


Struts Configuration & Hello World Program

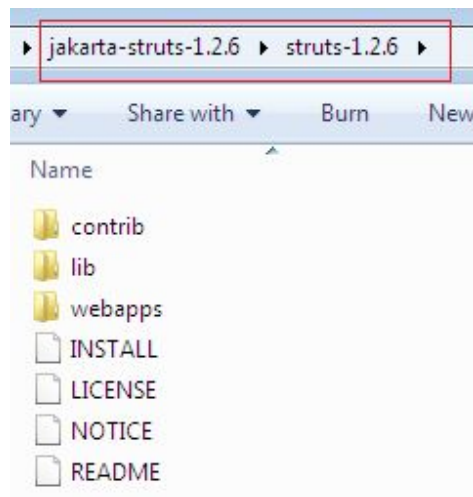
Configuring Struts with Eclipse

You can download jakarta-struts-1.2.6, from the official site of Apache.

Before starting this installation I'll assume that you have configured java, Tomcat 6.0(or higher versions) with eclipse Helios.

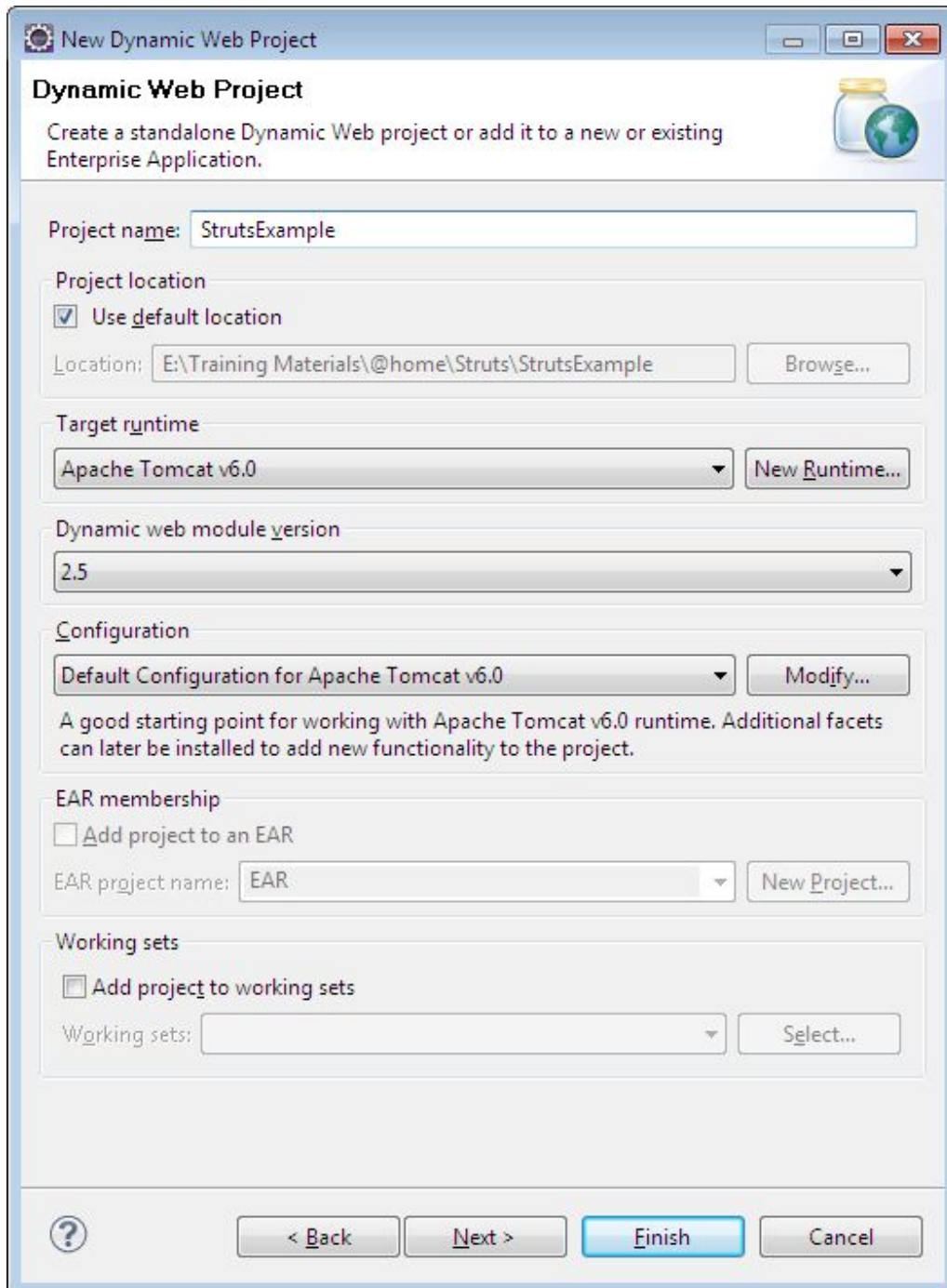
Extract  jakarta-struts-1.2.6

After extracting you should have the file content as shown below



After this create a dynamic project in Eclipse

Struts Configuration & Hello World Program



New Dynamic Web Project

Create a standalone Dynamic Web project or add it to a new or existing Enterprise Application.

