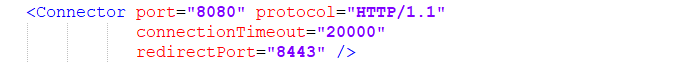
Javaweb学习总结（二）-- tomcat服务器学习和使用（一）

1. Tomcat服务器端口的配置

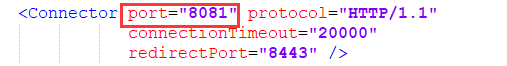
Tomcat的所有配置都放在conf文件之中，里面的server.xml文件是配置的核心文件。

如果想修改tomcat服务器的启动端口，则可以在server.xml配置文件中的connector节点进行的端口修改，例如：讲tomcat的服务器的启动端口由默认的8080改为8081端口。

Tomcat服务器启动端口默认设置：



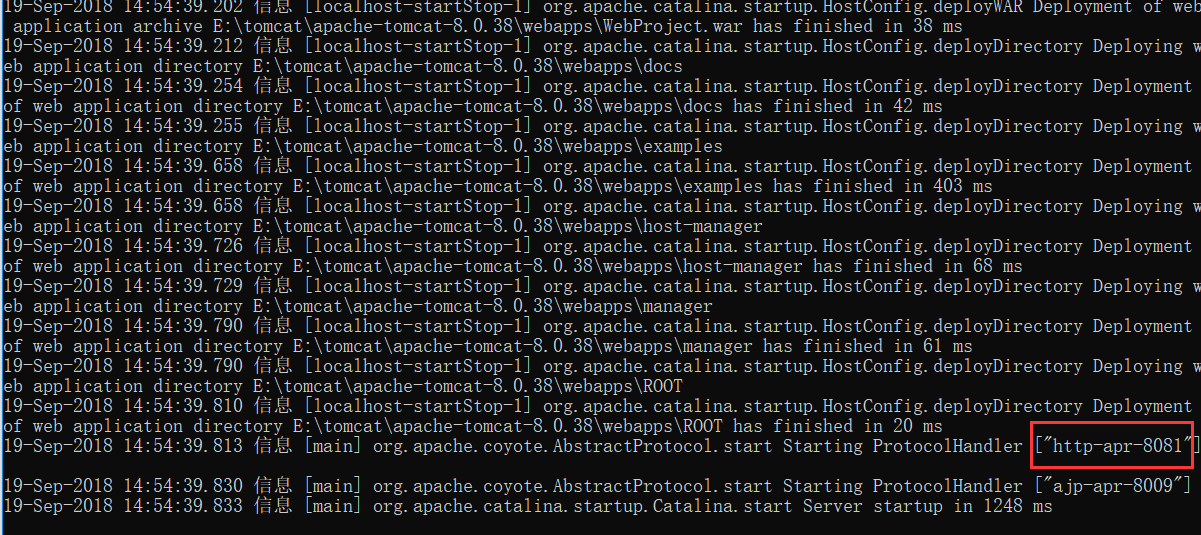
降tomcat服务器启动端口修改成8081的端口号：



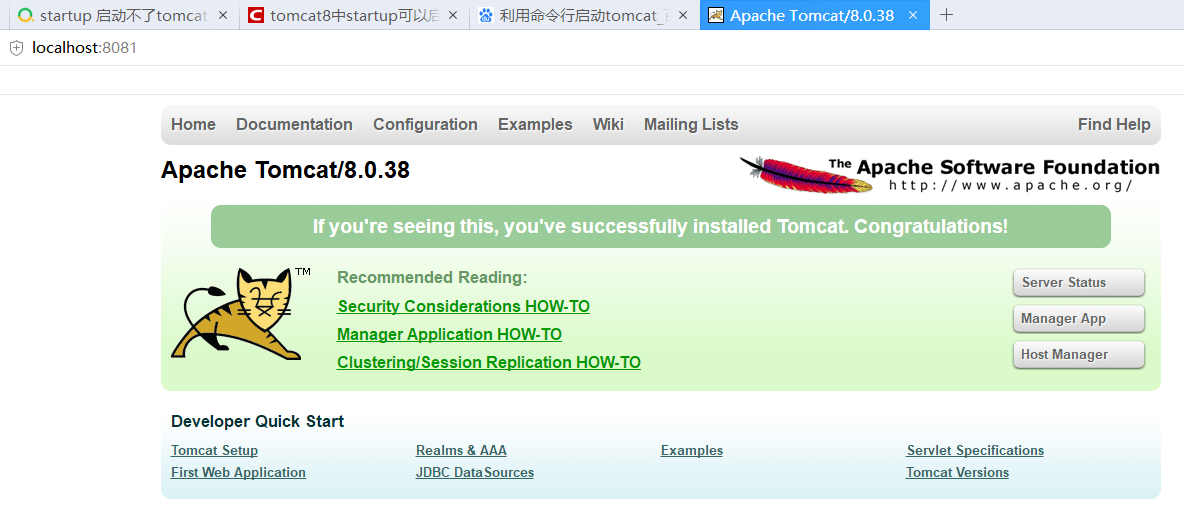
这样就把原来默认tomcat的8080端口改成了8081的端口了，需要注意的是，一旦服务器中的\*.xml文件改变了，则tomcat服务器必须重新启动，重新启动之后将重新读取新的配置信息。因为已经在server.xml文件中奖tomcat的启动端口修改成了8081，所以tomcat服务器启动时就以8081端口启动了， 如果下图所示：

