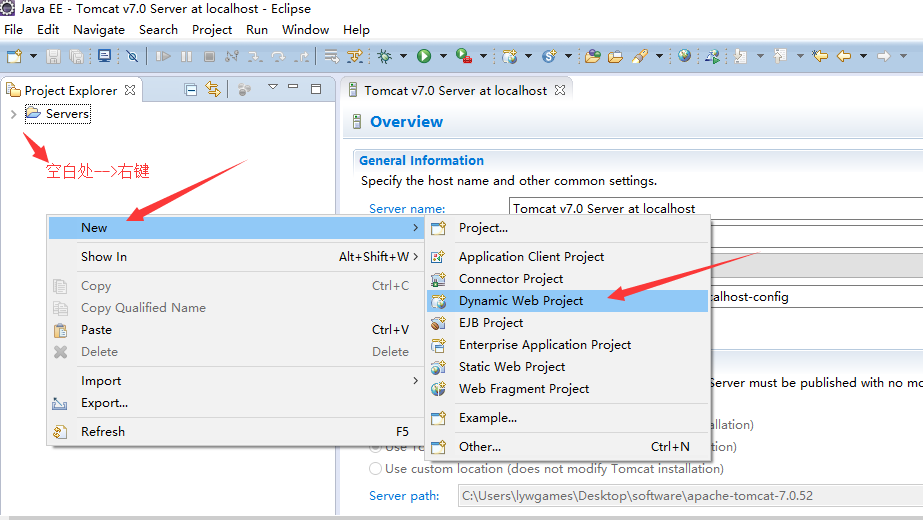
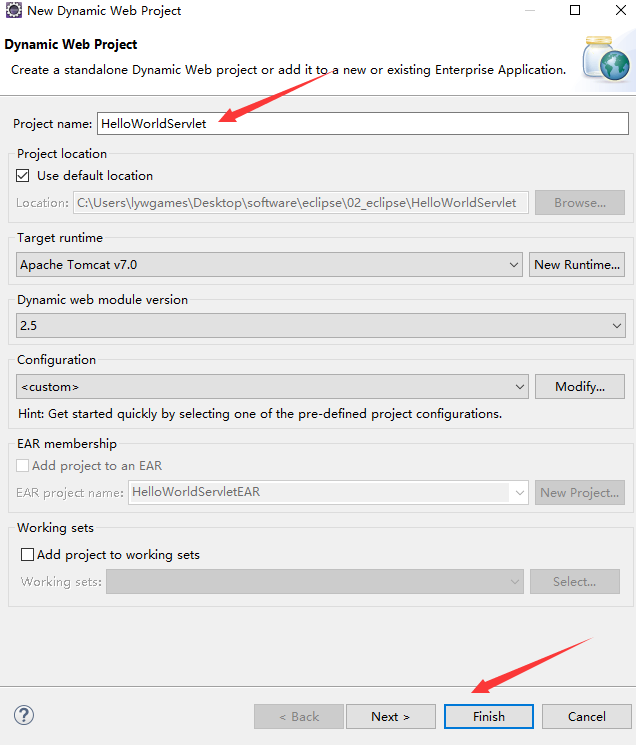
一.创建一个HelloWorldServlet的WEB工程

1.新建一个动态的Web工程



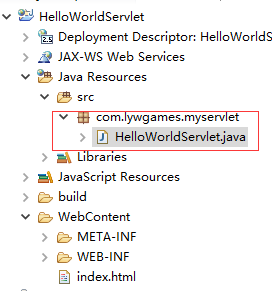
2.工程名字HelloWorldServlet



2.在WebContent下新建一个index.html

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta charset="UTF-8">  <title>Hello World</title>  </head>  <body>  <h1>Hello World</h1>  </body>  </html> |

