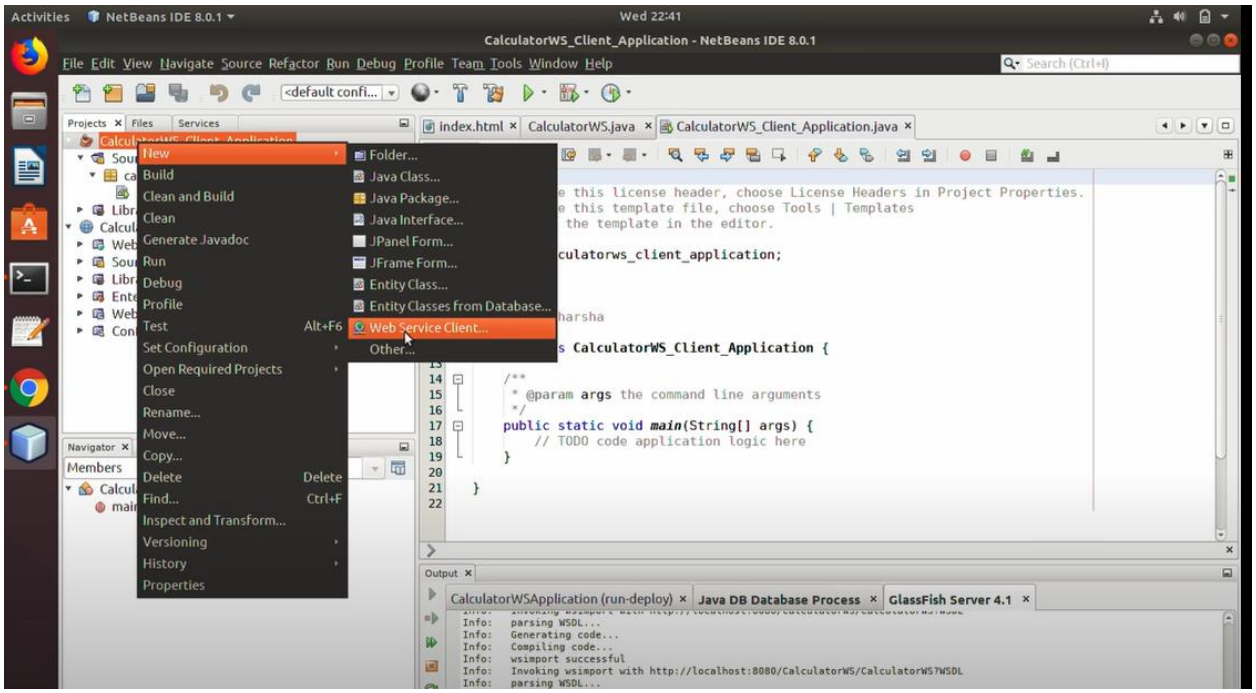
3) 

4)

