# Servlet&HTTP&Request

### 学习目标

1. 掌握Servlet的体系结构，了解其他Servlet配置
2. 理解HTTP协议及HTTP请求的结构（请求行，头，体）
3. 熟练掌握使用HttpServeltRequest获取常用的头信息，请求参数，请求参数乱码处理，转发的特点
4. 熟练掌握登陆案例

### 课程内容

* Servlet体系结构及配置
* HTTP协议
* HttpServletRequest对象
* 登陆案例

### 第一堂课

本节知识点

1.1 今日内容

1.2 Servlet体系结构

1.3 Servlet之urlpartten配置

视频时间

30分30秒

本节目标

知道今天的学习内容

掌握Servlet的体系结构

熟练应用Servlet之urlparttrn配置

#### 1.1 今日内容

##### 1.1.1知识概述

本节课主要梳理今天的学习内容，知道今天主要讲解的内容即可。

##### 1.1.2视频详情



##### 1.1.3总结与补充

* 1. Servlet
  2. HTTP协议
  3. Request

##### 1.1.4课堂提问与练习

无

##### 1.1.5习题答案

无

##### 1.1.6练习答案

无

##### 1.1.7视频缺陷

无

##### 1.1.8视频扩展

无

#### 1.2 Servlet体系结构

##### 1.2.1知识概述

本节课主要讲解Servlet的体系结构，Servlet的体系结构，其实Servlet体系就是书写一个Servlet的方式，Java基础班我们就知道做一个类，无外乎三种方式：第一种，自己写一个具有某些模拟功能的类；第二种，实现一个已经存在的接口；第三种，继承一个已经存在的类。那么Servlet怎么做尼？

很简单，实现Servlet接口,或者继承GenericServlet或者继承HttpServlet.

##### 1.2.2视频详情



##### 1.2.3总结与补充

1. 概念

2. 步骤

3. 执行原理

4. 生命周期

5. Servlet3.0 注解配置

6. Servlet的体系结构

Servlet -- 接口

|

GenericServlet -- 抽象类

|

HttpServlet -- 抽象类

\* GenericServlet：将Servlet接口中其他的方法做了默认空实现，只将service()方法作为抽象

\* 将来定义Servlet类时，可以继承GenericServlet，实现service()方法即可

\* HttpServlet：对http协议的一种封装，简化操作

1. 定义类继承HttpServlet

2. 复写doGet/doPost方法

##### 1.2.4课堂提问与练习

无

##### 1.2.5习题答案

无

##### 1.2.6练习答案

无

##### 1.2.7视频缺陷

无

##### 1.2.8视频扩展

无

#### 1.3 Servlet之urlpartten配置

##### 1.3.1知识概述

本节课主要讲解Servlet使用配置方式实现。重点讲解Servlet的访问路径的配置即urlpartten的配置。

##### 1.3.2视频详情



##### 1.3.3总结与补充

Servlet相关配置

1. urlpartten:Servlet访问路径

* 1. 一个Servlet可以定义多个访问路径 ： @WebServlet({"/d4","/dd4","/ddd4"})
  2. 路径定义规则：
  3. /xxx：路径匹配
  4. /xxx/xxx:多层路径，目录结构
  5. \*.do：扩展名匹配

##### 1.3.4课堂提问与练习

无

##### 1.3.5习题答案

无

##### 1.3.6 练习答案

无

##### 1.3.7 视频缺陷

无

##### 1.3.8 视频扩展

无

### 第二堂课

本节知识点：

2.1 HTTP\_概述

2.2 HTTP\_请求消息\_请求行

2.3 HTTP\_请求消息\_请求头&请求体

视频时间：

43分23秒

本节目标 ：

了解http协议内容

掌握http请求格式

#### 2.1 HTTP概述

##### 2.1.1知识概述

http协议是一个重要的协议，他是贯穿整合B/S架构的核心协议。因此学习http协议的时候一定要从请求和响应两个角度思考。注意方向性（请求：浏览器🡪服务器；响应：服务器🡪浏览器）。