Project name:

Project location

☒ Use default location

Location:

Target runtime

Dynamic web module version

Configuration

A good starting point for working with Apache Tomcat v6.0 runtime. Additional facets can later be installed to add new functionality to the project.

EAR membership

☐ Add project to an EAR

EAR project name:

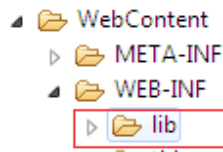
Working sets

☐ Add project to working sets

Working sets:

Click on finish, once after this we have to copy all the library files (i.e., jar) of Struts to lib folder of Eclipse

Struts Configuration & Hello World Program



As shown below

jakarta-struts-1.2.6 > struts-1.2.6 > lib			
Share with ▼ Burn New folder			
Name	Date modified	Type	Size
validator-rules	11/20/2004 1:43 PM	XML File	12 KB
struts-tiles	11/20/2004 1:43 PM	TLD File	8 KB
struts-nested	11/20/2004 1:43 PM	TLD File	69 KB
struts-logic	11/20/2004 1:43 PM	TLD File	15 KB
struts-html	11/20/2004 1:43 PM	TLD File	72 KB
struts-bean	11/20/2004 1:43 PM	TLD File	9 KB
struts	11/20/2004 1:43 PM	Executable Jar File	519 KB
jakarta-oro	11/20/2004 1:43 PM	Executable Jar File	64 KB
commons-validator	11/20/2004 1:43 PM	Executable Jar File	83 KB
commons-logging	11/20/2004 1:43 PM	Executable Jar File	38 KB
commons-fileupload	11/20/2004 1:43 PM	Executable Jar File	22 KB
commons-digester	11/20/2004 1:43 PM	Executable Jar File	165 KB
commons-beanutils	11/20/2004 1:43 PM	Executable Jar File	185 KB
antlr	11/20/2004 1:43 PM	Executable Jar File	380 KB
web-app_2_3	11/20/2004 1:43 PM	DTD File	33 KB
web-app_2_2	11/20/2004 1:43 PM	DTD File	16 KB
tiles-config_1_1	11/20/2004 1:43 PM	DTD File	13 KB
struts-config_1_2	11/20/2004 1:43 PM	DTD File	35 KB
struts-config_1_1	11/20/2004 1:43 PM	DTD File	35 KB
struts-config_1_0	11/20/2004 1:43 PM	DTD File	19 KB

1. Copy all Executable jar into lib folder
2. Copy all tld files in the folder folder tld(this is any name), create in WEB-INF and copy all TLD file
3. And copy struts-config_1_2.DTD in the tld folder

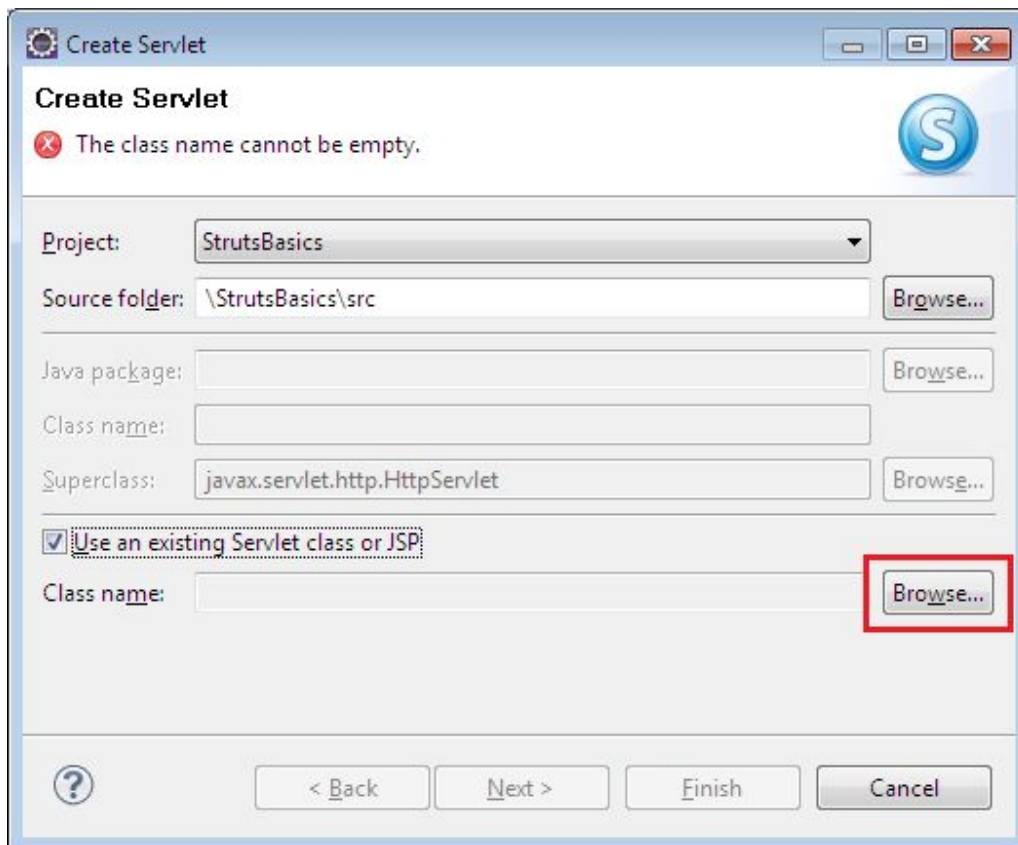
If you look at the web.xml file it does not contain any servlets