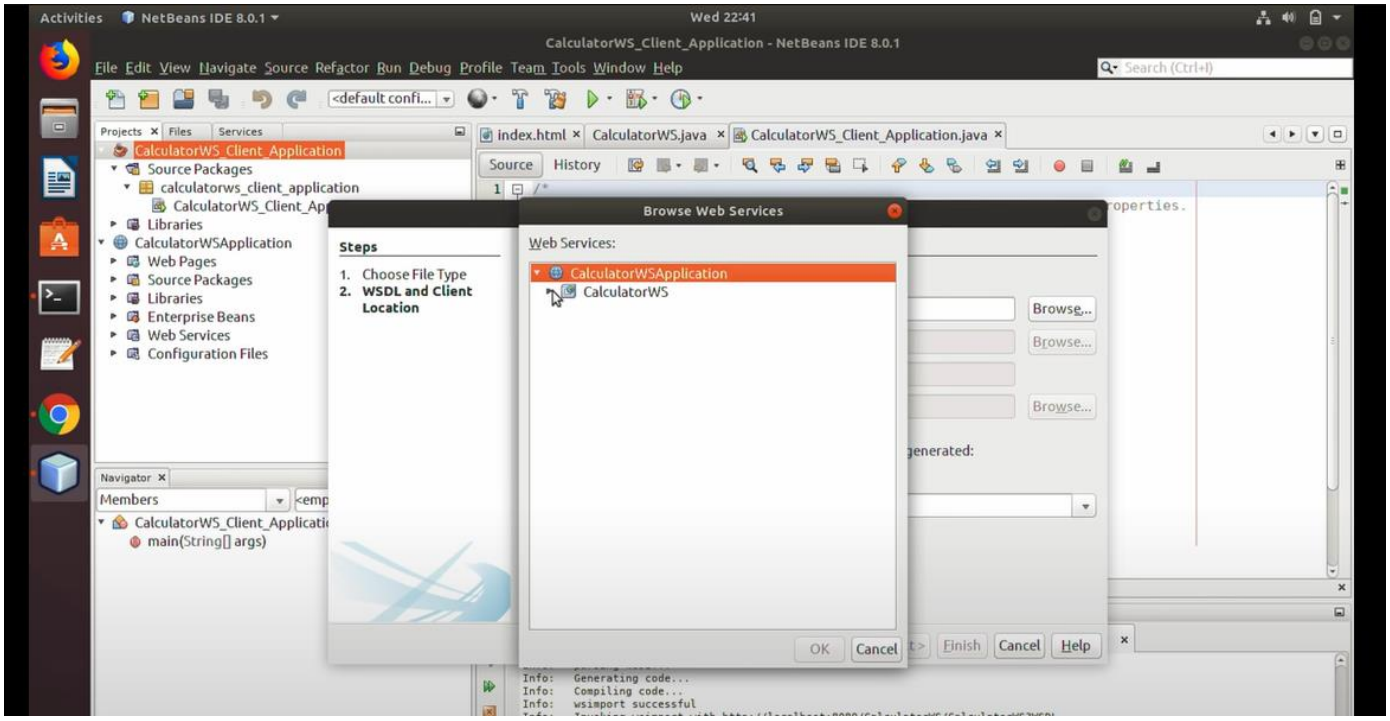


5) **Right click on new_Web_client App ->NEW->Web Service Client->browse the project and select String application under NewWebApp(earlier created).->Ok->finish.**

6)

