Top of Form

#### Creating Servlet Example in Eclipse

Eclipse is an open-source ide for developing JavaSE and JavaEE (J2EE) applications. You can download the eclipse ide from the eclipse website <http://www.eclipse.org/downloads/>.

You need to download the eclipse ide for JavaEE developers.

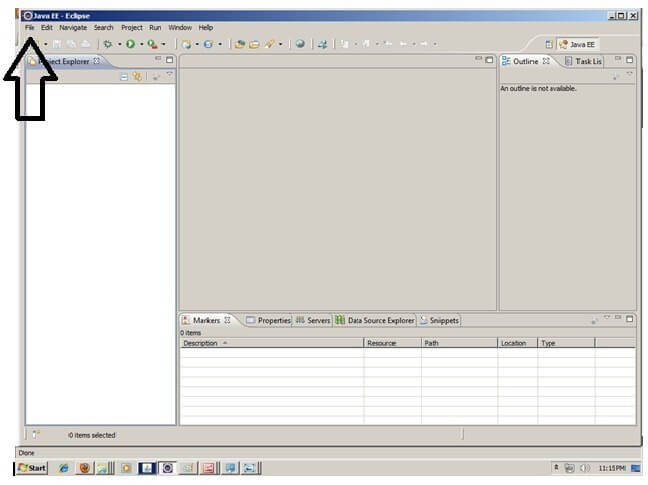
Creating **servlet example in eclipse ide**, saves a lot of work to be done. It is easy and simple to create a servlet example. Let's see the steps, you need to follow to create the first servlet example.

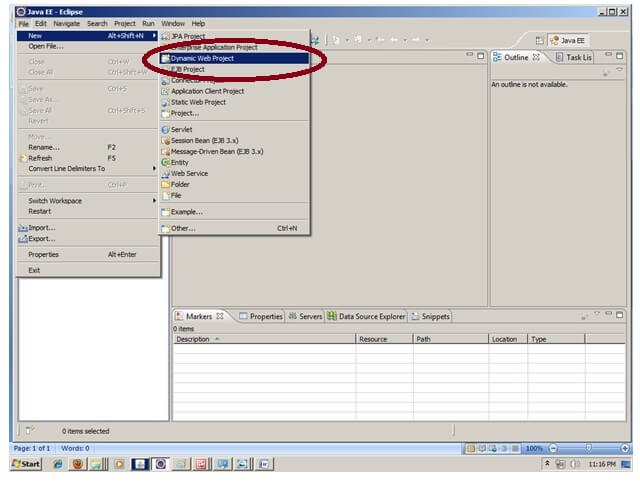
* Create a Dynamic web project
* create a servlet
* add servlet-api.jar file
* Run the servlet

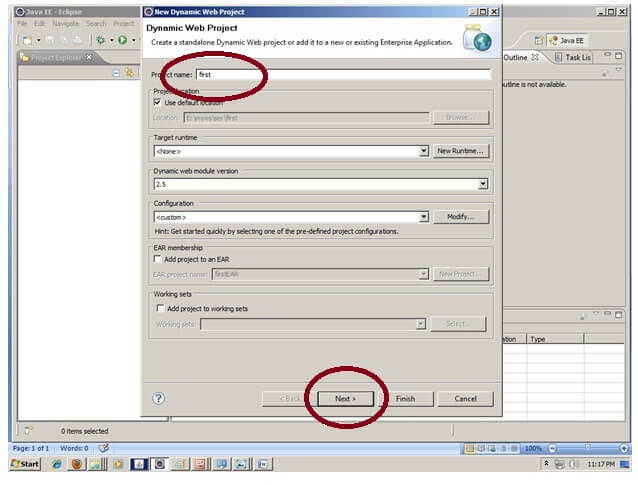
[download this example (developed in eclipse)](https://static.javatpoint.com/src/servlet/firstservleteclipse.zip)

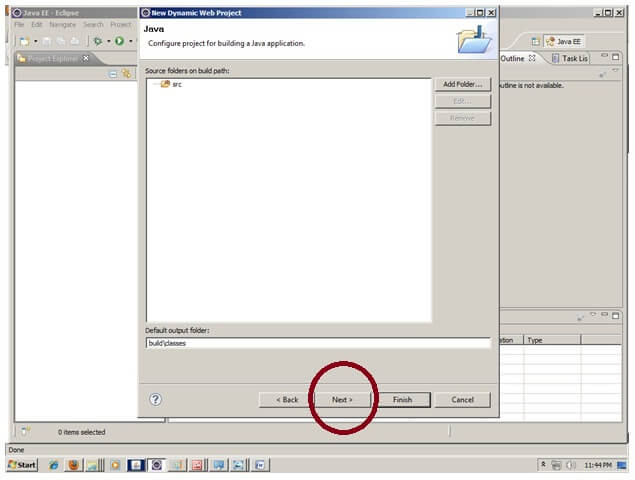
#### 1) Create the dynamic web project:

