Tomcat源码调试配置

一、下载

Tomcat的构建是基于Ant和Eclipse的，然而现在很多人都喜欢IDEA+Maven的项目构建方式，所以本文将基于这个环境来搭建源码的调试。我们需要以下工具：

Tomcat源码下载地址：https://mirrors.tuna.tsinghua.edu.cn/apache/tomcat/tomcat-8/v8.5.23/src/apache-tomcat-8.5.23-src.tar.gz

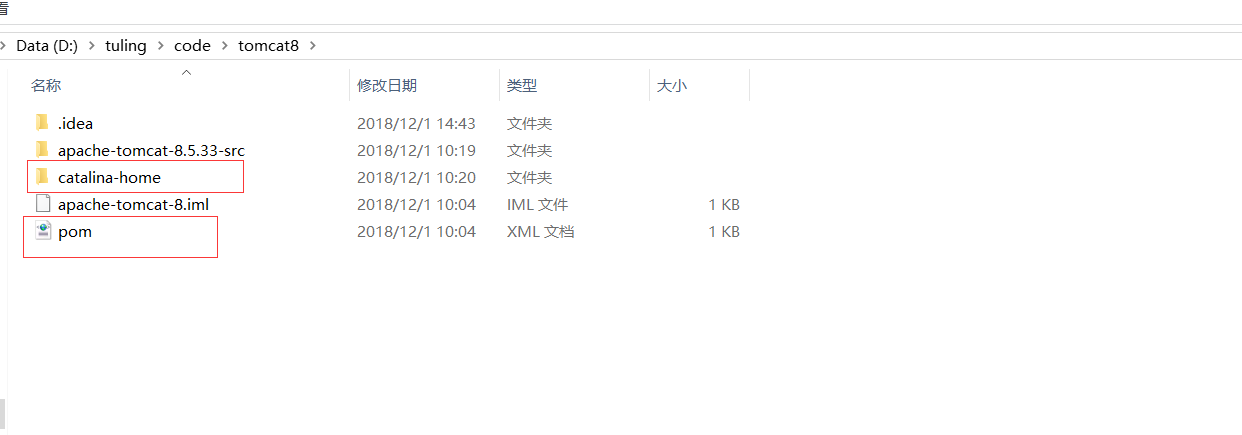
IDEA工具：https://www.jetbrains.com/idea/download

MAVEN：http://maven.apache.org/download.cgi

JDK：自然不用多提了，但是要按照所选源码要求的版本，这里用的是JDK8

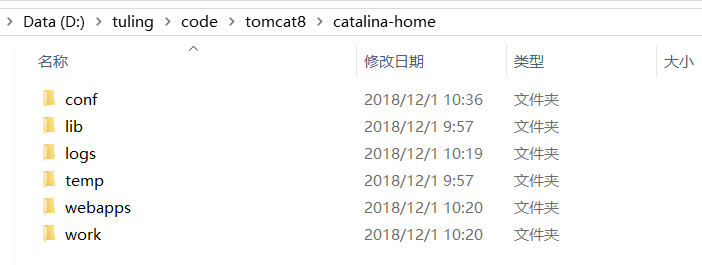
安装和下载这些软件包就可以开始搭建调试环境了。

1. 项目结构



新建一个目录，比如：code\tomcat8，然后将tomcat8的源码解压至该目录

新建catalina-home目录，然后将apache-tomcat-8.5.23-src目录下的 conf文件夹拷贝到此处，该目录结构如下



除了conf目录其他都是可选的，webapps用于我们应用默认的部署目录，work logs是启动Tomcat自动生成的，其结构跟我们下载的二进制Tomcat程序是一样的.

