Struts2 Hello World 示例

通过学习 Struts2 框架可以了解到，当你在 Struts2 的 web 应用程序里点击一个超链接或提交一个 HTML 表单时，会由控制器收集输入并发送一个叫 Actions 的 Java 类。Action 被执行后，Result 会选择一个资源给予响应。这个资源通常是一个 JSP，也可以是一个 PDF 文件，一个 Excel 表格，或者是一个 Java 小程序窗口。

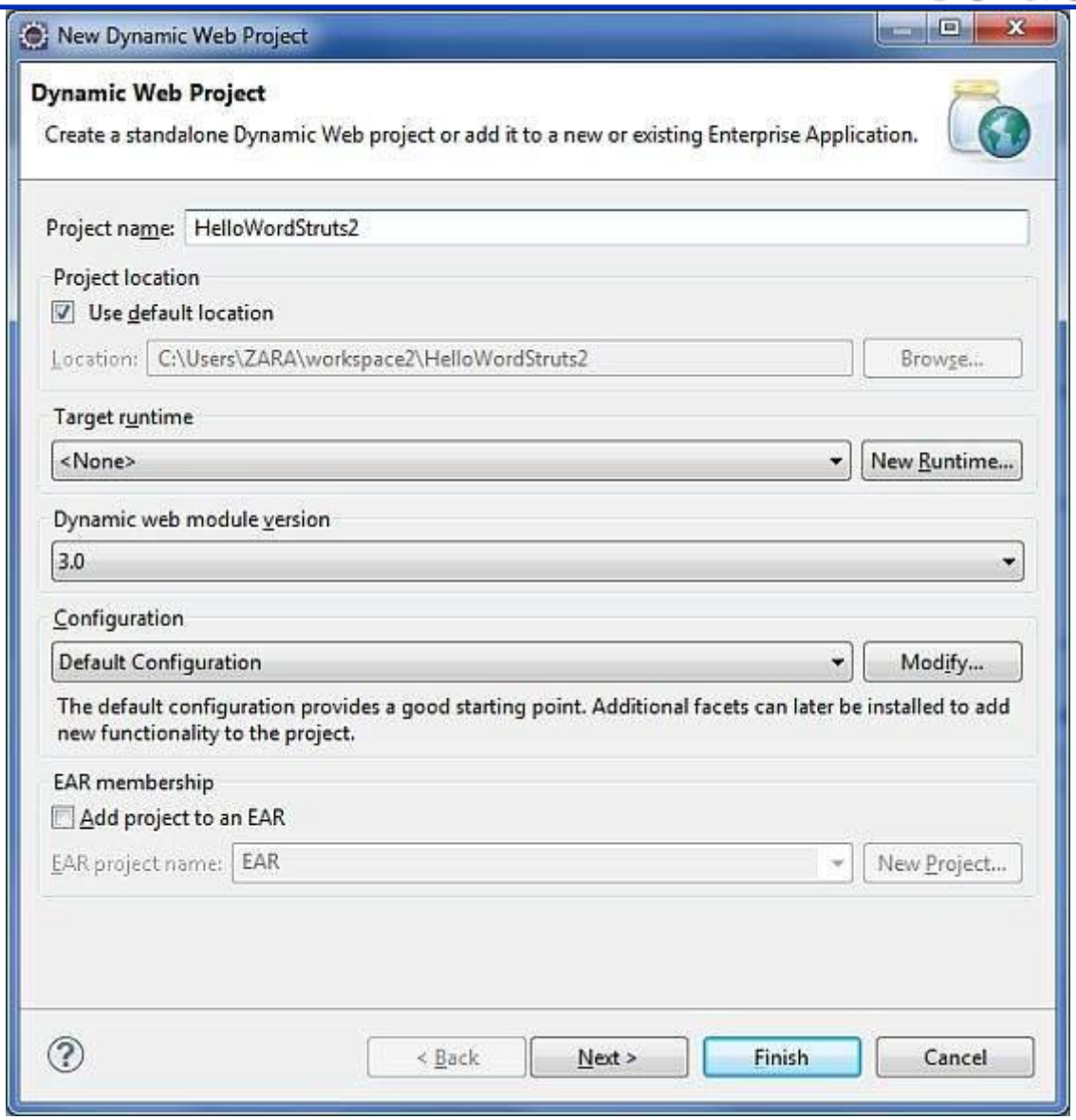
假设你已经建好了你的开发环境，那么现在让我们继续构建第一个 Struts2 项目：**Hello World** 。这个项目的目标是构建一个收集用户名并在用户名后跟随显示“Hello World”的 web 应用程序。我们需要为每个 Struts2 项目构建以下四个组件：