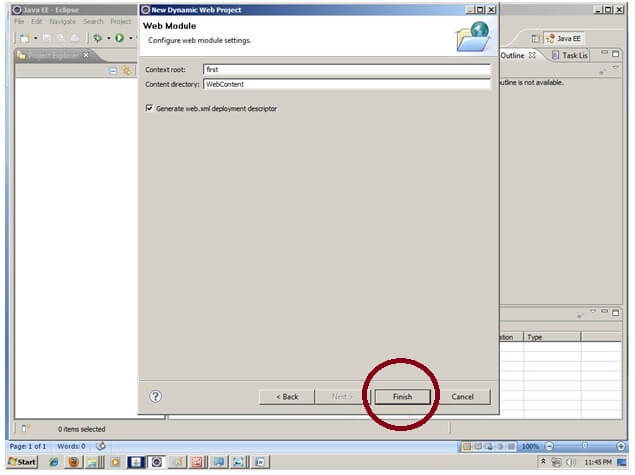
For creating a dynamic web project **click on File Menu -> New -> Project..-> Web -> dynamic web project -> write your project name e.g. first -> Finish**.

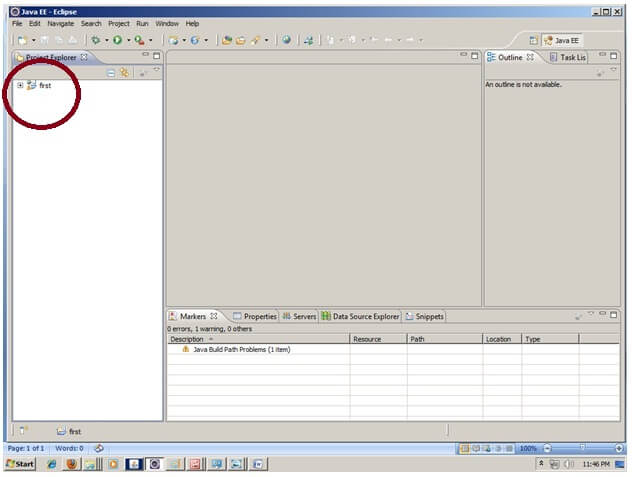






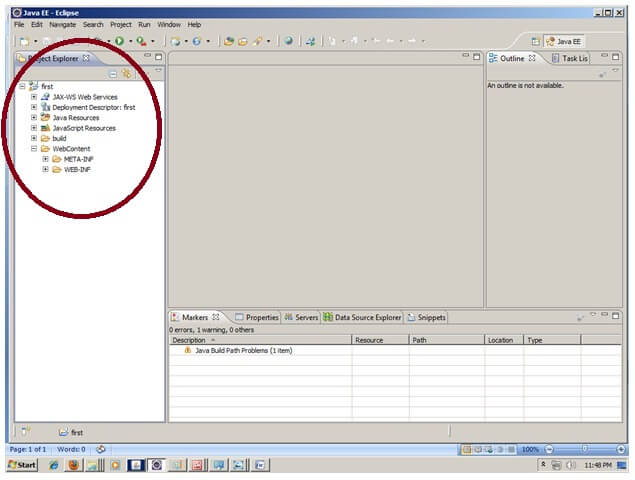


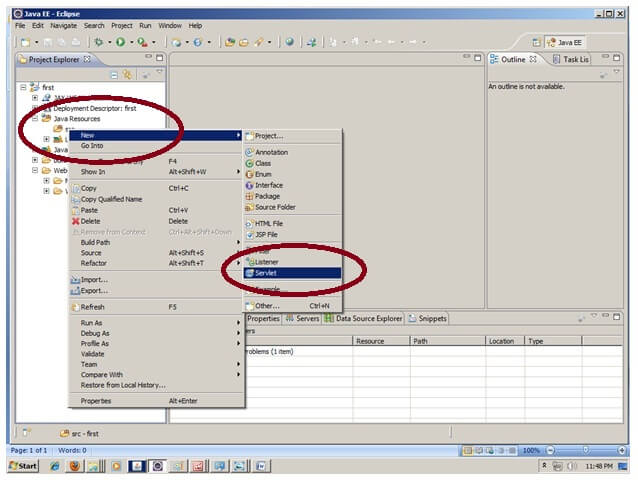


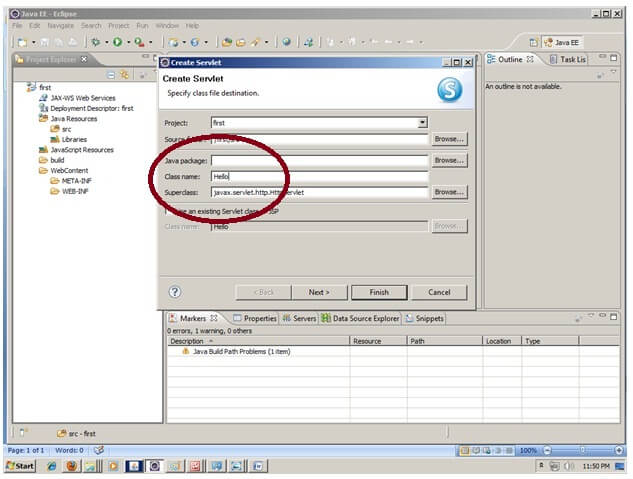


