

# 开发环境搭建

@M了个J  
李明杰

<https://github.com/CoderMJLee>

<http://cnblogs.com/mjios>



实力IT教育 [www.520it.com](http://www.520it.com)

码拉松



# Java开发必备

■ 要想顺利开发Java程序，有2个必须要安装的软件：JDK、JRE

■ JDK (Java Development Kit)：Java开发工具包

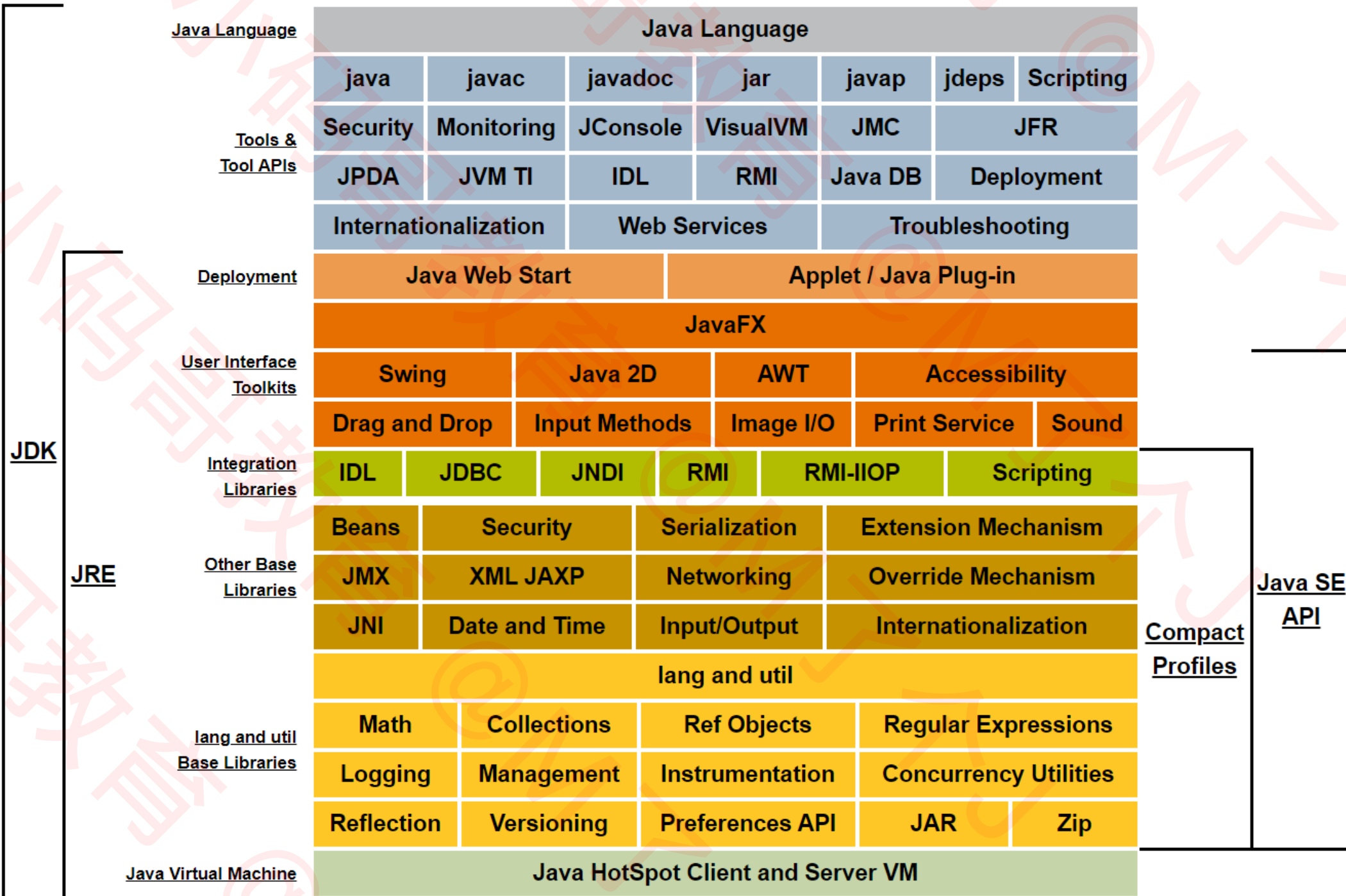
□ 提供了开发中要用到的各种常用工具

■ JRE (Java Runtime Environment)：Java运行环境

□ 将开发完的Java程序运行到JVM上



■ 关于JDK、JRE更具体的内容，在以后的课程中会进行详细讲解



# JDK的下载

■ JDK中已经包含了JRE，所以不需要单独下载JRE，只下载JDK即可（这里选择JDK8）

□ 下载地址：<https://www.oracle.com/technetwork/java/javase/downloads/index.html>

## Java SE 8u231

Java SE 8u231 includes important bug fixes. Oracle strongly recommends that all Java SE 8 users upgrade to this release.

[Learn more](#)

- [Installation Instructions](#)
- [Release Notes](#)
- [Oracle License](#)

JDK

DOWNLOAD

■ 根据你的操作系统选择对应的版本

□ Windows x86对应32bit Windows系统

□ Windows x64对应64bit Windows系统

□ Mac OS X x64对应Mac OS X系统

## Java SE Development Kit 8u231

You must accept the [Oracle Technology Network License Agreement for Oracle Java SE](#) to download this software.