序号	名称及描述
1	Action（操作） 创建一个动作类，包含完整的业务逻辑并控制用户、模型以及视图间的交互。
2	Interceptors（拦截器） 这是控制器的一部分，可依据需求创建拦截器，或使用现有的拦截器。
3	View（视图） 创建一个 JSP 与用户进行交互，获取输入并呈现最终信息。
4	Configuration Files（配置文件） 创建配置文件来连接动作、视图以及控制器，这些文件分别是 struts.xml、web.xml 以及 strut

我们如果打算使用 Eclipse IDE，那么所有必需的组件都要在动态 Web 项目（Dynamic Web Project）下创建。因此我们就先从创建动态 Web 项目开始。

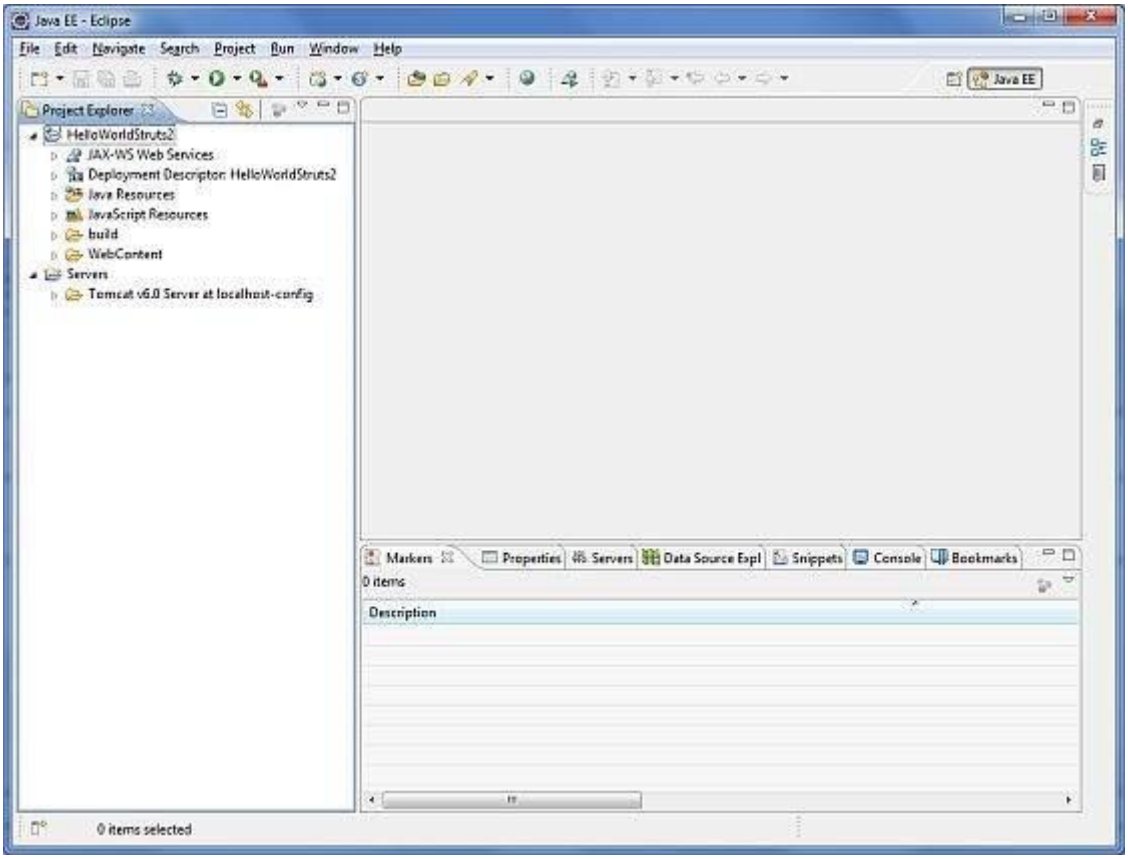
创建一个动态 Web 项目：

启动你的 Eclipse，然后打开“**File**”>“**New**”>“**Dynamic Web Project**”，输入“**HelloWorldStruts2**”的项目名称，参照下图设置其他选项：



按照下图选择所有默认选项，最后检查 **Generate Web.xml deployment descriptor** 选项。

这个将在 Eclipse 为你创建一个动态 web 项目。现在点击“**Windows**”>“**Show**”>“**View**”>“**Project Explorer**”，你就可以看到你的项目窗口，如下图：



现在从 Struts2 的 lib 文件夹 **C:\struts-2.2.3\lib** 中拷贝以下文件到项目的 **WEB-INF\lib** 文件夹里。你可以直接拖拽以下所有文件到 WEB-INF\lib 文件夹。

- commons-fileupload-x.y.z.jar
- commons-io-x.y.z.jar
- commons-lang-x.y.jar
- commons-logging-x.y.z.jar
- commons-logging-api-x.y.jar
- freemarker-x.y.z.jar
- javassist-x.y.z.GA
- ognl-x.y.z.jar
- struts2-core-x.y.z.jar

- xwork-core.x.y.z.jar

创建 Aciton 类

Action 类是 Struts2 应用程序的关键，我们通过它实现大部分的业务逻辑。那么让我们在**“Java Resources”>“src”**的类目下创建一个名称为“HelloWorldAction.java”的 java 文件夹，使用一个命名为**“cn.w3cschool.struts2”**并包含以下内容的资源包。