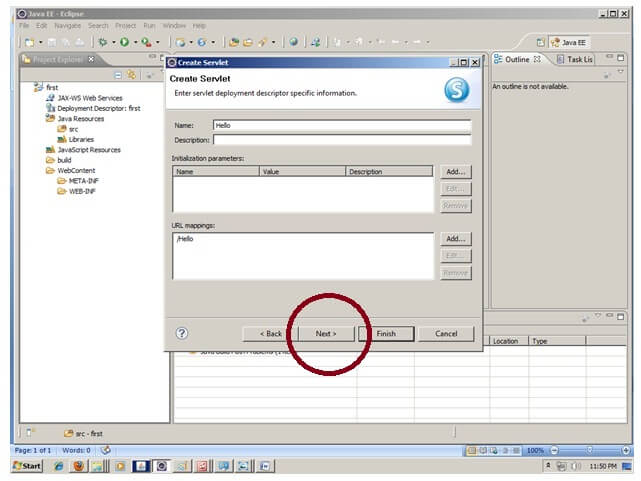
#### 2) Create the servlet in eclipse IDE:

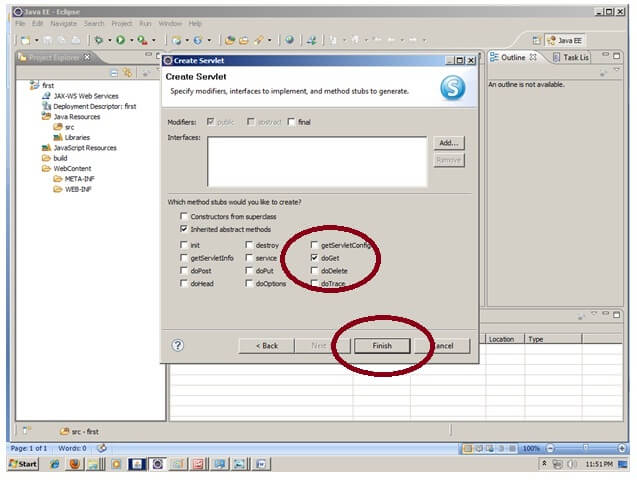
For creating a servlet, **explore the project by clicking the + icon -> explore the Java Resources -> right click on src -> New -> servlet -> write your servlet name e.g. Hello -> uncheck all the checkboxes except doGet() -> next -> Finish**.

