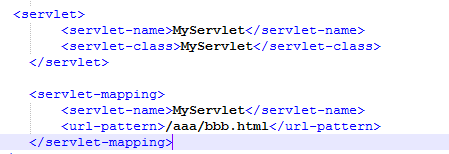
Servlet

1. **Servlet**是一个Java类，且是一个特殊的Java类；特殊性在于实现了**接口Servlet**。
2. **Web服务器**可以**接收多个不同路径的请求**，它可以把不同路径的请求分别映射给一个**Servlet**去处理。

如/aaa、/bbb等不同的web工程，利用Servlet去处理。

1. **原始方法**：把编译好的java文件的.class文件拷贝到Web应用的**WEB-INF文件夹中**，然后在WEB-INF文件夹中的web.xml中，声明一下：



就这样完成了java文件与Servlet的联系。

**注意配置路径的时候：<url-pattern>url模式的后面：**

**星号的前面不能有斜杠，这种写法是错误的，去掉.ddd。**

1. 对于java文件，主要是编写一个**MyServlet类**，此类只需要实现**javax.servlet.Servlet接口**，并实现其中的抽象方法，其中最主要的方法就是**service方法**，此方法有两个参数，一个是**ServletRequest request请求**,另外一个是 **ServletResponse response答复**。

需要导入包**servlet-api.jar包**。

理解： **service方法具有两个参数，一个是request对象，**是有内容的，包含了请求信息，**另外一个是response对象，**是没有内容的，相当于是空篮子，是用来装东西的。

1. 实现**javax.servlet.Servlet接口**中，最重要的方法就是service方法，但是还有其他必须实现的方法，如**init方法**即初始化方法。
2. 



设置内容为文本，解码利用encoding指定的方式进行解码。

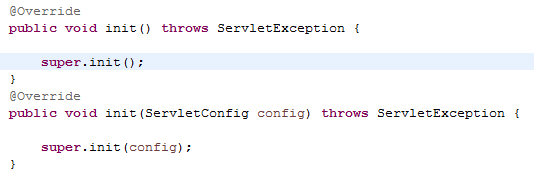
1. 如果直接利用打印流PrintWriter，利用ServletResponse获取的PrintWriter对象可以利用getWriter获取，直接打印的是字符串，但是并没有指定编码方式，默认的编码方式是。
2. **init方法用于初始化。**
3. **javax.servlet.GenericServlet:**既然创建一个Servlet需要实现javax.servlet.Servlet接口，且实现此接口存在很多的共性，所以直接提供了一个类**GenericServlet**，从而可以继承此类即可。



GenericServlet类实现Servlet接口，直接继承于java.lang.Object，它的直接子类是**HttpServlet类**。

1. **模板方法设计模式**：

就是把**共性的内容**提取出来，然后把**不同的部分**封装成一个小的方法，让子类去**继承并实现**。如：



父类中已经把**带参数的init方法的共性部分**写出来了，把**独特的部分**利用**空参数的init方法**代替了，简单的话，子类只需要重写父类的空参数的init方法即可，但是如果直接实现带参数的init方法也可以，相当于把父类的**带参数的init方法**覆盖了。这种设计方法称为**模板方法设计模式**。

1. HttpServlet：GenericServlet的直接子类。

五个重要的API。

1. **Servlet：**
2. **ServletContext：操作Servlet容器的接口。MIME类型。**
3. **ServletConfig：翻译为Servlet上下文或者Servlet环境。**
4. **ServletRequest：**
5. **ServletResponse：**
6. Servlet的生命周期：