* JSP基础学习：

1. jsp运行在服务器端
2. jsp：Java server page
3. jsp的基础是servlet
4. jsp是综合技术Jsp=html+java片段+jsp标签（servlet）+javascript（css）
5. jsp无需配置，直接使用，修改jsp文件，不需要重启服务器
6. jsp访问：<http://localhost:8080/web>应用名称/\*\*.jsp

Mvc模式=Jsp+java类（service、javabean）+servlet

1. jsp是一种用于开发动态web资源的技术

* 快速入门：

**练习1：**在jsp页面输出当前时间（用于讲解JSP的 运行原理jsp与servlet的关系）

翻译 编译

ShowTime.class

ShowTime\_servlet.java

showTime.Jsp

如果是第一次访问该jsp文件：web服务器会把showTime.jsp文件翻译成servlet(showTime.java)文件，再将其编译成一个showTime.class，然后把class文件加载到内存

如果是第二次及以后访问，直接访问装载内存中的实例

装载到内存实例

Web服务器

第二次访问

第一次访问

第二次访问浏览器：http://localhost:8080/jsp1/showTime.jsp

第一次访问浏览器：http://localhost:8080/jsp1/showTime.jsp

单例：所有请求都用一个对象来处理，比如我们常用的service和dao层的对象通常都是单例的

多例：每个请求用一个新的对象来处理，比如：action

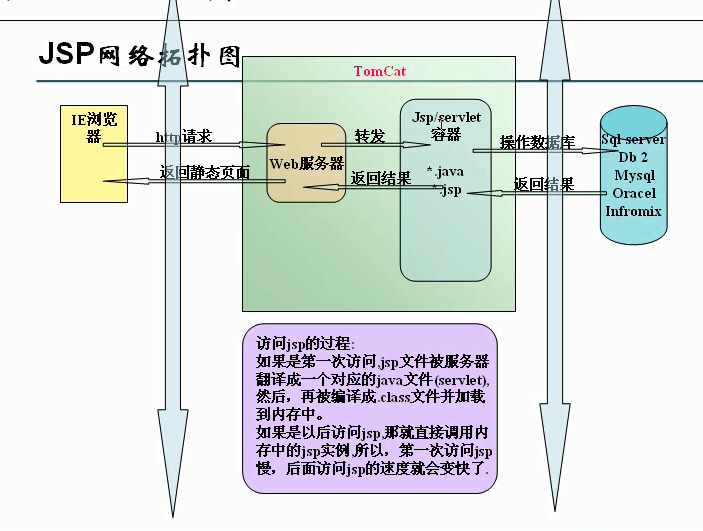
单例模式和多例模式说明：

1. 单例模式和多例模式属于对象模式
2. 单例模式的对象在整个系统中只有一份，多例模式可以有多个实例
3. 他们都不对外提供构造方法，即构造方法都为私有

综上所述：jsp为单例模式

如果jsp文件修改了，则跟第一次访问一样

* Jsp网络拓扑图



* 考虑问题：

1. Web服务器是如何调用并执行一个jsp页面（见上面描述）
2. Jsp页面中的html排版标签是如何被发送到客户端的？（转成servlet文件，out.write()）
3. Jsp页面中的java代码服务器是如何执行的？（转成servlet文件，去掉<%%>原样显示）
4. web服务器在调用jsp时，会给jsp提供一些什么java对象？（提供了九个内置对象：out:向客户端输出数据，字节流、request:接受客户的http请求、response:封装jsp的产生回应、session：用于保存用户的信息，跟踪用户的行为、application：多个用户共享对象、pageContext、exception、page、config）

修改jsp模板：

Window->Preferences->Web->JSP Files->Editor->Templates->New JSP File(html)->Edit

* JSP基础—语法（标签）
  1. 指令元素—用于从jsp发送一个信息到容器，比如设置全局变量，文字编码，引入包等

1. Page指令

<%@ page contentTye=”text/html;charset=utf-8”%>

Page指令中常用属性有：

Language=””jsp中嵌入的代码是XX，通常是java

Import=”包名”在该页面引入包或者某个具体的类

可引入多个包，中间用逗号隔开

Session=[true|false] 是否在jsp页面获取session对象

Buffer=[none|8k|指定大小]给out对象使用的缓存区是多大，默认8K

AutoFlash=[true|false]当buffer满后，是否自动刷新到浏览器

isThreadSafe=[true|false]默认true表示该jsp的线程安全由程序员控制，false则对应的Servlet将实现线程安全接口

errorPage=”相对JSP页面”，当jsp页面出现错误时，自动跳转到指定jsp页面，要求使用相对路径，当“/”开始时，表示当前web应用的根目录。如果不以“/”开头，则表示相对本jsp页面