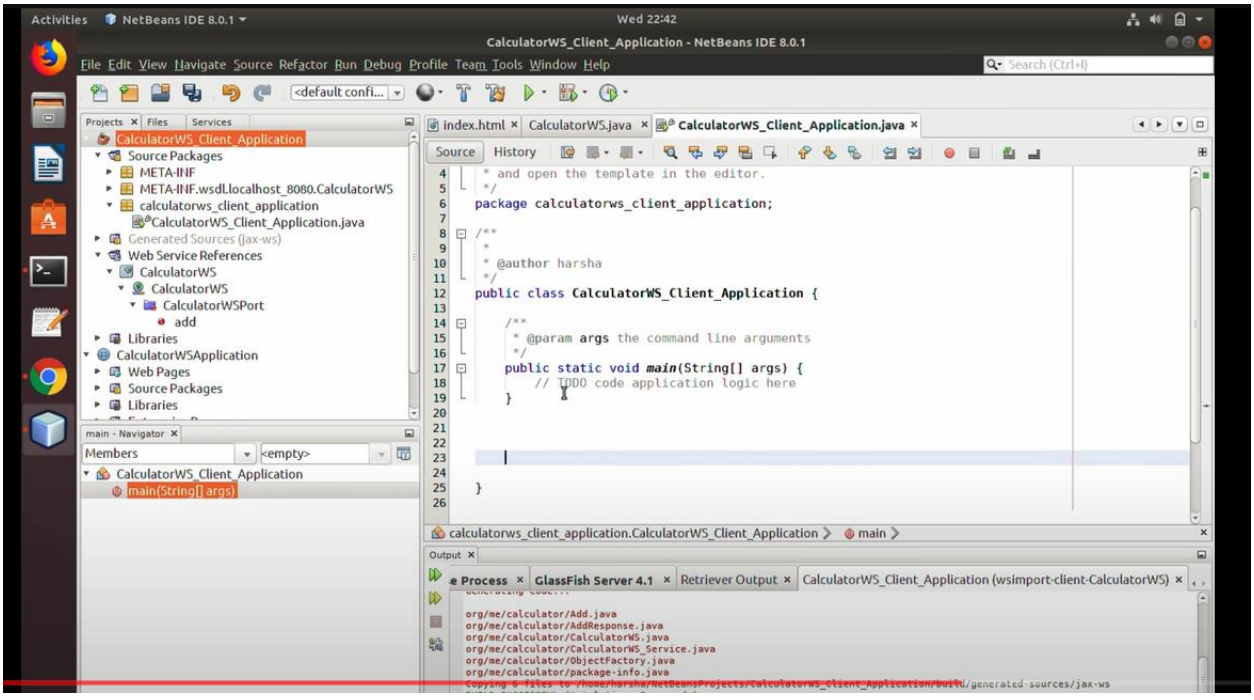


7)

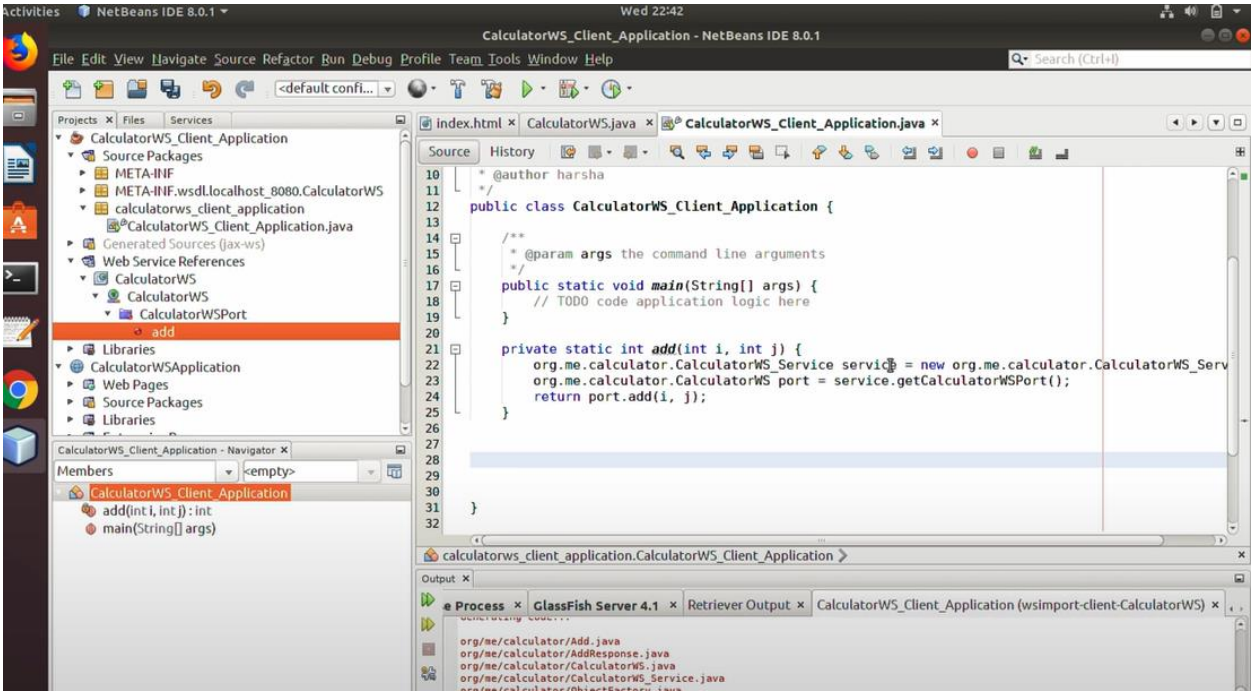


8) **Web Service refrences created ->go on destructuring the folder until the stringOps and then drag and drop the stringOps file into NewWebApp_Client file.**