3.在src目录下新建一个HelloWorldServlet.java



1. 让HelloWorldServlet.java实现Servlet接口, 并实现Servlet接口的方法, 在Service方法里打印Hello World

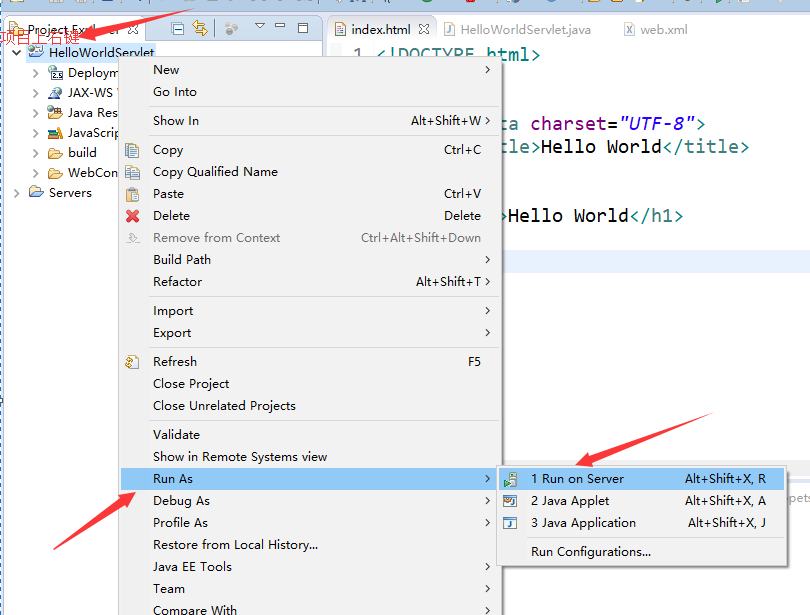
|  |
| --- |
| package com.lywgames.myservlet;  import java.io.IOException;  import javax.servlet.Servlet;  import javax.servlet.ServletConfig;  import javax.servlet.ServletException;  import javax.servlet.ServletRequest;  import javax.servlet.ServletResponse;  public class HelloWorldServlet implements Servlet {  @Override  public void init(ServletConfig config) throws ServletException {}  @Override  public ServletConfig getServletConfig() {  return null;  }  @Override  public void service(ServletRequest req, ServletResponse res) throws ServletException, IOException {  System.out.println("Hello World");  }  @Override  public String getServletInfo() {  return null;  }  @Override  public void destroy() {}  } |

6.在web.xml里注册我们的Servlet

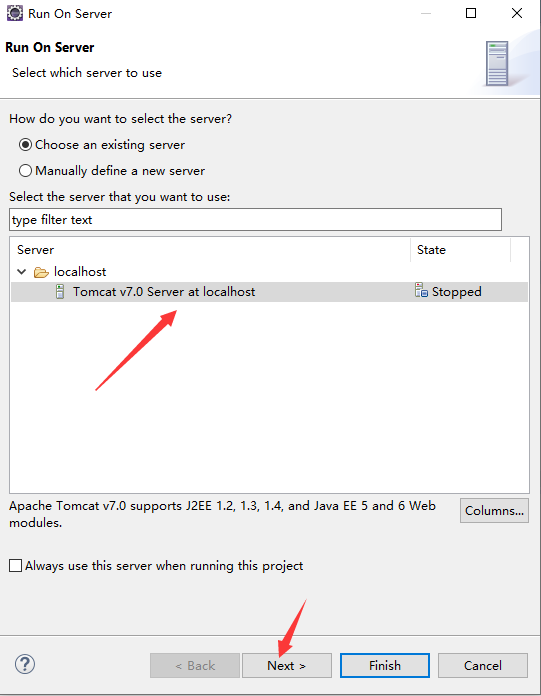
|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?>  <web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  xmlns="http://java.sun.com/xml/ns/javaee"  xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_2\_5.xsd"  id="WebApp\_ID" version="2.5">  <display-name>HelloWorldServlet</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>  <!-- 注册servlet, 名字叫做HWS, 具体的路径是com.lywgames.myservlet.HelloWorldServlet -->  <servlet>  <servlet-name>HWS</servlet-name>  <servlet-class>com.lywgames.myservlet.HelloWorldServlet</servlet-class>  </servlet>  <!-- 注册servlet的映射。 通过 servlet-name找到上面注册的具体servlet。 url-pattern: 在地址栏上的访问路径, 一定要以/打头 。-->  <servlet-mapping>  <servlet-name>HWS</servlet-name>  <url-pattern>/HWS.action</url-pattern>  </servlet-mapping>  </web-app> |

