

今日内容

- 1 1. JSP:
- 2 1. 指令
- 3 2. 注释
- 4 3. 内置对象
- 5
- 6 2. MVC开发模式
- 7 3. EL表达式
- 8 4. JSTL标签
- 9 5. 三层架构

JSP:

- 1 1. 指令
- 2 * 作用：用于配置JSP页面，导入资源文件
- 3 * 格式：
- 4 <%@ 指令名称 属性名1=属性值1 属性名2=属性值2 ... %>
- 5 * 分类：
- 6 1. **page** : 配置JSP页面的
- 7 * **contentType**: 等同于**response.setContentType()**
- 8 1. 设置响应体的**mime**类型以及字符集
- 9 2. 设置当前**jsp**页面的编码（只能是高级的**IDE**才能生效，如果使用低级工
- 具，则需要设置**pageEncoding**属性设置当前页面的字符集）
- 10 * **import**: 导包
- 11 * **errorPage**: 当前页面发生异常后，会自动跳转到指定的错误页面
- 12 * **isErrorPage**: 标识当前也是是否是错误页面。
- 13 * **true**: 是，可以使用内置对象**exception**
- 14 * **false**: 否。默认值。不可以使用内置对象**exception**

- 1 2. **include** : 页面包含的。导入页面的资源文件
- 2 * **<%@include file="top.jsp"%>**
- 3 3. **taglib** : 导入资源
- 4 * **<%@ taglib prefix="c"**
- uri="http://java.sun.com/jsp/jstl/core" %>
- 5 * **prefix**: 前缀，自定义的
- 6 2. 注释：
- 7 1. **html**注释：
- 8 **<!-- -->**: 只能注释**html**代码片段
- 9 2. **jsp**注释: 推荐使用
- 10 **<%-- --%>**: 可以注释所有

1	3. 内置对象		
2	* 在jsp页面中不需要创建，直接使用的对象		
3	* 一共有9个：		
4	变量名	真实类型	作用
5	* pageContext	PageContext	当前页面共享
6	数据，还可以获取其他八个内置对象		
7	* request	HttpServletRequest	一次请求访问
8	的多个资源(转发)		
9	* session	HttpSession	一次会话的多
10	个请求间		
11	* application	ServletContext	所有用户间共
12	享数据		
13	* response	HttpServletResponse	响应对象
14	* page	Object	当前页面
15	(Servlet)的对象 this		
16	* out	JspWriter	输出对象，数
17	据输出到页面上		
18	* config	ServletConfig	Servlet的配
19	置对象		
20	* exception	Throwable	异常对象

MVC：开发模式

1. jsp演变历史
 1. 早期只有servlet，只能使用response输出标签数据，非常麻烦
 2. 后来又jsp，简化了servlet的开发，如果过度使用jsp，在jsp中即写大量的java代码，有写html表，造成难于维护，难于分工协作
 3. 再后来，java的web开发，借鉴mvc开发模式，使得程序的设计更加合理性
2. MVC:
 1. M: Model，模型。JavaBean
 - * 完成具体的业务操作，如：查询数据库，封装对象
 2. V: View，视图。JSP
 - * 展示数据
 3. C: Controller，控制器。Servlet
 - * 获取用户的输入
 - * 调用模型
 - * 将数据交给视图进行展示

- * 优缺点:
 1. 优点:
 1. 耦合性低，方便维护，可以利于分工协作
 2. 重用性高
 2. 缺点:
 1. 使得项目架构变得复杂，对开发人员要求高

EL表达式

- 1 1. 概念: Expression Language 表达式语言
- 2 2. 作用: 替换和简化jsp页面中java代码的编写
- 3 3. 语法: \${表达式}
- 4 4. 注意:
 - 5 * jsp默认支持el表达式的。如果要忽略el表达式
 - 6 1. 设置jsp中page指令中: isELIgnored="true" 忽略当前jsp页面中所有的el表达式
 - 7 2. \\${表达式} : 忽略当前这个el表达式