##### 2.1.2视频详情



##### 2.1.3总结与补充

\* 概念：Hyper Text Transfer Protocol 超文本传输协议

\* 传输协议：定义了，客户端和服务器端通信时，发送数据的格式

\* 特点：

1. 基于TCP/IP的高级协议

2. 默认端口号:80

3. 基于请求/响应模型的:一次请求对应一次响应

4. 无状态的：每次请求之间相互独立，不能交互数据

\* 历史版本：

\* 1.0：每一次请求响应都会建立新的连接

\* 1.1：复用连接

##### 2.1.4课堂提问与练习

无

##### 2.1.5习题答案

无

##### 2.1.6 练习答案

无

##### 2.1.7 视频缺陷

无

##### 2.1.8 视频扩展

无

#### 2.2 HTTP请求消息-请求行

##### 2.2.1知识概述

本节课主要讲解HTTP协议的请求行相关的知识点，掌握请求行的格式和内容。

##### 2.2.2视频详情



##### 2.2.3总结与补充

请求消息数据格式

1. 请求行

请求方式 请求url 请求协议/版本

GET /login.html HTTP/1.1

\* 请求方式：

\* HTTP协议有7中请求方式，常用的有2种

\* GET：

1. 请求参数在请求行中，在url后。

2. 请求的url长度有限制的

3. 不太安全

\* POST：

1. 请求参数在请求体中

2. 请求的url长度没有限制的

3. 相对安全

##### 2.2.4课堂提问与练习

无

##### 2.2.5习题答案

无

##### 2.2.6 练习答案

无

##### 2.2.7 视频缺陷

无

##### 2.2.8 视频扩展

无

#### 2.3 HTTP\_请求消息\_请求头&请求体

##### 2.3.1知识概述

本节课主要讲解HTTP协议的请求头和请求体相关的知识点，掌握请求头和请求体的格式和内容。

##### 2.3.2视频详情



##### 2.3.3总结与补充

1. 请求头：客户端浏览器告诉服务器一些信息

请求头名称: 请求头值

\* 常见的请求头：

1. User-Agent：浏览器告诉服务器，我访问你使用的浏览器版本信息

\* 可以在服务器端获取该头的信息，解决浏览器的兼容性问题

2. Referer：http://localhost/login.html

\* 告诉服务器，我(当前请求)从哪里来？

\* 作用：

1. 防盗链：

2. 统计工作：

3. 请求空行

空行，就是用于分割POST请求的请求头，和请求体的。

4. 请求体(正文)：

\* 封装POST请求消息的请求参数的

\* 字符串格式：

POST /login.html HTTP/1.1

Host: localhost

User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64; rv:60.0) Gecko/20100101

