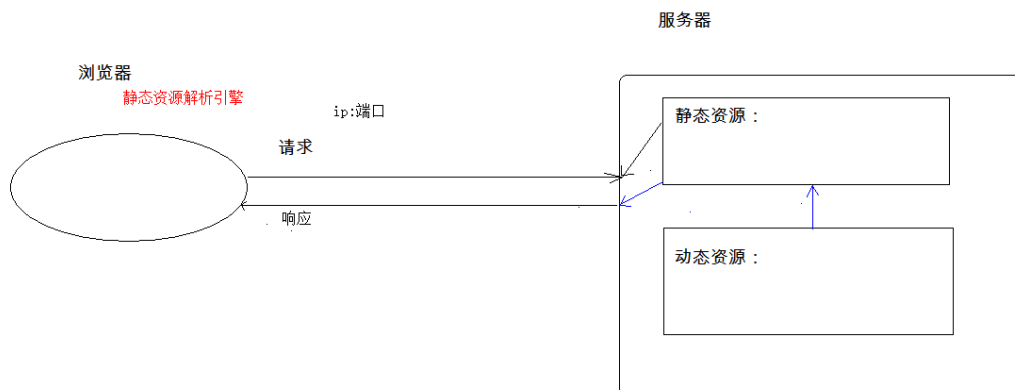


web相关概念



1. 软件架构

1. C/S: 客户端/服务器端
2. B/S: 浏览器/服务器端

2. 资源分类

1. 静态资源: 所有用户访问后, 得到的结果都是一样的, 称为静态资源. 静态资源可以直接被浏览器解析

* 如: `html,css,JavaScript`

2. 动态资源: 每个用户访问相同资源后, 得到的结果可能不一样. 称为动态资源. 动态资源被访问后, 需要先转换为静态资源, 在返回给浏览器

* 如: `servlet/jsp,php,asp....`

3. 网络通信三要素

1. IP: 电子设备(计算机)在网络中的唯一标识。
2. 端口: 应用程序在计算机中的唯一标识。 0~65536
3. 传输协议: 规定了数据传输的规则
 1. 基础协议:
 1. tcp: 安全协议, 三次握手。 速度稍慢
 2. udp: 不安全协议。 速度快

web服务器软件：

* 服务器：安装了服务器软件的计算机（本质上还是一台电脑，只不过这个电脑的性能非常好，硬盘大、内存大、cpu的个数多、cpu的等级比较高，总之配置很高； 这个电脑本身是一个硬件，得安装一个服务器软件后这个电脑才能被是服务器）

注：我们自己的电脑安装了mysql服务器软件后，我们的电脑就可以成为mysql服务器，成为了mysql服务器后，别人就可以通过登录我们的mysql账户密码来访问我们我们的mysql服务器及操作mysql类里面的内容

* 服务器软件：接收用户的请求，处理请求，做出响应

* web服务器软件：电脑上安装了web服务器软件后，电脑就是web服务器，也可以让别人访问到我们；接收用户的请求，处理请求，做出响应；动态资源只能运行在服务器软件上

* 在web服务器软件中，可以部署web项目，让用户通过浏览器来访问这些项目

* web容器：web服务器的别称； 即web服务器=web容器

* 常见的java相关的web服务器软件：

* webLogic: oracle公司，大型的JavaEE服务器，支持所有的JavaEE规范，收费的。

* webSphere: IBM公司，大型的JavaEE服务器，支持所有的JavaEE规范，收费的。


* JBOSS: JBOSS公司的，大型的JavaEE服务器，支持所有的JavaEE规范，收费的。


* Tomcat: Apache基金组织，中小型的JavaEE服务器，仅仅支持少量的JavaEE规范servlet/jsp。开源的，免费的。

Tomcat下载

1.进入<http://tomcat.apache.org/>

2.点击左方的 Download-Tomcat8





September 21-23
www.apachecon.com

[Save the date!](#)

Apache Tomcat

- Home
- Taglibs
- Maven Plugin

Download

- Which version?
- Tomcat 10
- Tomcat 9
- Tomcat 8
- Tomcat Migration Tool for Jakarta EE
- Tomcat Connectors
- Tomcat Native
- Taglibs
- Archives

Documentation

- Tomcat 10.1 (alpha)
- Tomcat 10.0
- Tomcat 9.0
- Tomcat 8.5
- Tomcat Connectors
- Tomcat Native
- Wiki
- Migration Guide
- Presentations

Apache Tomcat

The Apache Tomcat[®] software is an [Authentication](#) specifications. These

The Jakarta EE platform is the evolution developed as part of Java EE.

The Apache Tomcat software is developed by the best-of-breed developers from

Apache Tomcat software powers many of the [PoweredBy](#) wiki page.

Apache Tomcat, Tomcat, Apache, etc.

Tomcat 8.5.73 Released

The Apache Tomcat Project is proud to announce that compared to 8.5.72 include:

- Improvements to native/APR
- Improve robustness of JNDI

Full details of these changes, and a

Tomcat 9.0.55 Released

The Apache Tomcat Project is proud to announce that compared to 9.0.54 include:

- Experimental OpenSSL support

3. 点击core（核心版）---- 64-bit Windows zip 进行下载

Mirrors

You are currently using **<https://dldcn.apache.org/>**. If you encounter a problem with this mirror, mirrors list) that should be available.

Other mirrors:

8.5.73

Please see the [README](#) file for packaging information. It explains what every distribution contains.

Binary Distributions

- Core:
 - [zip](#) ([pgp](#), [sha512](#))
 - [tar.gz](#) ([pgp](#), [sha512](#))
 - [32-bit Windows zip](#) ([pgp](#), [sha512](#))
 - [64-bit Windows zip](#) ([pgp](#), [sha512](#))
 - [32-bit/64-bit Windows Service Installer](#) ([pgp](#), [sha512](#))
- Full documentation:
 - [tar.gz](#) ([pgp](#), [sha512](#))
- Deployer:
 - [zip](#) ([pgp](#), [sha512](#))
 - [tar.gz](#) ([pgp](#), [sha512](#))
- Extract:

Tomcat安装

- 解压压缩包即可
- 注意：安装的目录命名建议不要有中文与空格

Tomcat卸载

- 删除安装目录即可

Tomcat目录结构

bin	可执行文件	2018/5/31 14:50	文件夹	
conf	配置文件	2018/5/31 14:50	文件夹	
lib	依赖jar包	2018/5/31 14:50	文件夹	
logs	日志文件	2018/4/27 21:24	文件夹	
temp	临时文件	2018/5/31 14:50	文件夹	
webapps	存放web项目	2018/5/31 14:50	文件夹	
work	存放运行时的数据	2018/4/27 21:24	文件夹	
LICENSE		2018/4/27 21:24	文件	57 KB
NOTICE		2018/4/27 21:24	文件	2 KB
RELEASE-NOTES		2018/4/27 21:24	文件	8 KB
RUNNING.txt		2018/4/27 21:24	TXT 文件	17 KB

bin目录:

- xxx.bat: 批处理文件,双击即可运行
 - exe: 双击即可运行
 - sh: sh是在linux下可以执行的文件，windos下不能执行

剪贴板	组织	新建	打开	选择
↑ > Computer > LLH (D:) > apache-tomcat-8.0.52-64 > bin				
名称	修改日期	类型	大小	
bootstrap.jar	2018/4/28 19:27	Executable Jar File	35 KB	
catalina.bat	2018/4/28 19:27	Windows 批处理...	16 KB	
catalina.sh	2018/4/28 19:27	Shell Script	23 KB	
catalina-tasks.xml	2018/4/28 19:27	XML 文件	2 KB	
commons-daemon.jar	2018/4/28 19:27	Executable Jar File	25 KB	
commons-daemon-native.tar.gz	2018/4/28 19:27	WinRAR 压缩文件	203 KB	
configtest.bat	2018/4/28 19:27	Windows 批处理...	2 KB	
configtest.sh	2018/4/28 19:27	Shell Script	2 KB	
daemon.sh	2018/4/28 19:27	Shell Script	9 KB	
digest.bat	2018/4/28 19:27	Windows 批处理...	3 KB	
digest.sh	2018/4/28 19:27	Shell Script	2 KB	
service.bat	2018/4/28 19:27	Windows 批处理...	8 KB	
setclasspath.bat	2018/4/28 19:27	Windows 批处理...	4 KB	
setclasspath.sh	2018/4/28 19:27	Shell Script	4 KB	
shutdown.bat	2018/4/28 19:27	Windows 批处理...	2 KB	
shutdown.sh	2018/4/28 19:27	Shell Script	2 KB	
startup.bat	2018/4/28 19:27	Windows 批处理...	2 KB	
startup.sh	2018/4/28 19:27	Shell Script	2 KB	
tcnative-1.dll	2018/4/28 19:27	应用程序扩展	2,077 KB	
tomcat8.exe	2018/4/28 19:27	应用程序	112 KB	
tomcat8w.exe	2018/4/28 19:27	应用程序	116 KB	
tomcat-juli.jar	2018/4/28 19:27	Executable Jar File	48 KB	
tomcat-native.tar.gz	2018/4/28 19:27	WinRAR 压缩文件	396 KB	
tool-wrapper.bat	2018/4/28 19:27	Windows 批处理...	5 KB	
tool-wrapper.sh	2018/4/28 19:27	Shell Script	6 KB	
version.bat	2018/4/28 19:27	Windows 批处理...	2 KB	
version.sh	2018/4/28 19:27	Shell Script	2 KB	

conf目录:

- 程序员在使用tomcat时要和它交互一些信息，这需要conf里的配置文件来完成

- xml、properties、policy、xsd都是配置文件

Computer > LLH (D:) > apache-tomcat-8.0.52-64 > conf >				
名称	修改日期	类型	大小	
Catalina	2021/6/3 18:07	文件夹		
catalina.policy	2018/4/28 19:27	POLICY 文件	14 KB	
catalina.properties	2018/4/28 19:27	PROPERTIES 文件	8 KB	
context.xml	2018/4/28 19:27	XML 文件	2 KB	
logging.properties	2018/4/28 19:27	PROPERTIES 文件	4 KB	
server.xml	2021/8/26 13:03	XML 文件	7 KB	
tomcat-users.xml	2018/4/28 19:27	XML 文件	3 KB	
tomcat-users.xsd	2018/4/28 19:27	VCEXpress.xsd.1...	3 KB	
web.xml	2018/4/28 19:27	XML 文件	171 KB	

lib目录:

- 存放依赖jar包，tomcat启动运行的时候需要依赖于一些jar包才能运行

Computer > LLH (D:) > apache-tomcat-8.0.52-64 > lib				
名称	修改日期	类型	大小	
annotations-api.jar	2018/4/28 19:27	Executable Jar File	13 KB	
catalina.jar	2018/4/28 19:27	Executable Jar File	1,553 KB	
catalina-ant.jar	2018/4/28 19:27	Executable Jar File	54 KB	
catalina-ha.jar	2018/4/28 19:27	Executable Jar File	118 KB	
catalina-storeconfig.jar	2018/4/28 19:27	Executable Jar File	74 KB	
catalina-tribes.jar	2018/4/28 19:27	Executable Jar File	272 KB	
ecj-4.6.3.jar	2018/4/28 19:27	Executable Jar File	2,393 KB	
el-api.jar	2018/4/28 19:27	Executable Jar File	80 KB	
jasper.jar	2018/4/28 19:27	Executable Jar File	574 KB	
jasper-el.jar	2018/4/28 19:27	Executable Jar File	159 KB	
jsp-api.jar	2018/4/28 19:27	Executable Jar File	61 KB	
servlet-api.jar	2018/4/28 19:27	Executable Jar File	239 KB	
tomcat-api.jar	2018/4/28 19:27	Executable Jar File	10 KB	
tomcat-coyote.jar	2018/4/28 19:27	Executable Jar File	714 KB	
tomcat-dbcp.jar	2018/4/28 19:27	Executable Jar File	264 KB	
tomcat-i18n-es.jar	2018/4/28 19:27	Executable Jar File	67 KB	
tomcat-i18n-fr.jar	2018/4/28 19:27	Executable Jar File	41 KB	
tomcat-i18n-ja.jar	2018/4/28 19:27	Executable Jar File	43 KB	
tomcat-jdbc.jar	2018/4/28 19:27	Executable Jar File	142 KB	
tomcat-jni.jar	2018/4/28 19:27	Executable Jar File	31 KB	
tomcat-util.jar	2018/4/28 19:27	Executable Jar File	117 KB	
tomcat-util-scan.jar	2018/4/28 19:27	Executable Jar File	202 KB	
tomcat-websocket.jar	2018/4/28 19:27	Executable Jar File	222 KB	
websocket-api.jar	2018/4/28 19:27	Executable Jar File	37 KB	

logs目录:

- 我们可以通过logs目录里的日志文件来监控或者来调试程序（注：tomcat第一次下载解压到电脑时log目录里是没有任何文件的，第一次启动tomcat后就有log文件了）

剪贴板	组织	新建	打开	选择	
↑	Computer > LLH (D:) > apache-tomcat-8.0.52-64 > logs				
名称	修改日期	类型	大小		
catalina.2021-06-03.log	2021/6/3 18:38	文本文档	12 KB		
catalina.2021-08-25.log	2021/8/25 22:08	文本文档	25 KB		
catalina.2021-08-26.log	2021/8/26 22:05	文本文档	168 KB		
catalina.2021-08-27.log	2021/8/27 10:12	文本文档	13 KB		
commons-daemon.2021-06-03.log	2021/6/3 18:38	文本文档	1 KB		
host-manager.2021-06-03.log	2021/6/3 18:38	文本文档	0 KB		
host-manager.2021-08-25.log	2021/8/25 20:17	文本文档	0 KB		
host-manager.2021-08-26.log	2021/8/26 10:26	文本文档	0 KB		
host-manager.2021-08-27.log	2021/8/27 10:11	文本文档	0 KB		
localhost.2021-06-03.log	2021/6/3 18:38	文本文档	12 KB		
localhost.2021-08-25.log	2021/8/25 20:24	文本文档	24 KB		
localhost.2021-08-26.log	2021/8/26 22:05	文本文档	111 KB		
localhost.2021-08-27.log	2021/8/27 10:12	文本文档	13 KB		
localhost_access_log.2018-06-29.txt	2018/6/29 18:00	文本文档	1 KB		
localhost_access_log.2021-06-03.txt	2021/6/3 18:39	文本文档	1 KB		
localhost_access_log.2021-06-04.txt	2021/6/4 0:36	文本文档	1 KB		
localhost_access_log.2021-08-25.txt	2021/8/25 23:04	文本文档	15 KB		
localhost_access_log.2021-08-26.txt	2021/8/26 21:33	文本文档	4 KB		
localhost_access_log.2021-08-27.txt	2021/8/27 10:11	文本文档	0 KB		
manager.2021-06-03.log	2021/6/3 18:38	文本文档	0 KB		
manager.2021-08-25.log	2021/8/25 20:17	文本文档	0 KB		
manager.2021-08-26.log	2021/8/26 10:26	文本文档	0 KB		
manager.2021-08-27.log	2021/8/27 10:11	文本文档	0 KB		
tomcat8-stderr.2021-06-03.log	2021/6/3 18:38	文本文档	12 KB		
tomcat8-stdout.2021-06-03.log	2021/6/3 18:38	文本文档	1 KB		

temp目录：

- 这里面是用来存储一些临时文件数据的，这个目录不太常用

剪贴板	组织	新建	打开	选择	
↑	Computer > LLH (D:) > apache-tomcat-8.0.52-64 > temp				
名称	修改日期	类型	大小		
safeToDelete.tmp	2018/4/28 19:27	TMP 文件	0 KB		

webapps目录：

- 用来存放我们的web项目，tomcat自带以下6个项目，这些项目里放了一些图片、页面等文件

剪贴板	组织	新建	打开	选择
↑ > Computer > LLH (D:) > apache-tomcat-8.0.52-64 > webapps >				
名称	修改日期	类型	大小	
docs	2021/6/3 18:07	文件夹		
examples	2021/6/3 18:07	文件夹		
host-manager	2021/6/3 18:07	文件夹		
LZPMS	2021/6/3 18:07	文件夹		
manager	2021/6/3 18:07	文件夹		
ROOT	2021/6/3 18:07	文件夹		

work目录：

- 用来存放运行是的数据

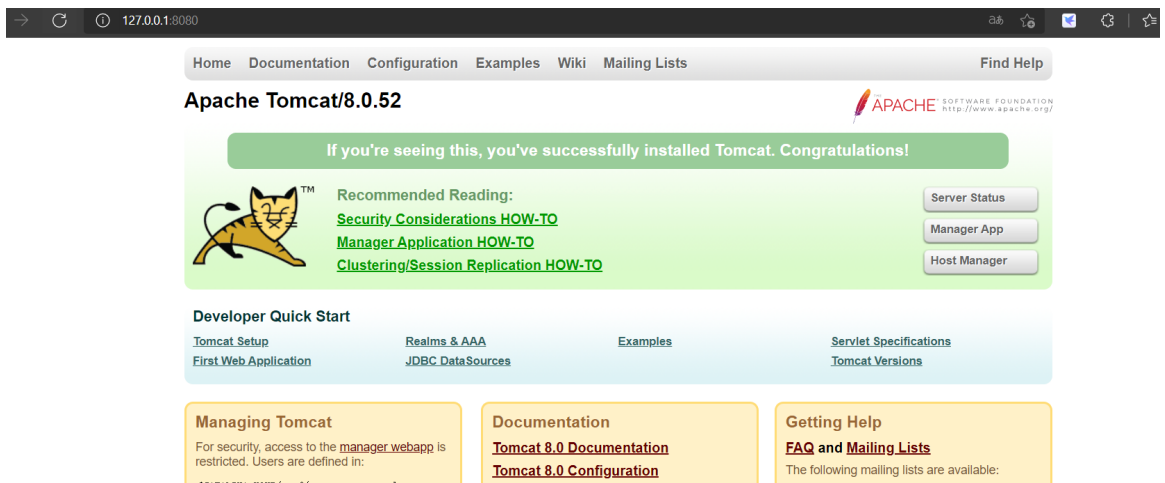
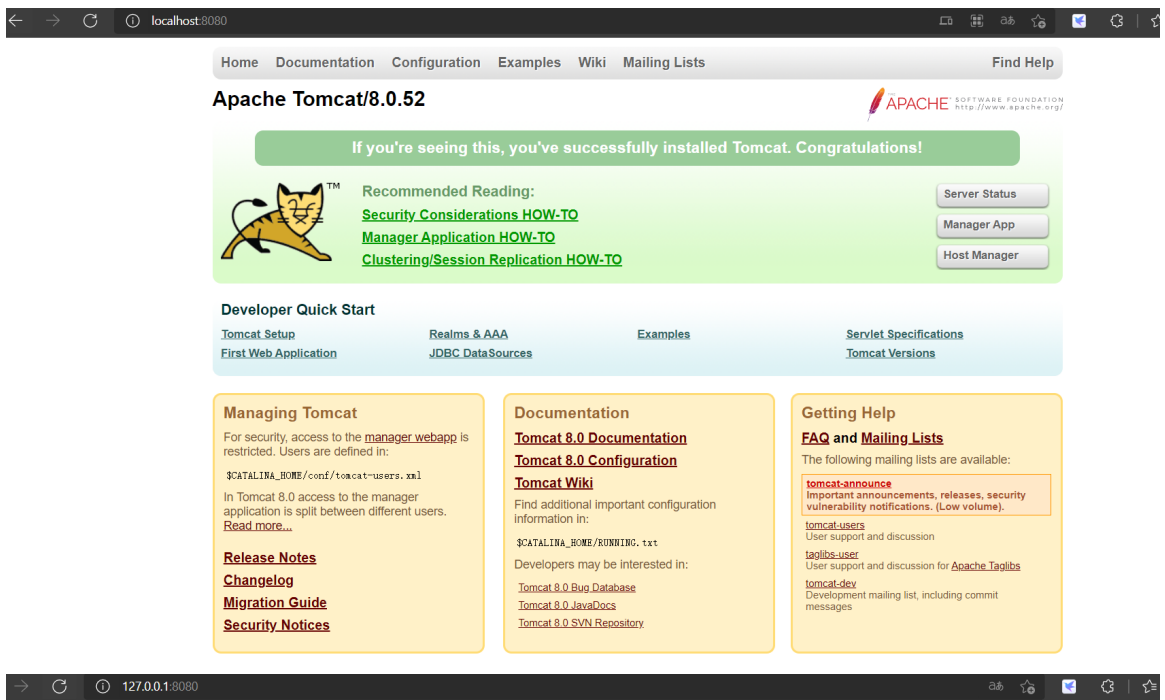
剪贴板	组织	新建	打开	选择
↑ > Computer > LLH (D:) > apache-tomcat-8.0.52-64 > work >				
名称	修改日期	类型	大小	
Catalina	2021/6/3 18:07	文件夹		

Tomcat启动

- 双击bin/startup.bat
- 会自动弹出命令行窗口-----一直在打印信息-----直到输出“Server start in xxx ms”则启动成功

Tomcat访问

- 自己访问自己：浏览器中输入<http://localhost:8080>（Tomcat默认端口是8080）或者127.0.0.1:8080或者localhost:8080（如果是高级浏览器，除了IE之外，http可以省略不用写）----然后可以访问到Tomcat自带的项目即webappps里的root目录----
--若网页打开成功则证明Tomcat启动成功



- 别人访问自己：cmd输入ipconfig-----可以看到自己的ip地址-----别人输入 <http://ip:8080> 则也可以访问到

```

C:\Users\15516535379>ipconfig

Windows IP 配置

无线局域网适配器 本地连接* 1:

    媒体状态 . . . . . : 媒体已断开连接
    连接特定的 DNS 后缀 . . . . . :

无线局域网适配器 本地连接* 2:

    连接特定的 DNS 后缀 . . . . . :
    本地链接 IPv6 地址. . . . . : fe80::fce9:94ff:fe89:b719%20
    IPv4 地址 . . . . . : 192.168.137.1
    子网掩码 . . . . . : 255.255.255.0
    默认网关. . . . . :

以太网适配器 VMware Network Adapter VMnet1:

    连接特定的 DNS 后缀 . . . . . :
    本地链接 IPv6 地址. . . . . : fe80::250:56ff:fec0:1%18
    IPv4 地址 . . . . . : 192.168.16.1
    子网掩码 . . . . . : 255.255.255.0
    默认网关. . . . . :

以太网适配器 VMware Network Adapter VMnet8:

    连接特定的 DNS 后缀 . . . . . :
    本地链接 IPv6 地址. . . . . : fe80::250:56ff:fec0:8%19
    IPv4 地址 . . . . . : 192.168.109.1
    子网掩码 . . . . . : 255.255.255.0
    默认网关. . . . . :

无线局域网适配器 WLAN:

    连接特定的 DNS 后缀 . . . . . :
    本地链接 IPv6 地址. . . . . : fe80::dee9:94ff:fe89:b719%17
    IPv4 地址 . . . . . : 10.37.43.39
    子网掩码 . . . . . : 255.255.0.0
    默认网关. . . . . : 10.37.0.1

以太网适配器 以太网 2:


    媒体状态 . . . . . : 媒体已断开连接
    连接特定的 DNS 后缀 . . . . . :

```


← → ↻ ⚠ 不安全 | 192.168.137.1:8080

Home Documentation Configuration Examples Wiki Mailing Lists Find Help

Apache Tomcat/8.0.52

 SOFTWARE FOUNDATION
http://www.apache.org/

If you're seeing this, you've successfully installed Tomcat. Congratulations!



Recommended Reading:

- [Security Considerations HOW-TO](#)
- [Manager Application HOW-TO](#)
- [Clustering/Session Replication HOW-TO](#)

Server Status

Manager App

Host Manager

Developer Quick Start

Tomcat Setup	Realms & AAA	Examples	Servlet Specifications
First Web Application	JDBC DataSources		Tomcat Versions

Managing Tomcat

For security, access to the `manager webapp` is restricted. Users are defined in:

```
$CATALINA_HOME/conf/tomcat-users.xml
```

In Tomcat 8.0 access to the manager application is split between different users.