- 1 5. 使用:
 - 2 1. 运算:
 - 3 * 运算符:
 - 4 1. 算数运算符: + - * /(div) %(mod)
 - 5 2. 比较运算符: > < >= <= == !=
 - 6 3. 逻辑运算符: &&(and) ||(or) !(not)
 - 7 4. 空运算符: empty
 - 8 * 功能: 用于判断字符串、集合、数组对象是否为null或者长度是否为0
 - 9 * \${empty list}:判断字符串、集合、数组对象是否为null或者长度为0
 - 10 * \${not empty str}:表示判断字符串、集合、数组对象是否不为null 并
且 长度>0
 - 11 2. 获取值
 - 12 1. el表达式只能从域对象中获取值
 - 13 2. 语法:
 - 14 1. \${域名称.键名}: 从指定域中获取指定键的值
 - 15 * 域名称:
 - 16 1. pageScope --> pageContext
 - 17 2. requestScope --> request
 - 18 3. sessionScope --> session
 - 19 4. applicationScope --> application (ServletContext)
 - 20 * 举例: 在request域中存储了name=张三
 - 21 * 获取: \${requestScope.name}
 - 22
 - 23 2. \${键名}: 表示依次从最小的域中查找是否有该键对应的值, 直到找到为止。

- 1 3. 获取对象、List集合、Map集合的值
 - 2 1. 对象: \${域名称.键名.属性名}
 - 3 * 本质上会去调用对象的getter方法
 - 4
 - 5 2. List集合: \${域名称.键名[索引]}
 - 6
 - 7 3. Map集合:
 - 8 * \${域名称.键名.key名称}
 - 9 * \${域名称.键名["key名称"]}

- 1 3. 隐式对象:
 - 2 * el表达式中有11个隐式对象
 - 3 * pageContext:
 - 4 * 获取jsp其他八个内置对象
 - 5 * \${pageContext.request.contextPath}: 动态获取虚拟目录

JSTL

- 1 1. 概念: **JavaServer Pages Tag Library** JSP标准标签库
- 2 * 是由Apache组织提供的开源的免费的jsp标签 <标签>
- 3
- 4 2. 作用: 用于简化和替换jsp页面上的java代码
- 5
- 6 3. 使用步骤:
- 7 1. 导入jstl相关jar包
- 8 2. 引入标签库: **taglib**指令: `<%@ taglib %>`
- 9 3. 使用标签
- 10
- 11 4. 常用的JSTL标签
- 12 1. **if**: 相当于java代码的if语句
- 13 1. 属性:
- 14 * **test** 必须属性, 接受boolean表达式
- 15 * 如果表达式为true, 则显示if标签体内容, 如果为false, 则不显示标签体内容
- 16 * 一般情况下, **test**属性值会结合el表达式一起使用
- 17 2. 注意:
- 18 * **c:if**标签没有else情况, 想要else情况, 则可以在定义一个**c:if**标签
- 19 2. **choose**: 相当于java代码的switch语句
- 20 1. 使用**choose**标签声明 相当于switch声明
- 21 2. 使用**when**标签做判断 相当于case
- 22 3. 使用**otherwise**标签做其他情况的声明 相当于default
- 23
- 24 3. **foreach**: 相当于java代码的for语句
- 25
- 26 5. 练习:
- 27 * 需求: 在request域中有一个存有User对象的List集合。需要使用jstl+el将list集合数据展示到jsp页面的表格table中

三层架构: 软件设计架构

- 1 1. 界面层(表示层): 用户看得界面。用户可以通过界面上的组件和服务器进行交互
- 2 2. 业务逻辑层: 处理业务逻辑的。
- 3 3. 数据访问层: 操作数据存储文件。

案例: 用户信息列表展示

- 1 1. 需求: 用户信息的增删改查操作
- 2 2. 设计:
- 3 1. 技术选型: **Servlet+JSP+MySQL+JDBCTemplate+Duid+BeanUtils+tomcat**
- 4 2. 数据库设计:
- 5 `create database day17; -- 创建数据库`

```
6      use day17;          -- 使用数据库
7      create table user(  -- 创建表
8          id int primary key auto_increment,
9          name varchar(20) not null,
10         gender varchar(5),
11         age int,
12         address varchar(32),
13         qq varchar(20),
14         email varchar(50)
15     );
16
17 3. 开发:
18     1. 环境搭建
19         1. 创建数据库环境
20         2. 创建项目, 导入需要的jar包
21
22     2. 编码
```

```
1 4. 测试
2 5. 部署运维
```