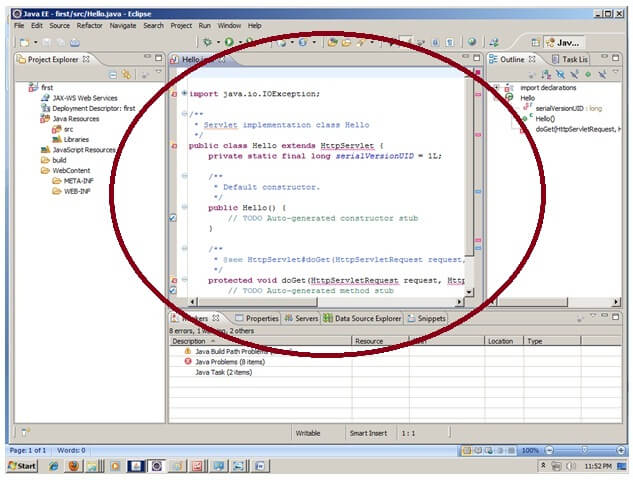






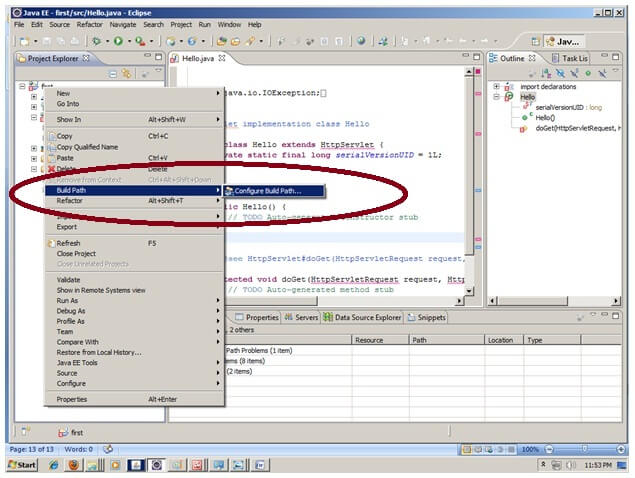


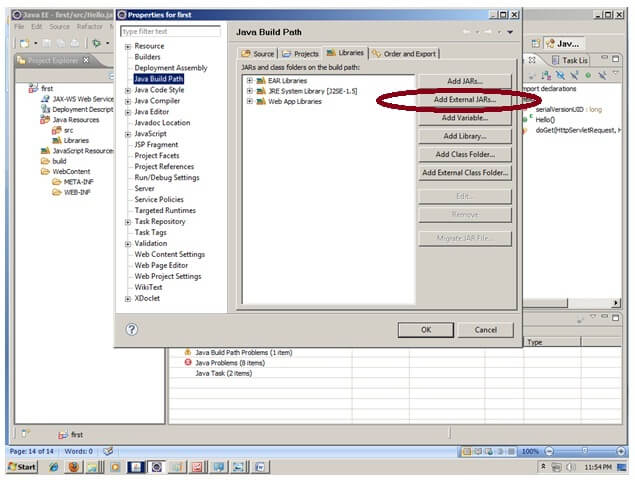


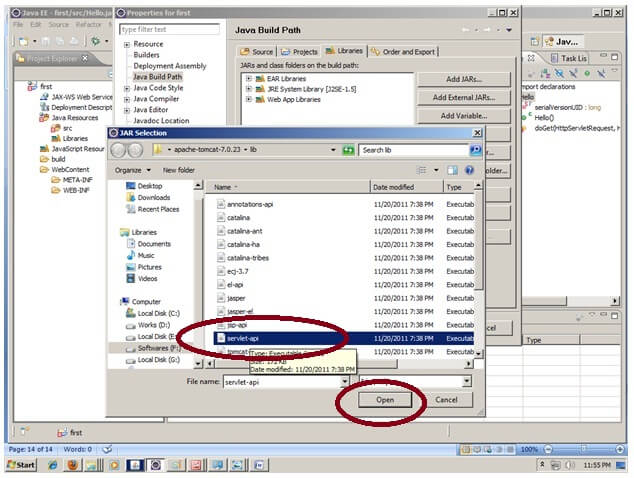


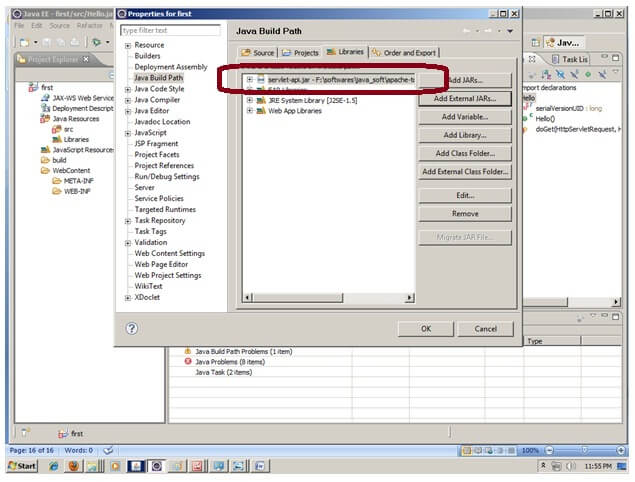
#### 3) add jar file in eclipse IDE:

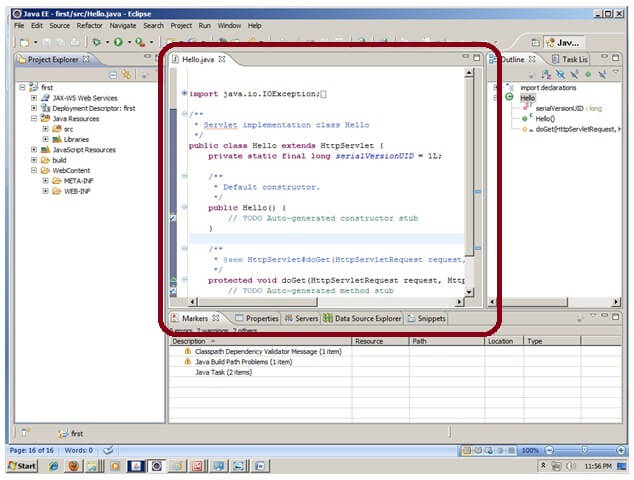
For adding a jar file, right click on your project -> Build Path -> Configure Build Path -> click on Libraries tab in Java Build Path -> click on Add External JARs button -> select the servlet-api.jar file under tomcat/lib -> ok.