[Read more...](#)

[Release Notes](#)

[Changelog](#)

Documentation

[Tomcat 8.0 Documentation](#)

[Tomcat 8.0 Configuration](#)

[Tomcat Wiki](#)

Find additional important configuration information in:

`$CATALINA_HOME/RUNNING.txt`

Developers may be interested in:

Getting Help

FAQ and Mailing Lists

The following mailing lists are available:

tomcat-announce

Important announcements, releases, security vulnerability notifications. (Low volume).

[tomcat-users](#)

User support and discussion

[taglibs-user](#)

User support and discussion for [Apache Taglibs](#)

[tomcat-dev](#)

- 自己访问别人：自己输入 `http://别人的ip:8080` 则也可以访问到

Tomcat问题:

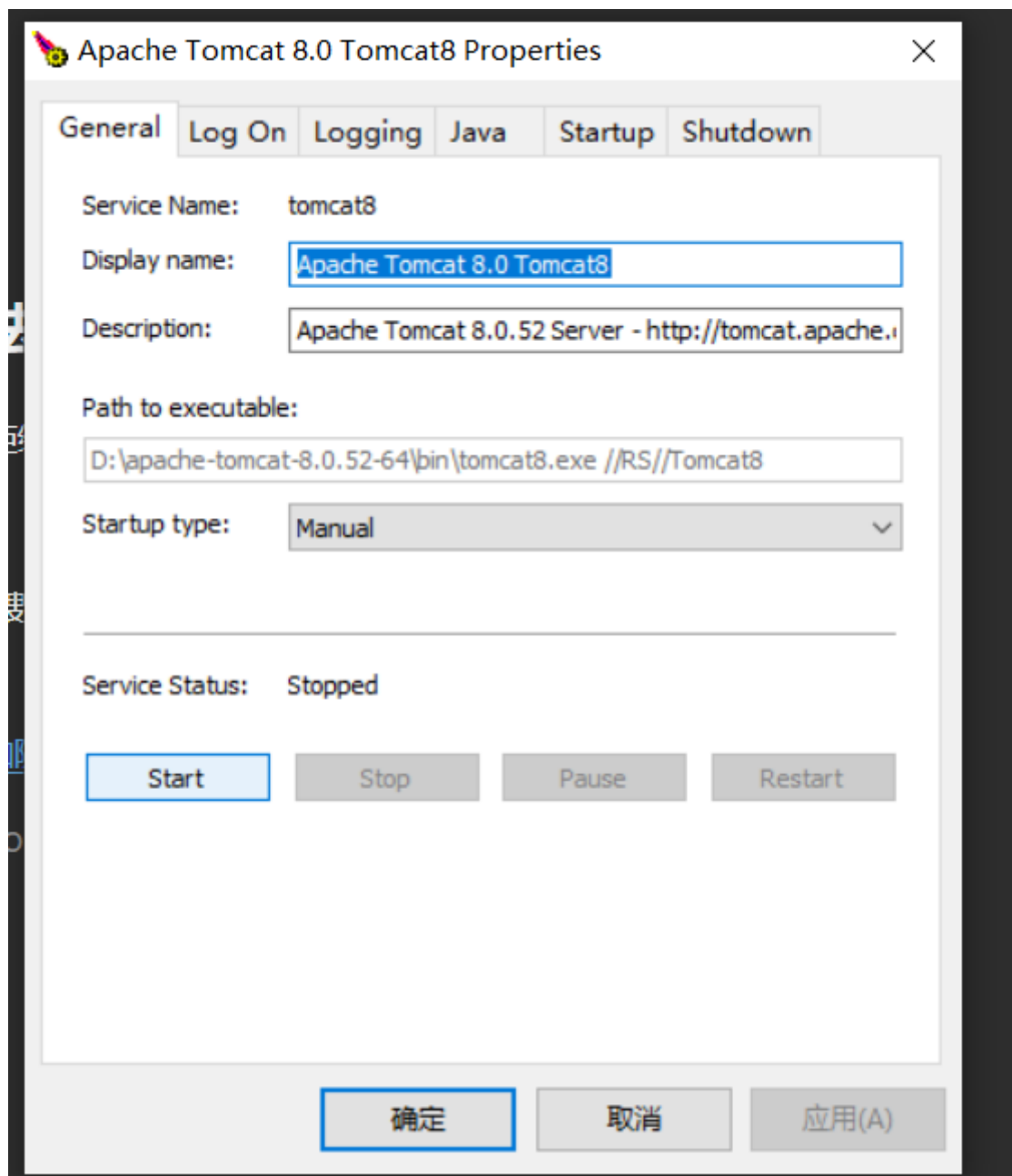
1.启动时出现以下界面，访问localhost:8080不成功

```
Tomcat
29-Nov-2021 00:31:42.656 信息 [localhost-startStop-1] org.apache.catalina.startup.HostConfig.deployDirectory Deploying web application directory D:\apache-tomcat-8.0.52-64\webapps\manager
29-Nov-2021 00:31:42.734 信息 [localhost-startStop-1] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web application directory D:\apache-tomcat-8.0.52-64\webapps\manager has finished in 78 ms
29-Nov-2021 00:31:42.740 信息 [localhost-startStop-1] org.apache.catalina.startup.HostConfig.deployDirectory Deploying web application directory D:\apache-tomcat-8.0.52-64\webapps\ROOT
29-Nov-2021 00:31:42.797 信息 [localhost-startStop-1] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web application directory D:\apache-tomcat-8.0.52-64\webapps\ROOT has finished in 58 ms
29-Nov-2021 00:31:42.814 信息 [main] org.apache.coyote.AbstractProtocol.start Starting ProtocolHandler ["http-apr-8080"]
29-Nov-2021 00:31:42.841 信息 [main] org.apache.coyote.AbstractProtocol.start Starting ProtocolHandler ["ajp-apr-8009"]
29-Nov-2021 00:31:42.852 信息 [main] org.apache.catalina.startup.Catalina.start Server startup in 20445 ms
29-Nov-2021 00:31:44.170 信息 [Abandoned connection cleanup thread] org.apache.catalina.loader.WebappClassLoaderBase.checkStateForResourceLoading Illegal access: this web application instance has been stopped already. Could not load []. The following stack trace is thrown for debugging purposes as well as to attempt to terminate the thread which caused the illegal access.
java.lang.IllegalStateException: Illegal access: this web application instance has been stopped already. Could not load []. The following stack trace is thrown for debugging purposes as well as to attempt to terminate the thread which caused the illegal access.
    at org.apache.catalina.loader.WebappClassLoaderBase.checkStateForResourceLoading(WebappClassLoaderBase.java:1372)
    at org.apache.catalina.loader.WebappClassLoaderBase.getResource(WebappClassLoaderBase.java:1042)
    at com.mysql.jdbc.AbandonedConnectionCleanupThread.checkContextClassLoaders(AbandonedConnectionCleanupThread.java:90)
    at com.mysql.jdbc.AbandonedConnectionCleanupThread.run(AbandonedConnectionCleanupThread.java:63)
    at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1149)
    at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:624)
    at java.lang.Thread.run(Thread.java:748)
```

解决方法:

- 运行 bin/tomcatw.exe-----点击start，可成功在网页成功访问出界面

setclasspath.sh	2018/4/28 19:27	Shell Script	4 KB
shutdown.bat	2018/4/28 19:27	Windows 批处理...	2 KB
shutdown.sh	2018/4/28 19:27	Shell Script	2 KB
startup.bat	2018/4/28 19:27	Windows 批处理...	2 KB
startup.sh	2018/4/28 19:27	Shell Script	2 KB
tcnative-1.dll	2018/4/28 19:27	应用程序扩展	2,077 KB
tomcat8.exe	2018/4/28 19:27	应用程序	112 KB
tomcat8w.exe	2018/4/28 19:27	应用程序	116 KB
tomcat-juli.jar	2018/4/28 19:27	Executable Jar File	48 KB
tomcat-native.tar.gz	2018/4/28 19:27	WinRAR 压缩文件	396 KB
tool-wrapper.bat	2018/4/28 19:27	Windows 批处理...	5 KB
tool-wrapper.sh	2018/4/28 19:27	Shell Script	6 KB



- 点击 bin/tomcatw.exe里的stop-----再次点击bin/startup.bat ,可成功访问网页
(注：再次运行bin/startup.bat后命令行弹出来的仍然是下列输出信息，但实则已经解决成功)

```

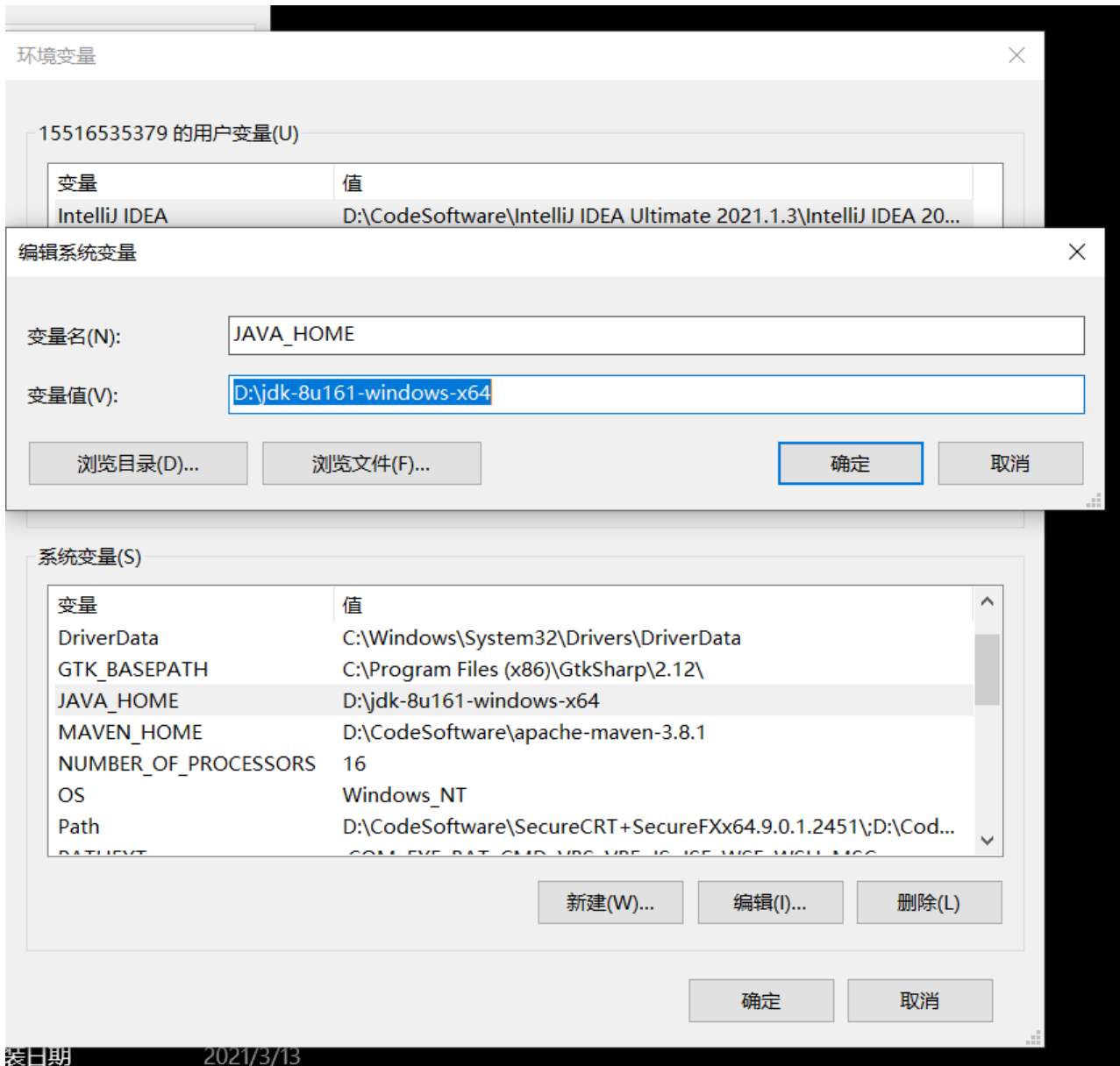
29-Nov-2021 09:26:04.453 信息 [localhost-startStop-1] org.apache.catalina.startup.HostConfig.deployDirectory Deployment
of web application directory D:\apache-tomcat-8.0.52-64\webapps\ROOT has finished in 46 ms
29-Nov-2021 09:26:04.464 信息 [main] org.apache.coyote.AbstractProtocol.start Starting ProtocolHandler ["http-apr-8080"]
29-Nov-2021 09:26:04.489 信息 [main] org.apache.coyote.AbstractProtocol.start Starting ProtocolHandler ["ajp-apr-8009"]
29-Nov-2021 09:26:04.503 信息 [main] org.apache.catalina.startup.Catalina.start Server startup in 19777 ms
29-Nov-2021 09:26:05.764 信息 [Abandoned connection cleanup thread] org.apache.catalina.loader.WebappClassLoaderBase.c
heckStateForResourceLoading Illegal access: this web application instance has been stopped already. Could not load []. The
following stack trace is thrown for debugging purposes as well as to attempt to terminate the thread which caused the i
llegal access.
java.lang.IllegalStateException: Illegal access: this web application instance has been stopped already. Could not load
[[]]. The following stack trace is thrown for debugging purposes as well as to attempt to terminate the thread which caus
ed the illegal access.
    at org.apache.catalina.loader.WebappClassLoaderBase.checkStateForResourceLoading(WebappClassLoaderBase.java:1372)
    at org.apache.catalina.loader.WebappClassLoaderBase.getResource(WebappClassLoaderBase.java:1042)
    at com.mysql.jdbc.AbandonedConnectionCleanupThread.checkContextClassLoaders(AbandonedConnectionCleanupThread.jav
a:90)
    at com.mysql.jdbc.AbandonedConnectionCleanupThread.run(AbandonedConnectionCleanupThread.java:63)
    at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1149)
    at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:624)
    at java.lang.Thread.run(Thread.java:748)

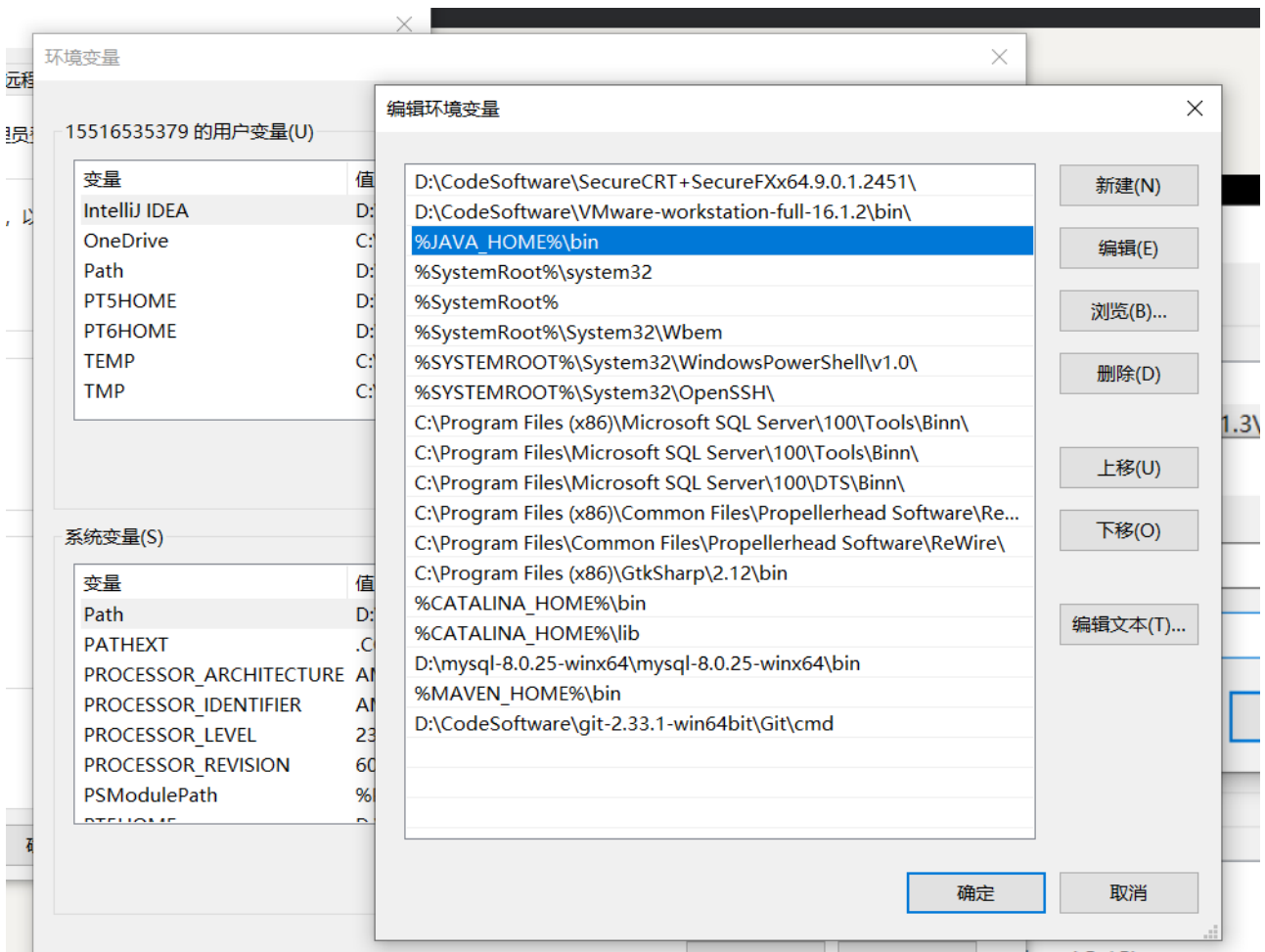
```

2.黑窗口一闪而过：即点击bin/start.bat后命令行黑窗口刚出现就立马消失

解决方法：

- 原因：没有正确配置JAVA_HOME环境变量（与JAVA中的配置一样），Tomcat是由java写的，Tomcat需要依赖JDK; start.bat实则是一个dos命令，需要去找JAVA
- 正确配置：系统变量下点击新建-----添加如下方框内的内容；
再点击系统变量里的Path-----新建蓝色横线里的内容
成功配置后，再次运行bin/start.bat后可成功





3.启动时出现问题:

- 已经同打开了start.bat,但没有关闭,于此同时又启动了一次start.bat,然后第二次的start.bat启动窗口输出一会信息后窗口会自动关闭,但是第一次打开的start.bat窗口仍然会存在(注:localhost:8080仍然可以成功访问网页)
- 错误信息可以在logs/日期最新的catalina.log文件 里看:蓝色横线里说明"启动失败"与""端口号冲突

名称	修改日期	类型	大小
catalina.2021-11-28.log	2021/11/28 23:45	文本文档	25 KB
catalina.2021-11-29.log	2021/11/29 10:07	文本文档	128 KB
commons-daemon.2021-11-29.log	2021/11/29 9:11	文本文档	1 KB
host-manager.2021-11-28.log	2021/11/28 23:39	文本文档	0 KB
host-manager.2021-11-29.log	2021/11/29 0:31	文本文档	0 KB
localhost.2021-11-28.log	2021/11/28 23:45	文本文档	25 KB
localhost.2021-11-29.log	2021/11/29 10:07	文本文档	111 KB
localhost_access_log.2018-06-29.txt	2018/6/29 18:00	文本文档	1 KB
localhost_access_log.2021-11-28.txt	2021/11/28 23:39	文本文档	0 KB
localhost_access_log.2021-11-29.txt	2021/11/29 10:08	文本文档	6 KB
manager.2021-11-28.log	2021/11/28 23:39	文本文档	0 KB
manager.2021-11-29.log	2021/11/29 0:31	文本文档	0 KB
tomcat8-stderr.2021-11-29.log	2021/11/29 9:11	文本文档	13 KB
tomcat8-stdout.2021-11-29.log	2021/11/29 9:10	文本文档	1 KB


```

at org.apache.catalina.util.LifecycleBase.init(LifecycleBase.java:107)
at org.apache.catalina.core.StandardServer.initInternal(StandardServer.java:871)
at org.apache.catalina.util.LifecycleBase.init(LifecycleBase.java:102)
at org.apache.catalina.startup.Catalina.load(Catalina.java:592)
at org.apache.catalina.startup.Catalina.load(Catalina.java:615)
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
at java.lang.reflect.Method.invoke(Method.java:498)
at org.apache.catalina.startup.Bootstrap.load(Bootstrap.java:308)
at org.apache.catalina.startup.Bootstrap.main(Bootstrap.java:482)

```

```

29-Nov-2021 10:07:38.453 严重 [main] org.apache.catalina.core.StandardService.initInternal Failed to initialize connector [Connector[HTTP/1.1-8080]]
org.apache.catalina.LifecycleException: Failed to initialize component [Connector[HTTP/1.1-8080]]

```

```

at org.apache.catalina.util.LifecycleBase.init(LifecycleBase.java:107)
at org.apache.catalina.core.StandardService.initInternal(StandardService.java:568)
at org.apache.catalina.util.LifecycleBase.init(LifecycleBase.java:102)
at org.apache.catalina.core.StandardServer.initInternal(StandardServer.java:871)
at org.apache.catalina.util.LifecycleBase.init(LifecycleBase.java:102)
at org.apache.catalina.startup.Catalina.load(Catalina.java:592)
at org.apache.catalina.startup.Catalina.load(Catalina.java:615)
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)

```

```
java.lang.Object.run(Native Method)
```

```
com.mchange.v2.async.ThreadPoolAsynchronousRunner$PoolThread.run(ThreadPoolAsynchronousRunner.java:534)
```

```
29-Nov-2021 10:07:58.897 信息 [localhost-startStop-1] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web applica
```

```
29-Nov-2021 10:07:58.898 信息 [localhost-startStop-1] org.apache.catalina.startup.HostConfig.deployDirectory Deploying web application
```

```
29-Nov-2021 10:07:58.954 信息 [localhost-startStop-1] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web applica
```

```
29-Nov-2021 10:07:58.954 信息 [localhost-startStop-1] org.apache.catalina.startup.HostConfig.deployDirectory Deploying web application
```

```
29-Nov-2021 10:07:59.005 信息 [localhost-startStop-1] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web applica
```

```
29-Nov-2021 10:07:59.014 信息 [main] org.apache.catalina.startup.Catalina.start Server startup in 20538 ms
```

```
29-Nov-2021 10:07:59.016 严重 [main] org.apache.catalina.core.StandardServer.await StandardServer.await: create[localhost:8005]:
```

```
java.net.BindException: Address already in use: JVM_Bind
```

```

at java.net.DualStackPlainSocketImpl.bind0(Native Method)
at java.net.DualStackPlainSocketImpl.socketBind(DualStackPlainSocketImpl.java:106)
at java.net.AbstractPlainSocketImpl.bind(AbstractPlainSocketImpl.java:387)
at java.net.PlainSocketImpl.bind(PlainSocketImpl.java:190)
at java.net.ServerSocket.bind(ServerSocket.java:375)
at java.net.ServerSocket.<init>(ServerSocket.java:237)
at org.apache.catalina.core.StandardServer.await(StandardServer.java:438)
at org.apache.catalina.startup.Catalina.await(Catalina.java:729)
at org.apache.catalina.startup.Catalina.start(Catalina.java:675)
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
at java.lang.reflect.Method.invoke(Method.java:498)
at org.apache.catalina.startup.Bootstrap.start(Bootstrap.java:349)

```

解决方案:

- 找到占用8080端口号的进程把它干死,杀死该进程
- cmd -----netstat -ano ;然后找到8080对应的PID进程id

```

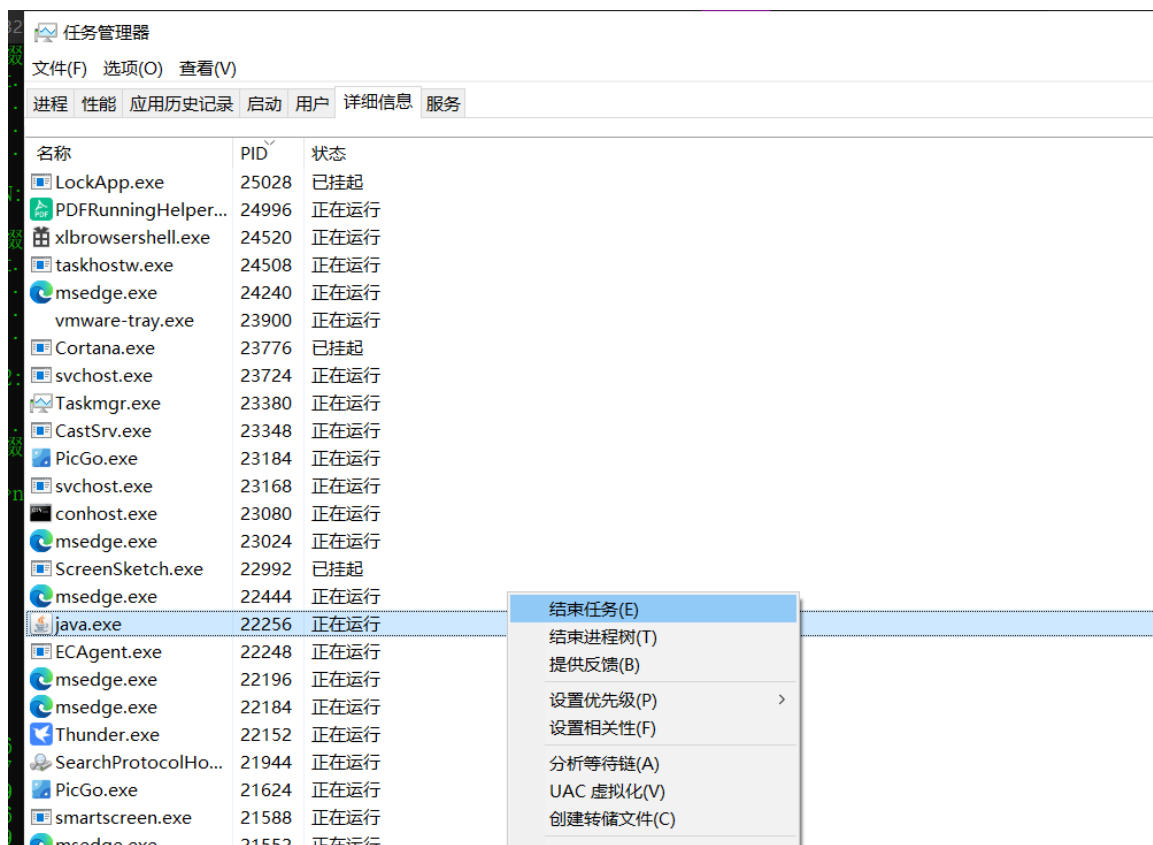
C:\Users\15516535379>netstat -ano

活动连接

 协议 本地地址           外部地址           状态           PID
TCP    0.0.0.0:135         0.0.0.0:0          LISTENING      1444
TCP    0.0.0.0:443         0.0.0.0:0          LISTENING      8068
TCP    0.0.0.0:445         0.0.0.0:0          LISTENING      4
TCP    0.0.0.0:554         0.0.0.0:0          LISTENING      5100
TCP    0.0.0.0:902         0.0.0.0:0          LISTENING      4276
TCP    0.0.0.0:912         0.0.0.0:0          LISTENING      4276
TCP    0.0.0.0:1696        0.0.0.0:0          LISTENING      17956
TCP    0.0.0.0:1797        0.0.0.0:0          LISTENING      11792
TCP    0.0.0.0:2869        0.0.0.0:0          LISTENING      4
TCP    0.0.0.0:3306        0.0.0.0:0          LISTENING      6200
TCP    0.0.0.0:3389        0.0.0.0:0          LISTENING      12732
TCP    0.0.0.0:5021        0.0.0.0:0          LISTENING      11632
TCP    0.0.0.0:5040        0.0.0.0:0          LISTENING      636
TCP    0.0.0.0:5357        0.0.0.0:0          LISTENING      4
TCP    0.0.0.0:8009        0.0.0.0:0          LISTENING      22256
TCP    0.0.0.0:8080        0.0.0.0:0          LISTENING      22256
TCP    0.0.0.0:8680        0.0.0.0:0          LISTENING      30440
TCP    0.0.0.0:10243       0.0.0.0:0          LISTENING      4
TCP    0.0.0.0:11200       0.0.0.0:0          LISTENING      5092

```

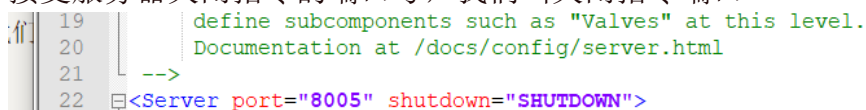
- 桌面底部----右键启动任务管理器-----点金详细信息---点击pid进行升序或者降序-
---右键点击结束任务



- 结束任务后8080端口就别杀死了,第一次打开的start.bat窗口也会立即关闭
- 再点击start.bat可以正常启动成功,localhost:8080也可以成功访问网页

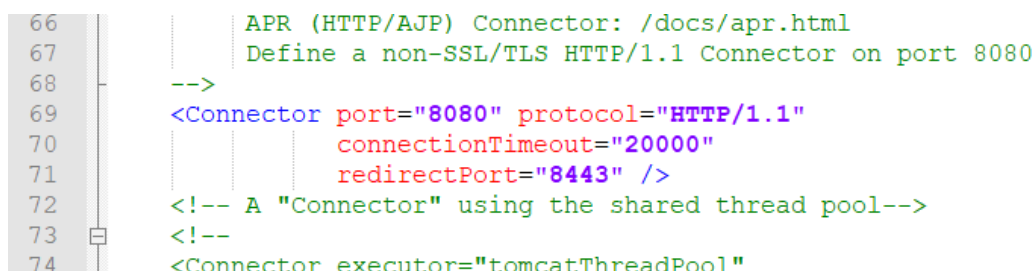
Tomcat端口介绍

- 在conf/server.xml里,以下图片都是tomcat默认时候设置的,没有做任何修改
- tomcat实际上就4个端口
- 接受服务器关闭指令的端口号,我们叫关闭指令端口



- port="8080"http:请求处理端口,我们在网页上输入的普通url地址包含的端口就是他.这个端口叫http端口;

redirectPort="https":请求的处理端口.这个端口叫https端口



- port="8009":接收AJP协议的处理端口.这个端口叫ajp端口;

redirectPort="https":请求的处理端口.这个端口叫https端口:https请求的处理端口

```
88 -->
89
90 <!-- Define an AJP 1.3 Connector on port 8009 -->
91 <Connector port="8009" protocol="AJP/1.3" redirectPort="8443" />
92
93
```

Tomcat端口修改

- 一般会把Tomcat端口号在conf/server.xml里把8080修改为80,原因:80端口号是http协议的默认端口号(修改时不能打开start.bat,需要在关闭状态下)
- 好处:在访问网页是不用输入端口号即可使用

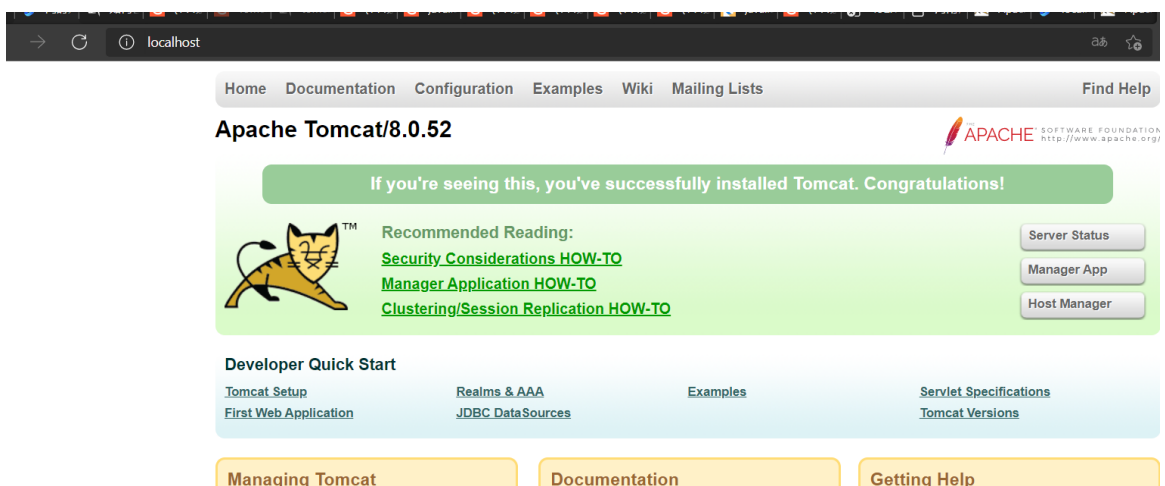
修改前:

```
66 APR (HTTP/AJP) Connector: /docs/apr.html
67 Define a non-SSL/TLS HTTP/1.1 Connector on port 8080
68 -->
69 <Connector port="8080" protocol="HTTP/1.1"
70 connectionTimeout="20000"
71 redirectPort="8443" />
72 <!-- A "Connector" using the shared thread pool-->
73 <!--
```

修改后:

```
65 Java AJP Connector: /docs/config/ajp.html
66 APR (HTTP/AJP) Connector: /docs/apr.html
67 Define a non-SSL/TLS HTTP/1.1 Connector on port 8080
68 -->
69 <Connector port="80" protocol="HTTP/1.1"
70 connectionTimeout="20000"
71 redirectPort="8443" />
72 <!-- A "Connector" using the shared thread pool-->
73 <!--
74 <Connector executor="tomcatThreadPool"
```

- 打开start.bat后在网页搜索localhost 或者 localhost: 或者 localhost:80 即可访问网页



- 市面上基本所有的http网站都是默认端口号都是80,比如搜索 www.itcast.cn 与 搜索 www.itcast.cn:80 是一样的
- 注:我的电脑的server.xml里依旧是8080,我没有去进行修改

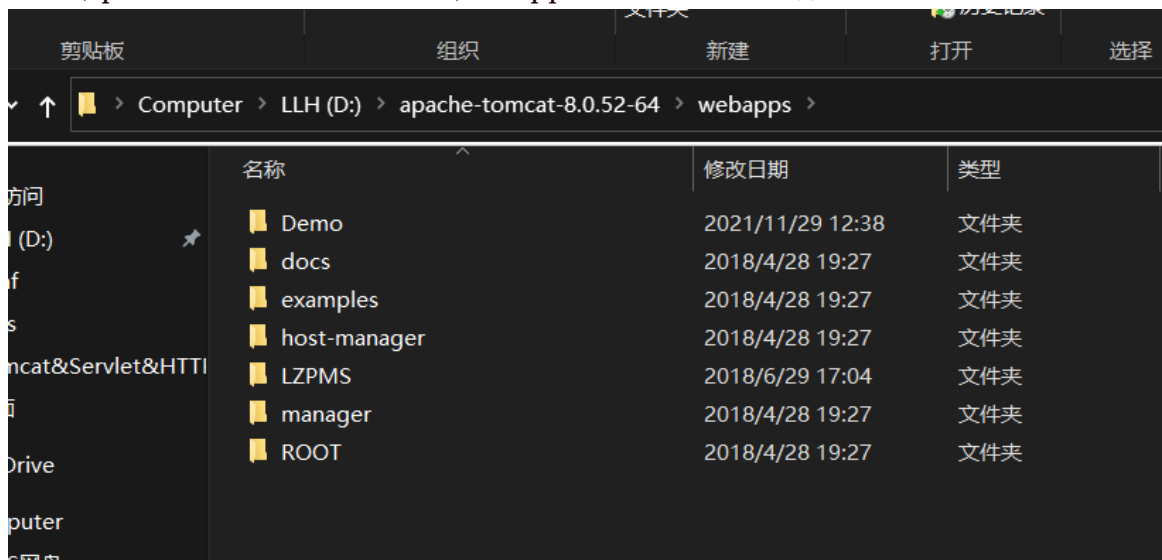
Tomcat关闭

- 正常关闭方法1:双击bin/shutdown.bat
- 正常关闭方法2:在start.bat启动窗口最后一行按ctrl+c 即可关闭
- 强制关闭:点击start.bat启动窗口的右上角的叉 (不推荐)
- 注:后期不会经常通过命令窗口的方式来启动Tomcat,而是在IDEA里进行启动与关闭

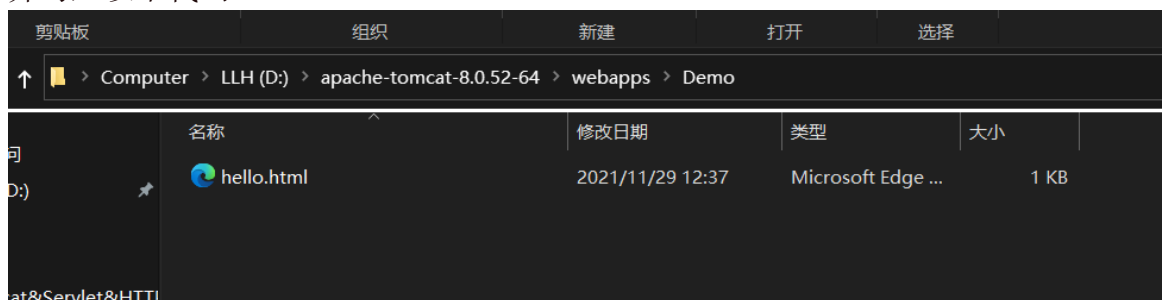
Tomcat部署项目

1.方式一:直接手工将项目放到webapps目录里

- 在D:\apache-tomcat-8.0.52-64\webapps里新建Demo文件夹

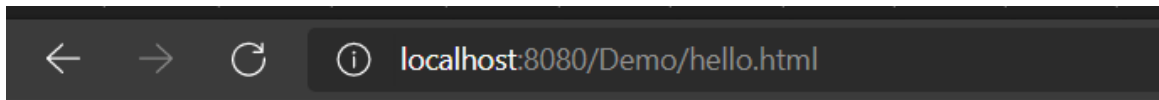


- 在D:\apache-tomcat-8.0.52-64\webapps\Demo文件夹里新建txt文本框,命名hello,并写入以下代码

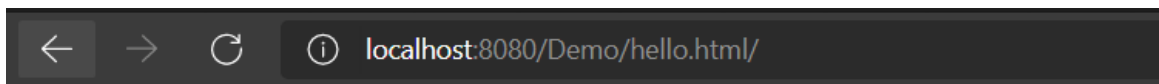


```
<font color='red'>
Hello,Tomcat
</font>
```

- 把hello.txt再重命名为hello.html
- 启动start.bat,网页搜索localhost:8080/Demo/hello.html
不要搜索localhost:8080/Demo/hello.html/ ,会搜不到的
不要搜索localhost:8080/Demo或者localhost:8080/Demo/ ,会搜不到的



Hello,Tomcat



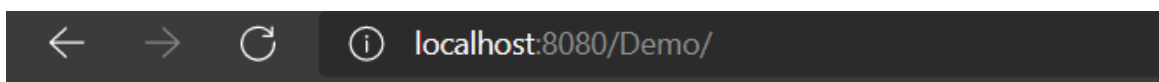
HTTP Status 404 - /Demo/hello.html/

type Status report

message /Demo/hello.html/

description The requested resource is not available.

Apache Tomcat/8.0.52



HTTP Status 404 - /Demo/

type Status report

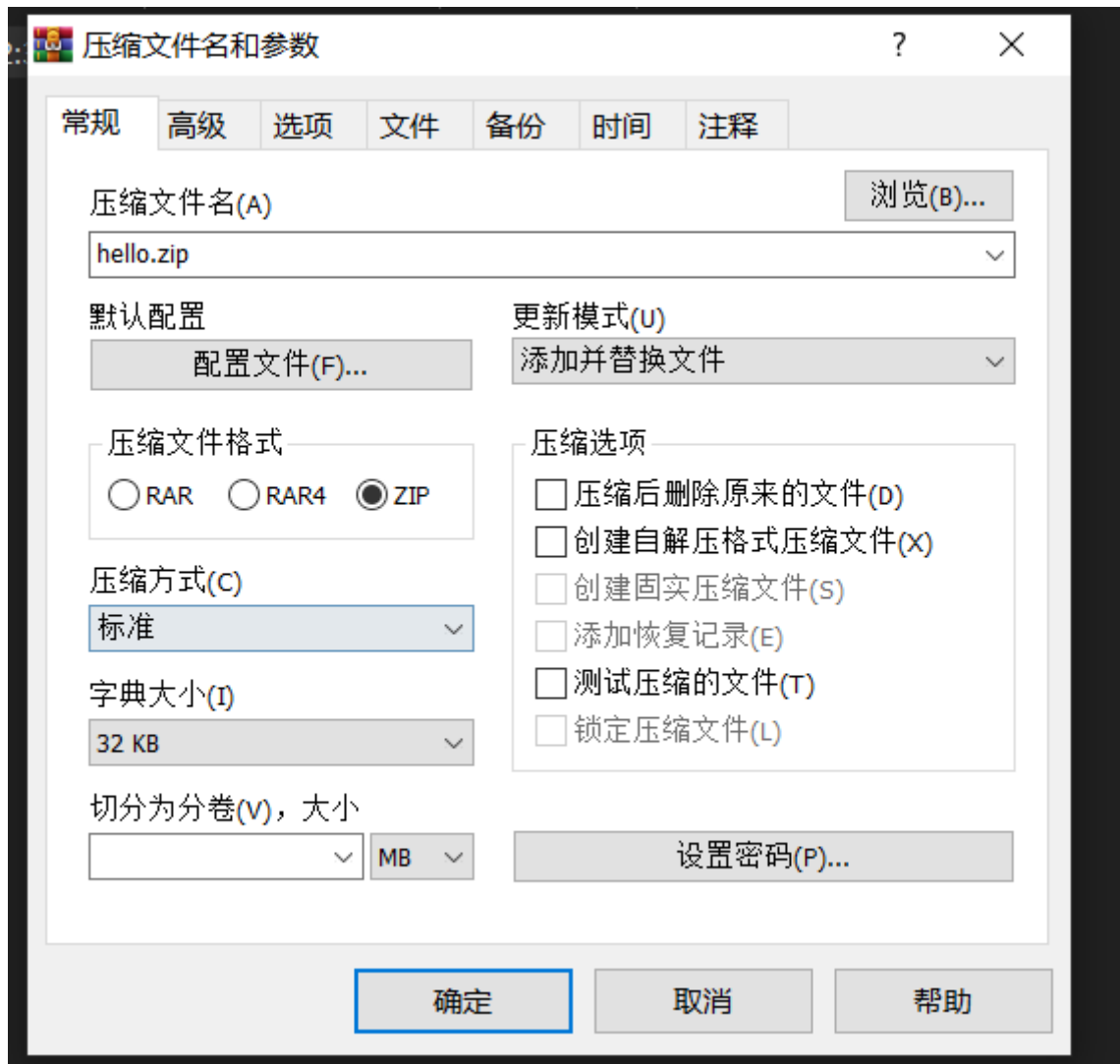
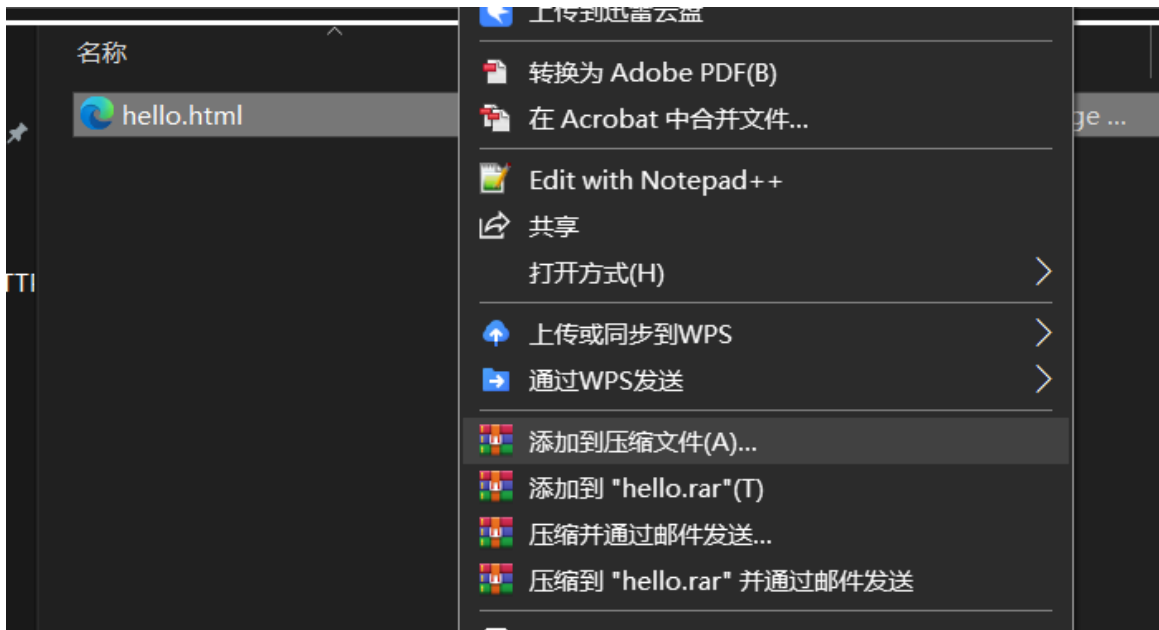
message /Demo/

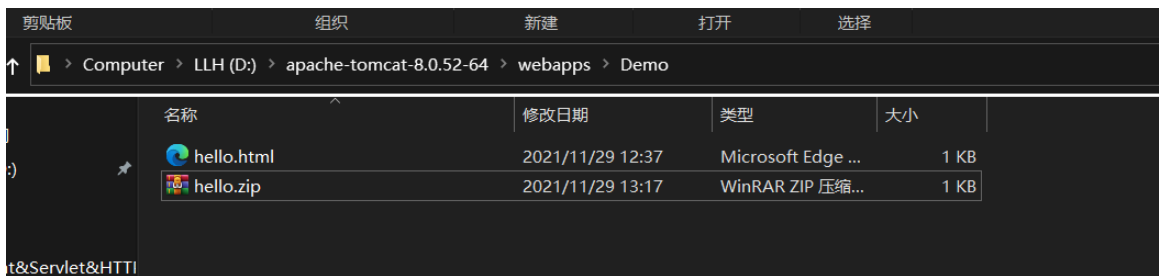
description The requested resource is not available.

