Creating a Web Service in Java using NetBeans IDE

This document provides step-by-step instructions to create and deploy a web service in Java using NetBeans IDE and GlassFish 4.0. In the project, we will create a calculation service.

Prerequisites:

NetBeans 8.0 IDE (Can be downloaded from here - Link)

Step 1: Create a Java Web Project

Open NetBeans IDE

Click on New Project and choose Java Web \rightarrow Web Application



Enter the Project Name: CalculationWS, using the default settings and then click on "Finish".

● ○ ●	New Web Application	
Steps	Server and Settings	
1. Choose Project 2. Name and Location	Add to Enterprise Application: <pre></pre>	Å T
4. Frameworks	Server: GlassFish Server 4 +	Add
	Java EE Version: Java EE 7 Web 💠	
	Context Path: /CalculationWS	
	Help < Back Next > Finish	Cancel

Now the Project has been created.

Step 2: Create a Web Service

Now go to the Project Tree Structure on the left side of the window.

Right click on the project and select "New" and then choose "Web Service"



Specify web service name "CalWS" and package name "CalculationWS". Click on "Finish".

00	New Web Service
Steps	Name and Location
 Choose File Type Name and Location 	Web Service Name: CalWS
	Project: CalculationWS
	Location: Source Packages \$
	Package: CalculationWS
	• Create Web Service from Scratch
	Create Web Service from Existing Session Bean
	Enterprise Bean: Browse
	Implement Web Service as Stateless Session Bean
	Help < Back

Open CalWS.java file, replace the original hello() function with the following code:

@WebMethod(operationName = "Addition")
public String Addition(@WebParam(name = "value1") String value1,@WebParam(name = "value2") String value2
<mark>) {</mark>
float value=Float.valueOf(value1)+Float.valueOf(value2);
return (Float.toString(value));
}

Now the web service is created.

Step 3: Deploy and Test Web Service

Right click on the project and select "Deploy"



This is to deploy all the web services in this project. If success, you will see:

<pre>23 @WebMethod(operationName = "Addition") public String Addition(@WebParam(name = "value1") String value1,@WebParam(name = "value2") String value2) { 25 float value=Float.value0f(value1)+Float.value0f(value2); 26 return (Float.toString(value)); 27 } </pre>					
Outpu					
	Retriever Output 🛛 Java DB Database Process 🛇 GlassFish Server 4 🛇 CalculationWS (run-deploy) ×				
	W ueps-mounte-jar.				
03	deps-ear-jar:				
20	library-inclusion-in-archive:				
	library-inclusion-in-manifest:				
	compile:				
	compile-jsps:				
	Undeploying				
	BUILD SUCCESSFUL (total time: 1 second)				

To test the web service, right click on the service and select "Test Web Service"



You will see:

	u ^z
Image: A state of the state	C Reader O
□ III Apple iCloud Facebook Twitter Wikipedia Yahoo News ▼ Popular ▼	+

CalWS 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 calculationws.CalWS.addition(java.lang.String java.lang.String)
addition
(
)

Right Click on the project and select "Clean and Build", a war file will be automatically generated under "dist" sub-directory.