当用户点击一个 URL 时，由 Action 类来响应用户操作。一个或多个 Action 类的方法被执行，并返回一个字符串结果。基于结果的值，会呈现一个特定的 JSP 页面。

```
package cn.w3cschool.struts2;


public class HelloWorldAction{

   private String name;


   public String execute() throws Exception {

      return "success";

   }


   public String getName() {

      return name;

   }


   public void setName(String name) {

      this.name = name;

   }
```

```
}
```

这是一个非常简单的具有“name”属性的类。对于“name”属性，我们用标准的 getter 和 setter 方法，以及一个返回“success”字符串的执行方法。

Struts2 框架将创建一个“HelloWorldAction”类的对象，并调用 execute 方法来响应用户的动作。你把你的业务逻辑放进 execute 方法里，最后会返回字符串常量。简单的描述每个 URL，你需要实现一个 Action 类，你也可以用类名直接作为你的动作名，或者如下面内容所示使用 struts.xml 文件映射到其他 name 上。

创建视图

我们需要一个 JSP 来呈现最终的信息，当一个预定义动作发生时这个页面将被 Struts2 框架调用，并且这个映像会定义到 struts.xml 文件里。那么让我们在你的 Eclipse 项目的 WebContent 文件夹里创建以下 JSP 文件 **HelloWorld.jsp** 。在 project explorer 中右键点击 WebContent 文件夹并选择**“New”>“JSP File”**。

```
<%@ page contentType="text/html; charset=UTF-8" %>
<%@ taglib prefix="s" uri="/struts-tags" %>
<html>
<head>
<title>Hello World</title>
</head>
<body>
   Hello World, <s:property value="name"/>
</body>
</html>
```

Taglib 指令告知 Servlet 容器这个页面将使用 Struts2 标签，并且这些标签会被 s 放在前面。s:property 标签显示 Action 类“name”属性的值，这个值是使用 HelloWorldAction 类的 **getName()** 方法返回的。

