JavaWeb与EE

1. **JavaWeb**

# (一) Servlet

Servlet API 包含以下4个Java包：

1.javax.servlet 其中包含定义**servlet和servlet容器之间契约的类和接口**。

2.javax.servlet.http 其中包含**定义HTTP Servlet 和Servlet容器之间的关系**。

3.javax.servlet.annotation 其中包含**标注servlet，Filter,Listener的标注。它还为被标注元件定义元数据。**

4.javax.servlet.descriptor，**其中包含提供程序化登录Web应用程序的配置信息的类型。**

# (二) servlet的生命周期

# servlet的生命周期就是从servlet出现到销毁的全过程。主要分为以下几个阶段：加载类—>实例化(为对象分配空间)—>初始化(为对象的属性赋值)—>请求处理(服务阶段)—>销毁

(三) **jsp和servlet的区别和联系**

1. **jsp经编译后就变成了Servlet.**  
   (JSP的本质就是Servlet，JVM只能识别java的类，不能识别JSP的代码,Web容器将JSP的代码编译成JVM能够识别的java类)  
   2.**jsp更擅长表现于页面显示,servlet更擅长于逻辑控制.**  
   3.**Servlet中没有内置对象，Jsp中的内置对象都是必须通过HttpServletRequest对象，HttpServletResponse对象以及HttpServlet对象得到.**Jsp是Servlet的一种简化，使用Jsp只需要完成程序员需要输出到客户端的内容，Jsp中的Java脚本如何镶嵌到一个类中，由Jsp容器完成。  
   而Servlet则是个完整的Java类，这个类的Service方法用于生成对客户端的响应。  
     
   联系：   
   JSP是Servlet技术的扩展，本质上就是Servlet的简易方式。**JSP编译后是“类servlet”。**Servlet和JSP最主要的不同点在于：  
   Servlet的应用逻辑是在Java文件中，并且完全从表示层中的HTML里分离开来。  
   而JSP的情况是Java和HTML可以组合成一个扩展名为.jsp的文件。  
   JSP侧重于视图，Servlet主要用于控制逻辑  
   **Servlet更多的是类似于一个Controller，用来做控制。**

**理解以下三点即可：**

**1、不同之处在哪？**

* Servlet在Java代码中通过HttpServletResponse对象动态输出HTML内容
* JSP在静态HTML内容中嵌入Java代码，Java代码被动态执行后生成HTML内容

**2、各自的特点**

* Servlet能够很好地组织业务逻辑代码，但是在Java源文件中通过字符串拼接的方式生成动态HTML内容会导致代码维护困难、可读性差
* JSP虽然规避了Servlet在生成HTML内容方面的劣势，但是在HTML中混入大量、复杂的业务逻辑同样也是不可取的

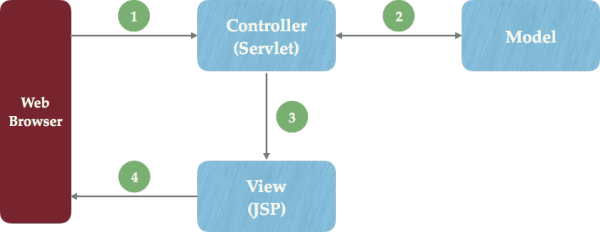
**3、通过MVC双剑合璧**

既然JSP和Servlet都有自身的适用环境，那么能否扬长避短，让它们发挥各自的优势呢？答案是肯定的——**MVC(Model-View-Controller)模式非常适合解决这一问题。**

MVC模式（**Model-View-Controller**）是[软件工程](https://link.zhihu.com/?target=https%3A//zh.wikipedia.org/wiki/%25E8%25BD%25AF%25E4%25BB%25B6%25E5%25B7%25A5%25E7%25A8%258B" \t "_blank)中的一种[软件架构](https://link.zhihu.com/?target=https%3A//zh.wikipedia.org/wiki/%25E8%25BD%25AF%25E4%25BB%25B6%25E6%259E%25B6%25E6%259E%2584)模式，把软件系统分为三个基本部分：模型（Model）、视图（View）和控制器（Controller）：

* Controller——负责转发请求，对请求进行处理
* View——负责界面显示
* Model——业务功能编写（例如算法实现）、数据库设计以及数据存取操
* 作实现

在JSP/Servlet开发的软件系统中，这三个部分的描述如下所示：



1. Web浏览器发送HTTP请求到服务端，被Controller(Servlet)获取并进行处理（例如参数解析、请求转发）
2. Controller(Servlet)调用核心业务逻辑——Model部分，获得结果
3. Controller(Servlet)将逻辑处理结果交给View（JSP），动态输出HTML内容
4. 动态生成的HTML内容返回到浏览器显示

MVC模式在Web开发中的好处是非常明显，它规避了JSP与Servlet各自的短板，Servlet只负责业务逻辑而不会通过out.append()动态生成HTML代码；JSP中也不会充斥着大量的业务代码。这大大提高了代码的可读性和可维护性。

1. **JavaEE**

## JavaEE主要技术

JavaEE 号称有十三种核心技术。它们分别是：**JDBC**、**JNDI**、**EJB**、RMI、**Servlet**、**JSP**、XML、JMS、Java IDL、JTS、JTA、JavaMail和JAF