Apache Tomcat/8.0.52

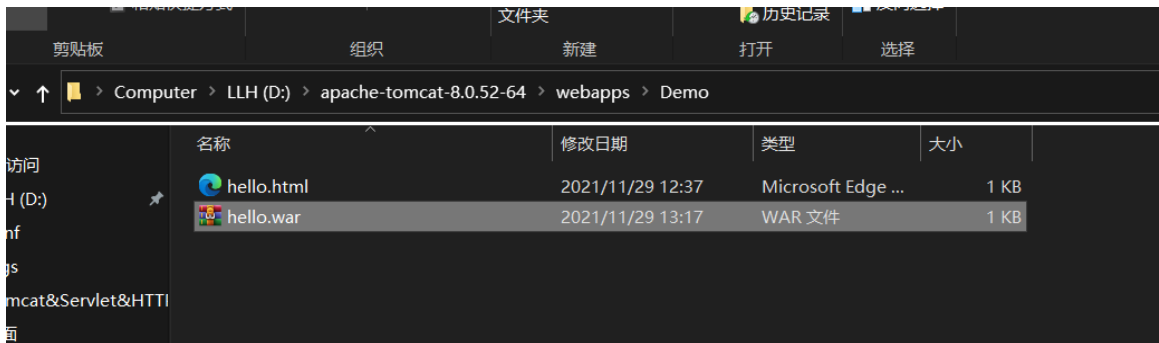
2.方式二:war包部署

- 在D:\apache-tomcat-8.0.52-64\webapps\Demo对hello.html右键进行zip打包



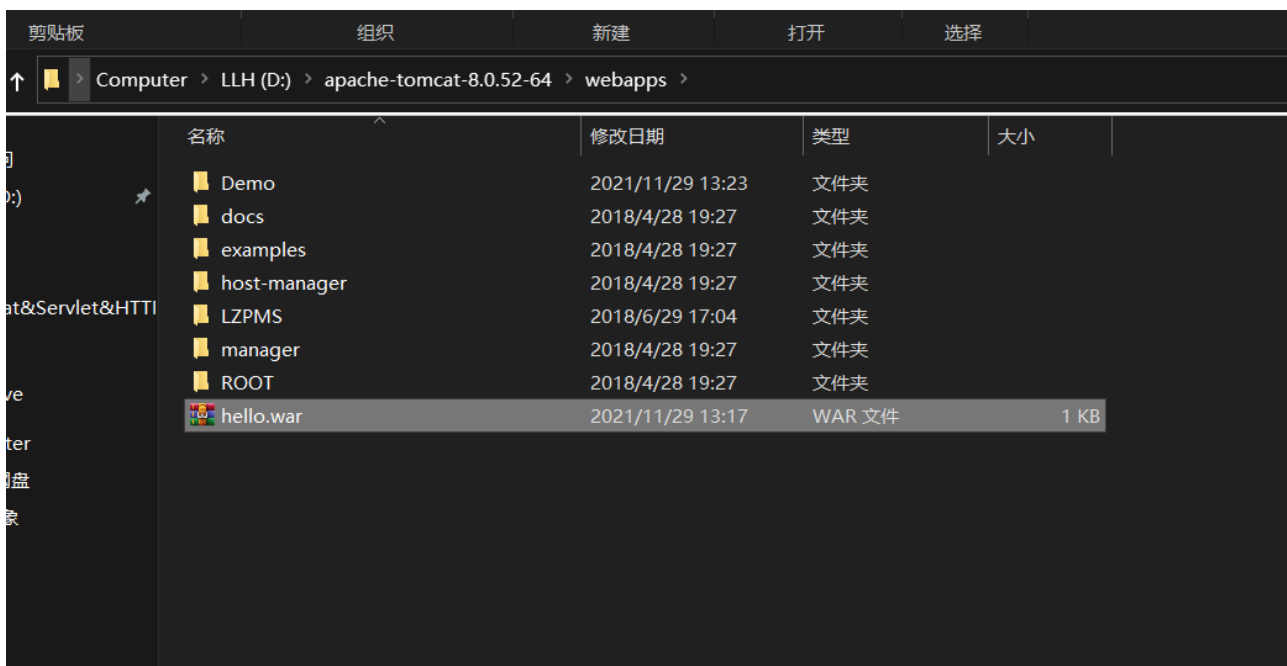


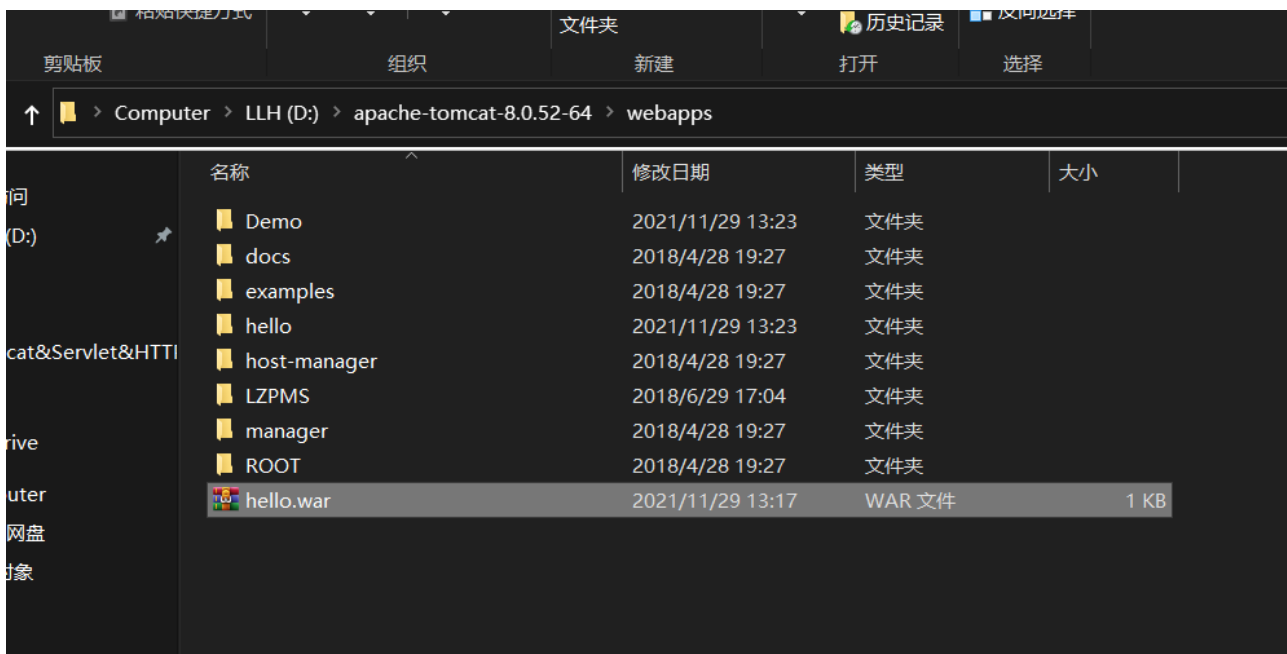
- 对"hello.zip" 重命名成 "hello.war"



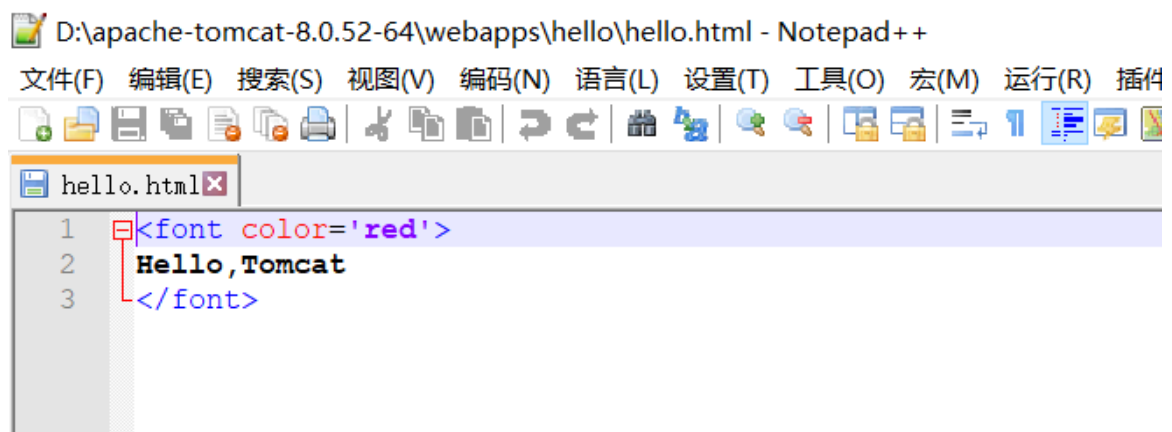
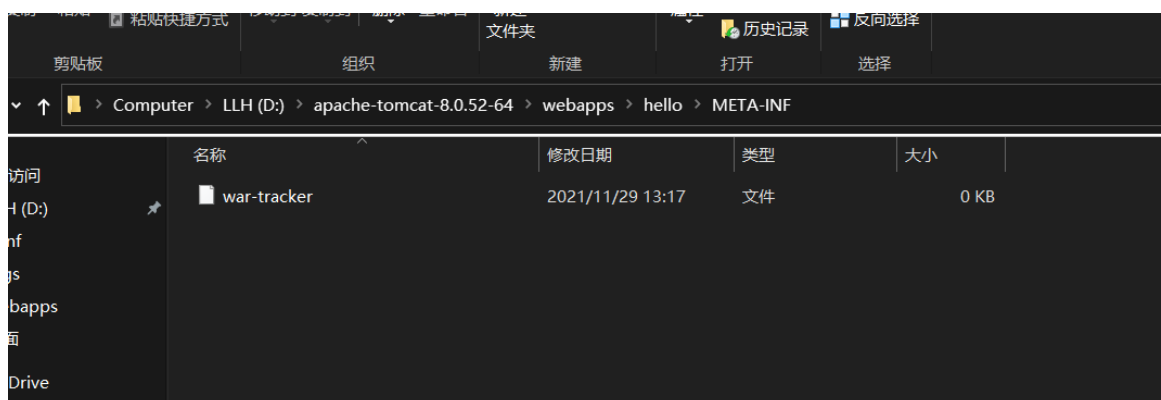
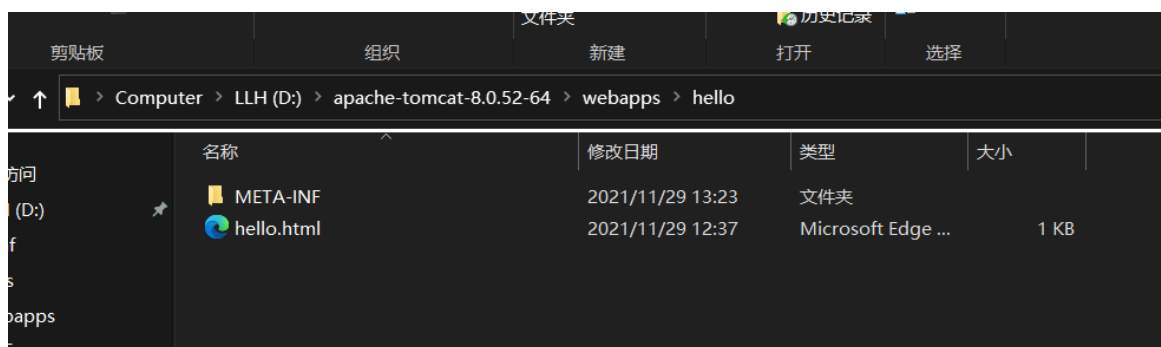
- 启动start.bat,把"hello.war"剪贴到D:\apache-tomcat-8.0.52-64\webapps里后,停一会后webapps里会自动生成一个hello文件夹;

注:自动生成必须在start.bat开启的状态下才能进行自动生成;若start.bat是关闭的,则不能进行自动生成

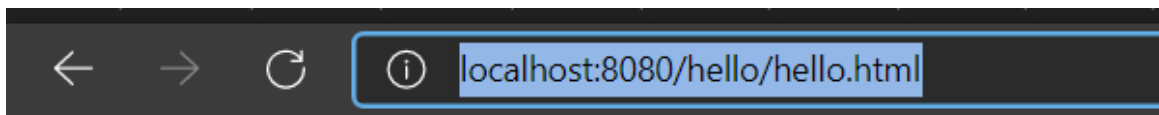




- 以下是自动生成的hello文件夹结构



- 网页输入 localhost:8080/hello/hello.html 可以进行访问



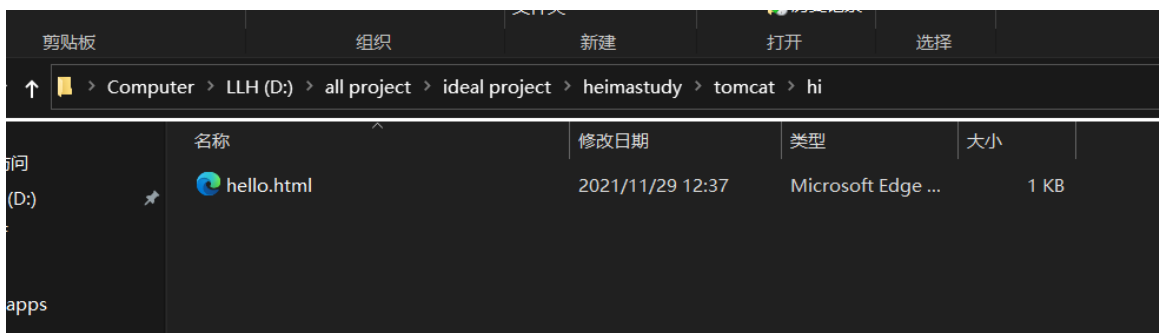
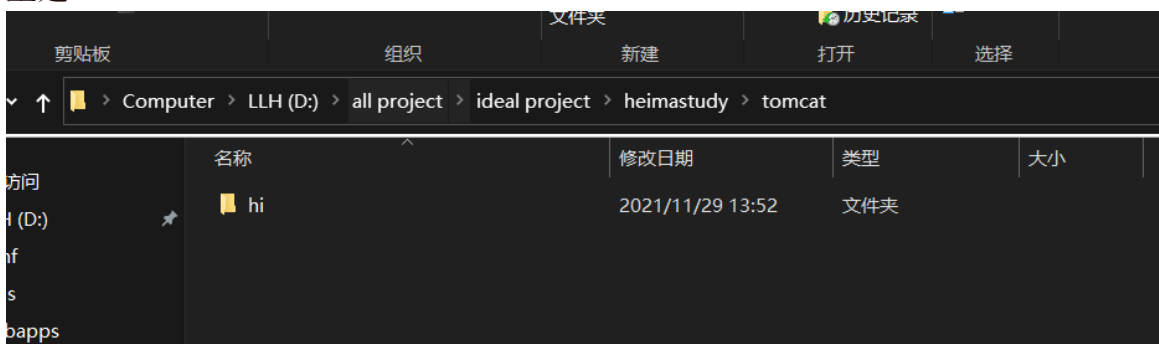
Hello,Tomcat

- 右键删除"hello.war"后,hello文件夹也会自动的进行删除

注:自动删除必须在start.bat开启的状态下才能进行自动删除;若start.bat是关闭的,则不能进行自动删除

3.方式三:配置文件虚拟目录配置

- 在D:\all project\ideal project\heimastudy\tomcat里新建文件夹"hi";在"hi"文件夹里建hello.html



- 进入conf/server.xml里在标签里添加以下代码(docBase:项目存放的目录;path:虚拟目录)

```
<Context docBase="D:\all project\ideal project\heimastudy\tomcat\hi" path="/hehe" />
```

添加前

```

121 </Realm>
122
123 <Host name="localhost" appBase="webapps"
124     unpackWARs="true" autoDeploy="true">
125
126     <!-- SingleSignOn valve, share authentication between web applications
127          Documentation at: /docs/config/valve.html -->
128     <!--
129     <Valve className="org.apache.catalina.authenticator.SingleSignOn" />
130     -->
131
132     <!-- Access log processes all example.
133          Documentation at: /docs/config/valve.html
134          Note: The pattern used is equivalent to using pattern="common" -->
135     <Valve className="org.apache.catalina.valves.AccessLogValve" directory="logs"
136          prefix="localhost_access_log" suffix=".txt"
137          pattern="%h %l %u %t "%r" %s %b" />
138
139     </Host>
140 </Engine>
141 </Service>
142 </Server>
143

```

添加后

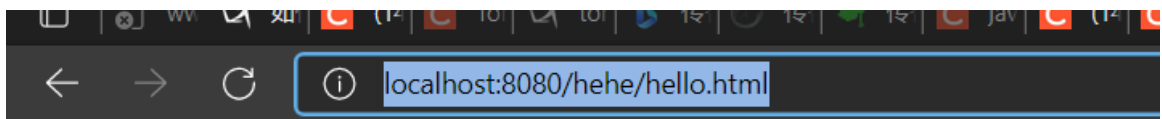
```

121 </Realm>
122
123 <Host name="localhost" appBase="webapps"
124     unpackWARs="true" autoDeploy="true">
125
126     <!-- SingleSignOn valve, share authentication between web applications
127          Documentation at: /docs/config/valve.html -->
128     <!--
129     <Valve className="org.apache.catalina.authenticator.SingleSignOn" />
130     -->
131
132     <!-- Access log processes all example.
133          Documentation at: /docs/config/valve.html
134          Note: The pattern used is equivalent to using pattern="common" -->
135     <Valve className="org.apache.catalina.valves.AccessLogValve" directory="logs"
136          prefix="localhost_access_log" suffix=".txt"
137          pattern="%h %l %u %t "%r" %s %b" />
138     <Context docBase="D:\all project\ideal project\heimastudy\tomcat\hi" path="/hehe" />
139
140 </Host>
141 </Engine>
142 </Service>
143 </Server>

```

- 启动start.bat,网页输入localhost:8080/hehe/hello.html会访问成功

注:这种方式webapps文件夹里是不会生成任何文件或文件夹的,而且也不会在D:\all project\ideal project\heimastudy\tomcat\hi里生成任何文件或文件夹

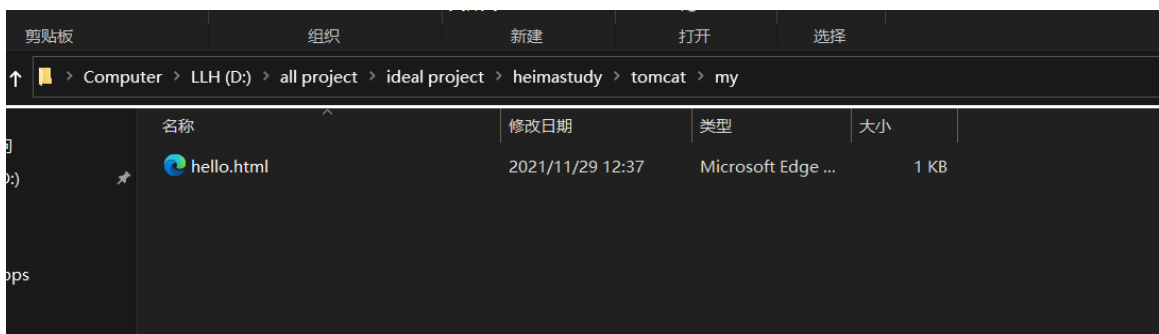


Hello,Tomcat

- 方式三的缺点:改动tomcat的文件会容易出问题,一个server.xml里可以虚拟部署很多个项目,容易出现问题.不建议使用(我的电脑的server.xml里已经取消此方式的使用)

4.方式四:

- 方式四相对于方式三安全了很多
- 在D:\all project\ideal project\heimastudy\tomcat里新建文件夹"my";在"my"文件夹里建hello.html

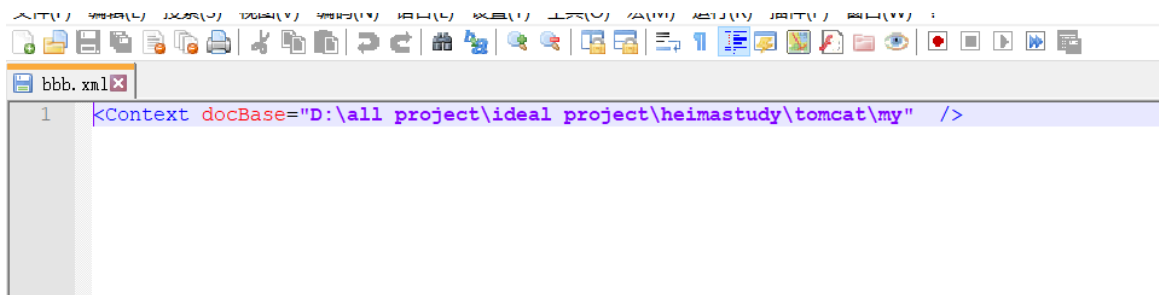


- 在D:\apache-tomcat-8.0.52-64\conf\Catalina\localhost里新建一个任意名称的txt文件,命名什么都可以,这里命名"bbb.txt",然后命名修改成"bbb.xml" (这个xml文件就是网页访问项目的虚拟名称)

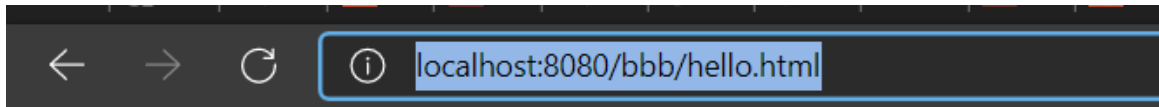


- 在"bbb.xml"里输入以下代码

```
<Context docBase="D:\all project\ideal project\heimastudy\tomcat\my" />
```



- 启动start.bat , 网页访问 localhost:8080/bbb/hello.html ,会成功



Hello,Tomcat

项目结构

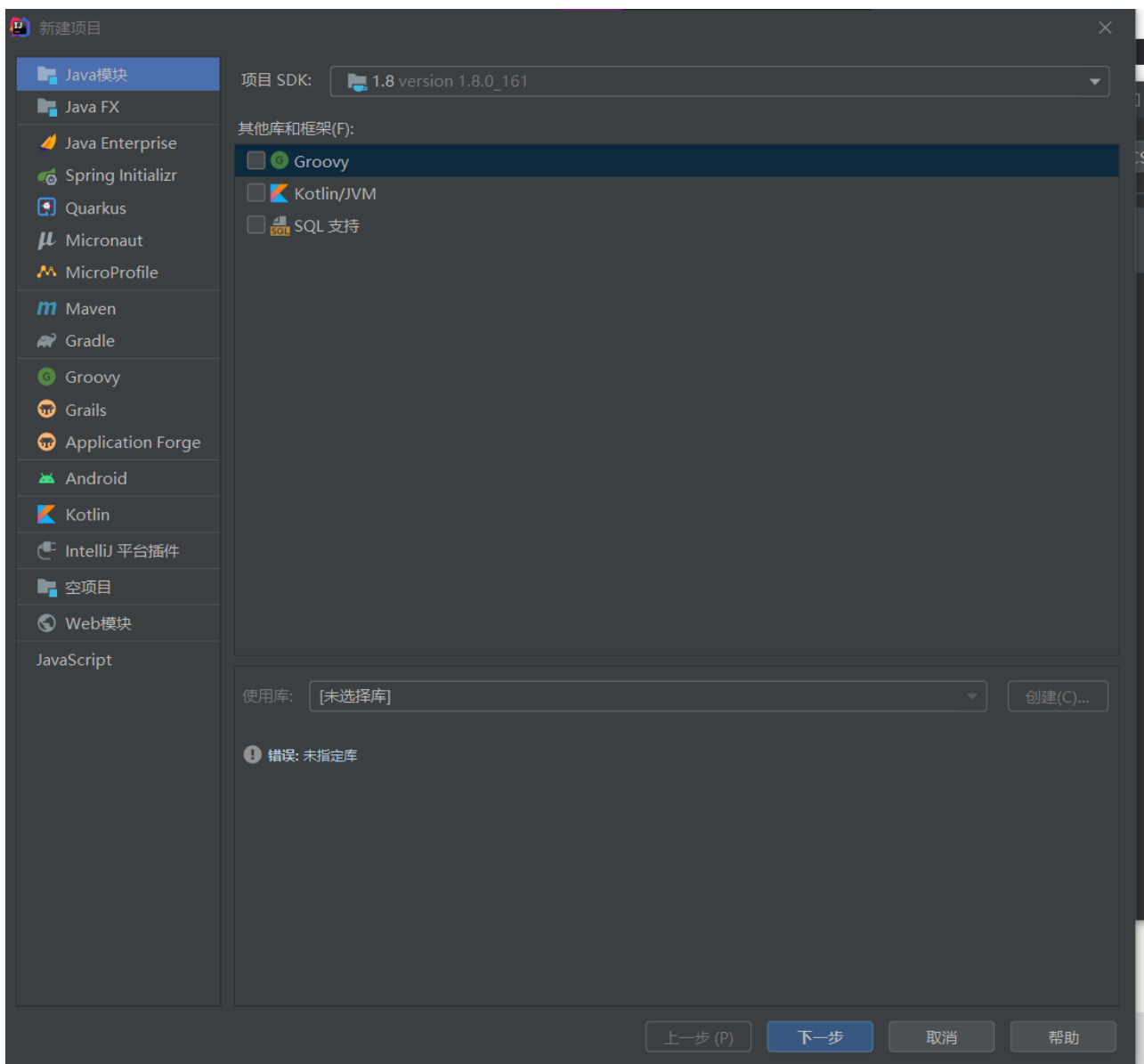
- 静态项目:这里面只能放静态资源,比如html\css\图片\文本\音频\视频等
- 动态项目:不仅可以放静态资源;还可以放动态资源,比如serve\jsp等(有WEB-INF目录的我们都称为动态项目)
- JAVA动态项目的目录结构:



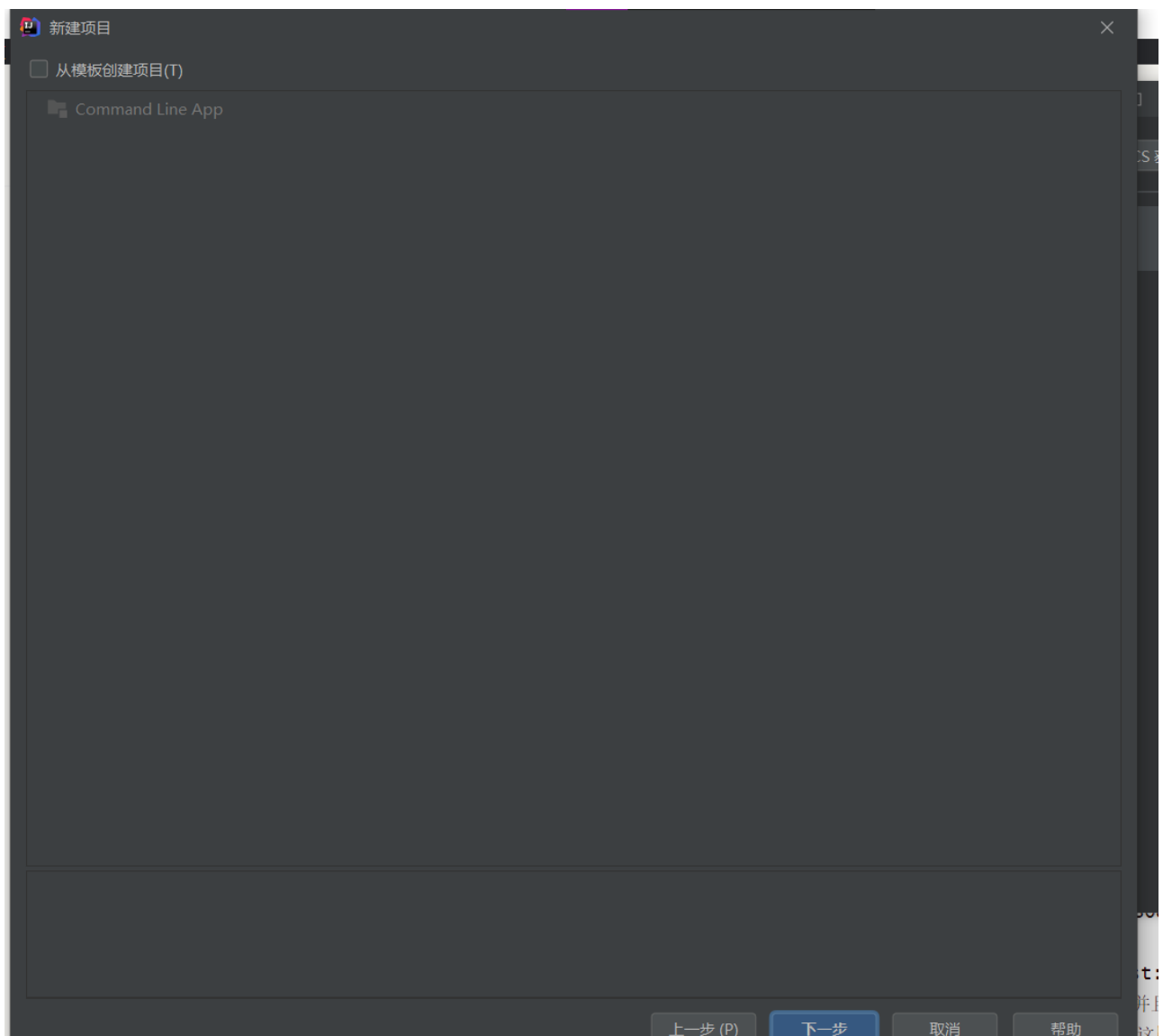
Tomcat集成IDEA

1.创建“Java模块”

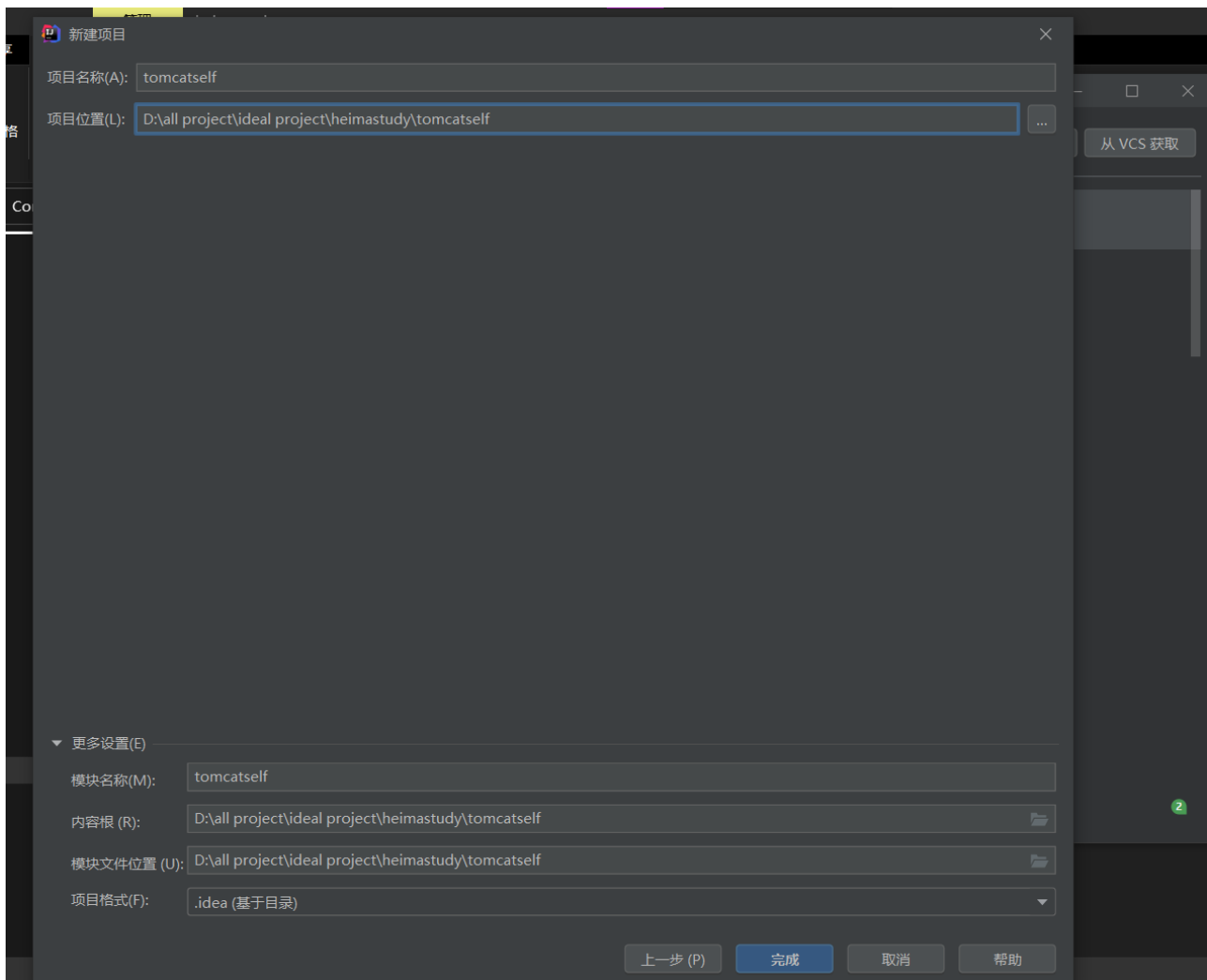
- 不用创建空项目, 点击“Java模块”直接创建



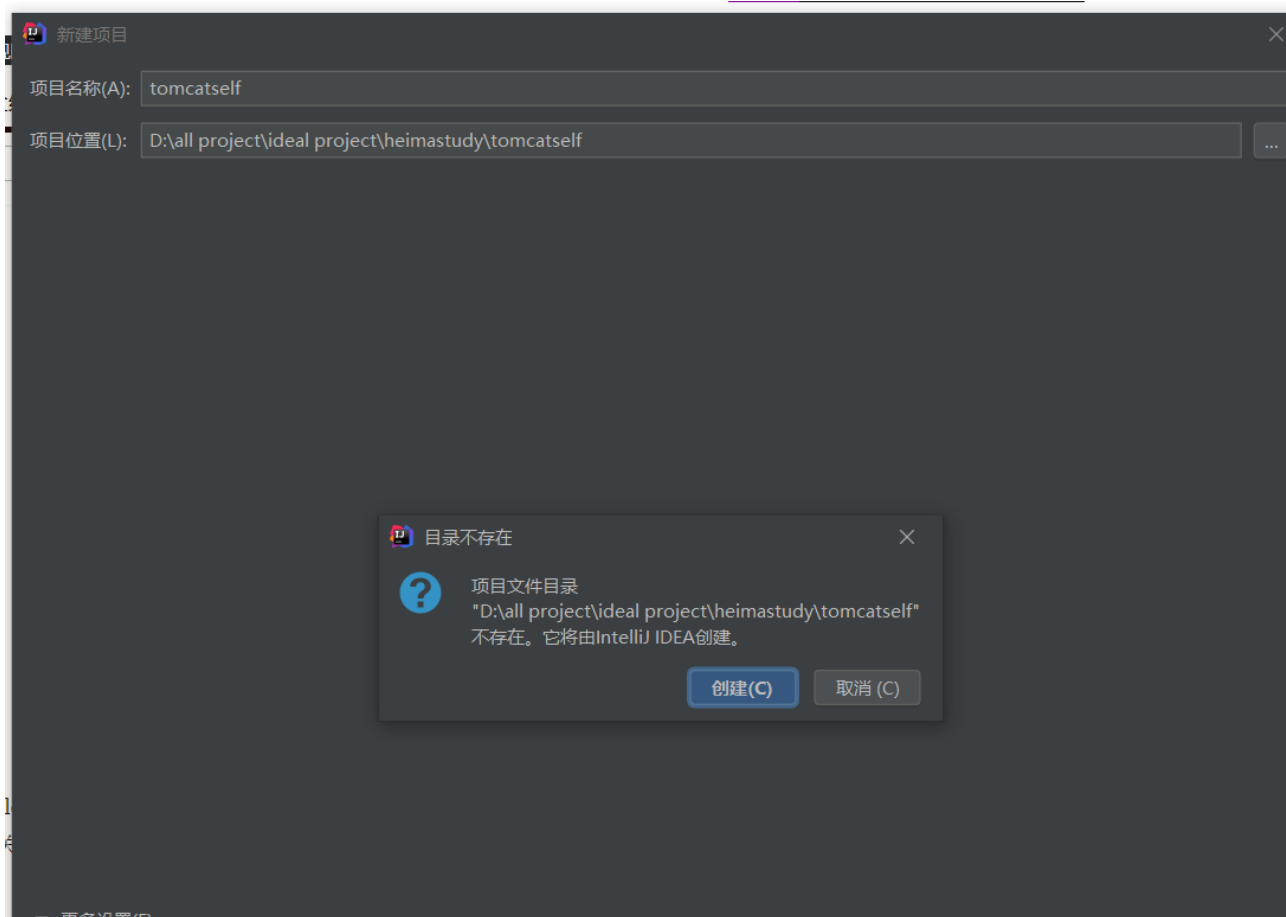
- 不点击从模板创建项目



- 如下命名

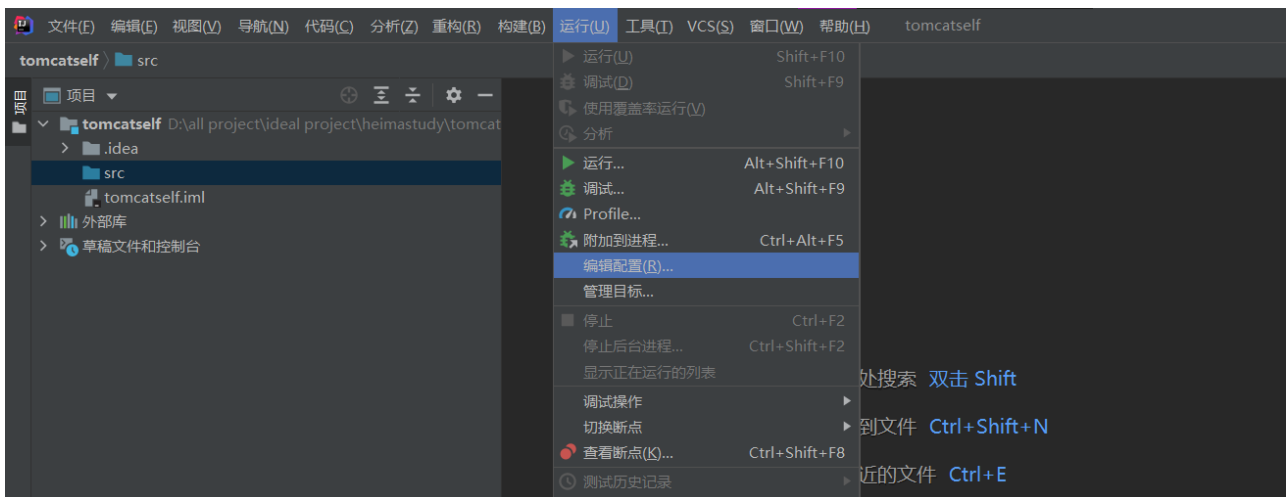


- 如下

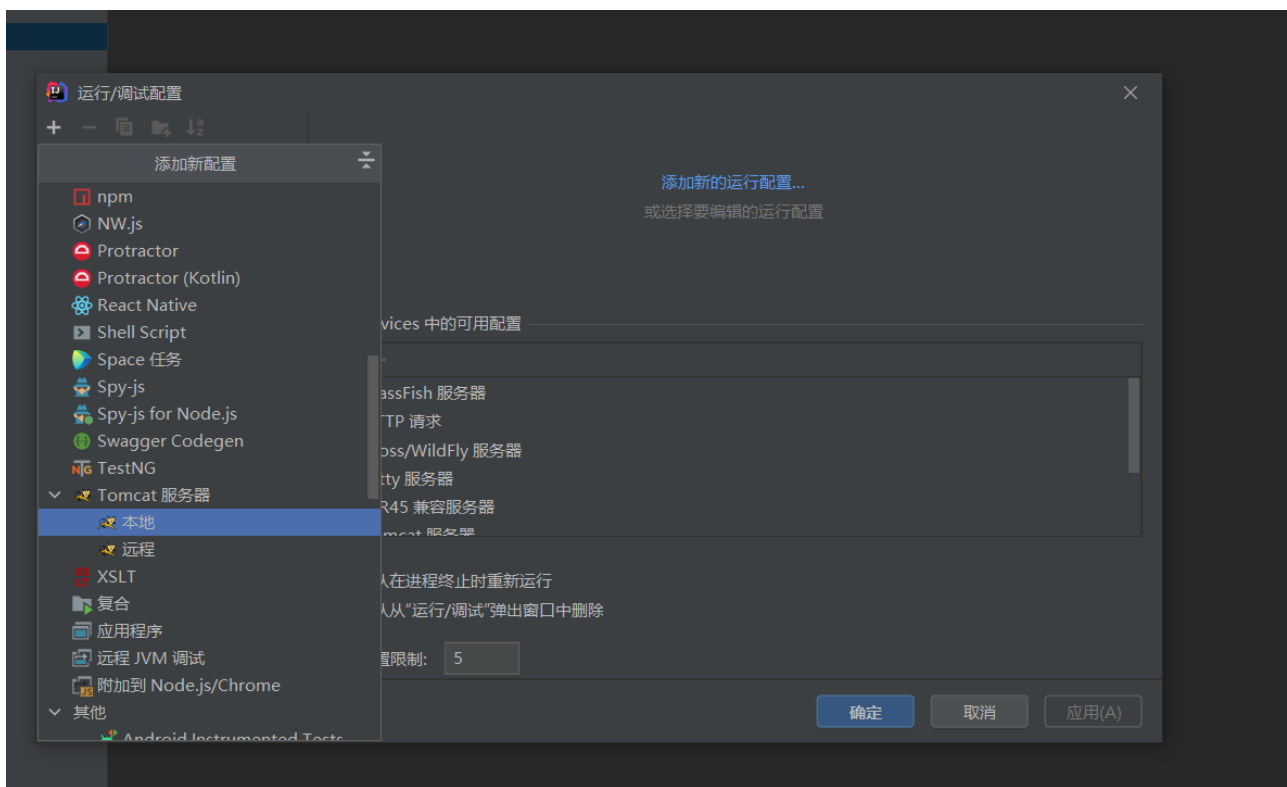


2.模块里配置Tomcat（接1.）

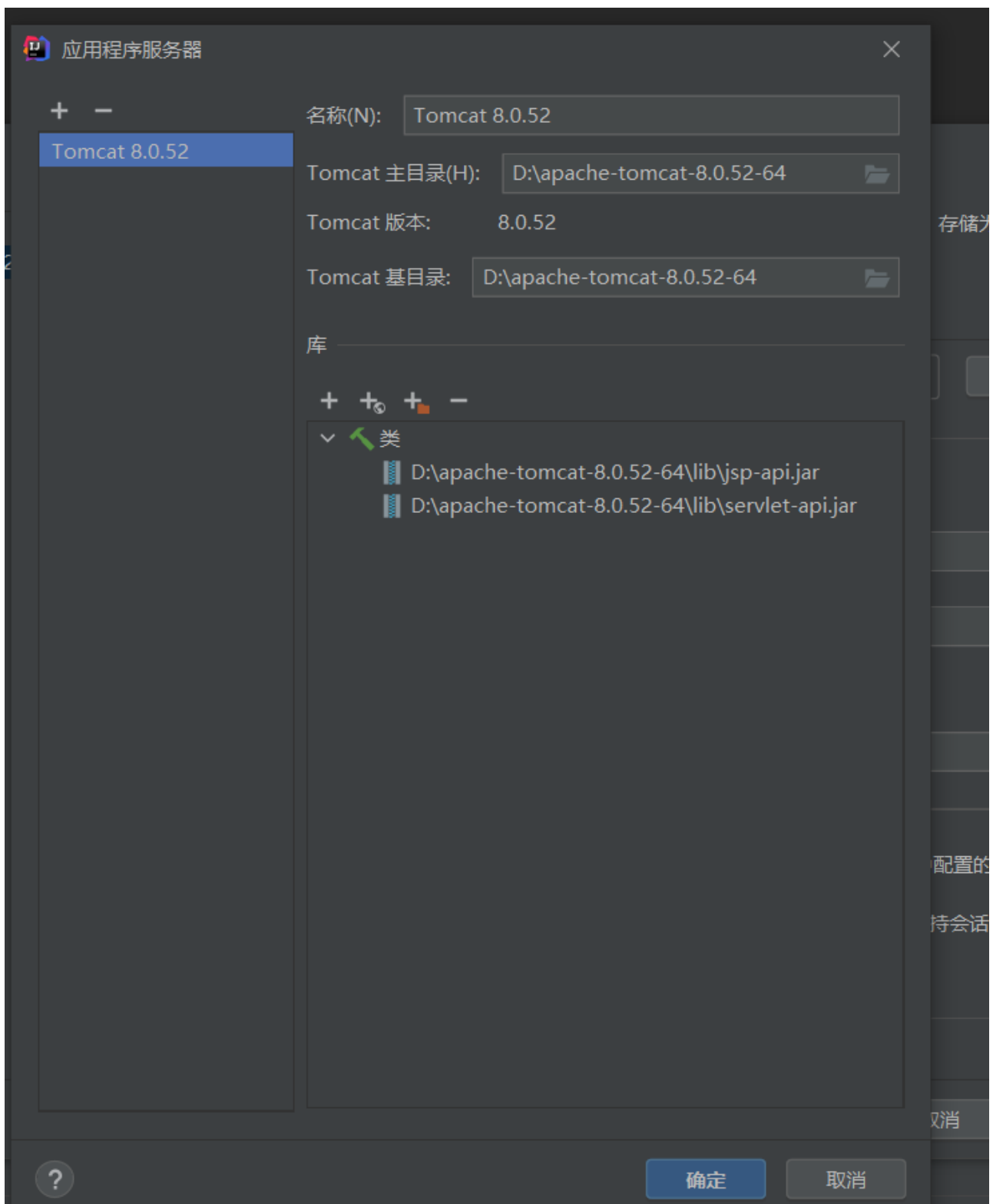
- 点击 运行----编辑配置



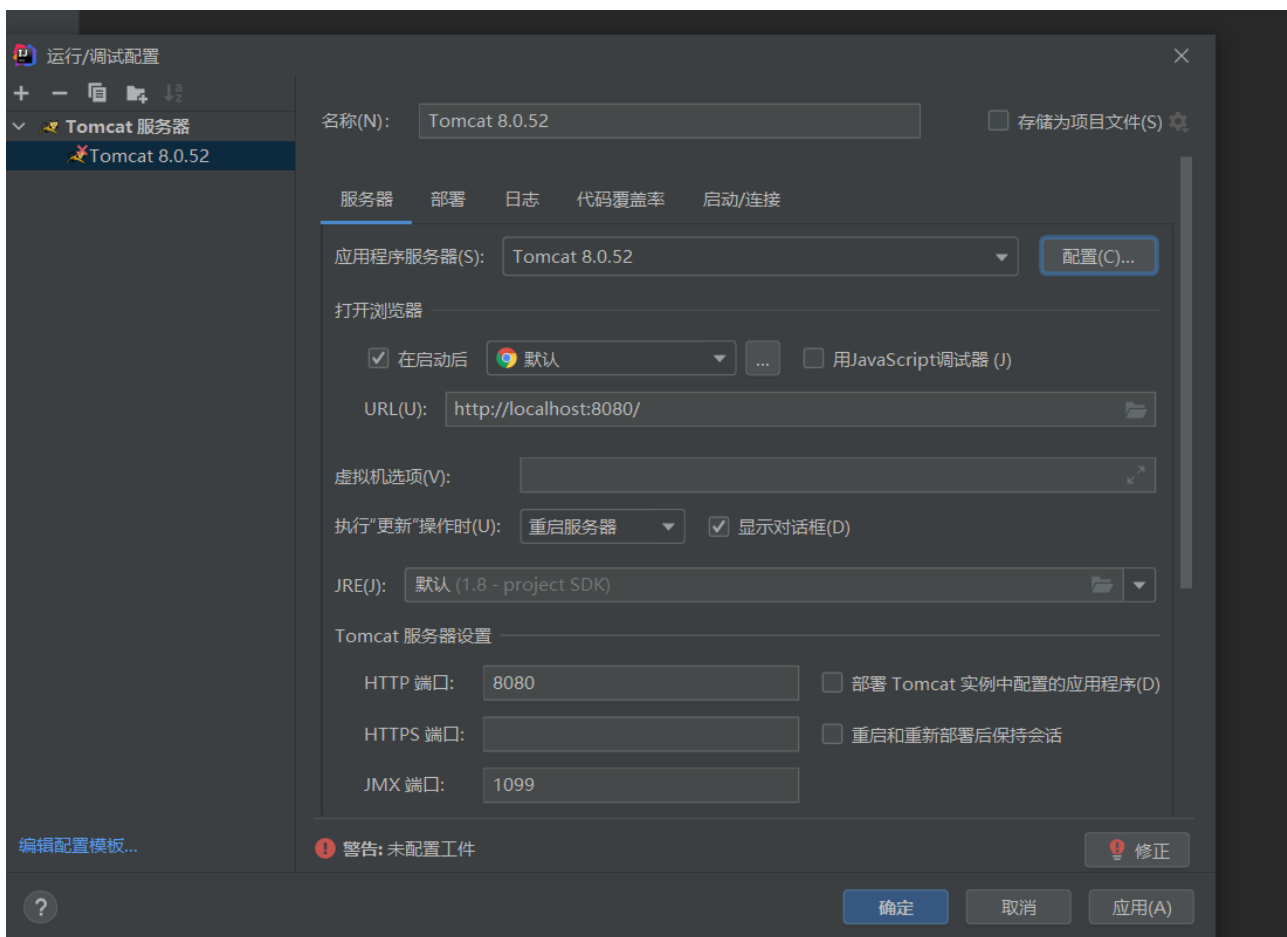
- 点击 +号----Tomcat服务器---本地 （注：不是Tomcat EE服务器）