☒ Accept License Agreement

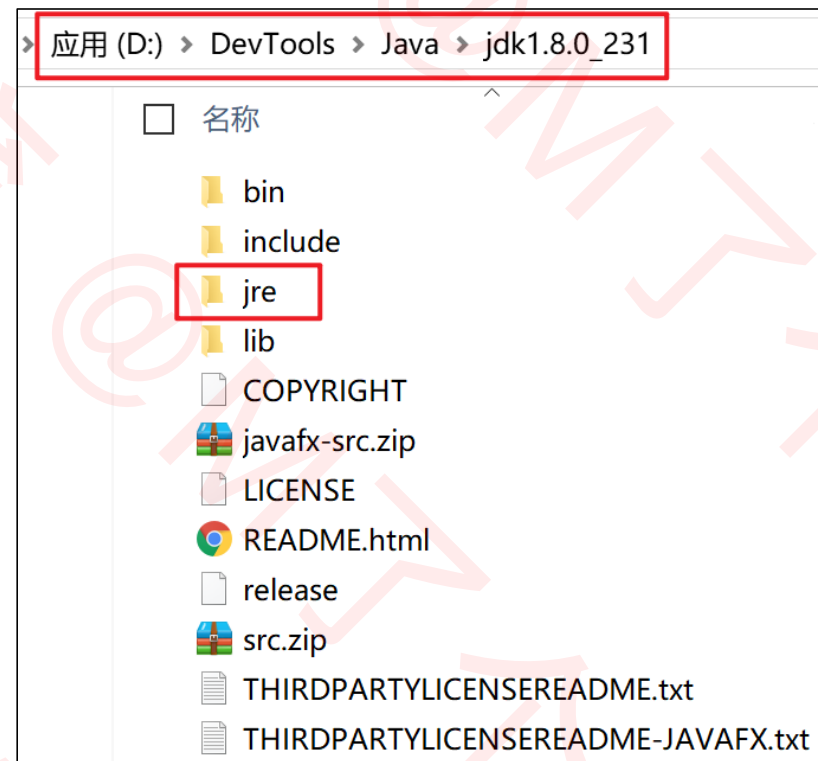
☐ Decline License Agreement

Product / File Description	File Size	Download
Linux ARM 32 Hard Float ABI	72.9 MB	<a href="#">jdk-8u231-linux-arm32-vfp-hflt.tar.gz</a>
Linux ARM 64 Hard Float ABI	69.8 MB	<a href="#">jdk-8u231-linux-arm64-vfp-hflt.tar.gz</a>
Linux x86	170.93 MB	<a href="#">jdk-8u231-linux-i586.rpm</a>
Linux x86	185.75 MB	<a href="#">jdk-8u231-linux-i586.tar.gz</a>
Linux x64	170.32 MB	<a href="#">jdk-8u231-linux-x64.rpm</a>
Linux x64	185.16 MB	<a href="#">jdk-8u231-linux-x64.tar.gz</a>
Mac OS X x64	253.4 MB	<a href="#">jdk-8u231-macosx-x64.dmg</a>
Solaris SPARC 64-bit (SVR4 package)	132.98 MB	<a href="#">jdk-8u231-solaris-sparcv9.tar.Z</a>
Solaris SPARC 64-bit	94.16 MB	<a href="#">jdk-8u231-solaris-sparcv9.tar.gz</a>