isErrorPage=[false|true]指定该jsp页面是否专门用于作为error页面，默认false，如果设为true，则可以在该页面直接使用exception内置对象(用的很少)

contentType=”text/html;charset=utf-8”指定网页以什么方式显示页面

pageEncoding=”utf-8”指定servlet引擎以什么方法翻译jsp-🡪servlet并指定网页以什么方式显示页面

1. Include指令

<%@ include file=”filename”%>

该指令用于引入一个文件（通常是jsp文件），jsp引擎会把两个jsp文件翻译成一个servlet文件，因此也称为静态引入

被引入的jsp文件，只需保留page指令即可，<html><body>均可省略

1. Taglib指令---这个指令允许在jsp页面使用自定义的标签

<myTag:yourTag num1=”123”>

* 1. 脚本元素---可以理解就是java的片段scriplet

1. Scriplet <%java代码%>
2. 表达式 <%=java表达式%> <%=rs.getString(1)%>
3. Declaration声明

<%! 成员变量声明 %> 如：<%! Int count=10%>

<%! 函数声明%>

<%! Int jisuanqi(int num){

Int result=0;

For(int i=0;i<num;i++){

Result=result+I;

}

}%>

备注：

使用<% int i=0%>局部变量

使用<%! Int i=0%>成员变量（该jsp对应的servlet的成员变量）

成员变量和局部变量的区别：

根据定义变量位置的不同，可以将变量分为成员变量和局部变量

成员变量是在类范围内定义的变量

局部变量是在一个方法内定义的变量

成员变量可以分为：

实例属性（不用static修饰）：随着实例属性的存在而存在

类属性（static修饰）：随着类的存在而存在

成员变量无需显式初始化，系统会自动对其进行默认初始化

局部变量可以分为：

形参：在整个方法内有效

方法局部变量：从定义这个变量开始到方法结束这一段时间

代码块局部变量：从定义这个变量开始到代码块结束这一段时间

面试题:

<%! Int i=90;%>

<%

Int j=90;

Out.println(“i=”+(++i)+”j=”+(++j));

%>

第一次访问输出：91 91

第二次访问输出：92 91

第三次访问输出：93 91

如果此时另外一台电脑访问此时输出：94 91

备注：函数不能再<%%>中定义

* 1. 动作元素---使用xml语法写的，是jsp规定的一系列标准动作，在容器处理jsp时，就执行相应的操作

<jsp:useBean>//创造一个JavaBean实例

<java:setProperty>//给一个javabean实例设置初始值

<java:param>//给一个jsp设置参数，常常与<jsp:include>结合使用

<jsp:include file=””>

<jsp:param name=”” value=””/>

</jsp:include>

<java:getProperty>//引入一个javabean实例的成员变量

<jsp:include>//引入另一个文件(动态引入：两个文件分开编译)

<jsp:include file=””>

</jsp:include>---动态引入

<%@ include file=””%>---静态引入

相同点：将一个文件引入另一个文件

不同点：静态引入将两个文件编译成一个文件，所被引入的文件不要包含body、html；动态引入时间两个文件分开编译

<jsp:plugin>//使用插件

<jsp:forward>//转发(用的较多)

<jsp:forward page=”0729.jsp”></jsp:forward>

作用：，在开发jsp的过程中，我们通常将jsp放在web-inf目录下，目的是为了防止用户直接访问jsp文件，因此在web-root下有一个入口，目的是转发

使用jsp：forward跳转到对应jsp文件

<jsp:forward page=”/WEB-INF/moveto.jsp”></jsp:forward>

<jsp:fallback>

* 1. Jsp文件的注释

<!-- -->html专用注释可用在jsp中

<%-- 注释内容 --%>jsp专用注释

* JSP九大内置对象：