

Servlet和Filter的url匹配url-pattern PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/245/2021_2022_Servlet_E5_92_8C_c104_245782.htm Servlet和filter是J2EE开发中常用的技术，

使用方便，配置简单，老少皆宜。估计大多数朋友都是直接配置用，也没有关心过具体的细节，今天遇到一个问题，上网查了servlet的规范才发现，servlet和filter中的url-pattern还是有一些文章在里面的，总结了一些东西，放出来供大家参考，以免遇到问题又要浪费时间。

一、servlet容器对url的匹配过程：当一个请求发送到servlet容器的时候，容器先会将请求的url减去当前应用上下文的路径作为servlet的映射url，比如我访问的是<http://localhost/test/aaa.html>，我的应用上下文是test，容器会将<http://localhost/test>去掉，剩下的/aaa.html部分拿来

做servlet的映射匹配。这个映射匹配过程是有顺序的，而且当有一个servlet匹配成功以后，就不会去理会剩下的servlet了（filter不同，后文会提到）。其匹配规则和顺序如下：1. 精确路径匹配。例子：比如servletA的url-pattern为/test，servletB的url-pattern为/*，这个时候，如果我访问的url

为<http://localhost/test>，这个时候容器就会先进行精确路径匹配，发现/test正好被servletA精确匹配，那么就去调用servletA，也不会去理会其他的servlet了。

2. 最长路径匹配。例子：servletA的url-pattern为/test/*，而servletB的url-pattern为/test/a/*，此时访问<http://localhost/test/a>时，容器会选择路径最长的servlet来匹配，也就是这里的servletB。

3. 扩展匹配，如果url最后一段包含扩展，容器将会根据扩展选择合适的servlet。例子：servletA的url-pattern：*.action 4. 如果前面三条

规则都没有找到一个servlet，容器会根据url选择对应的请求资源。如果应用定义了一个default servlet，则容器会将请求丢给default servlet（什么是default servlet？后面会讲）。根据这个规则表，就能很清楚的知道servlet的匹配过程，所以定义servlet的时候也要考虑url-pattern的写法，以免出错。对于filter，不会像servlet那样只匹配一个servlet，因为filter的集合是一个链，所以只会有处理的顺序不同，而不会出现只选择一个filter。Filter的处理顺序和filter-mapping在web.xml中定义的顺序相同。

二、url-pattern详解

在web.xml文件中，以下语法用于定义映射：以“/”开头和以“/*”结尾的是用来做路径映射的。以前缀“*.”开头的是用来做扩展映射的。“”是用来定义default servlet映射的。剩下的都是用来定义详细映射的。比如：/aa/bb/cc.action 所以，为什么定义“/*.action”这样一个看起来很正常的匹配会错？因为这个匹配即属于路径映射，也属于扩展映射，导致容器无法判断。

100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com