

1 TOMCAT服务器

零、服务器

1、什么是服务器

服务器分为服务器软件和服务器硬件

服务器硬件：一台配置很高的计算机(一台计算也可以作为服务器使用)

服务器软件：需要在服务器硬件上安装服务器软件，这台电脑才可以对外提供服务器，让外界来访问。

服务器还分为Web服务器(tomcat)、数据库服务器(mysql)、邮件服务器...

WebLogic

WebSphere

tomcat

一、tomcat下载、安装

1、tomcat下载、安装

下载地址：<http://tomcat.apache.org>

tomcat分为很多的版本,有windows版本和Linux版本,还有32位和64位,还分为安装版和解压版(绿色版)

tomcat也需要JDK版本的支持:

tomcat7 jdk 1.6及以上版本支持

tomcat8 jdk 1.7及以上版本支持

...

我们采用的是解压版(tomcat8.0),解压之后就可以使用!

2、tomcat启动、配置

在启动服务器之前需要检查是否配置了JAVA_HOME环境变量

JAVA_HOME环境变量指向JDK的根目录(bin的父目录)

该变量的作用是用于通知tomcat服务器JDK安装在哪一个位置!!!

变量名:JAVA_HOME

变量值:D:\Java\jdk1.8.0_45

启动tomcat: 通过[tomcat安装目录]/bin/startup.bat 启动服务器

关闭tomcat: 通过[tomcat安装目录]/bin/shutdown.bat 关闭服务器(或者直接将窗口关闭!)

如果服务器启动之后,可以通过访问:<http://localhost:8080>,如果看到tomcat服务器的主页,则证明服务器安装启动成功了!!

3、修改tomcat默认端口

tomcat服务器默认监听的端口为8080,这意味着每次在访问服务器时都需要在路径后面加上 :8080,如果在访问时,想省略端口,可以将tomcat监听的端口改为80(只有80可以省略不写!!)

找到[tomcat]/conf/server.xml文件,在文件的69行中,修改port属性的值改为80即可。

修改后需要重启服务器才可以生效!!!

4、常见问题:端口占用问题

在服务器启动时,可能会遇到端口占用问题:

java.net.BindException: Address already in use: bind

出现该问题的原因有两种:

(1)之前的服务器并没有完全关闭(仍然在占用80端口),导致现在服务器启动失败

解决方式: 运行shutdown.bat文件,将服务器按照正常的关闭流程再关闭一次. 再次启动服务器就可以启动成功了

(2)可能是别的程序在占用80端口,需要将别的程序先关闭,再启动服务器就OK了!

在cmd窗口中输入命令: netstat -ano 查看当前系统活动的进程,找到占用80端口的进程ID

再通过 taskkill /f /pid 进程ID 将占用80端口进程结束即可!!

二、tomcat目录结构

bin -- tomcat服务器所需要的一些批处理文件的存放目录

conf -- tomcat服务器配置文件的存放目录

其中server.xml文件的tomcat的核心配置文件

lib -- tomcat服务器运行时所依赖的一些jar包的存放目录

logs -- tomcat服务器日志文件的存放目录

temp -- tomcat服务器临时文件的存放目录

webapps -- localhost虚拟主机默认管理的目录,将组织好的Web应用放在该目录下,相当于发布到了localhost

虚拟主机中, 就可以通过浏览器来访问了

其实webapps目录就是服务器存放资源的目录

可以将一些资源文件(比如html、图片、jsp...)放在该目录下, 外界就可以通过浏览器来访问了.

work -- tomcat服务器工作文件的存放目录(JSP翻译为Servlet后的文件、session对象的序列化文件..)

三、虚拟主机和Web应用

1、什么是虚拟主机

虚拟主机就是tomcat服务器配置一个站点

tomcat可以配置多个站点, 每一个站点都是一台虚拟主机.

2、什么是Web应用

Web应用其实就是一个目录, 其中可以存放很多的资源文件(比如html、css、js、jsp、图片、servlet等), 将Web应用的目录放在服务器中运行, 外界就可以访问目录中的资源文件了

四、Web应用(!!!重点)

1、Web应用的目录结构

Web应用就是一个目录, 该目录中可以存放很多资源文件

news -- Web应用 (目录)

|
|-- WEB-INF以外的目录: 静态web资源(htmlcssjs), jsp, 图片、servlet等 直接放在WEB-INF以外的目录下, 可以通过浏览器直接访问

|-- WEB-INF目录: 放在这个目录中的资源将被保护起来, 浏览器不能直接访问

|--classes: 用来存放动态web资源(Servlet/jsp)中的class文件

|--lib: 用来存放动态web资源资源(Servlet/jsp)所依赖的一些jar包

|--web.xml: 当前web应用的核心的配置文件, web应用的很多配置操作都需要在这个文件中进行

2、部署Web应用到虚拟主机中

比如现在组织了一个Web应用叫做news, 希望将news Web应用部署到localhost虚拟主机中, 只需要将news目录直接拷贝到localhost主机默认管理的目录(webapps)下即可!!