Firefox/60.0

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;q=0.8

Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2

Accept-Encoding: gzip, deflate

Referer: http://localhost/login.html

Connection: keep-alive

Upgrade-Insecure-Requests: 1

username=zhangsan

##### 2.3.4课堂提问与练习

无

##### 2.3.5习题答案

无

##### 2.3.6 练习答案

无

##### 2.3.7 视频缺陷

无

##### 2.3.8 视频扩展

无

### 第三堂课

本节知识点：

3.1 Request原理

3.2 Request继承体系

3.3 Request获取请求行数据\_方法介绍

3.4 Request获取请求行数据\_代码演示

3.5 Request\_获取请求头数据

3.6 Request\_获取请求体数据

视频时间：

60分24秒

本节目标 ：

了解Request原理

掌握Request的常用API

#### 3.1 Request原理

##### 3.1.1知识概述

request对象和response对象的原理

1. request和response对象是由服务器创建的。我们来使用它们

2. request对象是来获取请求消息，response对象是来设置响应消息

##### 3.1.2视频详情



##### 3.1.3总结与补充

request对象和response对象的原理

1. request和response对象是由服务器创建的。我们来使用它们

2. request对象是来获取请求消息，response对象是来设置响应消息

##### 3.1.4课堂提问与练习

无

##### 3.1.5习题答案

无

##### 3.1.6练习答案

无

##### 3.1.7视频缺陷

无

##### 3.1.8视频扩展

无

#### 3.2 Request继承体系

##### 3.2.1知识概述

request对象继承体系结构：

ServletRequest -- 接口

| 继承

HttpServletRequest -- 接口

| 实现

org.apache.catalina.connector.RequestFacade 类(tomcat)

##### 3.2.2视频详情



##### 3.2.3总结与补充

request对象继承体系结构：

ServletRequest -- 接口

| 继承

HttpServletRequest -- 接口

| 实现

org.apache.catalina.connector.RequestFacade 类(tomcat)

##### 3.2.4课堂提问与练习

无

##### 3.2.5习题答案

无

##### 练习答案

无

##### 3.2.7视频缺陷

无

##### 3.2.8视频扩展

无

#### 3.3 Request获取请求行数据\_方法介绍

##### 3.3.1知识概述

1. 获取请求行数据

\* GET /day14/demo1?name=zhangsan HTTP/1.1

\* 方法：

1. 获取请求方式 ：GET

\* String getMethod()

2. (\*)获取虚拟目录：/day14

\* String getContextPath()

3. 获取Servlet路径: /demo1

\* String getServletPath()

4. 获取get方式请求参数：name=zhangsan

\* String getQueryString()

5. (\*)获取请求URI：/day14/demo1

\* String getRequestURI(): /day14/demo1

\* StringBuffer getRequestURL() :http://localhost/day14/demo1

\* URL:统一资源定位符 ： http://localhost/day14/demo1 中华人民共和国

\* URI：统一资源标识符 : /day14/demo1 共和国

6. 获取协议及版本：HTTP/1.1

\* String getProtocol()

7. 获取客户机的IP地址：

\* String getRemoteAddr()

##### 3.3.2视频详情



##### 3.3.3总结与补充

\* GET /day14/demo1?name=zhangsan HTTP/1.1

\* 方法：

1. 获取请求方式 ：GET

\* String getMethod()

2. (\*)获取虚拟目录：/day14

\* String getContextPath()

3. 获取Servlet路径: /demo1

\* String getServletPath()

4. 获取get方式请求参数：name=zhangsan

\* String getQueryString()

5. (\*)获取请求URI：/day14/demo1

\* String getRequestURI(): /day14/demo1

\* StringBuffer getRequestURL() :http://localhost/day14/demo1

\* URL:统一资源定位符 ： http://localhost/day14/demo1 中华人民共和国

\* URI：统一资源标识符 : /day14/demo1 共和国

6. 获取协议及版本：HTTP/1.1

\* String getProtocol()

7. 获取客户机的IP地址：

\* String getRemoteAddr()

##### 3.3.4课堂提问与练习

无

##### 3.3.5习题答案

无

##### 3.3.6 练习答案

无

##### 3.3.7 视频缺陷

无

##### 3.3.8 视频扩展

无

#### 3.4 Request\_获取请求行数据\_代码演示

##### 3.4.1知识概述

本节主要演示Request对象获取请求行数据代码演示。

##### 3.4.2视频详情



##### 3.4.3总结与补充

无

##### 3.4.4课堂提问与练习

无

##### 3.4.5习题答案

无

##### 3.4.6 练习答案

无

##### 3.4.7 视频缺陷

无

##### 3.4.8 视频扩展

无

#### 3.5 Request\_获取请求头数据

##### 3.5.1知识概述

获取请求头数据

\* 方法：

\* (\*)String getHeader(String name):通过请求头的名称获取请求头的值

\* Enumeration<String> getHeaderNames():获取所有的请求头名称

##### 3.5.2视频详情



##### 3.5.3总结与补充

获取请求头数据

\* 方法：

\* (\*)String getHeader(String name):通过请求头的名称获取请求头的值

\* Enumeration<String> getHeaderNames():获取所有的请求头名称

##### 3.5.4课堂提问与练习

无

##### 3.5.5习题答案

无

##### 3.5.6 练习答案

无

##### 3.5.7 视频缺陷

无

##### 3.5.8 视频扩展

无

#### 3.6 Request\_获取请求体数据

##### 3.6.1知识概述

获取请求体数据:

\* 请求体：只有POST请求方式，才有请求体，在请求体中封装了POST请求的请求参数

\* 步骤：

1. 获取流对象

\* BufferedReader getReader()：获取字符输入流，只能操作字符数据

\* ServletInputStream getInputStream()：获取字节输入流，可以操作所有类型数据

案例：用户登录

\* 在文件上传知识点后讲解

2. 再从流对象中拿数据

##### 3.6.2视频详情



##### 3.6.3总结与补充

获取请求体数据:

\* 请求体：只有POST请求方式，才有请求体，在请求体中封装了POST请求的请求参数

\* 步骤：

1. 获取流对象

\* BufferedReader getReader()：获取字符输入流，只能操作字符数据

\* ServletInputStream getInputStream()：获取字节输入流，可以操作所有类型数据

案例：用户登录

\* 在文件上传知识点后讲解

2. 再从流对象中拿数据

##### 3.6.4课堂提问与练习

无

##### 3.6.5习题答案

无

##### 3.6.6 练习答案

无

##### 3.6.7 视频缺陷

无

##### 3.6.8 视频扩展

无

### 第四堂课

本节知识点：

4.1 Request\_获取请求参数通用方式介绍

4.2 Request\_获取请求参数通用方式演示

4.3 Request\_获取请求参数中文乱码问题处理

4.4 Request\_请求转发

4.5 Request\_共享数据(域对象)

4.6 Request\_获取ServletContext

视频时间

48分00秒

本节目标 ：

1. 掌握使用Request获取请求参数
2. 掌握处理请求中文乱码问题
3. 了解请求转发
4. 了解Request域对象

#### 4.1 Request\_获取请求参数通用方式介绍

##### 4.1.1知识概述

1. 获取请求参数通用方式：不论get还是post请求方式都可以使用下列方法来获取请求参数

1. String getParameter(String name):根据参数名称获取参数值 username=zs&password=123

2. String[] getParameterValues(String name):根据参数名称获取参数值的数组

hobby=xx&hobby=game

3. Enumeration<String> getParameterNames():获取所有请求的参数名称

4. Map<String,String[]> getParameterMap():获取所有参数的map集合

##### 4.1.2视频详情



##### 4.1.3总结与补充

1. 获取请求参数通用方式：不论get还是post请求方式都可以使用下列方法来获取请求参数

1. String getParameter(String name):根据参数名称获取参数值 username=zs&password=123

2. String[] getParameterValues(String name):根据参数名称获取参数值的数组

hobby=xx&hobby=game

3. Enumeration<String> getParameterNames():获取所有请求的参数名称

4. Map<String,String[]> getParameterMap():获取所有参数的map集合

##### 4.1.4课堂提问与练习

无

##### 4.1.5习题答案

无

##### 练习答案

无

##### 视频缺陷

无

##### 视频扩展

无

#### 4.2 Request\_获取请求参数通用方式演示

##### 4.2.1知识概述

演示Request获取请求参数代码。

##### 4.2.2视频详情



##### 4.2.3总结与补充

无

##### 4.2.4课堂提问与练习

无

##### 4.2.5习题答案

无

##### 练习答案

无

##### 4.2.7视频缺陷

无

##### 4.2.8视频扩展

无

#### 4.3 Request\_获取请求参数中文乱码问题处理

##### 4.3.1知识概述

中文乱码问题：

\* get方式：tomcat 8 已经将get方式乱码问题解决了

\* post方式：会乱码

\* 解决：在获取参数前，设置request的编码request.setCharacterEncoding("utf-8");

