

# GlassFish Server

Type	----- Application server software
Vendor	----- SunMicroSystem(Oracle corporation) (open source software)
Version	----- 4.x(compatible with jdk 1.7)
Default port no	----- 8080(http) 4848(admin console)
To download software	----- <a href="http://www.glassfish.org">www.glassfish.org</a> (zip file)

->Allows to create domains ,default domain name is “domain1”

## Procedure to create user-defined domain in glassfish server

=====

<GlassFish\_Home>\glassfish\bin>

asadmin create-domain –user=GlassFish --adminport=4343 FirstDomain

Enter Password : GlassFish

Retype Password : GlassFish

## Procedure to perform console deployment in **FirstDomain** Domain of GlassFish

=====

### Server

=====

1.create war file(VoterApp.war)

2.start **FirstDomain** domain server of glassfish4.x

<GlassFish\_Home>\glassfish\bin\asadmin start-domain FirstDomain

3.Open admin consle of FirstDomain domain server

<http://localhost:4343> ---->username:GlassFish password:GlassFish

4.Deploy the web application

Application --> deploy -->browser(to select war file) -->ok

5.Test the web application

[url:http://localhost:9090/VoterApp/input.html](http://localhost:9090/VoterApp/input.html)

## **Note:**

====

- >GlassFish Server internally uses the Tomcat server supplied Servlet Container, JSP Containers.
- >GlassFish\_Home>\glassfish\domain\domains\FirstDomain\auto deploy folder
- >In GlassFish server only war file based hard deployment is possible that means Directory based hard deployment is not possible.

## **Procedure to configure GlassFish Server with Eclipse IDE**

=====

### **1.Install GlassFish Server adapter to Eclipse IDE**

### **2.configure GlassFish5.x Domain Server**

Window -> Preference -> server ->Runtime Environment -> Add

-> GlassFish -> create a new local server -> next ->

**Name** :GlassFish

**GlassFish location** : F:\SERVERS\GlassFish\glassfish-5.0\glassfish5\glassfish

**Java Location** : C:\Program Files\Java\jdk

-> next

**Name** : GlassFish 5 Domain

**Host name** : localhost

**Domain Path** : F:\SERVERS\GlassFish\glassfish-5.0\glassfish5\glassfish\domains\domain1

**Admin name** : GlassFish

**Admin Password** : GlassFish

->Finish

## Procedure to create DataSource pointing to JDBC connection pool for oracle FirstDomain domain of GlassFish4.x server

**Step1 :** place ojdbc6.jar in <GlassFish\_Home>\domain\FirstDomain\lib\ext folder

**Step2 :** start FirstDomain domain server of glassfish using  
<GlassFish\_Home>\bin>asadmin start-domain FirstDomain

**Step3:** open admin console:

<http://localhost:4343>

username : GlassFish

password : GlassFish

**step4:** create JDBC connection pool for oracle

Admin screen console -> resource -> JDBC -> JDBC connection pool -> new  
-> general setting

Pool name	: pool1
Resource type	: java.sql.Driver
DataBase vendor	: Oracle -> next
Initial and min pool size	: 8
Max pool size	: 100
Pool Resize Quantity	: 2
Idle Timeout	: 300 sec
Max wait Time	: 6000
url	: jdbc:oracle:thin:@localhost:1521:orcl
user	: system
password	: manager
finish -> launch pool -> ping -> save	

**step5:** create DataSource pointing to the above JDBC connection pool

Admin console -> Resources -> JDBC -> JDBC resources(data source) -> new  
-> resources -> (datasource name:ds1) -> Jndi name:DsJndi  
-> Pool name : pool1 -> ok

**Note:** Having the JNDI name “DsJndi” the DataSource object reference will be placed in JNDI registry of GlassFish server.

## Important points:

=====

- **By Default Tomcat server does not support CrossContext Communication**  
Because getContext(-) method is not implemented properly in Tomcatserver.  
To enable cross context communication we can place  
**<Context crossContext="true"/>** in **<Tomcat\_Home>\config\context.xml** file
- Since **GlassFish, Jboss servers** internally use Tomcat server this crossContext communication is also not possible in this two servers.
- ServletContext of one web application which is accessed from the ServletContext object of another web application is called **Foreign Context**.