9)

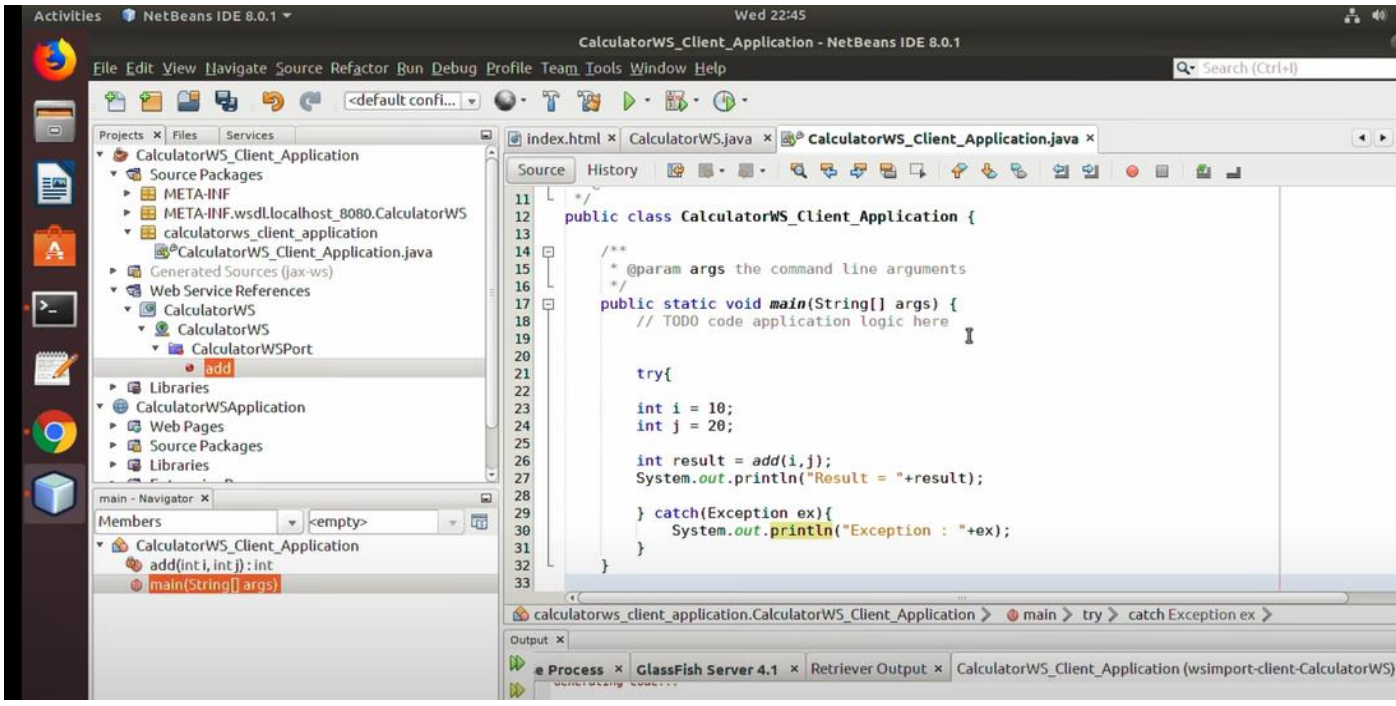


10)