##### 4.3.2视频详情



##### 4.3.3总结与补充

中文乱码问题：

\* get方式：tomcat 8 已经将get方式乱码问题解决了

\* post方式：会乱码

\* 解决：在获取参数前，设置request的编码request.setCharacterEncoding("utf-8");

##### 4.3.4课堂提问与练习

无

##### 4.3.5习题答案

无

##### 4.3.6练习答案

无

##### 4.3.7视频缺陷

无

##### 4.3.8视频扩展

无

#### 4.4 Request\_请求转发

##### 4.4.1知识概述

请求转发：一种在服务器内部的资源跳转方式

1. 步骤：

1. 通过request对象获取请求转发器对象：RequestDispatcher

getRequestDispatcher(String path)

2. 使用RequestDispatcher对象来进行转发：forward(ServletRequest request,

ServletResponse response)

2. 特点：

1. 浏览器地址栏路径不发生变化

2. 只能转发到当前服务器内部资源中。

3. 转发是一次请求

##### 4.4.2视频详情



##### 4.4.3总结与补充

请求转发：一种在服务器内部的资源跳转方式

1. 步骤：

1. 通过request对象获取请求转发器对象：RequestDispatcher

getRequestDispatcher(String path)

2. 使用RequestDispatcher对象来进行转发：forward(ServletRequest request,

ServletResponse response)

2. 特点：

1. 浏览器地址栏路径不发生变化

2. 只能转发到当前服务器内部资源中。

3. 转发是一次请求

##### 4.4.4课堂提问与练习

无

##### 4.4.5习题答案

无

##### 4.4.6练习答案

无

##### 4.4.7视频缺陷

无

##### 4.4.8 视频扩展

#### 4.5 Request\_共享数据(域对象）

##### 4.5.1知识概述

共享数据：

\* 域对象：一个有作用范围的对象，可以在范围内共享数据

\* request域：代表一次请求的范围，一般用于请求转发的多个资源中共享数据

\* 方法：

1. void setAttribute(String name,Object obj):存储数据

2. Object getAttitude(String name):通过键获取值

3. void removeAttribute(String name):通过键移除键值对

##### 4.5.2视频详情



##### 4.5.3总结与补充

共享数据：

\* 域对象：一个有作用范围的对象，可以在范围内共享数据

\* request域：代表一次请求的范围，一般用于请求转发的多个资源中共享数据

\* 方法：

1. void setAttribute(String name,Object obj):存储数据

2. Object getAttitude(String name):通过键获取值

3. void removeAttribute(String name):通过键移除键值对

##### 4.5.4课堂提问与练习

无

##### 4.5.5习题答案

无

##### 4.5.6练习答案

无

##### 4.5.7视频缺陷

无

##### 4.5.8 视频扩展

#### 4.6 Request\_获取ServletContext

##### 4.6.1知识概述

获取ServletContext：

\* ServletContext getServletContext()

##### 4.6.2视频详情



##### 4.6.3总结与补充

获取ServletContext：

\* ServletContext getServletContext()