Struts Configuration & Hello World Program

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://java.sun.com/
<display-name>StrutsBasics</display-name>
<welcome-file-list>
  <welcome-file>index.html</welcome-file>
  <welcome-file>index.htm</welcome-file>
  <welcome-file>index.jsp</welcome-file>
  <welcome-file>default.html</welcome-file>
  <welcome-file>default.htm</welcome-file>
  <welcome-file>default.jsp</welcome-file>
</welcome-file-list>

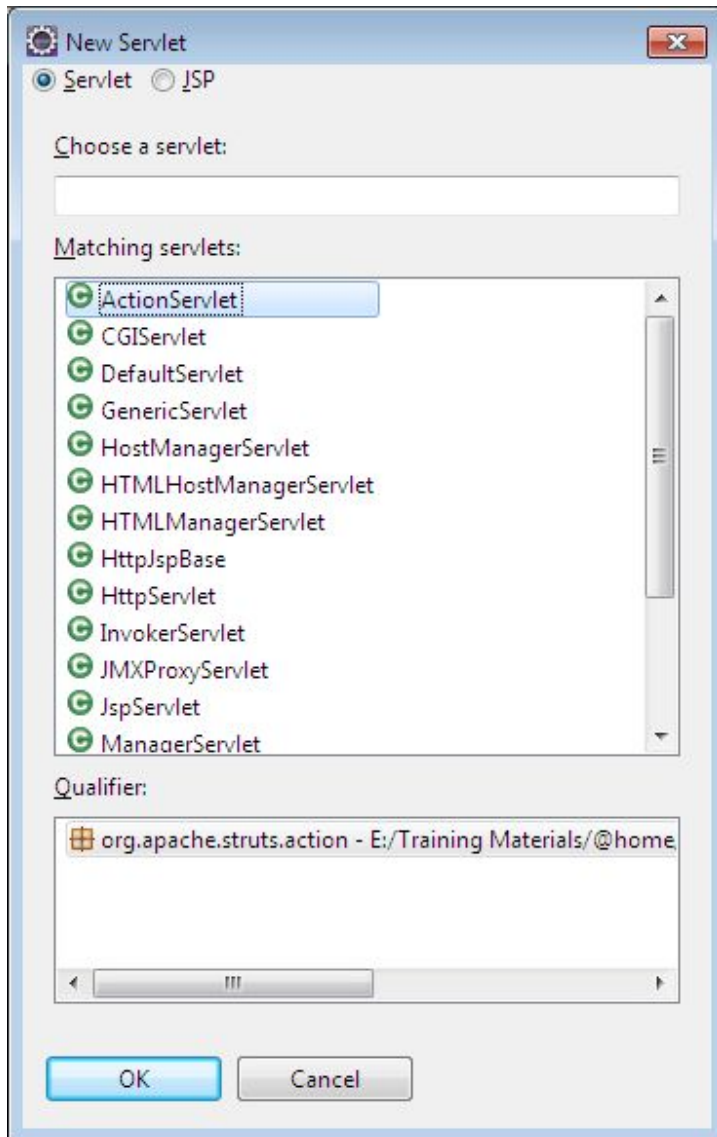
</web-app>
```

we have to have the front servlet for doing this, create a new servlet on the project, and select existing servlet in the popup window



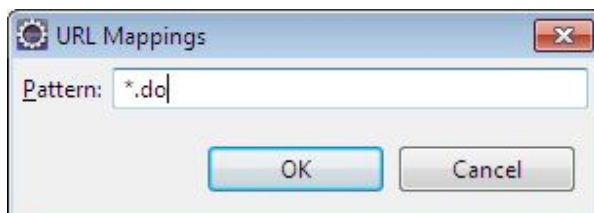
This servlet is given by struts.jar.

Struts Configuration & Hello World Program



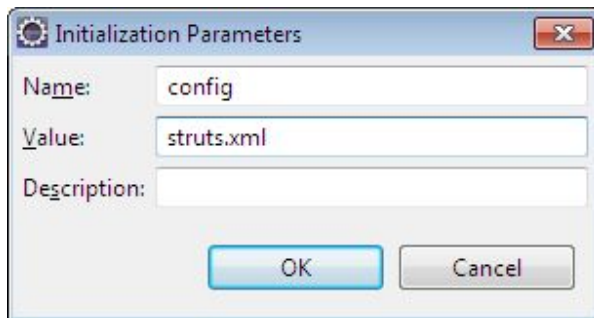
Select action servlet and click ok.

Give the URL Mappings to any file ending with customized extension, this can be changed to any thing.



Struts Configuration & Hello World Program

Initial parameters



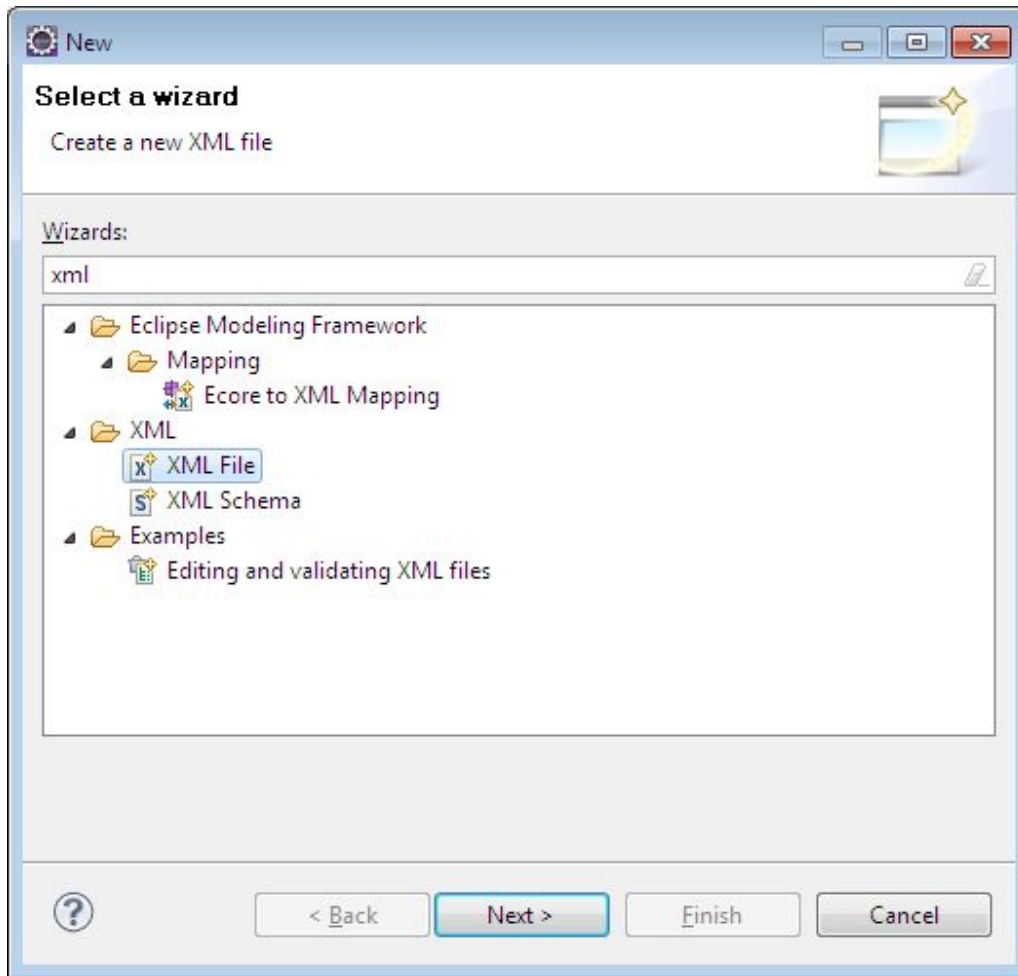
The file name is not mandatory to be struts.xml (in some books you will see it as struts-config.xml this is default), in such cast this initialization parameter is optional.

Now web.xml has an entry of servlet

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://java.sun.com
  <display-name>StrutsBasics</display-name>
  <welcome-file-list>
    <welcome-file>index.html</welcome-file>
    <welcome-file>index.htm</welcome-file>
    <welcome-file>index.jsp</welcome-file>
    <welcome-file>default.html</welcome-file>
    <welcome-file>default.htm</welcome-file>
    <welcome-file>default.jsp</welcome-file>
  </welcome-file-list>
  <servlet>
    <description></description>
    <display-name>ActionServlet</display-name>
    <servlet-name>ActionServlet</servlet-name>
    <servlet-class>org.apache.struts.action.ActionServlet</servlet-class>
    <init-param>
      <description></description>
      <param-name>config</param-name>
      <param-value>/WEB-INF/struts.xml</param-value>
    </init-param>
  </servlet>
  <servlet-mapping>
    <servlet-name>ActionServlet</servlet-name>
    <url-pattern>*.do</url-pattern>
  </servlet-mapping>
</web-app>
```