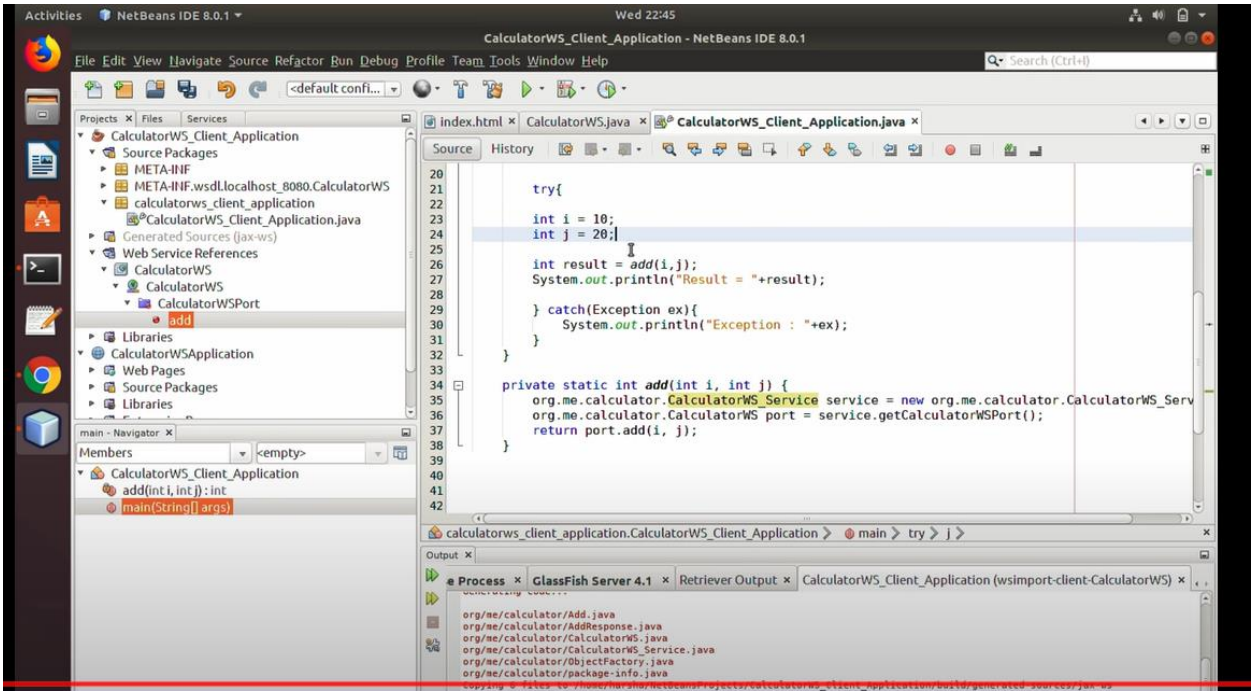


11) **Update the code and call the operations.**

12)