##### 4.6.4课堂提问与练习

无

##### 4.6.5习题答案

无

##### 4.6.6练习答案

无

##### 4.6.7视频缺陷

无

##### 4.6.8 视频扩展

### 第五堂课

本节知识点：

5.1 登录案例\_需求

5.2 登录案例\_分析

5.3 登录案例\_代码实现1

视频时间

30分03秒

本节目标 ：

熟练应用所学技术完成登陆案例

#### 5.1 登录案例\_需求

##### 5.1.1知识概述

用户登录案例需求：

1.编写login.html登录页面

username & password 两个输入框

2.使用Druid数据库连接池技术,操作mysql，day14数据库中user表

3.使用JdbcTemplate技术封装JDBC

4.登录成功跳转到SuccessServlet展示：登录成功！用户名,欢迎您

5.登录失败跳转到FailServlet展示：登录失败，用户名或密码错误

##### 5.1.2视频详情



##### 5.1.3总结与补充

无

##### 5.1.4课堂提问与练习

无

##### 5.1.5习题答案

无

##### 5.1.6练习答案

无

##### 5.1.7视频缺陷

无

##### 5.1.8视频扩展

无

#### 5.2 登录案例\_分析

##### 5.2.1知识概述

分析开发步骤：

* + 1. 创建数据库
    2. 创建工程
    3. 代码开发

##### 5.2.2视频详情



##### 5.2.3总结与补充

无

##### 5.2.4课堂提问与练习

无

##### 5.2.5习题答案

无

##### 5.2.6练习答案

无

##### 5.2.7视频缺陷

无

##### 5.2.8视频扩展

无

#### 5.3 登录案例\_代码实现1

##### 5.3.1知识概述

本节课主要是实现具体的

##### 5.3.2视频详情



##### 5.3.3总结与补充

##### 5.3.4课堂提问与练习

无

##### 5.3.5习题答案

无

##### 5.3.6练习答案

无

##### 5.3.7视频缺陷

无

##### 5.3.8视频扩展

无

### 第六堂课

本节知识点：

6.1 登录案例\_代码实现2

6.2 登录案例\_BeanUtils基本使用

6.3 登录案例\_BeanUtils介绍

视频时间

27分40秒

本节目标 ：

能够完成登陆案例代码

能够熟练使用BeanUtils

#### 6.1 登录案例\_代码实现2

##### 6.1.1知识概述

完善登陆代码

##### 6.1.2视频详情



##### 6.1.3总结与补充

无

##### 6.1.4课堂提问与练习

无

##### 6.1.5习题答案

无

##### 6.1.6练习答案

无

##### 6.1.7视频缺陷

无

##### 6.1.8视频扩展

无

#### 6.2 登录案例\_BeanUtils基本使用

##### 6.2.1知识概述

BeanUtils工具类，简化数据封装

\* 用于封装JavaBean的

1. JavaBean：标准的Java类

1. 要求：

1. 类必须被public修饰

2. 必须提供空参的构造器

3. 成员变量必须使用private修饰

4. 提供公共setter和getter方法

2. 功能：封装数据

##### 6.2.2视频详情



##### 6.2.3总结与补充

BeanUtils工具类，简化数据封装

\* 用于封装JavaBean的

1. JavaBean：标准的Java类

1. 要求：

1. 类必须被public修饰

2. 必须提供空参的构造器

3. 成员变量必须使用private修饰

4. 提供公共setter和getter方法

2. 功能：封装数据

##### 6.2.4课堂提问与练习

无

##### 6.2..5习题答案

无

##### 6.2.6练习答案

无

##### 6.2.7视频缺陷

无

##### 6.2.8视频扩展

无

#### 6.3登录案例\_BeanUtils介绍

##### 6.3.1知识概述

概念：

成员变量：

属性：setter和getter方法截取后的产物

例如：getUsername() --> Username--> username

3. 方法：

1. setProperty()

2. getProperty()

3. populate(Object obj , Map map):将map集合的键值对信息，封装到对应的JavaBean对象中

##### 6.3.2视频详情



##### 6.3.3总结与补充

概念：

成员变量：

属性：setter和getter方法截取后的产物

例如：getUsername() --> Username--> username

3. 方法：

1. setProperty()

2. getProperty()

3. populate(Object obj , Map map):将map集合的键值对信息，封装到对应的JavaBean对象中

##### 6.3.4课堂提问与练习

无

##### 6.3..5习题答案

无

##### 6.3.6练习答案

无

##### 6.3.7视频缺陷

无

##### 6.3.8 视频扩展

无