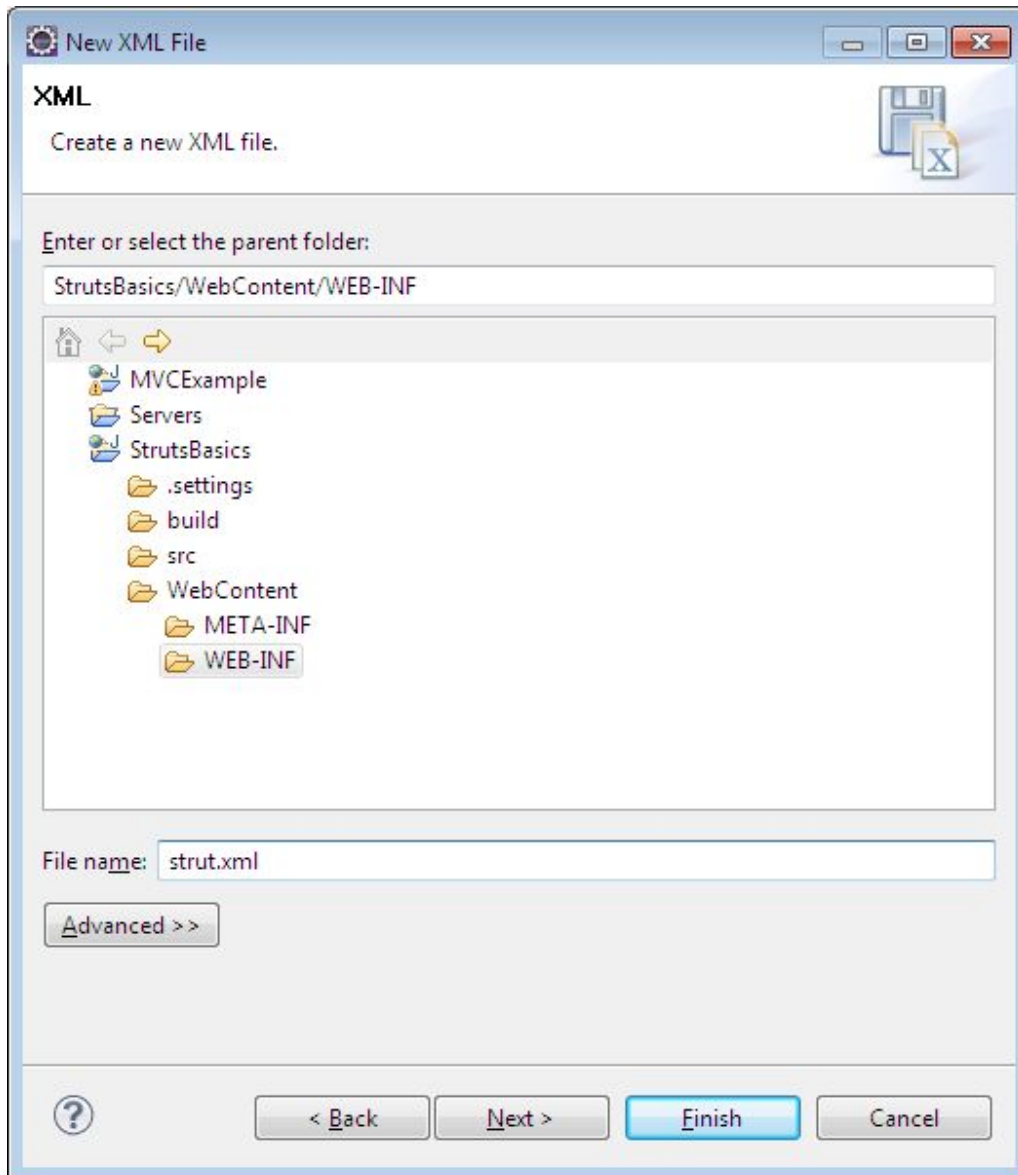
Create a xml file by name struts.xml in WEB-INF.