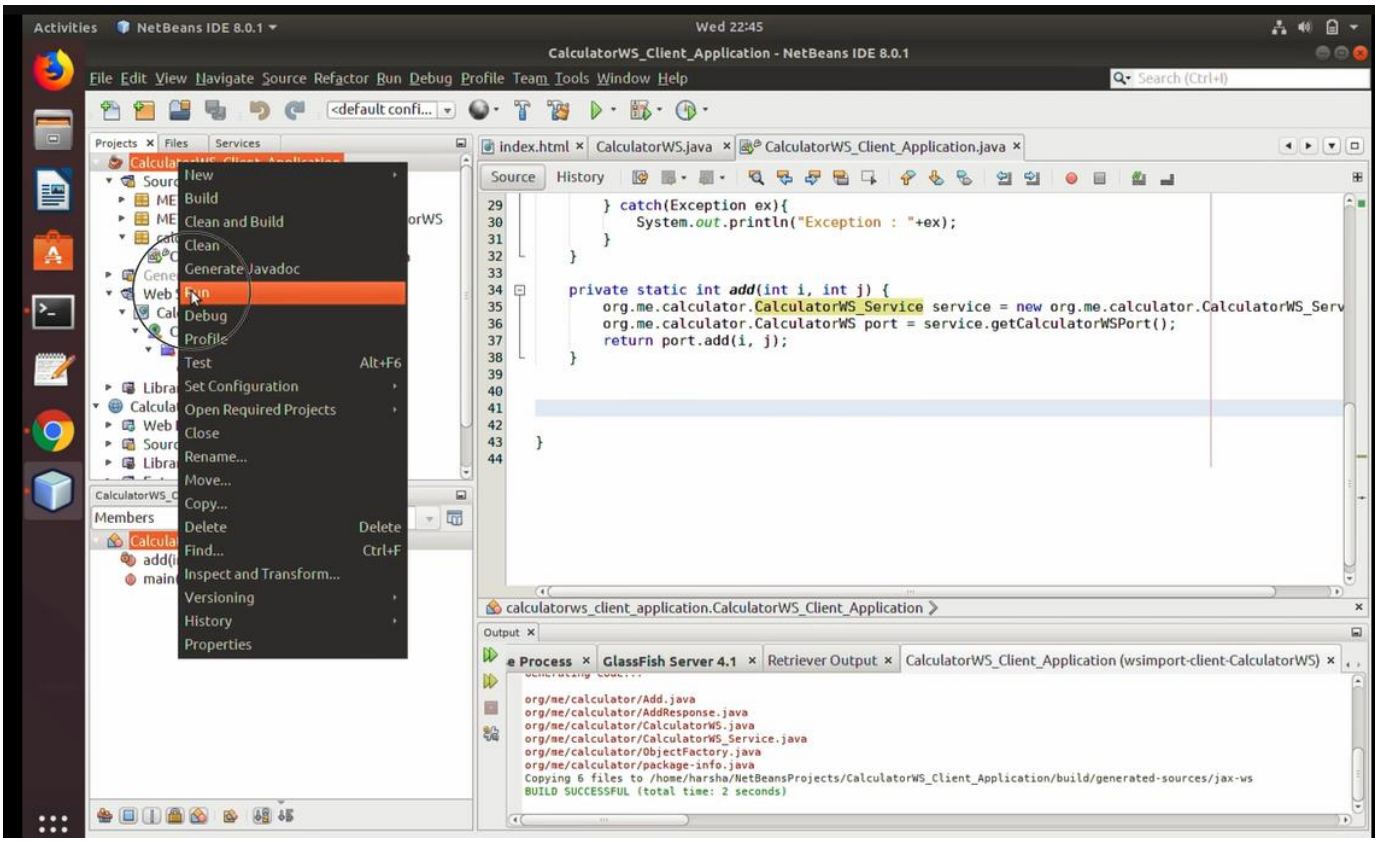


13)