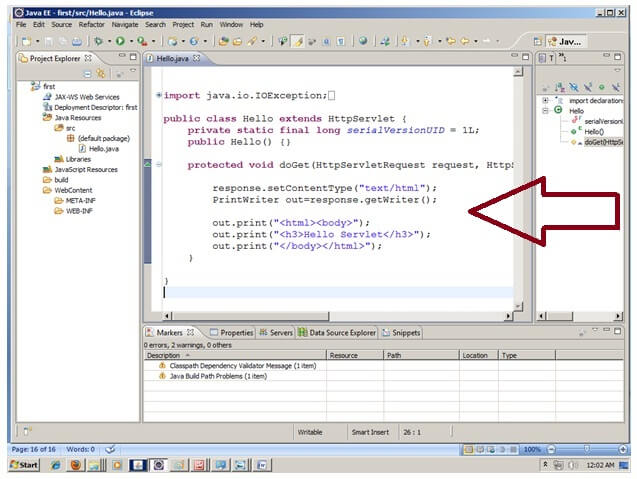






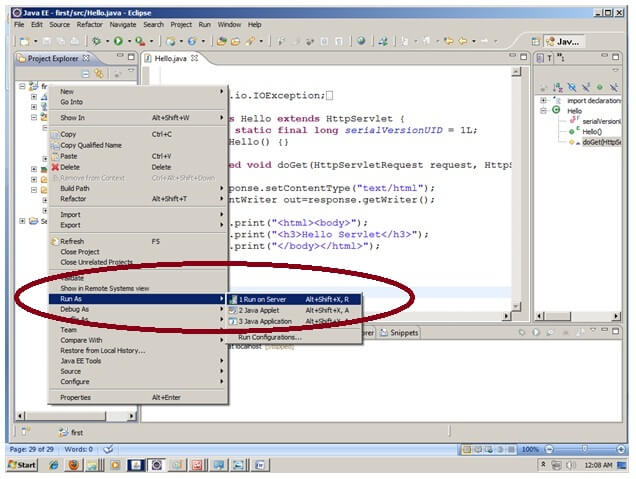


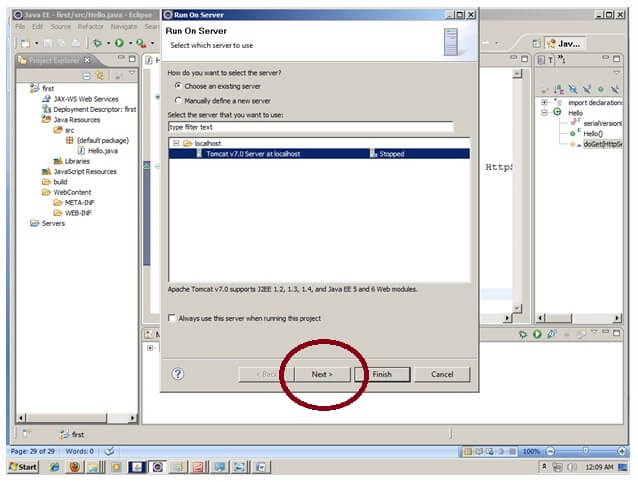
Now servlet has been created, Let's write the first servlet code.

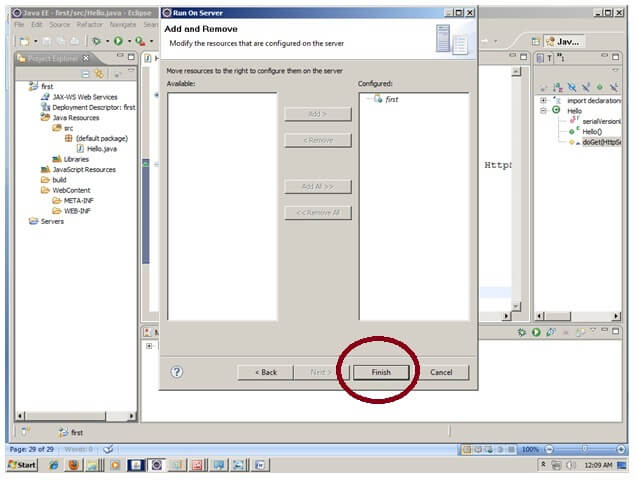


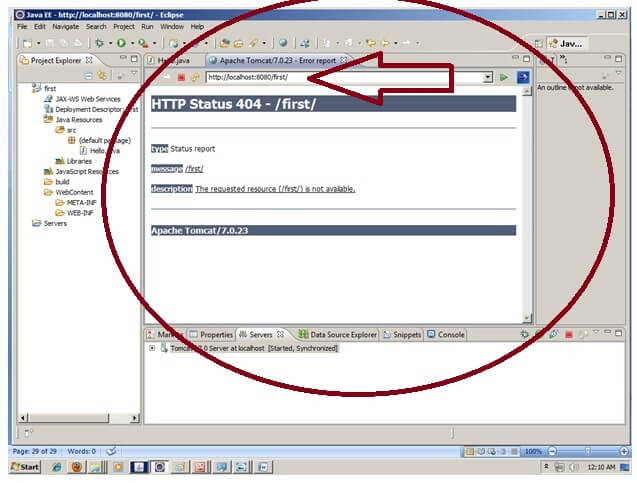
#### 4) Start the server and deploy the project:

For starting the server and deploying the project in one step, **Right click on your project -> Run As -> Run on Server -> choose tomcat server -> next -> addAll -> finish.**

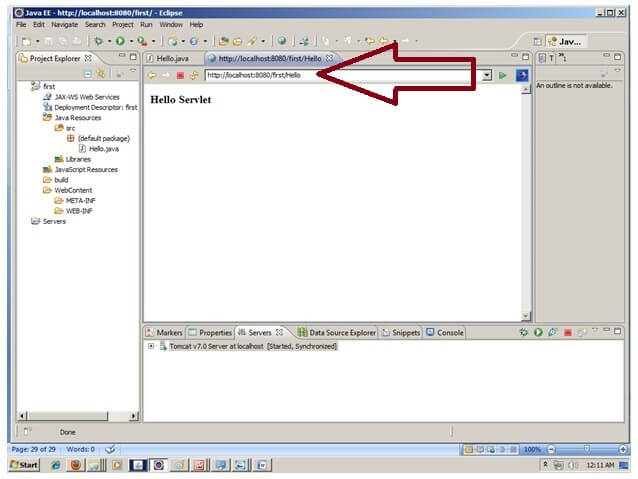








Now tomcat server has been started and project is deployed. To access the servlet write the url pattern name in the URL bar of the browser. In this case Hello then enter.