配置Maven依赖

我们采用module的形式来组织目录,首先在根目录(D:\code\tomcat8)下创建pom.xml,其内容如下:

<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>gxf</groupId>

<artifactId>apache-tomcat-8</artifactId>

<name>apache-tomcat-8-source</name>

<version>1.0</version>

<packaging>pom</packaging>

<modules>

<module>apache-tomcat-8.5.33-src</module>

</modules>

</project>

这里主要指定module为Tomcat的源码目录,然后在apache-tomcat-8.5.23-src配置Tomcat源码额外的依赖，在该目录创建pom.xml

<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>org.apache.tomcat</groupId>

<artifactId>Tomcat8.5.33</artifactId>

<name>Tomcat8.5.33</name>

<version>8.5</version>

<build>

<finalName>Tomcat8.0</finalName>

<sourceDirectory>java</sourceDirectory>

<testSourceDirectory>test</testSourceDirectory>

<resources>

<resource>

<directory>java</directory>

</resource>

</resources>

<testResources>

<testResource>

<directory>test</directory>

</testResource>

</testResources>

<plugins>

<plugin>

<groupId>org.apache.maven.plugins</groupId>

<artifactId>maven-compiler-plugin</artifactId>

<version>2.0.2</version>

<configuration>

<encoding>UTF-8</encoding>

<source>1.8</source>

<target>1.8</target>

</configuration>

</plugin>

</plugins>

</build>

<dependencies>

<dependency>

<groupId>org.easymock</groupId>

<artifactId>easymock</artifactId>

<version>3.5</version>

<scope>test</scope>

</dependency>

<dependency>

<groupId>junit</groupId>

<artifactId>junit</artifactId>

<version>4.12</version>

<scope>test</scope>

</dependency>

<dependency>

<groupId>ant</groupId>

<artifactId>ant</artifactId>

<version>1.7.0</version>

</dependency>

<dependency>

<groupId>wsdl4j</groupId>

<artifactId>wsdl4j</artifactId>

<version>1.6.2</version>

</dependency>

<dependency>

<groupId>javax.xml</groupId>

<artifactId>jaxrpc</artifactId>

<version>1.1</version>

</dependency>

<dependency>

<groupId>org.eclipse.jdt.core.compiler</groupId>

<artifactId>ecj</artifactId>

<version>4.6.1</version>

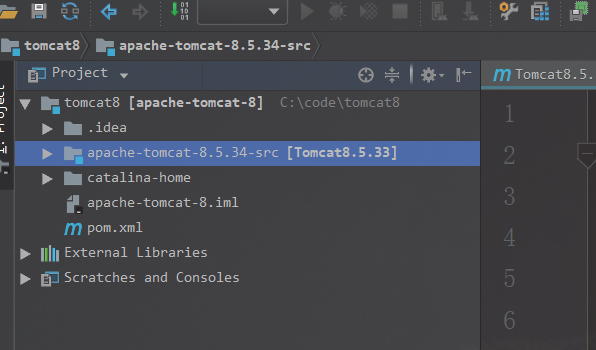
</dependency>

</dependencies>

</project>

三，准备构建

使用IDEA打开tomcat8下面的pom.xml



配置编译环境

说明：如果编译build的时候出现Test测试代码报错，删除该代码即可。本文中的Tomcat源码util.TestCookieFilter类会报错，将其删除即可。