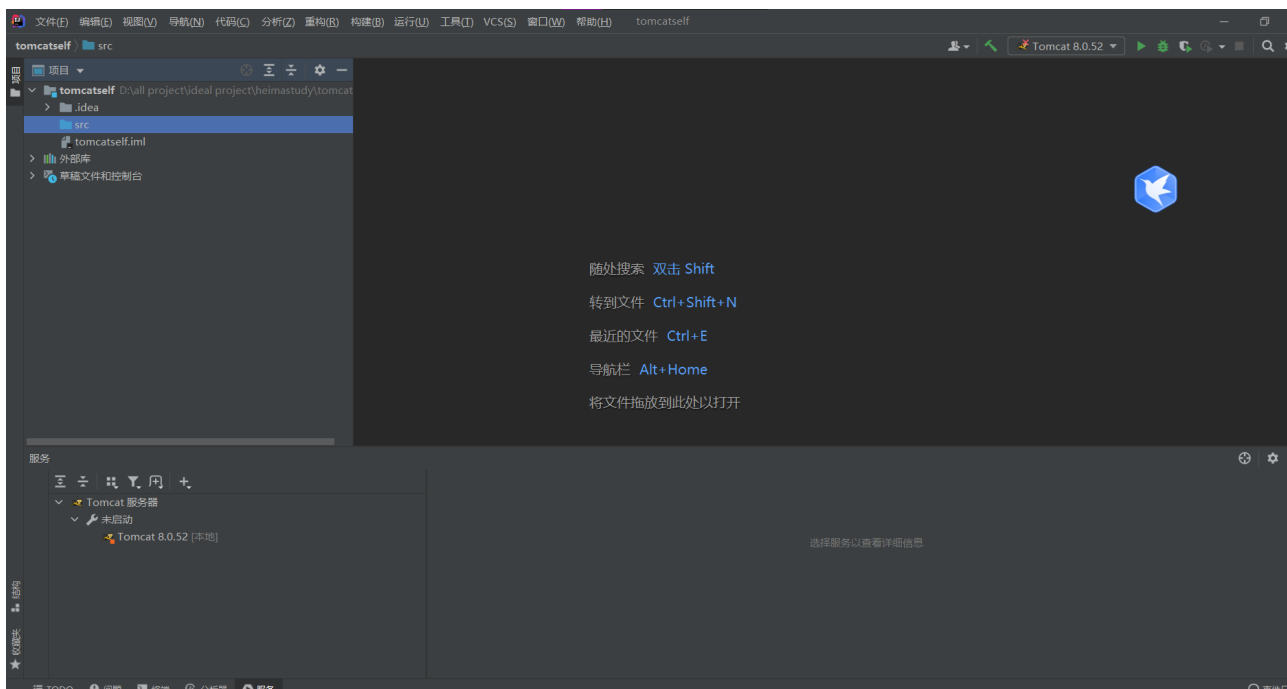
- 点击“应用程序服务器”后面的“配置”进行如下路径配置，配置完成后系统会自动加载两个jar包



- 配置完路径后，会出现如下默认界面，不用需改

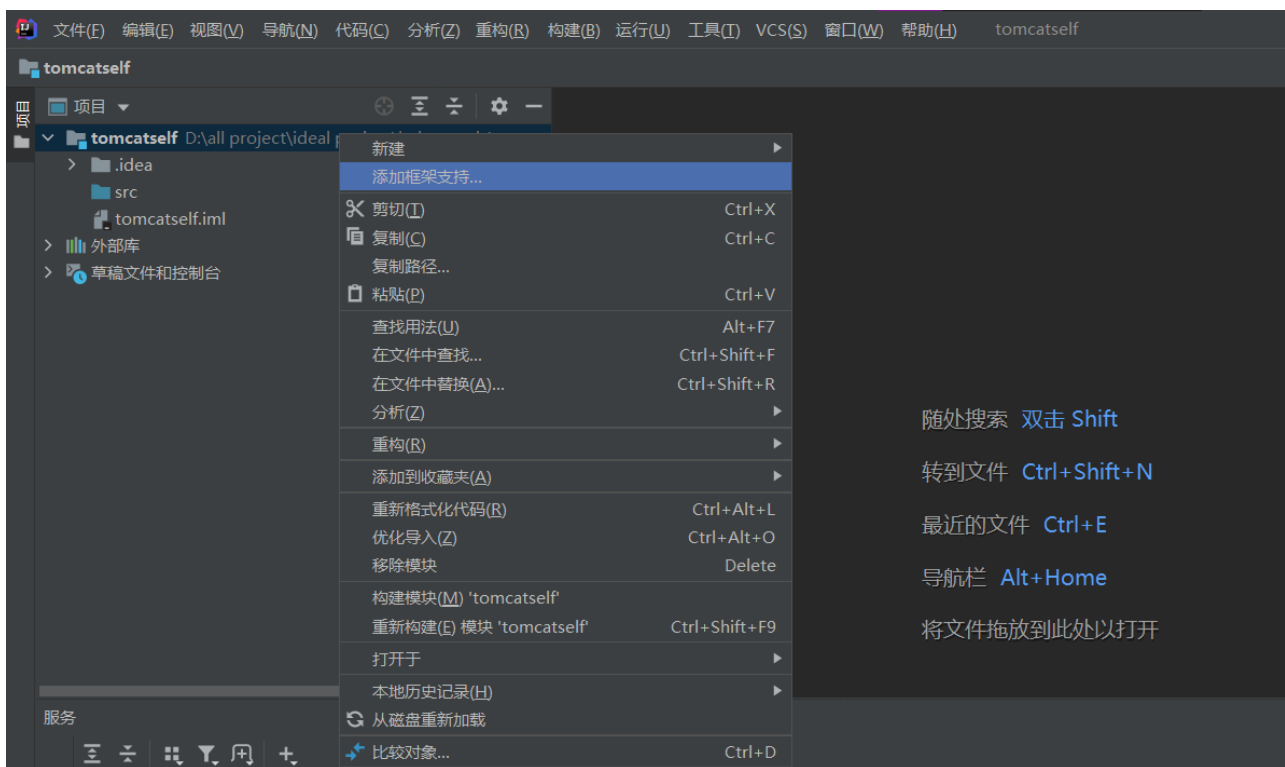


- 主界面左下角会弹出Tomcat窗口

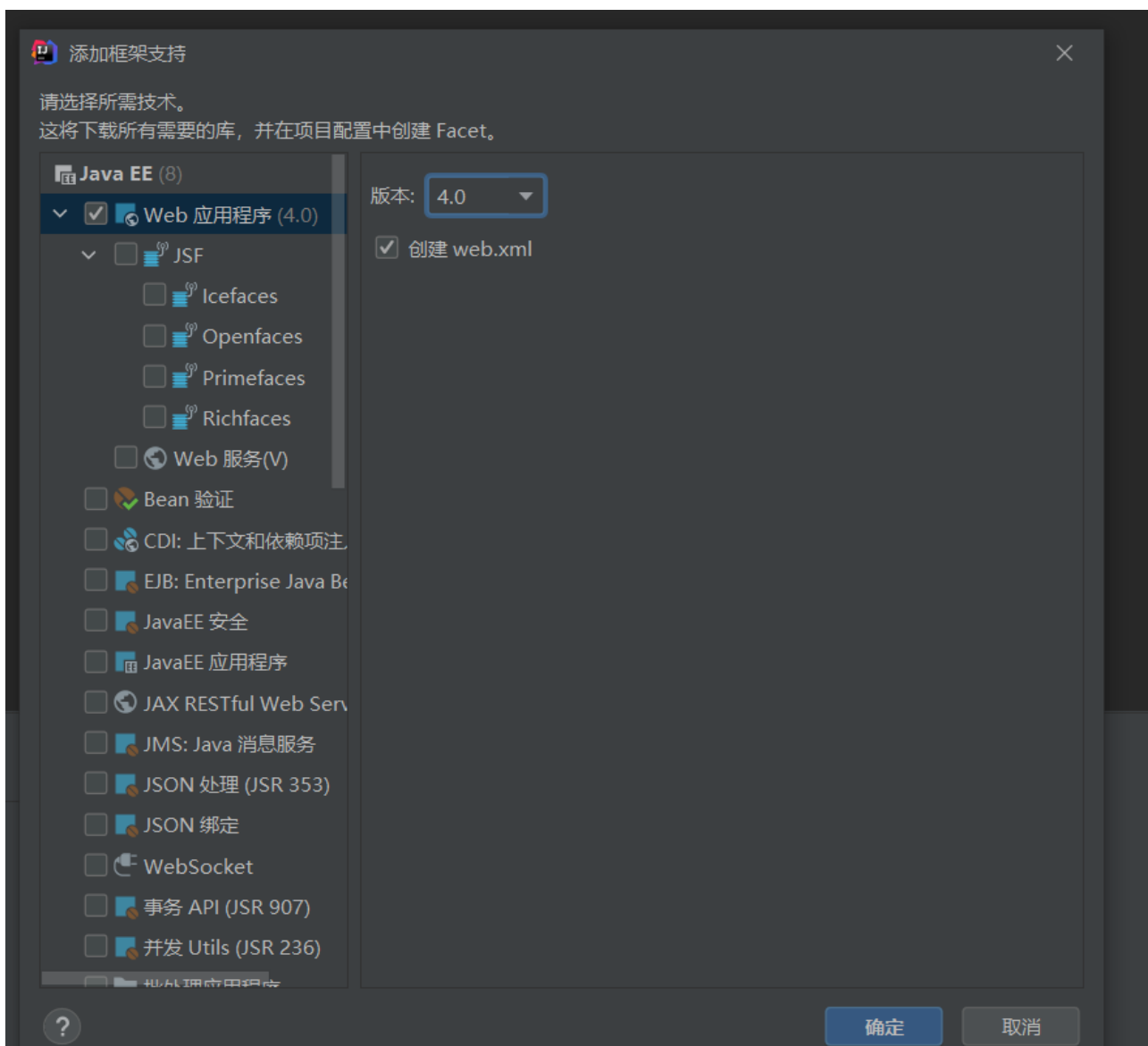


3.模块添加Web框架（接2.）

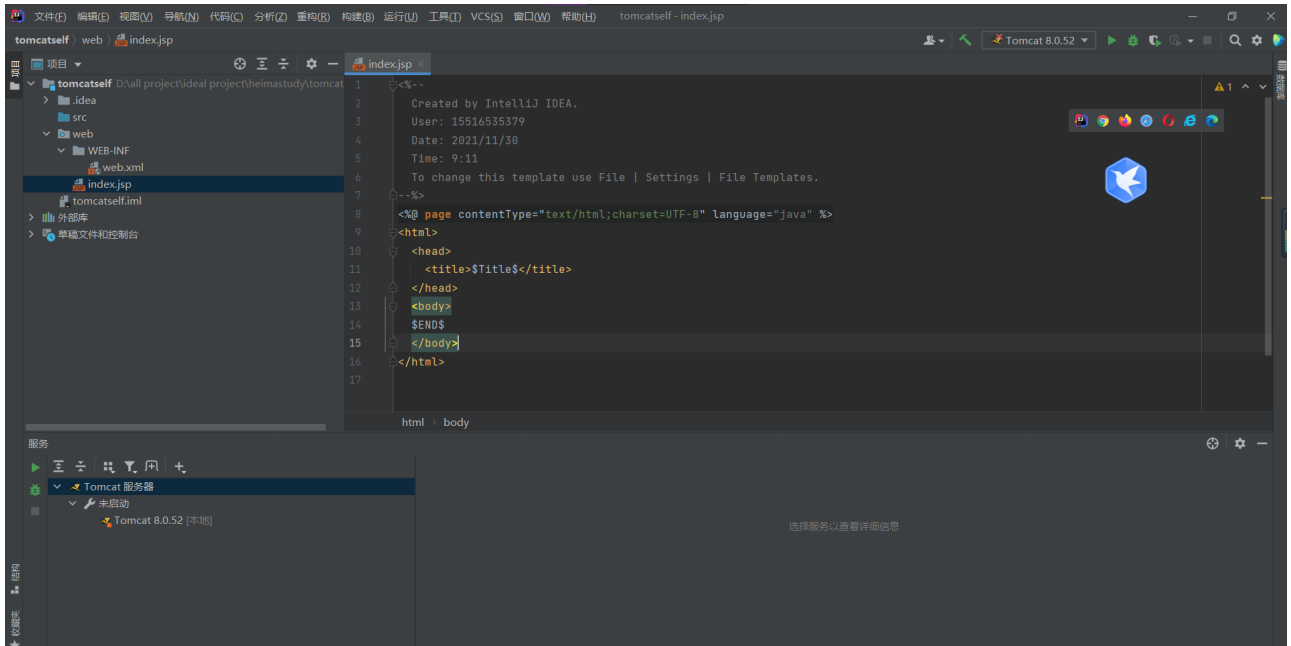
- 右键深蓝色行“tomcatself”----添加框架支持



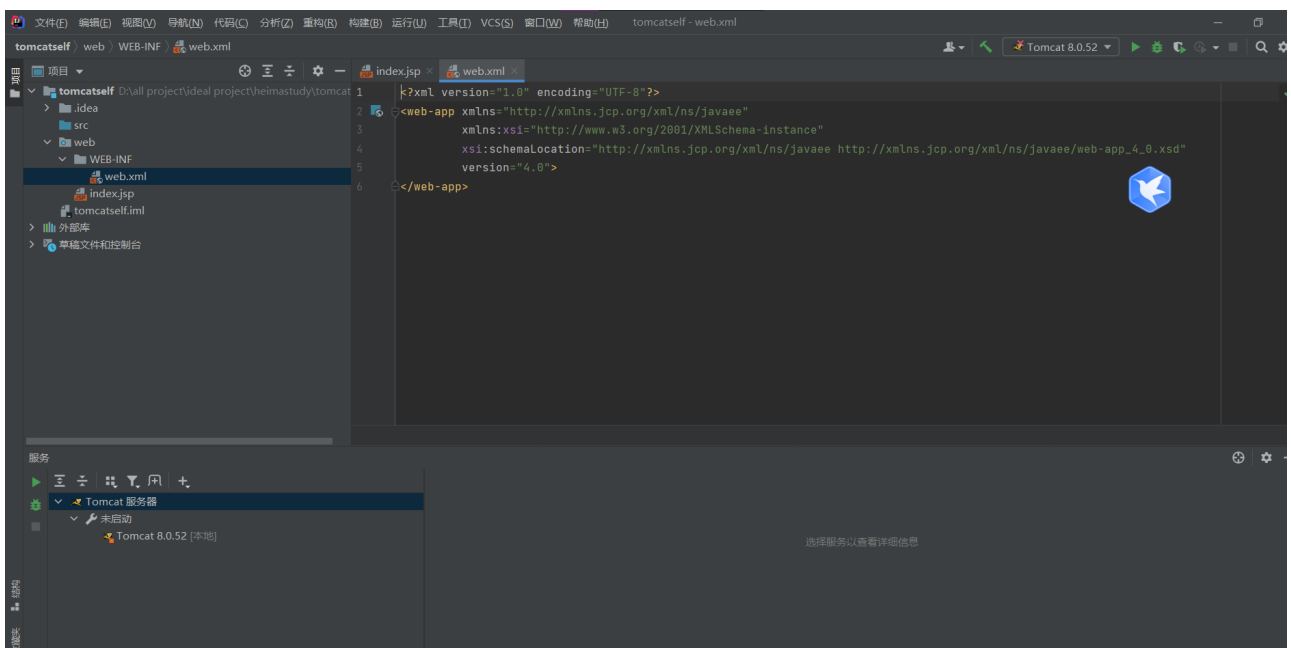
- 选择 "Web应用程序"