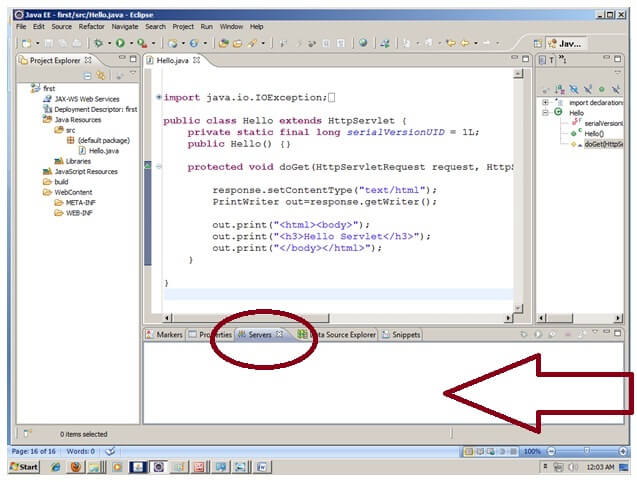


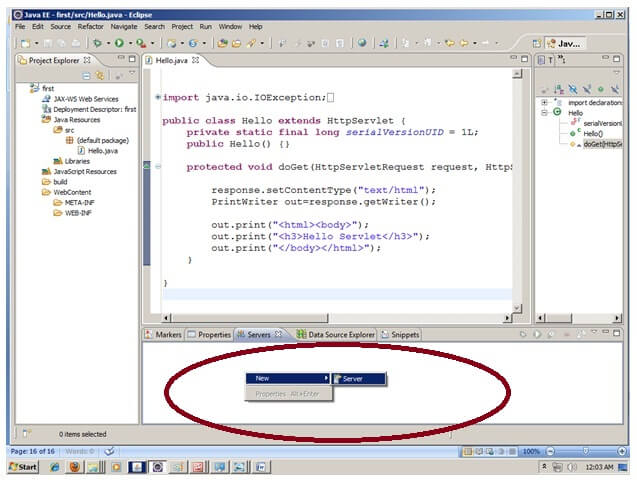
[download this example](https://static.javatpoint.com/src/servlet/firstservleteclipse.zip)

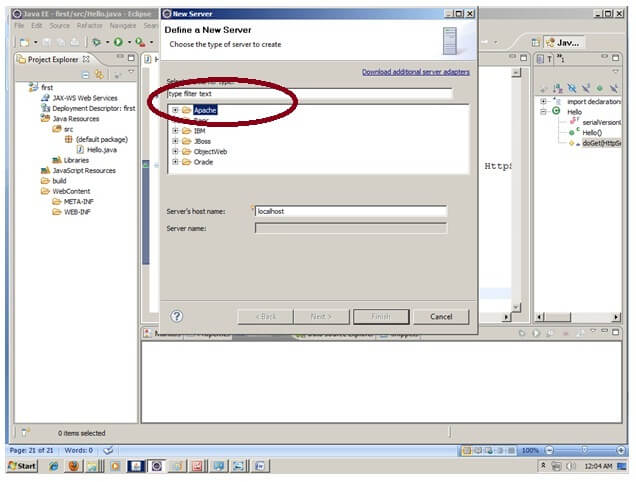
#### How to configure tomcat server in Eclipse ? (One time Requirement)

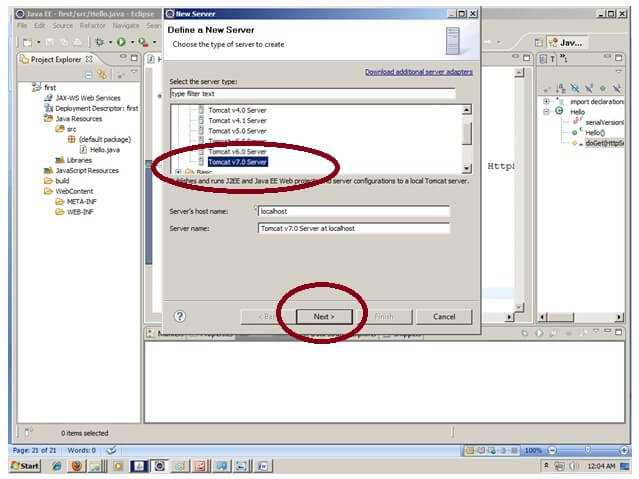
If you are using **Eclipse IDE first time**, you need to configure the tomcat server First.