访问的路径为: http://localhost/news/hello.html

3、扩展: 部署缺省的Web应用

在没有将news配置为缺省的(默认的)Web应用之前, 访问路径为:

http://localhost/news/hello.html

如果将news配置为缺省的Web应用(其实就是把目录名改为大写的ROOT), 访问的路径为:

http://localhost/hello.html

4、配置Web应用的主页

在配置主页之前, 访问hello.html路径为:

http://localhost/hello.html

如果把hello.html配置为主页: 再访问hello.html路径为:

http://localhost

五、其他相关

1、打war包

war包(jar包)其实就是一种压缩包格式(rar、zip、7z...)

可以将一个Web应用中的所有资源打成一个war包:

(1)将很多的文件和目录打成一个war包(单个文件)便于传输

(2)将文件和目录打成一个压缩包, 可以减少文件的体积!!

(3)war包这种格式, tomcat服务器能识别, 将war文件直接发布到tomcat中, 会自动解压发布!!!

打war包的方式:

进入Web应用的根目录, 选中所有文件和目录, 将所有文件和目录打成一个xxx.zip包, 再将后缀名改为war即可!!

需要注意的是:

(1)Web应用中的文件名和目录名不能包含中文

(2)先打成一个xxx.zip包(不要打成rar包), 否则可能会失败!

(3)打成的war包名称不能和已发布的Web应用重名!

2 HTTP协议

一、Http协议简介

1、什么是HTTP协议

协议可以理解为 规则、格式、标准

Http协议是用于规定浏览器和服务器的通信格式

在Http协议中规定了浏览器该按照什么格式给服务器发送请求信息

也规定了服务器该按照什么格式给浏览器发送响应信息

2、HTTP协议的三个原则

基于请求响应模型

一次请求对应一次响应

请求只能由浏览器发出(发起)，服务器只能被动的等待请求，做出回应!!

二、Http协议详解

1、Http请求信息格式

请求行

GET /news/hello.html HTTP/1.1

GET:请求方式为GET提交,常用的还有POST

/news/hello.html: 请求资源的路径

HTTP/1.1: 协议和版本

若干请求头(了解)

Host: localhost

— 通知服务器访问的是哪一台虚拟主机

Accept-Language: zh-cn, zh;q=0.8, en-us;q=0.5, en;q=0.3

— 通知服务器浏览器支持的语言环境

Cookie: cookie技术相关的一个头

...

(一个空白行)

请求实体内容

包含的是浏览器要发送给服务器的数据

如果请求方式是GET提交，请求实体中没有内容

如果请求方式是POST提交，并有数据发送给服务器，请求实体才会有内容!!

2、Http响应信息格式

状态行

HTTP/1.1 200 OK

HTTP/1.1: 协议和版本

200: 三位的数值，表示请求处理的结果

200: 表示请求处理成功

404: 表示浏览器请求的资源不存在

500: 表示服务器处理的过程中抛出了异常

...

OK: 描述短语,可以忽略

若干响应头

Content-Type: text/html; charset=utf-8

Content-Length: 389 单位是字节

Refresh: 定时刷新

3;url=http://www.baidu.com

注册之后，提示用户3秒之后跳转到首页

Set-Cookie: cookie技术相关的一个头

...

(一个空白行)

响应实体内容

就是浏览器请求的资源文件的内容

比如浏览器请求的是一个图片，响应实体中就是文件的内容

如果浏览器请求的是一个html文件，响应实体中就是html文件的内容

3、GET提交和POST提交

在HTTP协议中规定了7中提交方式，其中5种都不常用
只用GET和POST提交

3.1. GET提交和POST提交的应用

什么时候是POST提交?

只有使用表单(form)并且在表单上明确的通过method="POST"才是POST提交，其他方式都是GET提交

3.2. GET提交和POST提交的区别

主要体现在请求参数传输过程的不相同

GET提交的特点:

- (1)通过地址栏发送数据给服务器，相对不安全
- (2)通过地址栏发送数据,数据量不能太大(不超过1KB或4KB)

POST提交的特点:

- (1)在请求实体中发送数据给服务器，相比更安全
- (2)通过请求实体发送数据，数据量理论上是没有限制的!