可以把Test目录删除

-Dcatalina.home=catalina-home1

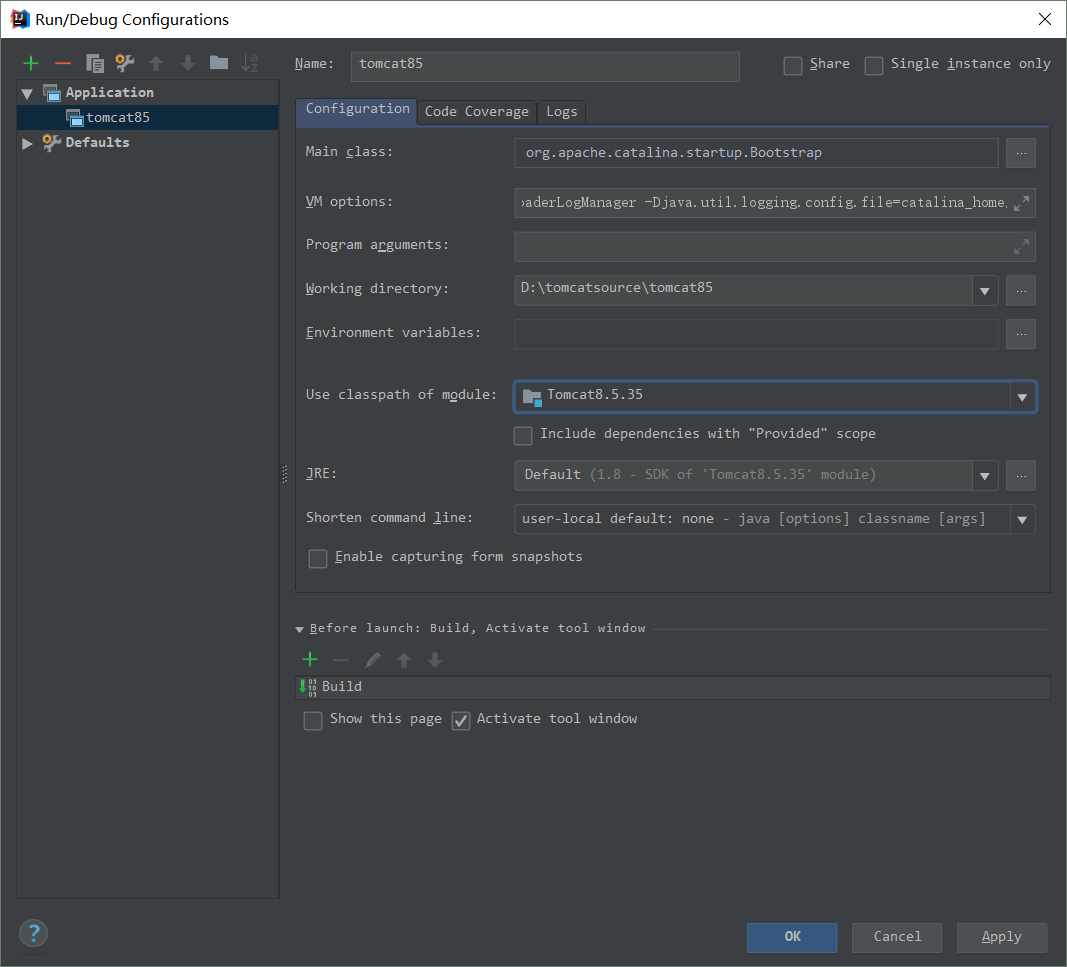
-Dcatalina.base=catalina-home1

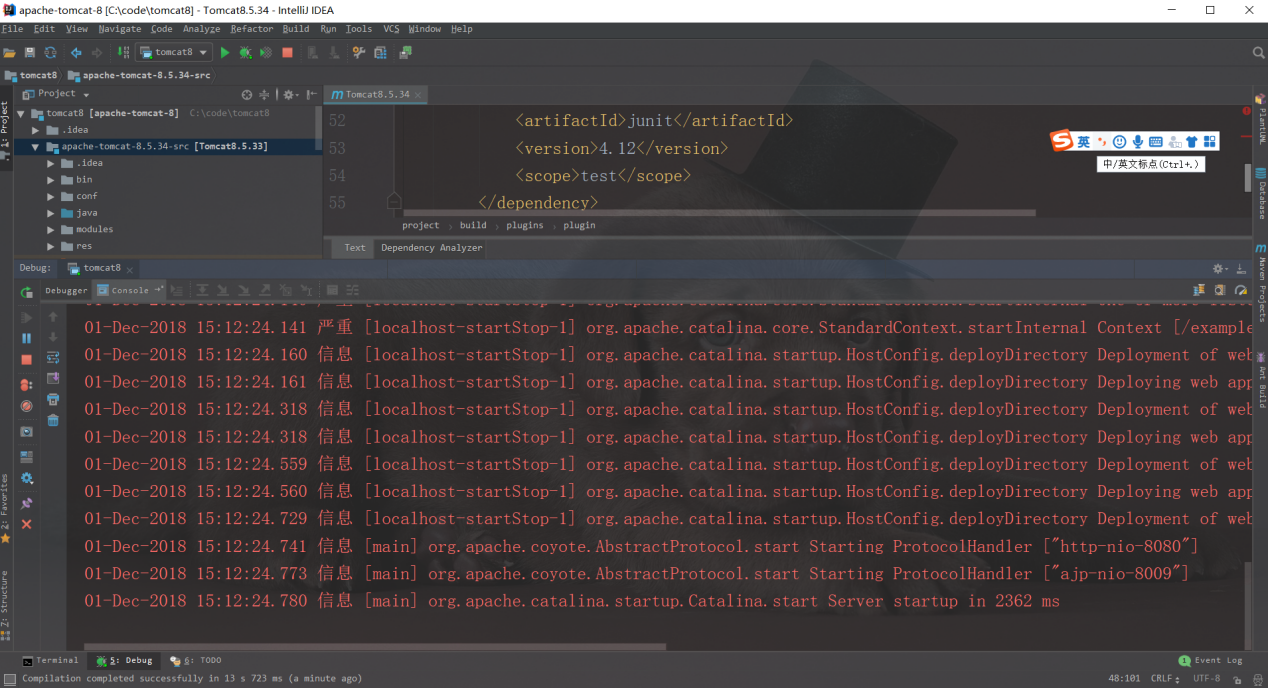
-Djava.endorsed.dirs=catalina-home1/endorsed