创建主页

在 WebContent 文件夹里，我们还需要创建 **index.jsp** 文件，这个文件是用作初始的 action URL。用户可以通过点击它命令 Struts2 框架去调用 HelloWorldAction 类的定义方法并呈现 HelloWorld.jsp 视图。

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
   pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="s" uri="/struts-tags"%>
   <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title>Hello World</title>
</head>
<body>
   <h1>Hello World From Struts2</h1>
   <form action="hello">
      <label for="name">Please enter your name</label><br/>
      <input type="text" name="name"/>
      <input type="submit" value="Say Hello"/>
   </form>
```

```
</body>

</html>
```

上面视图文件里定义的 **hello** action 将通过 struts.xml 文件影射到 HelloWorldAction 类及其 execute 方法。当用户点击提交按钮时，将使得 Struts2 框架运行 HelloWorldAction 类中的 execute 方法，并基于该方法的返回值，选择一个适当的视图作为响应进行呈现。

配置文件

我们需要一个映像把 URL、HelloWorldAction 类（模型）以及 HelloWorld.jsp（视图）联系在一起。映像告知 Struts2 框架哪个类将响应用户的动作（URL），类里的哪个方法将要执行，以及基于方法所返回的字符串结果，会呈现怎样的视图。

那么接下来让我们创建一个名为 **struts.xml** 的文件。因为 Struts2 要求 strust.xml 文件显示在 classes 的文件夹里，所以我们要在 WebContent/WEB-INF/classes 的文件夹下创建 struts.xml 文件。Eclipse 并没有默认创建“classes”文件夹，因此你需要自己创建。在 project explorer 里右键点击 WEB-INF 文件夹并选择**“New”>“Folder”**，你的 struts.xml 文件应该如下：

```
<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE struts PUBLIC

   "-//Apache Software Foundation//DTD Struts Configuration 2.0//EN"

   "http://struts.apache.org/dtds/struts-2.0.dtd">

<struts>

<constant name="struts.devMode" value="true" />

   <package name="helloworld" extends="struts-default">
```

```
      <action name="hello"
            class="cn.w3cschool.struts2.HelloWorldAction"
            method="execute">
            <result name="success">/HelloWorld.jsp</result>
      </action>
   </package>
</struts>
```

这里说几句关于上述的配置文件。这里我们设定常数 **struts.devMode** 的值为**真**，因为我们是在开发环境下工作，需要查看一些有用的日志消息。然后，我们定义一个名为 **helloworld** 的数据包。当你想要把你的 Actions 集合在一起时，创建一个数据包是非常有用的。在我们的示例中，我们命名我们的动作为“hello”，与 URL **/hello.action** 保持一致，由 **HelloWorldAction.class** 进行备份。**HelloWorldAction.class** 的 **execute** 方法就是当 URL **/hello.action** 被调用时运行。如果 execute 方法返回的结果为“success”，那么我们带用户进入 **HelloWorld.jsp**。

下一步是创建一个 **web.xml** 文件，这是一个适用于 Struts2 任何请求的接入点。在部署描述符（web.xml）中，Struts2 应用程序的接入点将会定义为一个过滤器。因此我们将在 web.xml 里定义一个 org.apache.struts2.dispatcher.FilterDispatcher 类的接入点，而 web.xml 文件需要在 WebContent 的 WEB-INF 文件夹下创建。Eclipse 已经创建了一个基础的 web.xml 文件，你在创建项目的时候可以使用。那么，让我们参照以下内容做修改：

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xmlns="http://java.sun.com/xml/ns/javaee"
   xmlns:web="http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
```

```
   xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
   http://java.sun.com/xml/ns/javaee/web-app_3_0.xsd"
   id="WebApp_ID" version="3.0">

   <display-name>Struts 2</display-name>
   <welcome-file-list>
      <welcome-file>index.jsp</welcome-file>
   </welcome-file-list>
   <filter>
      <filter-name>struts2</filter-name>
      <filter-class>
         org.apache.struts2.dispatcher.FilterDispatcher
      </filter-class>
   </filter>

   <filter-mapping>
      <filter-name>struts2</filter-name>
      <url-pattern>/*</url-pattern>
   </filter-mapping>
</web-app>
```