Struts Configuration & Hello World Program



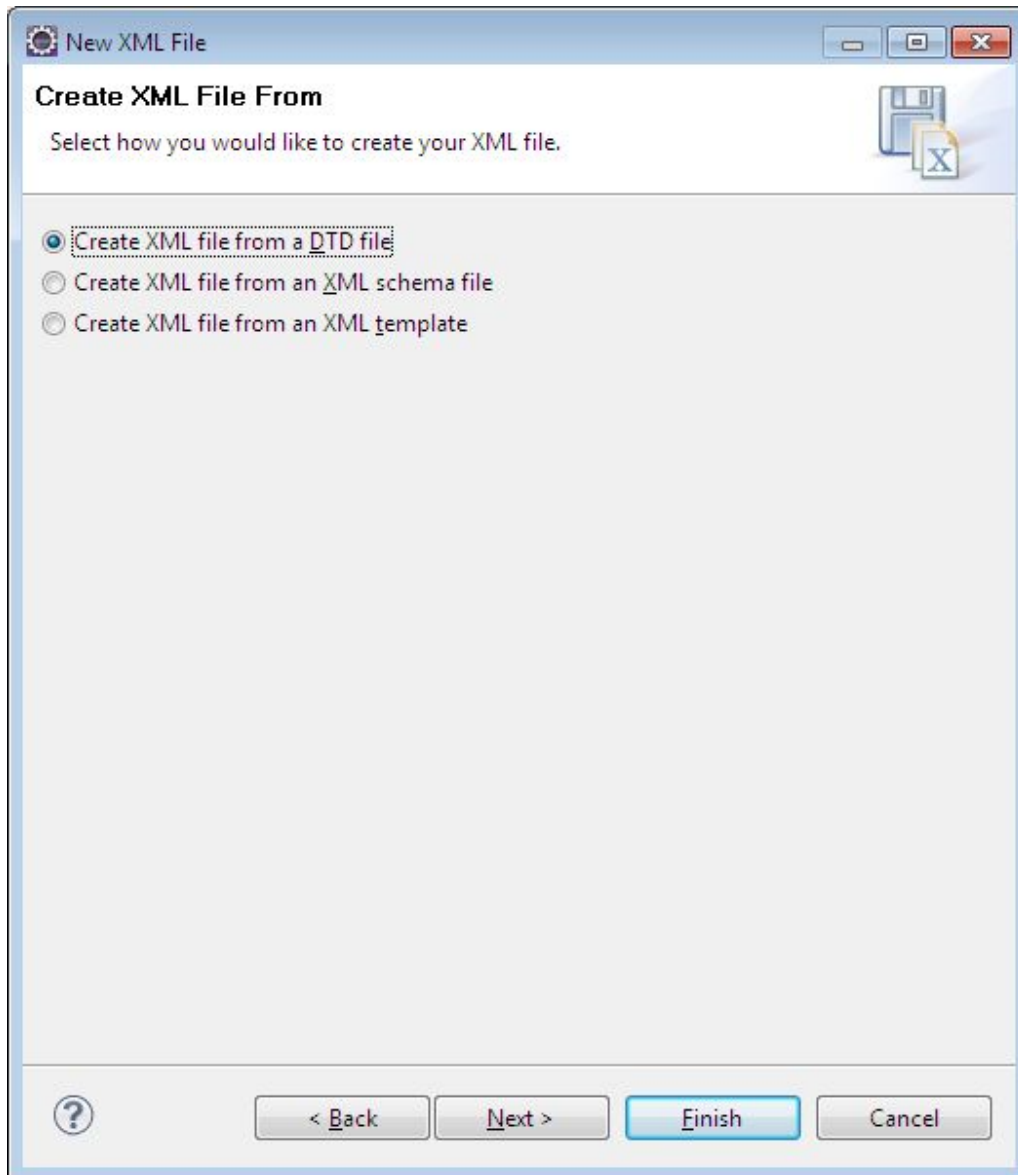
Name it as struts.xml

Struts Configuration & Hello World Program



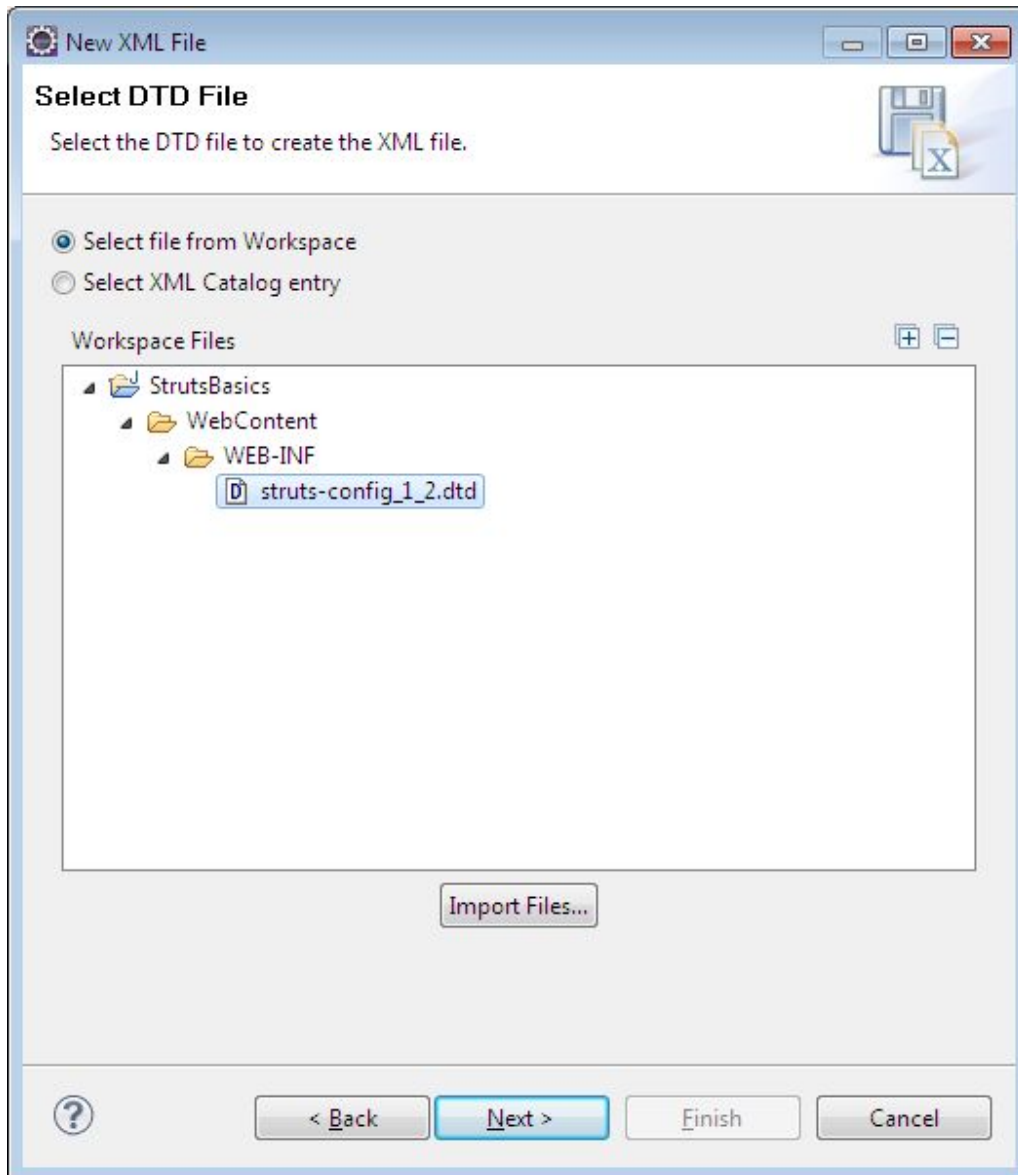
Select the option create XML from DTD

Struts Configuration & Hello World Program



Select the DTD file which was included

Struts Configuration & Hello World Program



And all other configurations keep it default.