-Djava.io.tmpdir=catalina-home1/temp

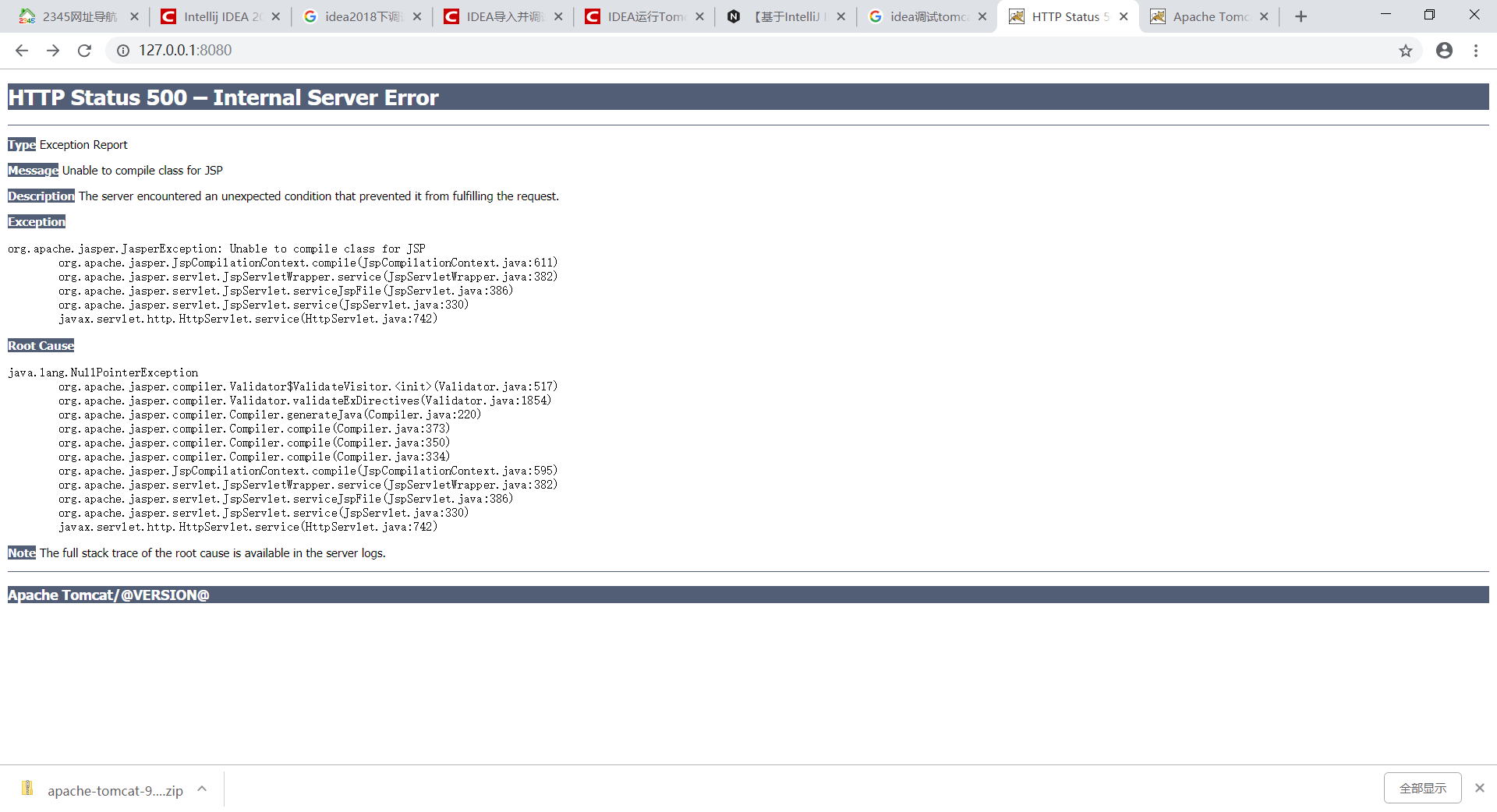
-Djava.util.logging.manager=org.apache.juli.ClassLoaderLogManager