- 添加框架后，项目文件夹会如下变化；以及 如下为 index.jsp里面的初始默认内容

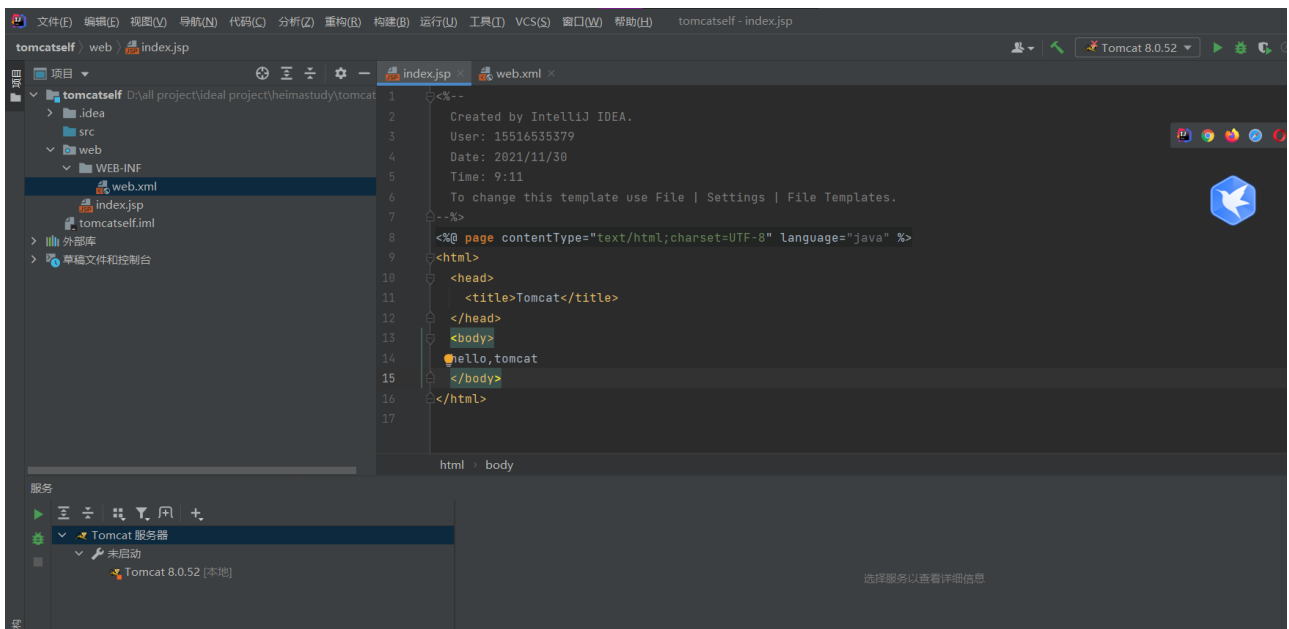


- 如下为 web.xml里面的初始默认内容

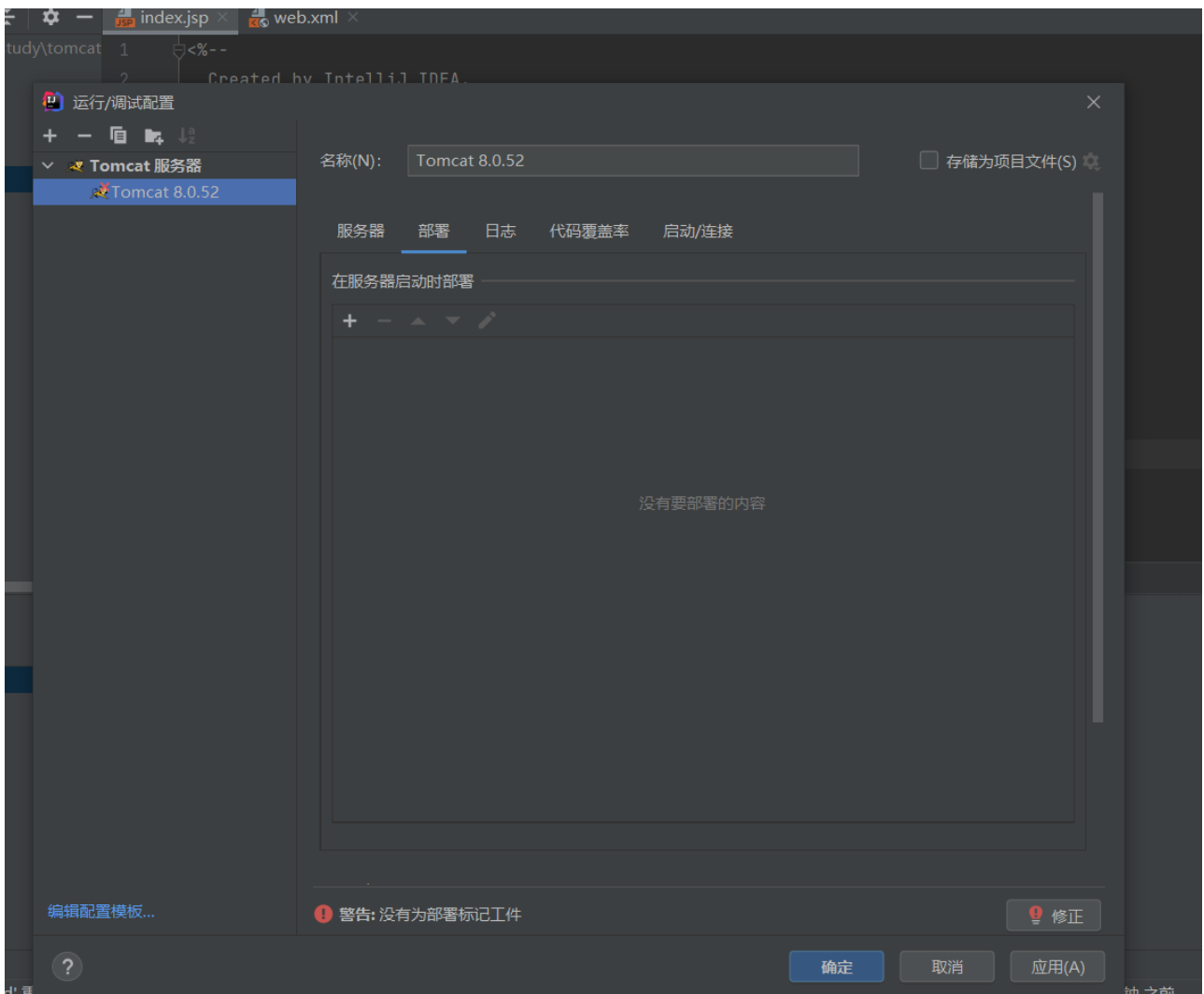


4.web与tomcat进行集合（接3.）

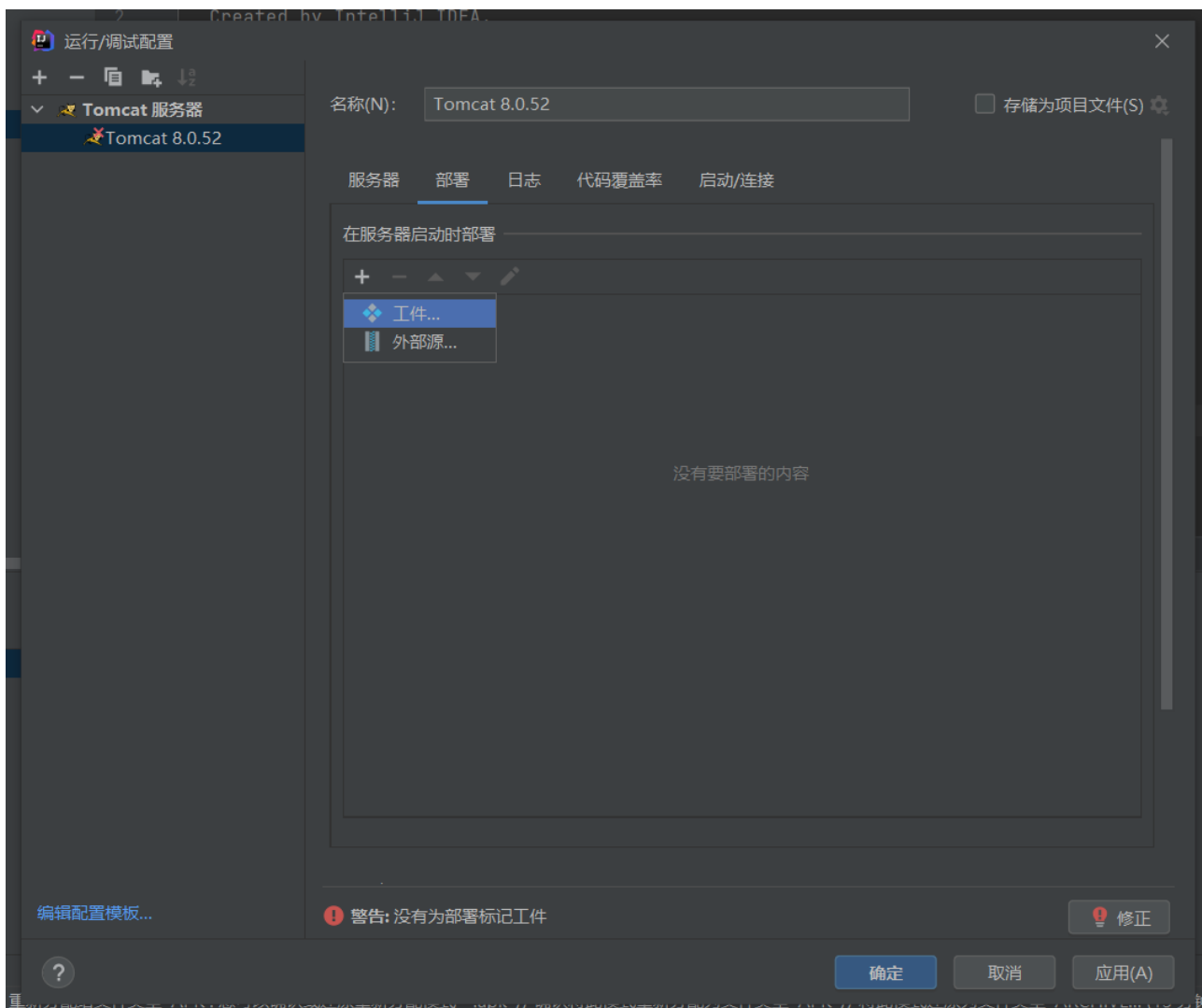
- 修改index.jsp后是如下界面



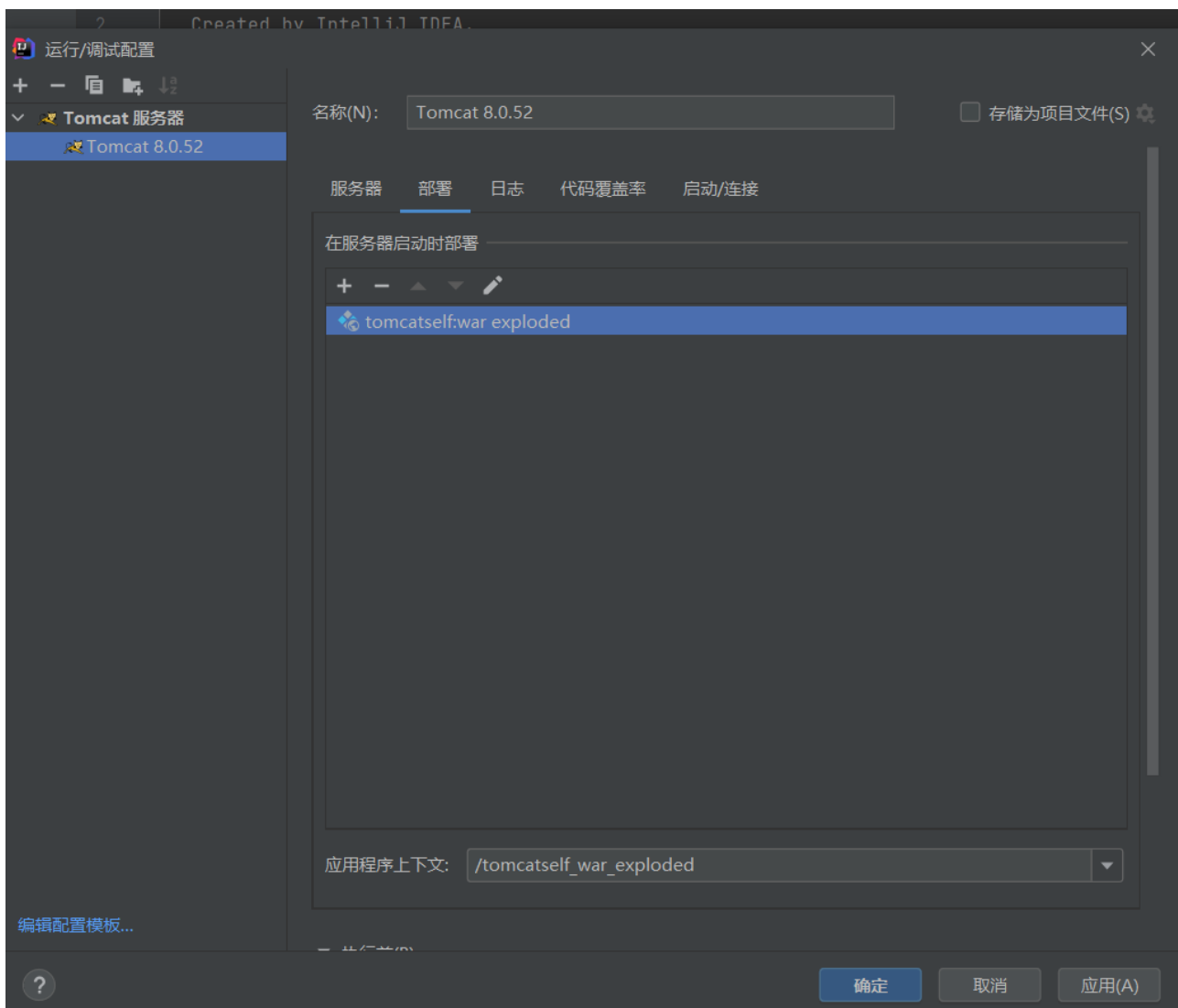
- 第一次点击部署时里面没有任何内容



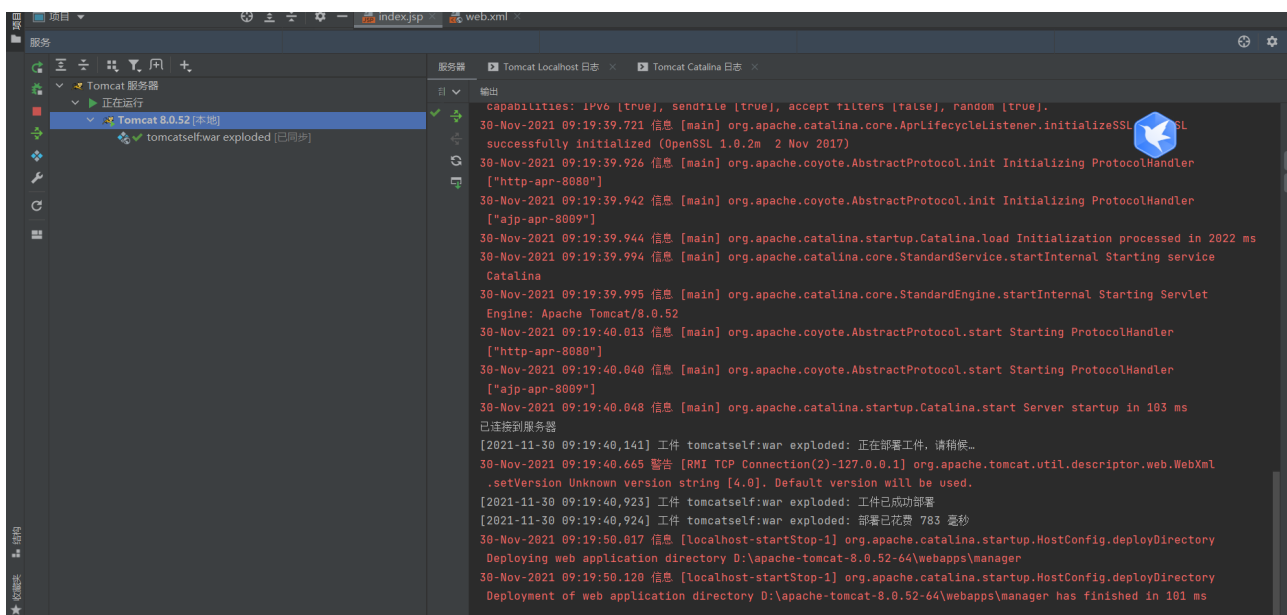
- 点击 +号----- 工件



- 弹出如下：（应用程序上下文时虚拟目录）



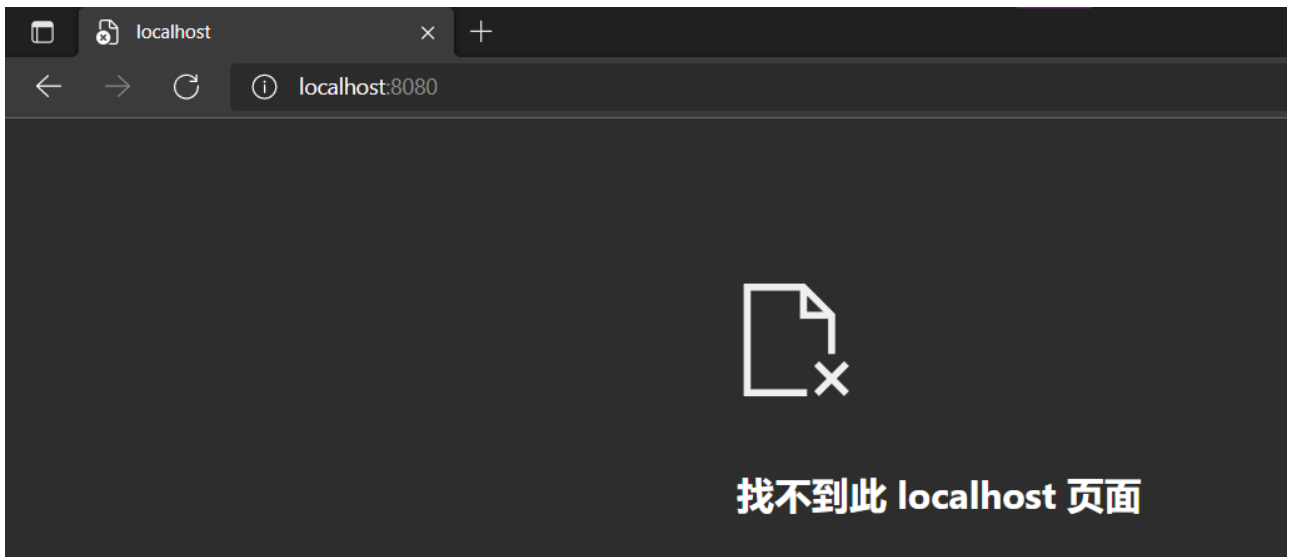
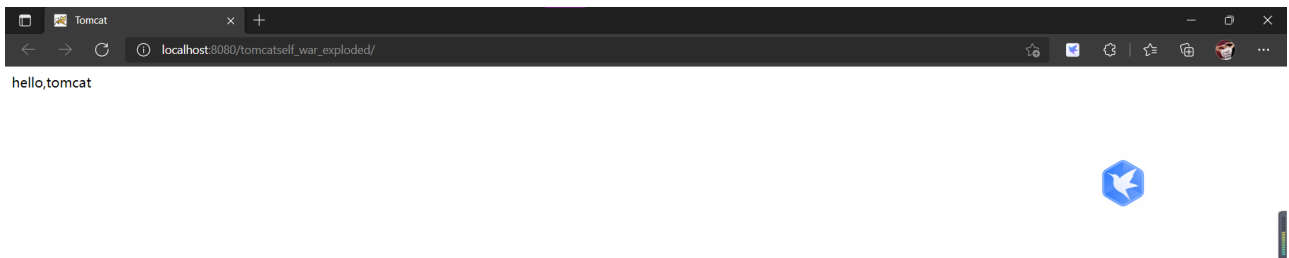
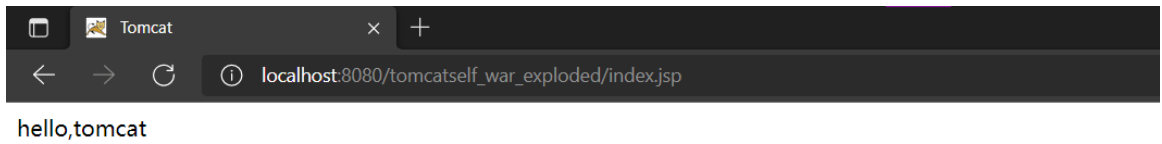
- 主界面右上角点击“绿色三角形后”控制台输出如下信息

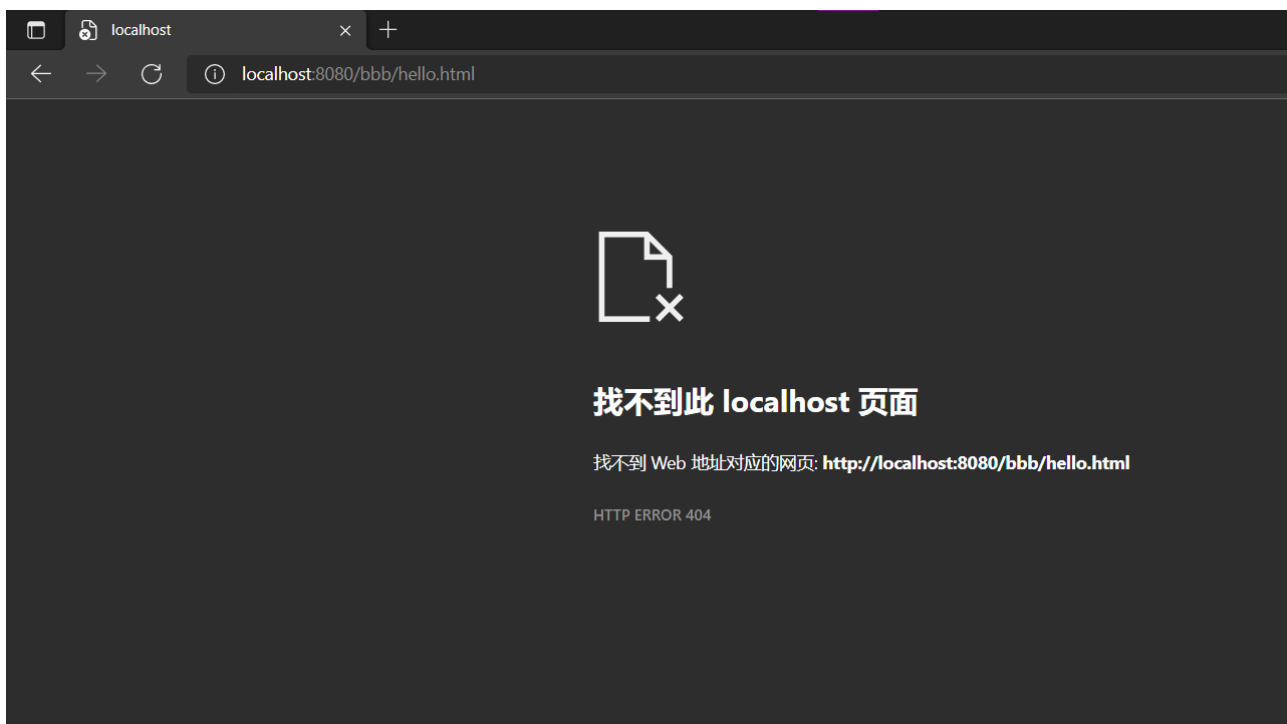


- 信息输出完成后，浏览器会自动弹出来，且如下路径(http://localhost:8080/tomcatself_war_exploded/)或者 (http://localhost:8080/tomcatself_war_exploded/index.jsp)

注：此时只能访问此页，不能启动其他任何页面

注：项目目录结构会出先一个out的目录



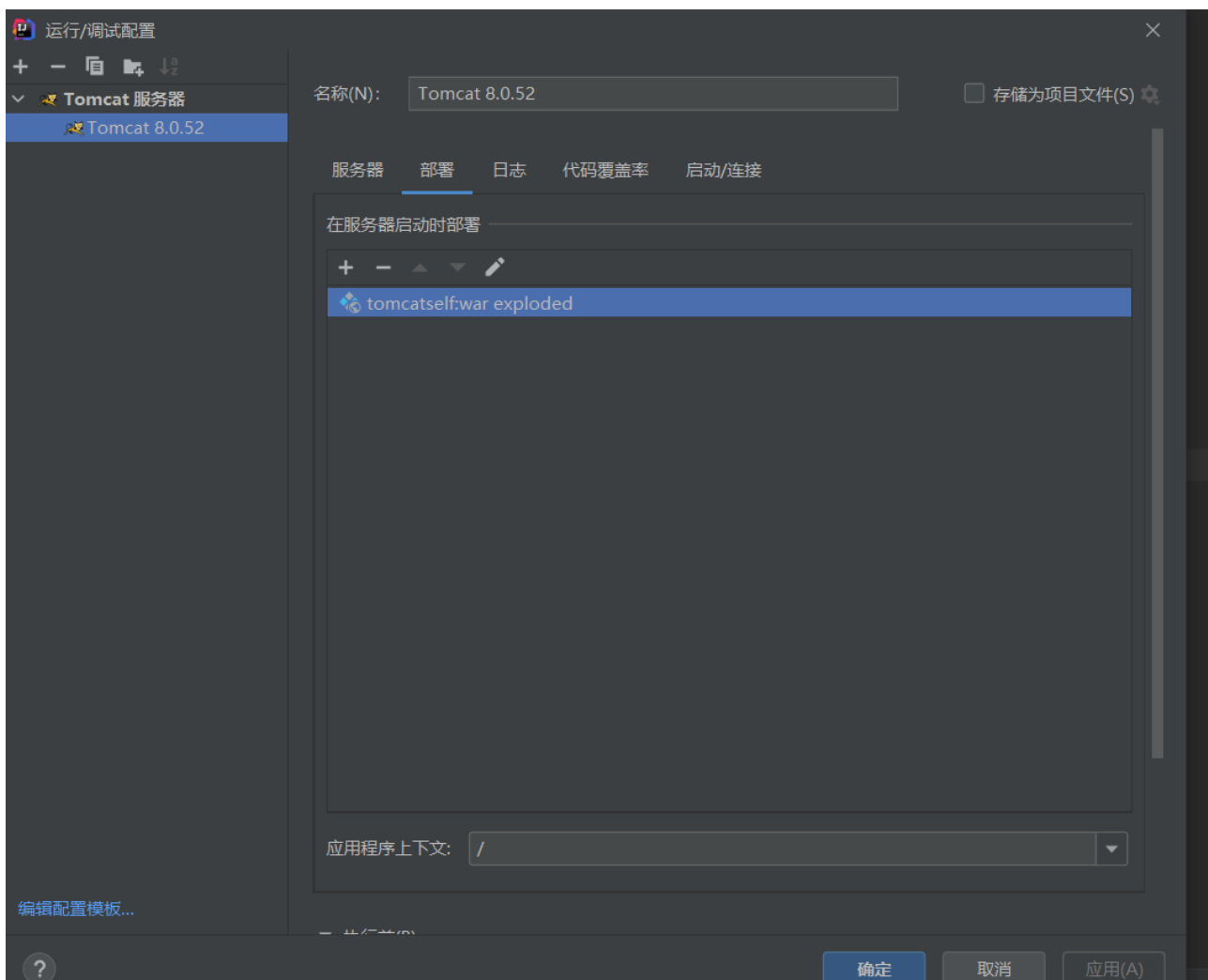


- 停止tomcat服务器后，修改应用程序上下文；然后 再启动服务器，信息输出完成后，浏览器会自动弹出来，且如下路径(<http://localhost:8080>)或者（<http://localhost:8080/index.jsp>）

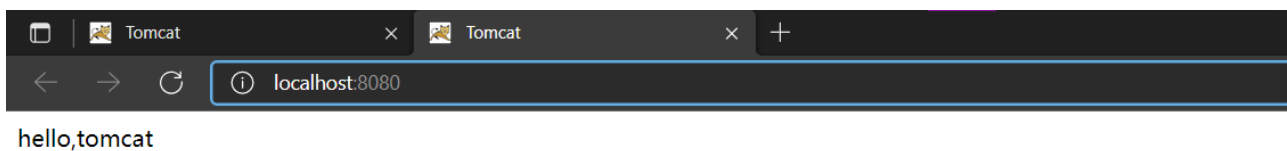
注：此时只能访问此页，不能启动其他任何页面；

`index.jsp`是默认的首页

必须得停止服务器后才能进行修改

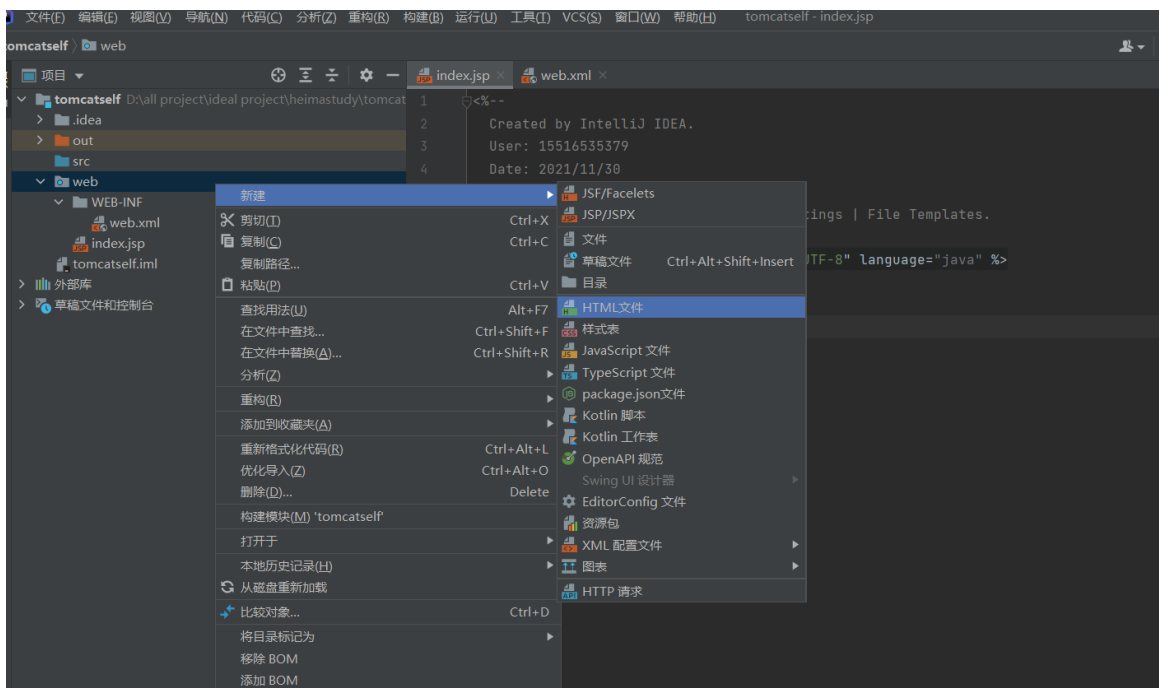


•

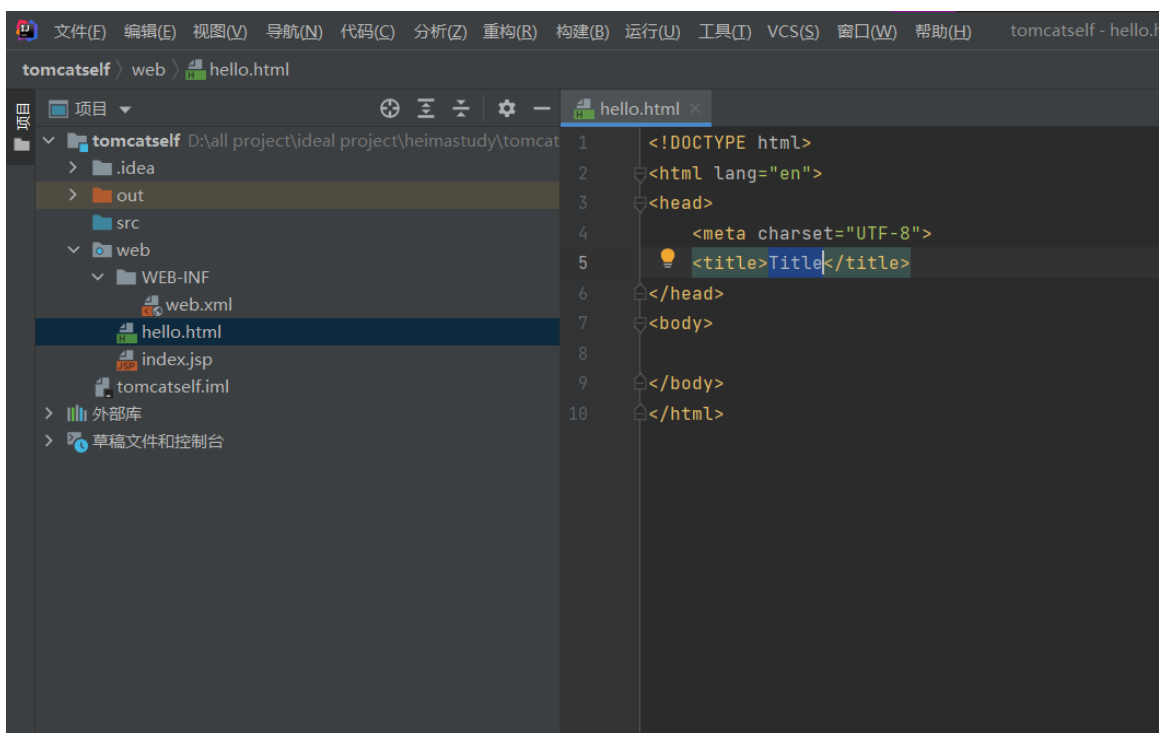


5.web下新建页面并访问（接4.）

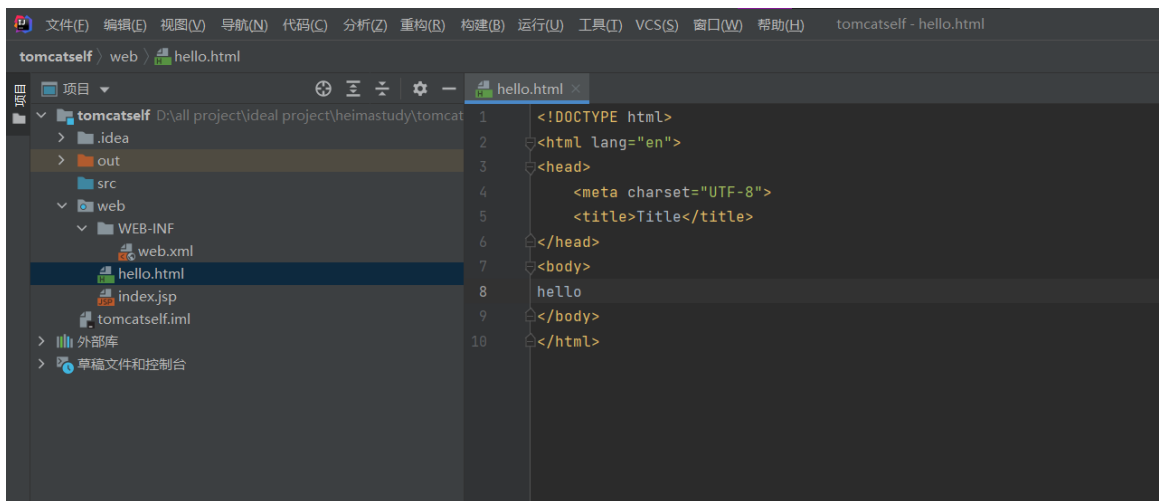
- 停止服务器后，右键web----新建----HTML文件----命名hello