二.运行HelloWorldServlet项目

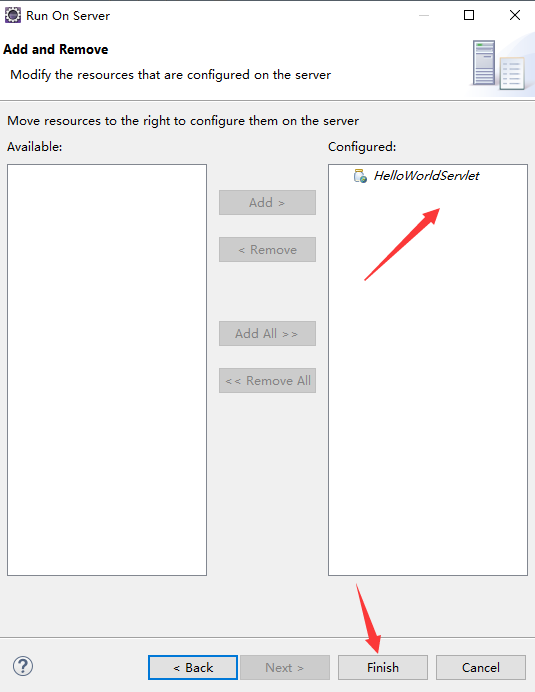
1.项目上右键——>Run As——>Run on Server



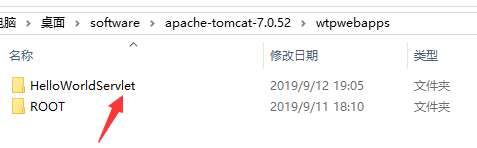
2.使用Tomcat7运行项目



3.把项目添加到server, 点击Finish按钮



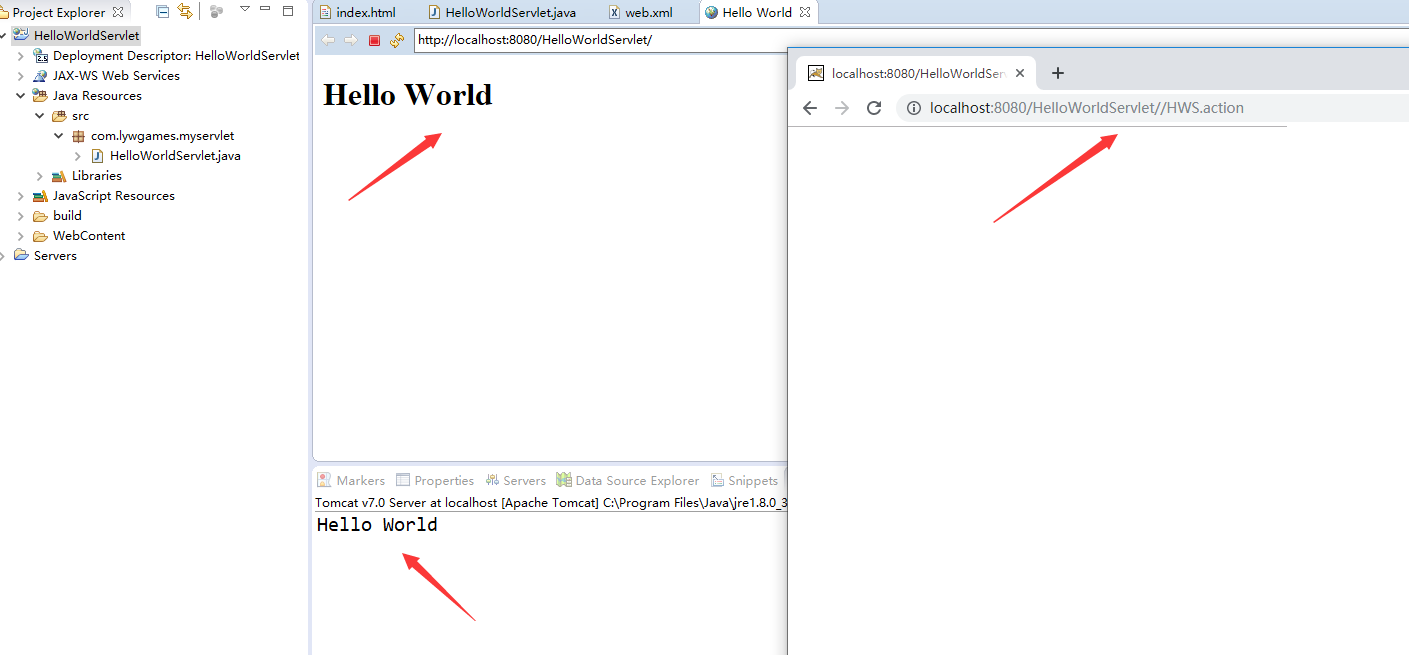
4.在Tomcat的wtpwebapps目录下, 查看有没有自己的项目



5.在WEB-INF的classes目录下, 查看HelloWorldServlet.class文件有没有生成



1. 运行成功后, Eclipse的浏览器会执行到index.html, 页面显示Hello World。在浏览器地址栏输入:http://localhost:8080/HelloWorldServlet/HWS.action, 回车查看控制台输出:Hello World。



1. HelloWorldServlet项目的执行过程

1.启动我们的Tomcat7加载我们的项目

2.Tomcat7启动并且加载我们的项目成功

3.Tomcat7加载我们项目中的web.xml文件, 读取到欢迎文件列表这个配置<welcome-file-list></welcome-file-list>, 发现欢迎文件列表中配置的有index.html, 而且我们项目的WebContent下有index.html, 因此项目启动成功, 浏览器就解释执行到了我们的index.html文件, 网页显示Hello World。

4.当我们在浏览器的地址栏中输入:

http://localhost:8080/HelloWorldServlet/HWS.action

5.Tomcat7就又在web.xml文件里寻找有没有哪个url-pattern是HWS.action, 然后通过servlet-name找到具体的类, 通过反射创建类, 执行里面的方法, 执行到了HelloWorldServlet.java的Service方法, 因此打印了Hello World。

6.HelloWorldServlet项目执行过程图解