启动命令：

在tomcat的bin目录下使用 .\startup.bat 命令启动tomcat



访问tomcat服务器也必须以新的访问端口去访问：<http://localhost:8081>,如下图所示：

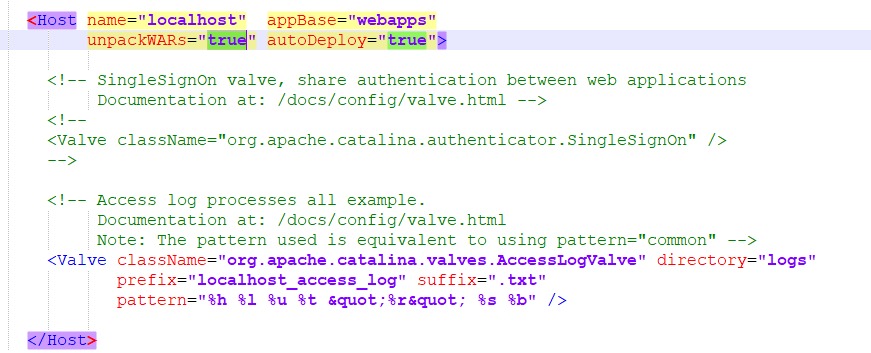


1. Tomcat服务器虚拟映射目录的映射方式

web应用开发好后，弱项提供外界访问，需要把web应用所在的目录交给web服务器管理，这个过程称之为虚拟目录的映射。那么在tomcat服务器中，如何进行虚拟目录的映射呢？总共有如下的集中方式：

2.1 虚拟目录的映射方式一：在server.xml的host元素中配置

找到server.xml文件的host元素，如下图所示：



在<Host></Host>这对标签加上

<Context path=”/JavaWebApp” docBse=”F:\JavaWebDemoProject”>

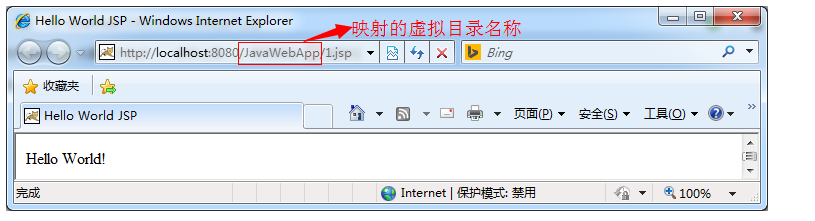
即可将在f盘下的JavaWebDemoProject这个JavaWeb应用映射到JavaWebApp这个虚拟目录上，JavaWebApp这个虚拟目录是由Tomcat服务器管理的，JavaWebApp是一个硬盘上不存在的目录，是我们随便写的一个目录，也就是虚拟的一个目录，所以称之为“虚拟目录”，代码如下：

其中，Context表示上下文，代表的就是一个javaweb应用，Context元素有两个属性，

Path：用来配置虚拟目录，必须以“/”开头

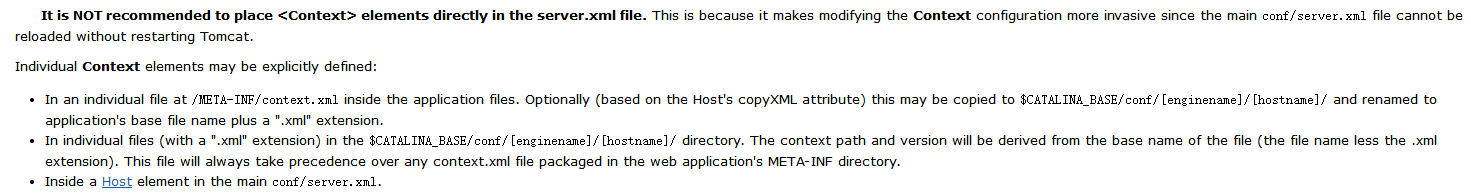
docBase：配置此虚拟目录对应着硬盘上的web应用所在的目录

使用浏览器访问“/JavaWebApp”这个虚拟目录下的1.jap这个web资源，访问结果如下：



1. jsp可以正常访问，这说明我们已经成功地将在F盘下的JavaWebDemoProject这个JavaWeb应用映射到JavaWebApp这个虚拟目录上了，**访问"/JavaWebApp/1.jsp"就相当于访问"F:\JavaWebDemoProject\1.jsp"。**

注意：在tomcat6之后找那个， 不在建议在server.xml文件中使用配置context元素的方式来添加虚拟目录的映射，因为每次修改server.xml文件后，tomcat服务器就必须要重新启动后才能重新加载server.xml文件。在tomcat服务器的文档<http://localhost:8080/docs/conf/context.html>中有这样的说明：



2.2 虚拟目录的映射方式二：让tomcat服务器自动映射

Tomcat服务器会自动管理webapps目录下的所有web应用，并把它映射成虚拟目录中的web应用，外界可以直接访问。

例如：把F盘下的JavaWebDemoProject这个javaweb应用直接copy到tomcat服务器webapps目录中，如下图所示：