

J2EE开发基础培训

银海软件 开发中心

殷飞 系统架构师

2006年7月

培训目标

- ❖ 了解XML文件结构
- ❖ 掌握简单的网页和javascript函数编写
- ❖ 掌握JSP编译运行机理
- ❖ 了解J2EE Web应用目录结构，常用技术
- ❖ 掌握request/response/session等概念
- ❖ 能够使用JSP、Servlet、struts编写应用
- ❖ 了解MVC模式
- ❖ 了解filter的运行机理
- ❖ 了解taglib的运行机理
- ❖ 了解B/S结构应用开发跟C/S开发的差异

培训计划

❖ 以应用的完善为主线

❖ 技术点引入步骤

◇ Html

◇ Javascript

◇ JSP

◇ Servlet

◇ Struts

◇ Filter/Taglib

HTML

- ❖ 网页只是简单的文本文件
- ❖ 浏览器对HTML进行解释执行
- ❖ HTML是一种不严格的标记语言
 - ◇ 标签不配对浏览器也可能会支持
- ❖ HTML标记不区分大小写
- ❖ HTML文件可以HTM或HTML为扩展名
- ❖ HTML页面框架
 - ◇ html
 - ◇ head
 - ◇ title
 - ◇ body

常用HTML标记

- ❖ meta-元数据
- ❖ p-段落
- ❖ font-字体
- ❖ table/tr/td-表格
- ❖ a-超链接
- ❖ br-换行

表单—form

❖ 常用的表单输入控件

◇ input-输入控件

◇ select-选择控件

◇ button-按钮

◇ submit-提交按钮

◇ reset-重置按钮

❖ 提交的是表单而不是页面

❖ 一个HTML页面可以有多个表单

javascript

- ❖ 浏览器是javascript的解释执行环境
- ❖ javascript区分大小写
- ❖ javascript变量为弱类型
- ❖ html页面的几个常用对象
 - ◇ document
 - ◇ window
 - ◇ form
 - ◇ input

html控件常用属性和事件

- ❖ **readOnly**: 输入控件是否只读
- ❖ **disabled**: 输入控件是否被disabled
- ❖ **value**: 获取或设置输入控件的值
- ❖ **onclick**: 输入控件或按钮被点击时触发
- ❖ **onchange**: 输入控件的值改变时触发
- ❖ **onfocus**: 输入控件或按钮获得焦点时触发

练习1

- ❖ 创建新增开发人员信息页面
- ❖ 录入控件全部采用text录入
- ❖ 控件name使用class中的成员变量名称
- ❖ 在提交数据之前添加javascript数据校验
 - ◇ 所有属性都不能为空
 - ◇ 性别只能为男和女

JSP

- ❖ 声明JSP元数据: `<%@ page %>`
 - ◇ `contentType`: 内容类型
 - ◇ `import`: 引入包
- ❖ 包含公用文件: `<%@ include file%>`
- ❖ 服务器端代码嵌入: `<%嵌入java代码%>`
- ❖ `Jsp`在服务端执行, 输出HTML, 再返回给客户端浏览器, 由浏览器解释HTML
- ❖ 获取提交的表单数据: `request.getParameter`

表单数据提交的两种方式

❖ Get

- ◇ 表单数据在url中
- ◇ 支持的数据量较小

❖ Post

- ◇ 表单数据不在url中
- ◇ 支持的数据量较大

❖ Get方式提交表单数据和在超链接url中包含参数没有任何区别

- ◇ `location?param1=value1¶m2=value2`

练习2

- ❖ 创建保存开发人员信息页面
- ❖ 使用`request.getParameter`获取表单参数
- ❖ 转换字符集
- ❖ 调用已有的Dao保存数据到数据库中

Servlet

- ❖ 基类—HttpServlet
- ❖ 覆写service方法
 - ◇ 使用response.getWriter()直接输出HTML
 - ◇ forward到JSP上，由JSP生成HTML
- ❖ web.xml配置
 - ◇ servlet: 定义servlet
 - ◆ servlet-name
 - ◆ servlet-class
 - ◆ init-param
 - + param-name
 - + param-value
 - ◇ servlet-mapping: 定义url映射
 - ◆ servlet-name
 - ◆ url-pattern