-Djava.util.logging.config.file=catalina-home1/conf/logging.properties

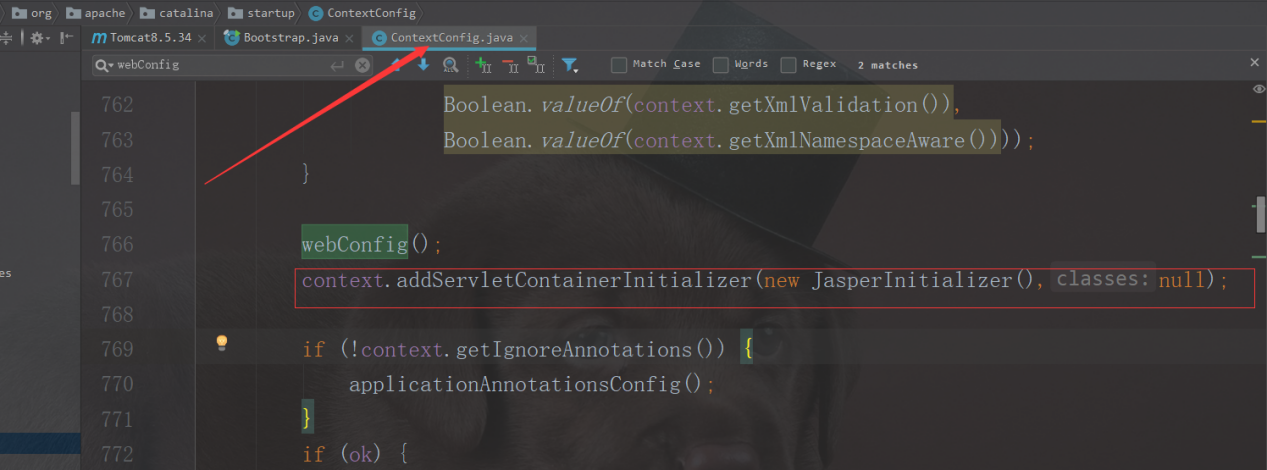




HTTP：//127.0.0.1:8080/



原因是我们直接启动org.apache.catalina.startup.Bootstrap的时候没有加载org.apache.jasper.servlet.JasperInitializer，从而无法编译JSP。这在Tomcat6/7是没有这个问题的。解决办法是在tomcat的源码org.apache.catalina.startup.ContextConfig中手动将JSP解析器初始化：



默认情况下我们必须把应用程序部署到catalina-home/webapps下，如何直接部署访问外部的应用程序?我们只需要修改server.xml和web.xml配置即可。

1. 编辑
2. catalina-home/conf/server.xml

<Context path="/tomcatsrc-web"

reloadable="true"

debug="0" docBase="c:\code\work\tomcatsrc2018\tomcatsrc-web\target\tomcatsrc-web-1.0-SNAPSHOT" workDir="c:\code\work\tomcatsrc2018\tomcatsrc-web\target\tomcatsrc-web-1.0-SNAPSHOT\work"

crossContext="true" />