我们指定了 index.jsp 作为我们的欢迎文件，那么我们已经配置好了在所有的 URL（列如：所有匹配/*模式的 URL）上运行 Struts2 过滤器。

注意：

如果它是 struts2-core-2.5.jar，那么将 web.xml 中的过滤器类标记值更改为

```
<filter-class>
```

```
    org.apache.struts2.dispatcher.filter.StrutsPrepareAndExecuteFilter
</filter-class>
```

如果它是 struts2-core-2.1.3.jar，那么将 web.xml 中的过滤器类标记值更改为

```
<filter-class>
    org.apache.struts2.dispatcher.FilterDispatcher
</filter-class>
```

自 Struts 2.1.3 以来，FilterDispatcher 就不推荐使用了。如果您使用的是较旧的版本，则用户高于解决方案。

如果它是 struts2-core-2.3.X.jar，那么将 web.xml 中的过滤器类标记值更改为

```
<filter-class>
    org.apache.struts2.dispatcher.ng.filter.StrutsPrepareAndExecuteFilter
</filter-class>
```

启用详细日志

通过在 **WEB-INF/classes** 文件夹下创建 **logging.properties** 文件，可以实现在使用 Struts 2 时启用完整的日志记录功能。 属性文件中需保留以下两行：：

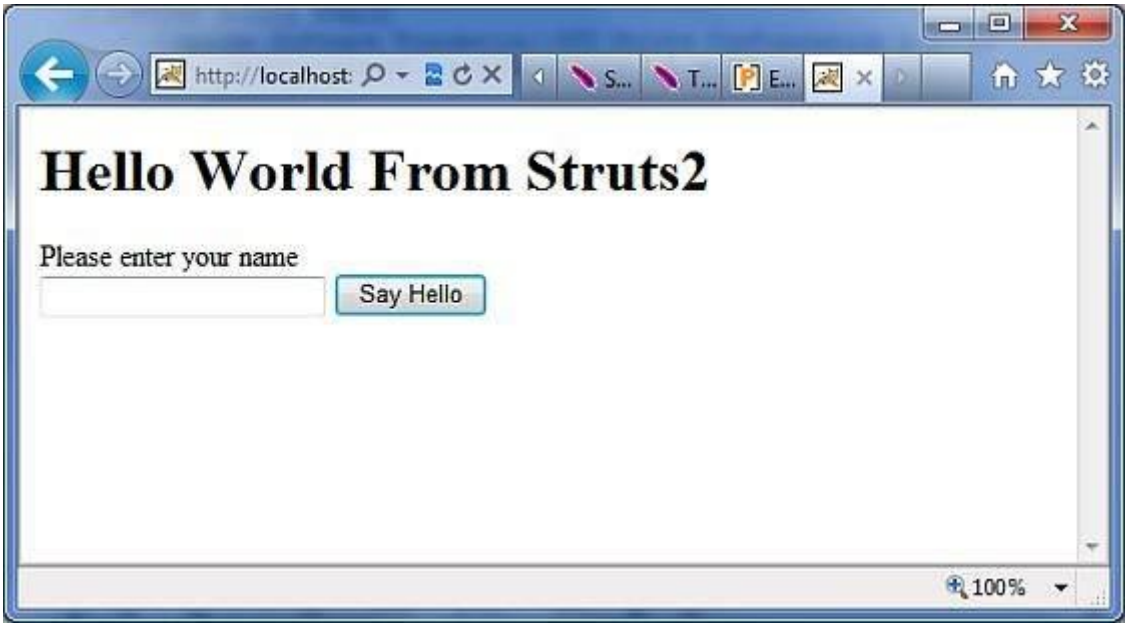
```
org.apache.catalina.core.ContainerBase.[Catalina].level = INFO
org.apache.catalina.core.ContainerBase.[Catalina].handlers = \
                            java.util.logging.ConsoleHandler
```