Solaris x64 (SVR4 package)	133.73 MB	<a href="#">jdk-8u231-solaris-x64.tar.Z</a>
Solaris x64	91.96 MB	<a href="#">jdk-8u231-solaris-x64.tar.gz</a>
Windows x86	200.22 MB	<a href="#">jdk-8u231-windows-i586.exe</a>
Windows x64	210.18 MB	<a href="#">jdk-8u231-windows-x64.exe</a>

# JDK的安装

■ JDK的安装路径不建议包含中文、空格等特殊符号

□ 下图的【开发工具】中已经包含了JRE，因此【公共JRE】是可以不安装的，取消即可



■ 从目录结构可以看得出来：JDK中已经包含了JRE

- 常见的Java IDE有：NetBeans、Eclipse、MyEclipse、IntelliJ IDEA
- 目前企业中用得最多的Java IDE是Eclipse、IntelliJ IDEA
- 在JavaSE学习阶段，将使用Eclipse

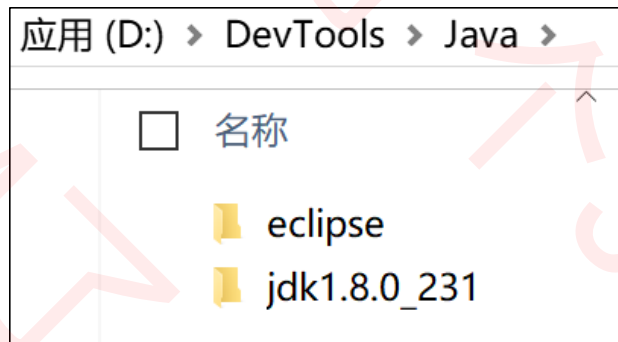
# Eclipse的下载和安装

- 下载地址: <https://www.eclipse.org/downloads/packages/>
- JavaSE阶段, 选择【Eclipse IDE for Java Developers】版本即可



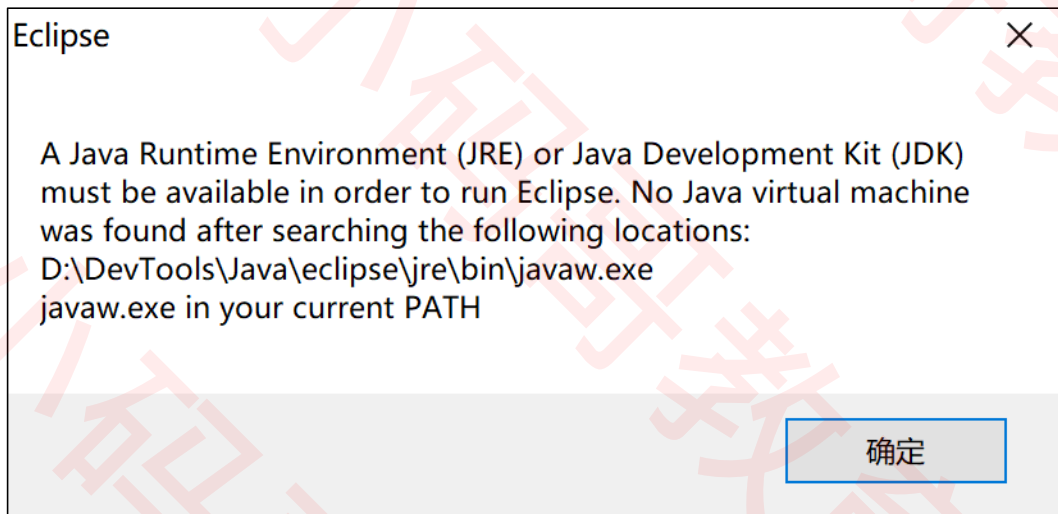
- 最新版的Eclipse已经不支持32bit, 旧版本的Eclipse才支持32bit
- <https://www.eclipse.org/downloads/packages/release/photon/r>