Once after than make an entry of the path and execute method in struts.xml file

Struts Configuration & Hello World Program

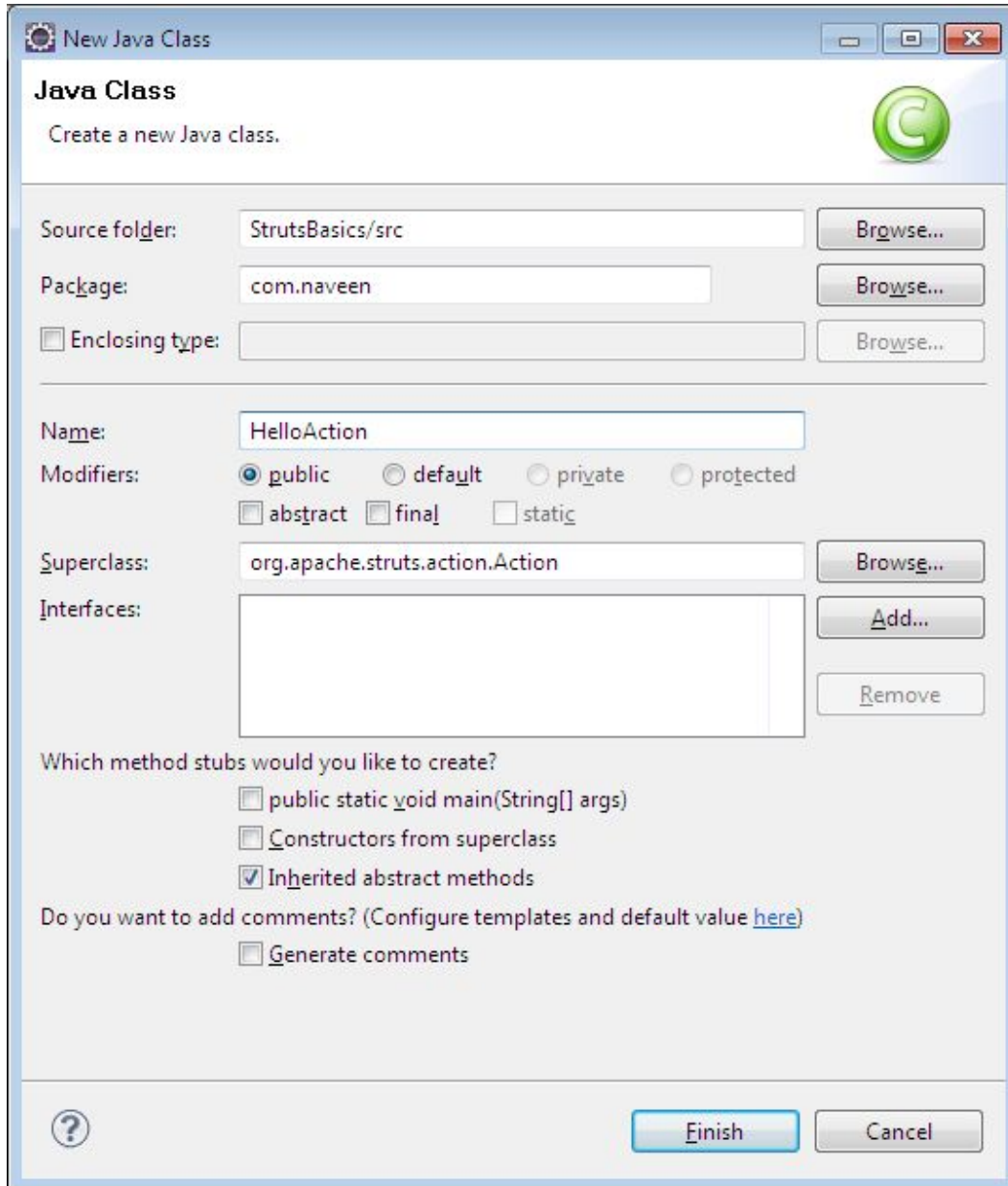
```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE struts-config SYSTEM "struts-config_1_2.dtd" >
<struts-config>
  <action-mappings>
    <action path="/hello" type="com.naveen.HelloAction">
      <forward name="view1" path="/hello.jsp" />
    </action>
  </action-mappings>
</struts-config>
```

URL

Execute Class

And create a class which extends Action, this is a kind of contract with the struts

Struts Configuration & Hello World Program



New Java Class

Create a new Java class.

Source folder: StrutsBasics/src Browse...

Package: com.naveen Browse...

☐ Enclosing type: Browse...

Name: HelloAction

Modifiers: ☒ public ☐ default ☐ private ☐ protected
☐ abstract ☐ final ☐ static

Superclass: org.apache.struts.action.Action Browse...

Interfaces: Add...
Remove

Which method stubs would you like to create?

☐ public static void main(String[] args)
☐ Constructors from superclass
☒ Inherited abstract methods

Do you want to add comments? (Configure templates and default value [here](#))
☐ Generate comments

? Finish Cancel

Then override execute method in the class,

Struts Configuration & Hello World Program

```
public class HelloAction extends Action {

    @Override
    public ActionForward execute(ActionMapping mapping, ActionForm form,
        HttpServletRequest request, HttpServletResponse response)
        throws Exception {

        String message = "hello Naveen ";
        request.setAttribute("message", message);
        return mapping.findForward("view1");
    }

}
```

And the view.jsp is created

```
<%@ taglib uri="/WEB-INF/tld/struts-bean.tld" prefix="bean" %>

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Insert title here</title>
</head>
<body>

<h1>Message from Friend</h1>

<bean:write name="message" scope="request" />
</body>
</html>
```

Finally we will have index.html which will call

```
<body>
<h1>Home</h1>

<a href="hello.do">Check a message</a>
</body>
```