默认由 logging.properties 指定一个 ConsoleHandler 将日志记录按指定路线发送给 stdout 和 FileHandler。程序运行日志的级别阈值可以使用 SEVERE，WARNING，INFO，CONFIG，FINE，FINER，FINEST 或者 ALL。

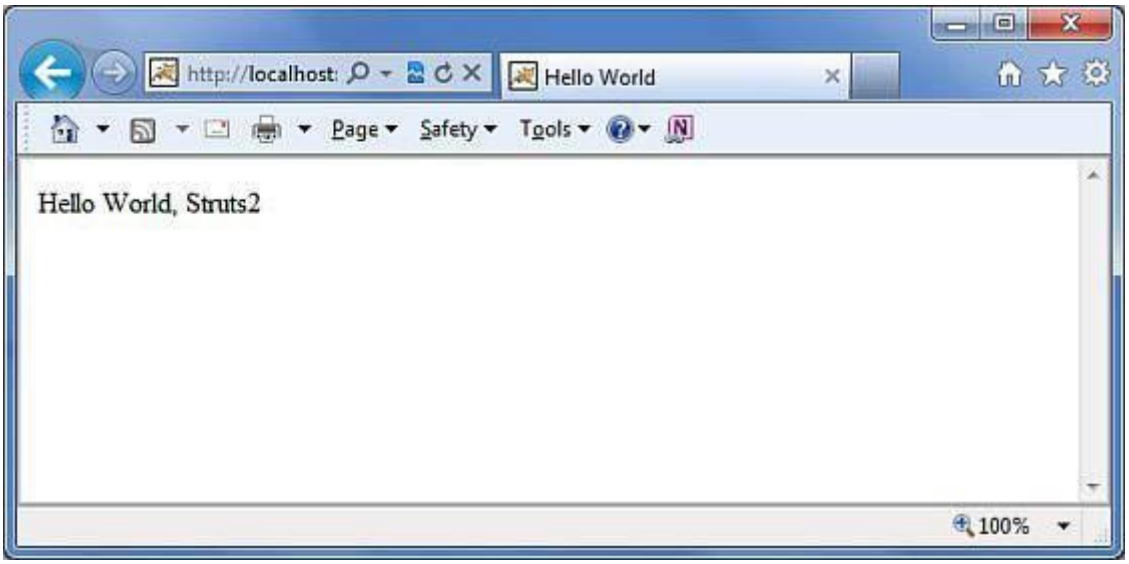
这样，我们就准备好使用 Struts 2 运行我们的 Hello World 程序了。

执行应用程序

右键点击项目名称，接着点击“**Export**”>“**WAR File**”创建 WAR 文件，然后将 WAR 部署到 Tomcat 的 webapps 目录中。最后，启动 Tomcat 服务器并尝试访问 URL http://localhost:8080/HelloWorldStruts2/index.jsp ，将会呈现如下图所示的结果：



输入一个“Struts2”值并提交页面，你可以看到以下页面



注意，你可以在 struts.xml 文件中定义一个索引作为操作，这样你可以调用索引页面 http://localhost:8080/HelloWorldStruts2/index.action 。查看下面是怎样定义索引作为操作：

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE struts PUBLIC
   "-//Apache Software Foundation//DTD Struts Configuration 2.0//EN"
   "http://struts.apache.org/dtds/struts-2.0.dtd">
<struts>
<constant name="struts.devMode" value="true" />
   <package name="helloworld" extends="struts-default">

      <action name="index">
            <result >/index.jsp</result>
      </action>

      <action name="hello"
            class="cn.w3cschool.struts2.HelloWorldAction"
            method="execute">
            <result name="success">/HelloWorld.jsp</result>
      </action>

   </package>
</struts>
```