服务器端的页面跳转

❖ forward

- ◇ 把处理请求直接转发到指定的url

❖ redirect

- ◇ 把要跳转的url发给客户端浏览器，由浏览器发起访问请求

练习3

- ❖ 把检查参数和保存数据的操作移到servlet
- ❖ 在servlet里面根据保存成功与否转向到不同的jsp显示成功或失败

JSP与Servlet

- ❖ **JSP在运行时会被编译成为servlet源文件，再编译为class执行**
- ❖ **JSP适合于做界面**
- ❖ **通常在Servlet对表单参数进行处理，并调用业务逻辑进行处理，再把需展现数据存入request中，根据需要forward到相应的JSP上去**
- ❖ **JSP不直接调用业务逻辑**

request/response

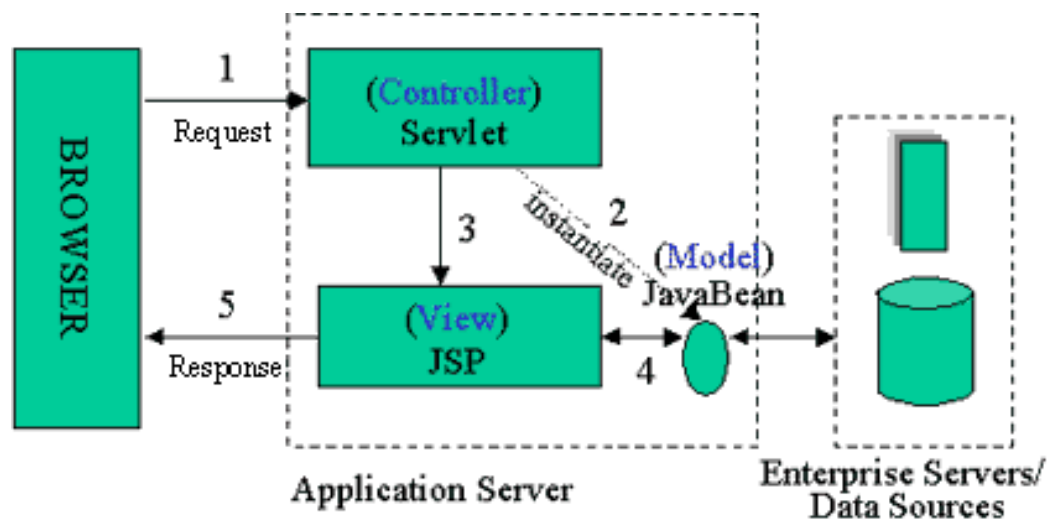
- ❖ **request**代表一次客户端请求
- ❖ 在一次请求期间控制流可以在**Servlet**和**JSP**之间进行转发
- ❖ **getParameter**函数用于获取表单参数
- ❖ **get/setAttribute**函数用于在**request**中存放和传递数据
- ❖ **response**代表服务器端对请求的响应
- ❖ **setContentType**设定输出内容类型
- ❖ **getWriter**用于向客户端输出内容

session

- ❖ 通过request.getSession函数获取到session
- ❖ 我们需要session—有些数据需要在多次交互之间保存，比如认证信息
- ❖ session机制需要判断发生多次交互的是否同一个客户端
- ❖ http协议是无状态协议，而不是常连接
- ❖ 服务器分配一个id给客户端浏览器，并通过该id来确定是否同一个session
- ❖ sessionid可能在url中或cookie中
- ❖ session通常有个超时时间
- ❖ 服务器端不能主动向客户端发起请求