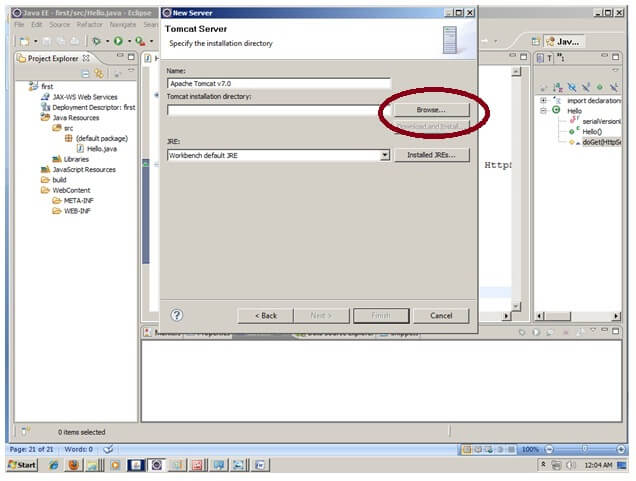
For configuring the tomcat server in eclipse IDE, **click on servers tab at the bottom side of the IDE -> right click on blank area -> New -> Servers -> choose tomcat then its version -> next -> click on Browse button -> select the apache tomcat root folder previous to bin -> next -> addAll -> Finish.**

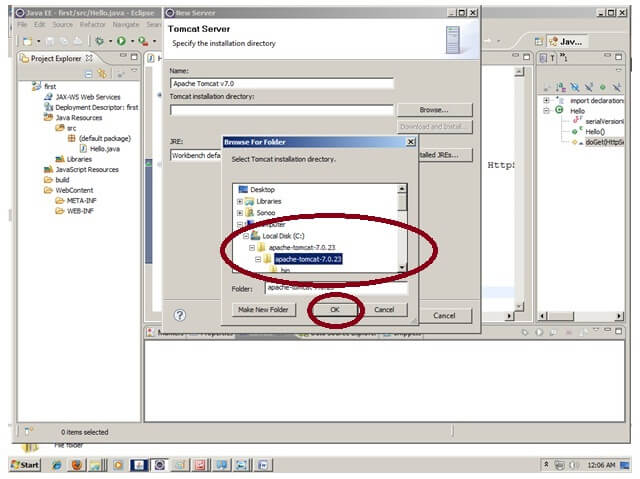


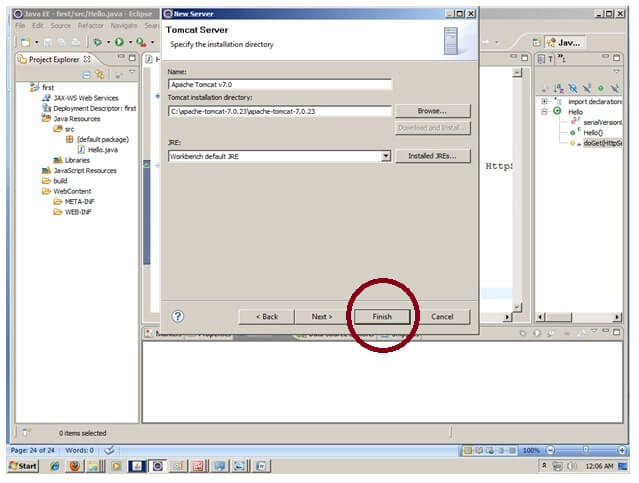




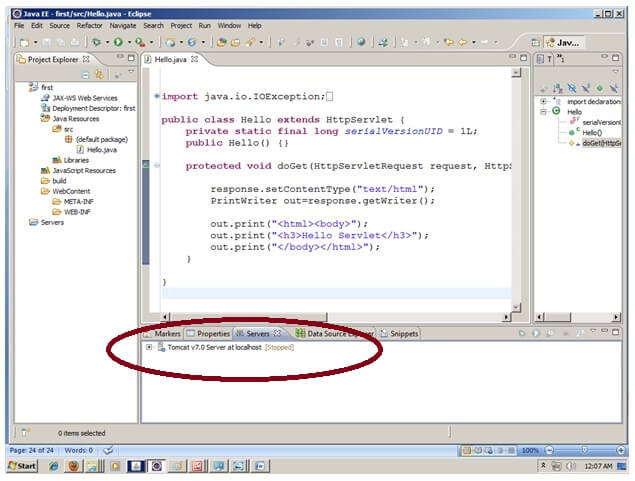








Now tomcat7 server has been configured in eclipse IDE.



Bottom of Form

Top of Form

#### ServletConfig Interface

An object of ServletConfig is created by the web container for each servlet. This object can be used to get configuration information from web.xml file.

If the configuration information is modified from the web.xml file, we don't need to change the servlet. So it is easier to manage the web application if any specific content is modified from time to time.

#### Advantage of ServletConfig

The core advantage of ServletConfig is that you don't need to edit the servlet file if information is modified from the web.xml file.

#### Methods of ServletConfig interface

1. **public String getInitParameter(String name):**Returns the parameter value for the specified parameter name.
2. **public Enumeration getInitParameterNames():**Returns an enumeration of all the initialization parameter names.
3. **public String getServletName():**Returns the name of the servlet.
4. **public ServletContext getServletContext():**Returns an object of ServletContext.