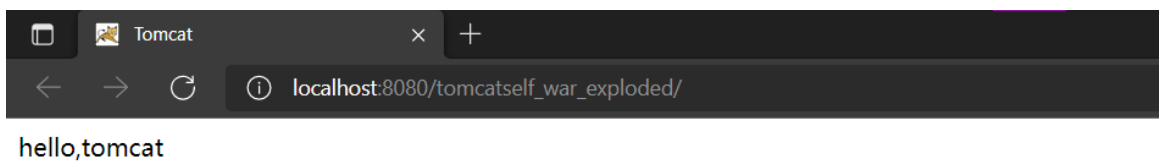
- 现在的文件结构及初始默认hello.html内容



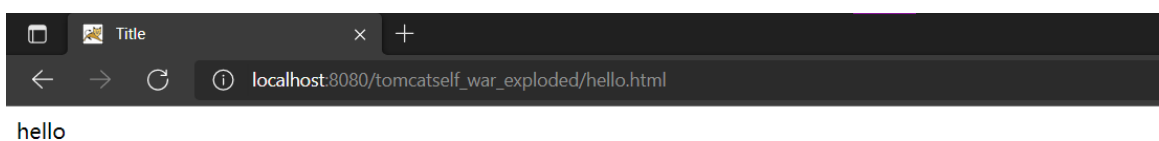
- 修改hello.html如下：



- 启动tomcat服务器，网址(http://localhost:8080/tomcatself_war_exploded)弹出

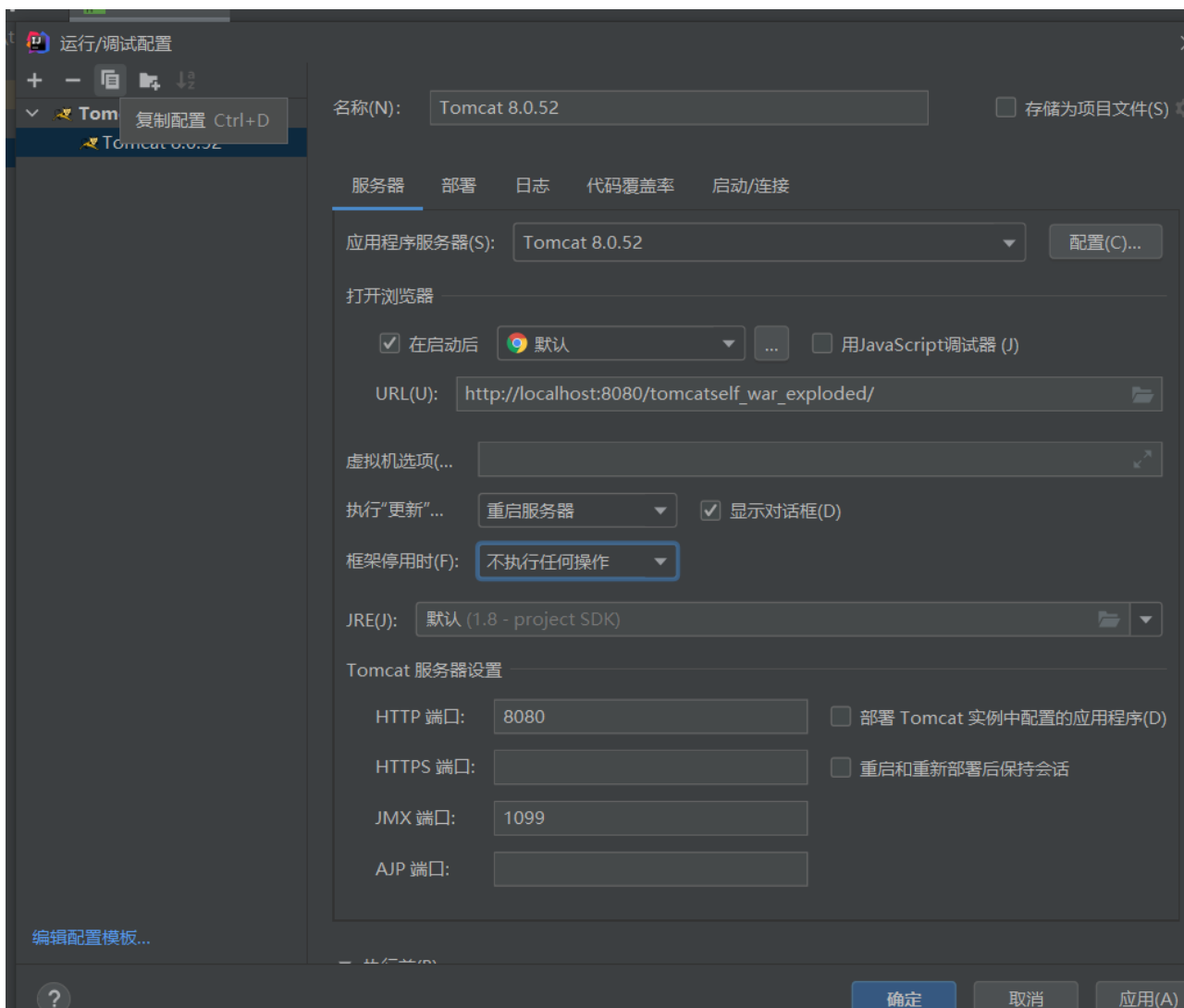


- 网页输入（http://localhost:8080/tomcatself_war_exploded/hello.html）即可访问



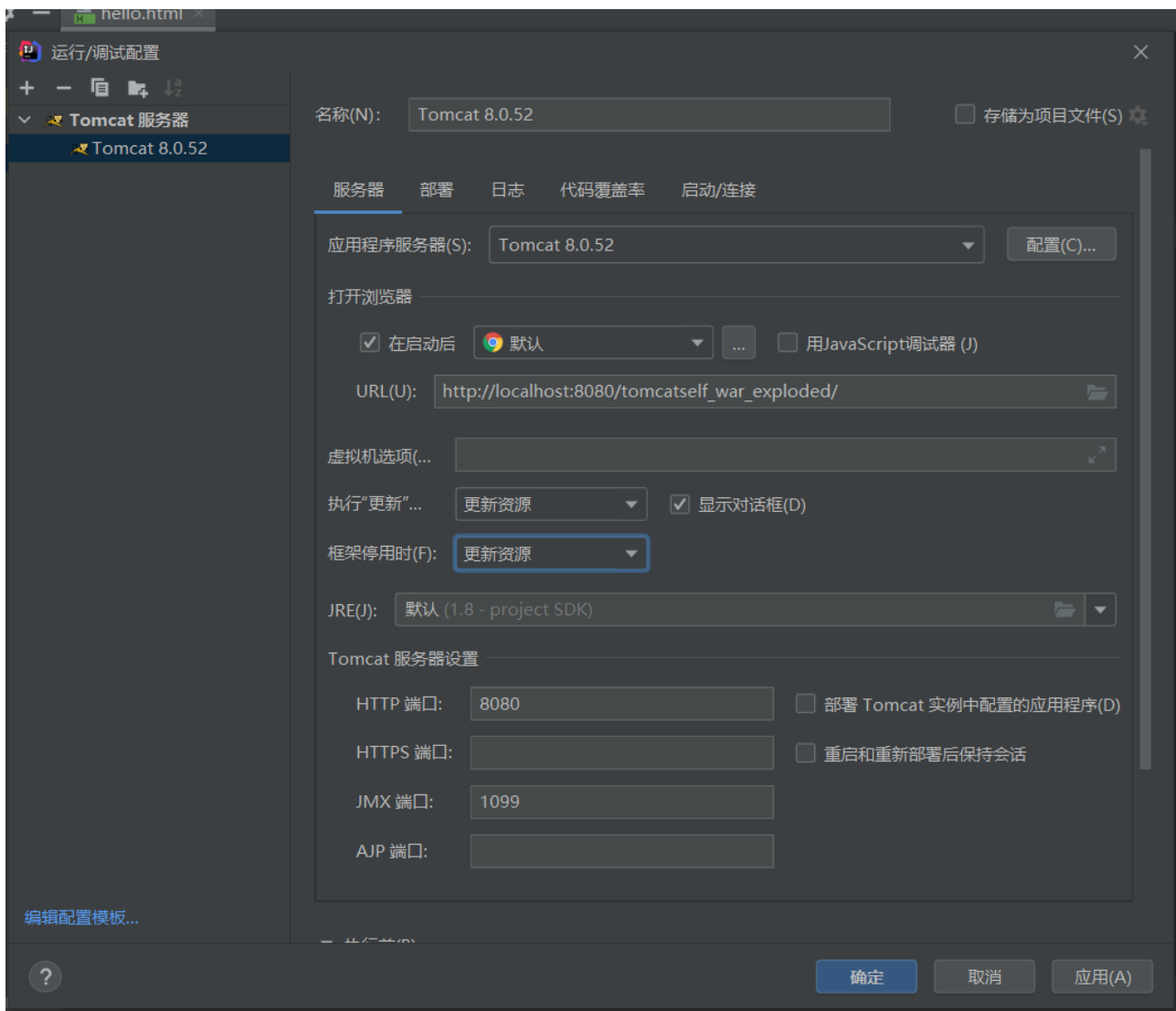
6.热部署快速访问（接5.）

- 以下是默认配置：之前只要改动项目的任何访问界面的信息都必须得需要停止服务器再启动服务器才能访问到新界面

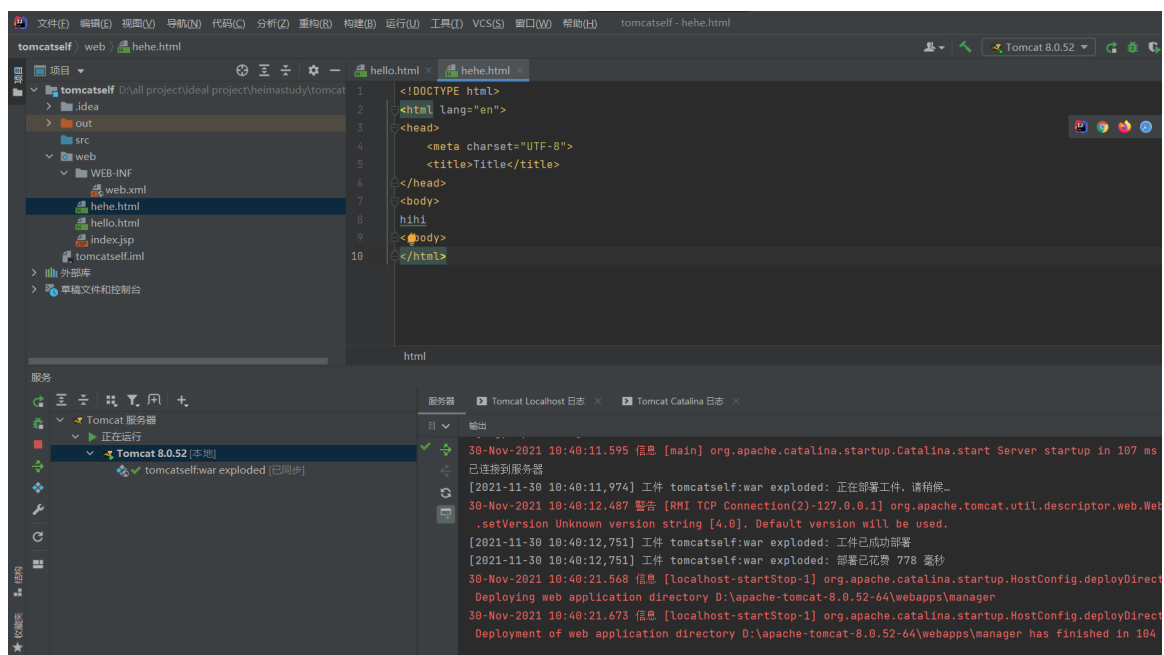


- 停止tomcat，改成以下配置后如果更新了资源则不再需要停止服务器，直接搜索网页即可搜到（不选择“更新类和资源”的原因是：一般类的改动都比较大，自动更新类的话会影响性能）

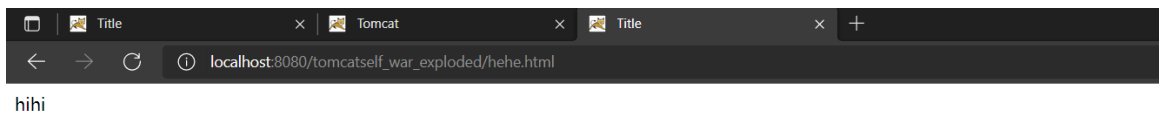
注：如果真的选择了更新类和资源，则在启动时不要选择绿三角形，而是选择调试模式即点击小虫子（也会自动弹出网页），在更新了src下面的代码后，再次访问访问网页时多刷新几次，代码即可会自动更新（注：即使刷新网页idea也会自动更新代码，但是得停好久才能更新，所以最好自己去刷新一下网站；）（注：更新类的意思是：只能在已经创建的类中更新，如果又新创了一个java，是不会自动更新的；想要可以自动更新类，则需要配置成重新部署，然后点击小虫子开启等步骤）



- 启动服务器，然后右键web---新建html文件---命名hehe，并修改内容



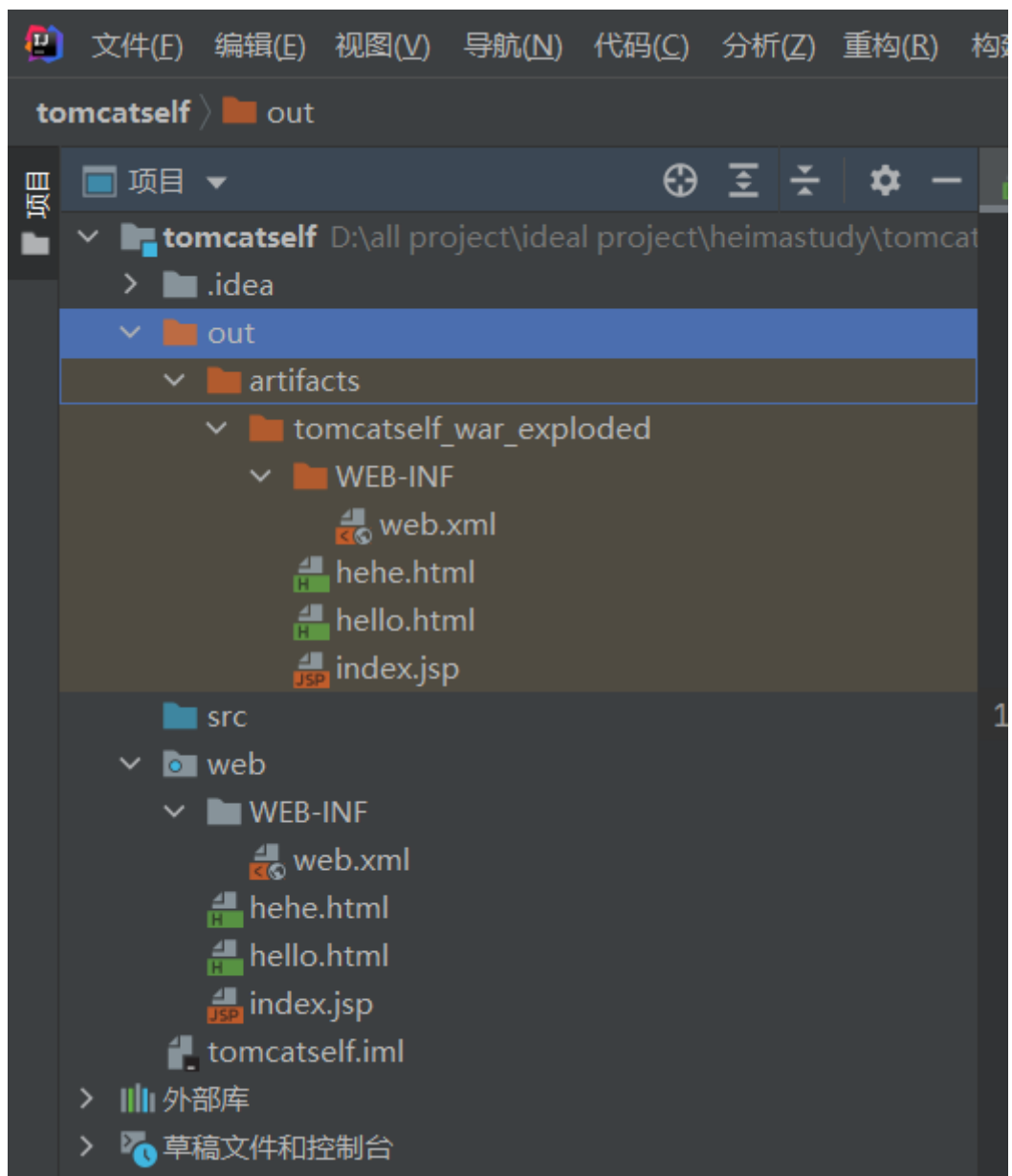
- 不需要停止服务器，可以直接在网页搜索（http://localhost:8080/tomcatself_war_exploded/hehe.html）进行访问



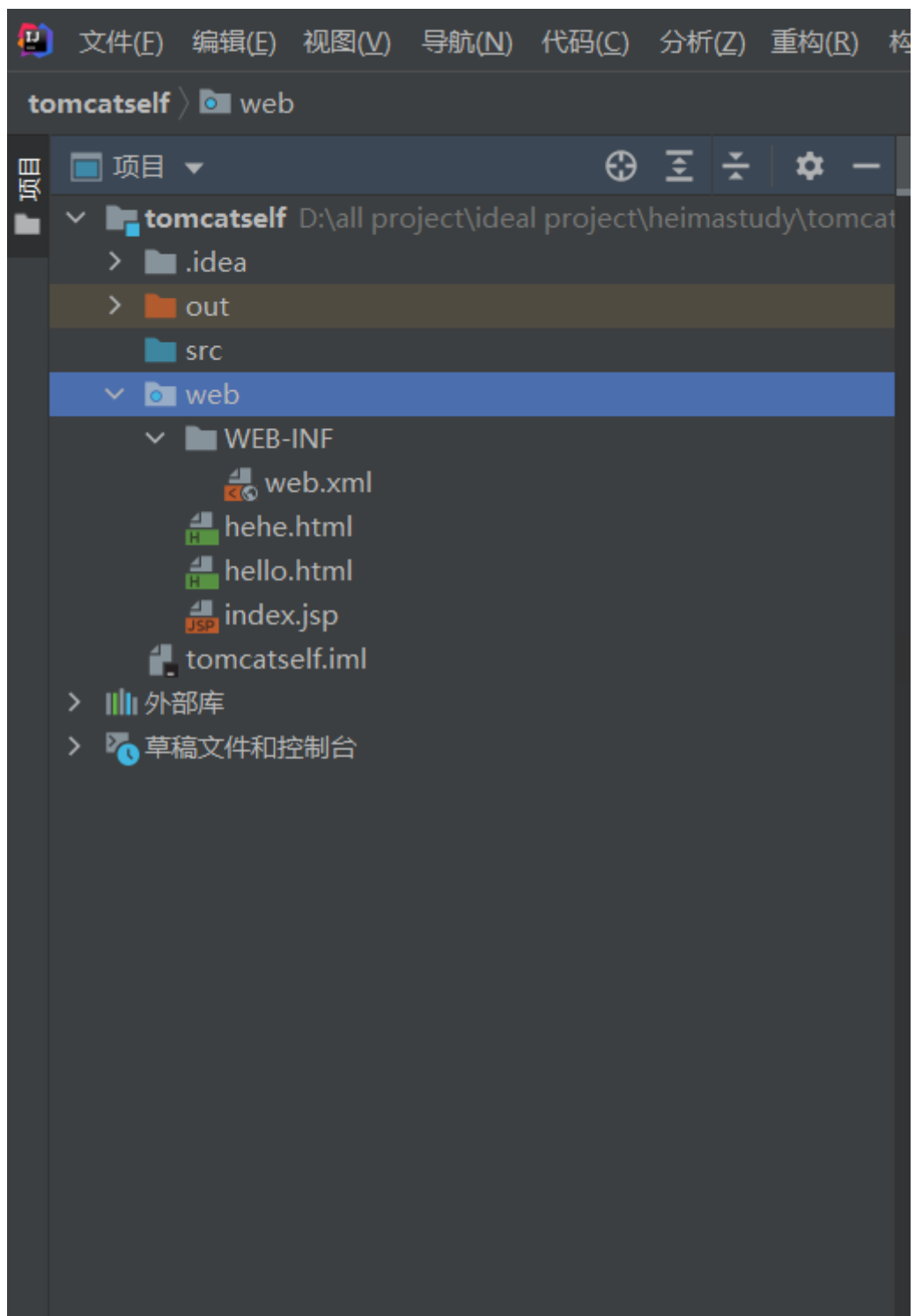
IDEA与tomcat的相关配置

- 可以在IDEA中的编辑配置里的HTTP端口进行对端口号进行修改
- WEB-INF目录下的资源不能被浏览器访问. (所以尽量不要把HTML等资源给放到WEB-INF下面, 尽量在WEB下面创建)
- idea中项目有两个存储的地方存在于两个地方"工作空间项目" 和"tomcat部署的web项目", 它俩存储的位置不一样

tomcat部署的web项目存在于如下图的out目录 (只有原项目的web文件夹) 里:



工作空间项目是如下图的tomcatself里的web文件夹：



- tomcat真正访问的是“tomcat部署的web项目”（out下面）
- "tomcat部署的web项目"对应着"工作空间项目"的web目录下的所有资源（正常目录下面的）
- Tomcat里的断点调试：使用"小虫子"启动 debug 启动（普通程序也是小虫子），点击小虫子后tomcat会启动，会正常输出一大堆信息；但是当在网页里搜索进行断点的程序后，控制台会出现断点调试的界面（比如servlet实现类）；当在网页里搜索不进行断点的程序后，控制台不会出现断点调试的界面