MVC模式

- ❖ M—Model, 领域模型层
- ❖ C—Controller, 控制器
- ❖ V—Viewer, 界面展示层



XML

- ❖ 严格的标记语言
 - ◇ 开始标记和结束标记必须配套
 - ◇ 区分大小写
- ❖ 内容非常的结构化
- ❖ 常用于数据传输和配置文件

struts

❖ 成熟的第三方开源项目

❖ 充当Controller的角色

◇ Controller的职责

- ◆ 针对客户端提交的表单数据调用相应的业务处理
- ◆ 根据业务处理的结果跳转到相应的展示页面

❖ struts实现的功能

- ◇ 表单数据的收集
- ◇ 跳转页面的管理

使用struts

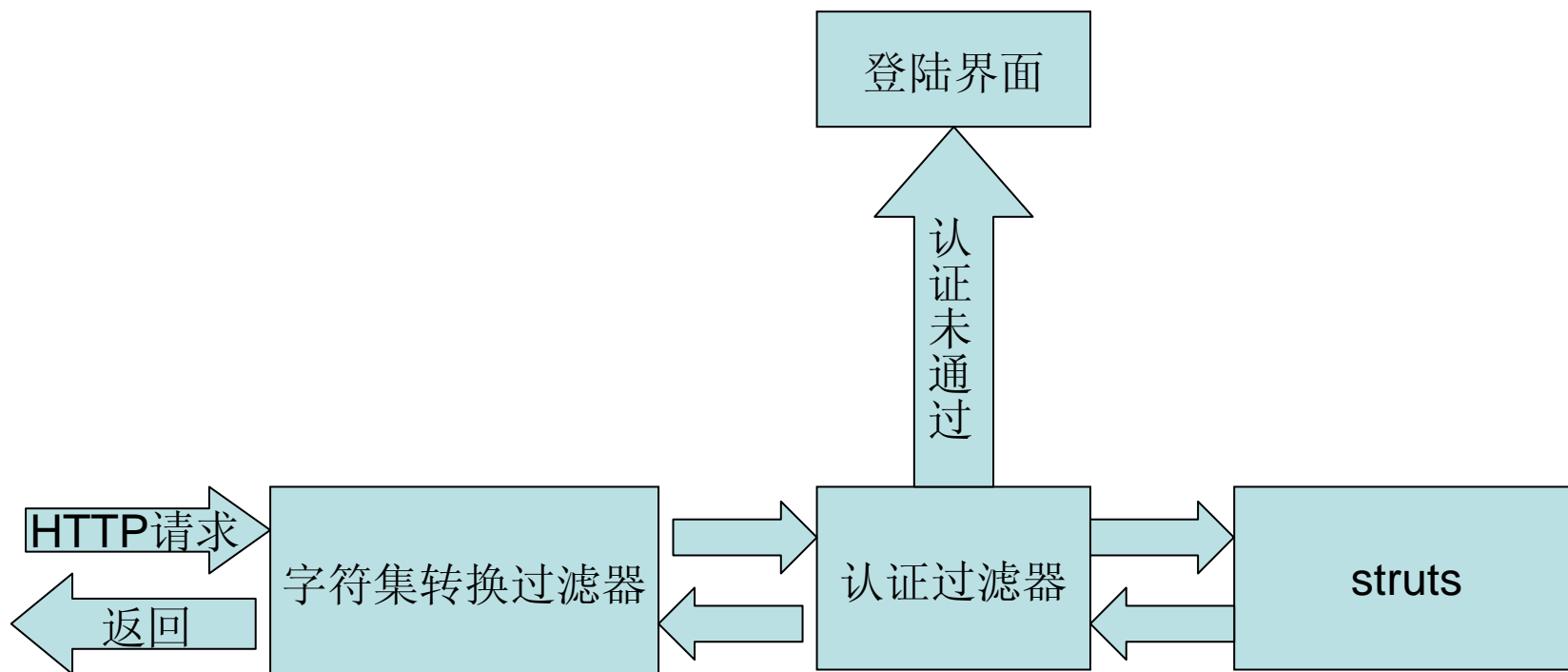
- ❖ 在web.xml中添加servlet配置
- ❖ 写Action类
- ❖ 写ActionForm类
- ❖ 在struts-config.xml中添加form-bean配置
- ❖ 在struts-config.xml中添加action/forward配置

练习4

- ❖ 把练习3中servlet的控制器功能使用struts进行替代

filter

❖ 过滤器例



filter

- ❖ **filter先于servlet执行**，可以对提交的表单数据先行进行处理，甚至可以选择不交给下一个servlet或filter处理
- ❖ **filter后于servlet返回**，可以对返回数据进行再加工
- ❖ **filter通常完成相对较简单**，切面较为明显的工作

taglib

- ❖ Taglib是封装界面元素的class
- ❖ Taglib是J2EE标准的一部分
- ❖ Taglib是一个接口
- ❖ Taglib的开发
 - ◇ Taglib class
 - ◇ Tld文件
- ❖ Taglib的使用
 - ◇ Web.xml和jsp里面声明Tld文件
 - ◇ 在jsp中使用