#### How to get the object of ServletConfig

1. **getServletConfig() method** of Servlet interface returns the object of ServletConfig.

#### Syntax of getServletConfig() method

1. **public** ServletConfig getServletConfig();

#### Example of getServletConfig() method

1. ServletConfig config=getServletConfig();
2. //Now we can call the methods of ServletConfig interface

#### Syntax to provide the initialization parameter for a servlet

The init-param sub-element of servlet is used to specify the initialization parameter for a servlet.

1. <web-app>
2. <servlet>
3. ......
5. <init-param>
6. <param-name>parametername</param-name>
7. <param-value>parametervalue</param-value>
8. </init-param>
9. ......
10. </servlet>
11. </web-app>

#### Example of ServletConfig to get initialization parameter

In this example, we are getting the one initialization parameter from the web.xml file and printing this information in the servlet.

**DemoServlet.java**

1. **import** java.io.\*;
2. **import** javax.servlet.\*;
3. **import** javax.servlet.http.\*;
5. **public** **class** DemoServlet **extends** HttpServlet {
6. **public** **void** doGet(HttpServletRequest request, HttpServletResponse response)
7. **throws** ServletException, IOException {
9. response.setContentType("text/html");
10. PrintWriter out = response.getWriter();
12. ServletConfig config=getServletConfig();
13. String driver=config.getInitParameter("driver");
14. out.print("Driver is: "+driver);
16. out.close();
17. }
19. }

**web.xml**

1. <web-app>
3. <servlet>
4. <servlet-name>DemoServlet</servlet-name>
5. <servlet-**class**>DemoServlet</servlet-**class**>
7. <init-param>
8. <param-name>driver</param-name>
9. <param-value>sun.jdbc.odbc.JdbcOdbcDriver</param-value>
10. </init-param>
12. </servlet>
14. <servlet-mapping>
15. <servlet-name>DemoServlet</servlet-name>
16. <url-pattern>/servlet1</url-pattern>
17. </servlet-mapping>
19. </web-app>

[download this example (developed in Myeclipse IDE)](https://static.javatpoint.com/src/servlet/config1.zip)

[download this example(developed in Eclipse IDE)](https://static.javatpoint.com/src/servlet/eclipse/config1.zip)

[download this example(developed in Netbeans IDE)](https://static.javatpoint.com/src/servlet/netbeans/config4.zip)

#### Example of ServletConfig to get all the initialization parameters

In this example, we are getting all the initialization parameter from the web.xml file and printing this information in the servlet.

**DemoServlet.java**

1. **import** java.io.IOException;
2. **import** java.io.PrintWriter;
3. **import** java.util.Enumeration;
5. **import** javax.servlet.ServletConfig;
6. **import** javax.servlet.ServletException;
7. **import** javax.servlet.http.HttpServlet;
8. **import** javax.servlet.http.HttpServletRequest;
9. **import** javax.servlet.http.HttpServletResponse;

12. **public** **class** DemoServlet **extends** HttpServlet {
13. **public** **void** doGet(HttpServletRequest request, HttpServletResponse response)
14. **throws** ServletException, IOException {
16. response.setContentType("text/html");
17. PrintWriter out = response.getWriter();
19. ServletConfig config=getServletConfig();
20. Enumeration<String> e=config.getInitParameterNames();
22. String str="";
23. **while**(e.hasMoreElements()){
24. str=e.nextElement();
25. out.print("<br>Name: "+str);
26. out.print(" value: "+config.getInitParameter(str));
27. }
29. out.close();
30. }
32. }

**web.xml**

1. <web-app>
3. <servlet>
4. <servlet-name>DemoServlet</servlet-name>
5. <servlet-**class**>DemoServlet</servlet-**class**>
7. <init-param>
8. <param-name>username</param-name>
9. <param-value>system</param-value>
10. </init-param>
12. <init-param>
13. <param-name>password</param-name>
14. <param-value>oracle</param-value>
15. </init-param>
17. </servlet>
19. <servlet-mapping>
20. <servlet-name>DemoServlet</servlet-name>
21. <url-pattern>/servlet1</url-pattern>
22. </servlet-mapping>
24. </web-app>

[download this example (developed in Myeclipse IDE)](https://static.javatpoint.com/src/servlet/config2.zip)

[download this example(developed in Eclipse IDE)](https://static.javatpoint.com/src/servlet/eclipse/config2.zip)

[download this example(developed in Netbeans IDE)](https://static.javatpoint.com/src/servlet/netbeans/config5.zip)

Bottom of Form

Top of Form

**Video Lectures :**

https://youtu.be/ewiOaDitBBw

**Reference Links:**

1.http://docs.oracle.com/cd/B31017\_01/web.1013/b28959/sess ions.htm

2.http://www.tutorialspoint.com/jsp/jsp\_cookies\_handling.htm

3.    <http://www.javatpoint.com/sonoojaiswal/servletconfig>

4.    https://www.javatpoint.com/creating-servlet-in-eclipse-ide

Bottom of Form