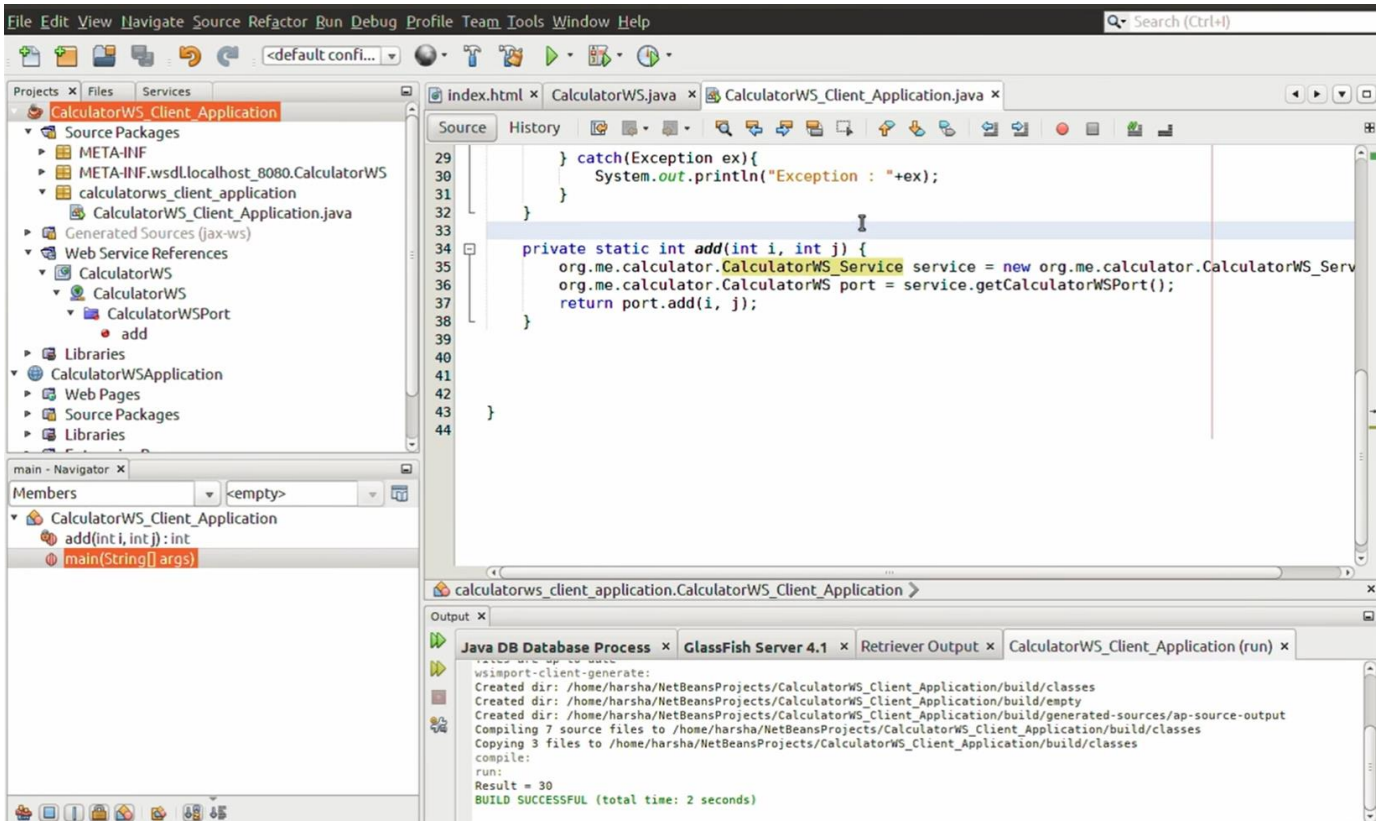


14) Right click on client file(into NewWebApp_Client) and run .

15)



16)


File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Search (Ctrl+I)
Projects
Files
Services
CalculatorWS_Client_Application
Source Packages
META-INF
META-INF.wsdl.localhost_8080.CalculatorWS
calculatorws_client_application
CalculatorWS_Client_Application.java
Generated Sources (jax-ws)
Web Service References
CalculatorWS
CalculatorWS
CalculatorWSPort
add
Libraries
CalculatorWSApplication
Web Pages
Source Packages
Libraries
main - Navigator
Members
<empty>
CalculatorWS_Client_Application
add(int i, int j) : int
main(String[] args)
index.html
CalculatorWS.java
CalculatorWS_Client_Application.java
Source
History
29 } catch(Exception ex){
30 System.out.println("Exception : "+ex);
31 }
32 }
33
34 private static int add(int i, int j) {
35 org.me.calculator.CalculatorWS_Service service = new org.me.calculator.CalculatorWS_Serv
36 org.me.calculator.CalculatorWS port = service.getCalculatorWSPort();
37 return port.add(i, j);
38 }
39
40
41
42
43 }
44
calculatorws_client_application.CalculatorWS_Client_Application
Output
Java DB Database Process
GlassFish Server 4.1
Retriever Output
CalculatorWS_Client_Application (run)
wsimport-client-generate:
Created dir: /home/harsha/NetBeansProjects/CalculatorWS_Client_Application/build/classes
Created dir: /home/harsha/NetBeansProjects/CalculatorWS_Client_Application/build/empty
Created dir: /home/harsha/NetBeansProjects/CalculatorWS_Client_Application/build/generated-sources/ap-source-output
Compiling 7 source files to /home/harsha/NetBeansProjects/CalculatorWS_Client_Application/build/classes
Copying 3 files to /home/harsha/NetBeansProjects/CalculatorWS_Client_Application/build/classes
compile:
run:
Result = 30
BUILD SUCCESSFUL (total time: 2 seconds)