- 下载完毕后, Eclipse是个zip压缩包, 直接解压即可使用
- Eclipse的解压路径不建议包含中文、空格等特殊符号
- 可以考虑跟JDK放在同一目录





- 在Windows中打开Eclipse可能会弹出下面的对话框



- 解决方案

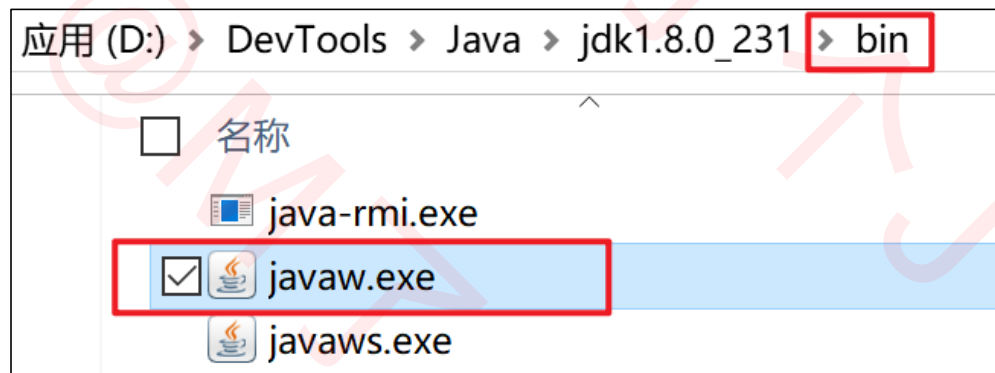
- 将JDK的bin目录配置到电脑的PATH中去, 这样Eclipse就可以在PATH中找到javaw.exe了

- 大概意思是: 在PATH中找不到javaw.exe

- 说明Eclipse启动的时候会去执行javaw.exe程序

- 那javaw.exe在哪呢?

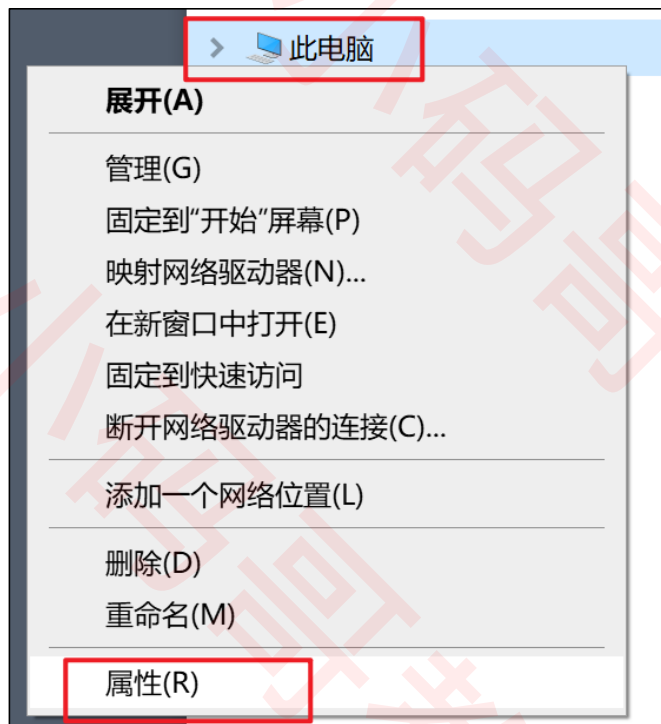
- 在JDK的bin目录下





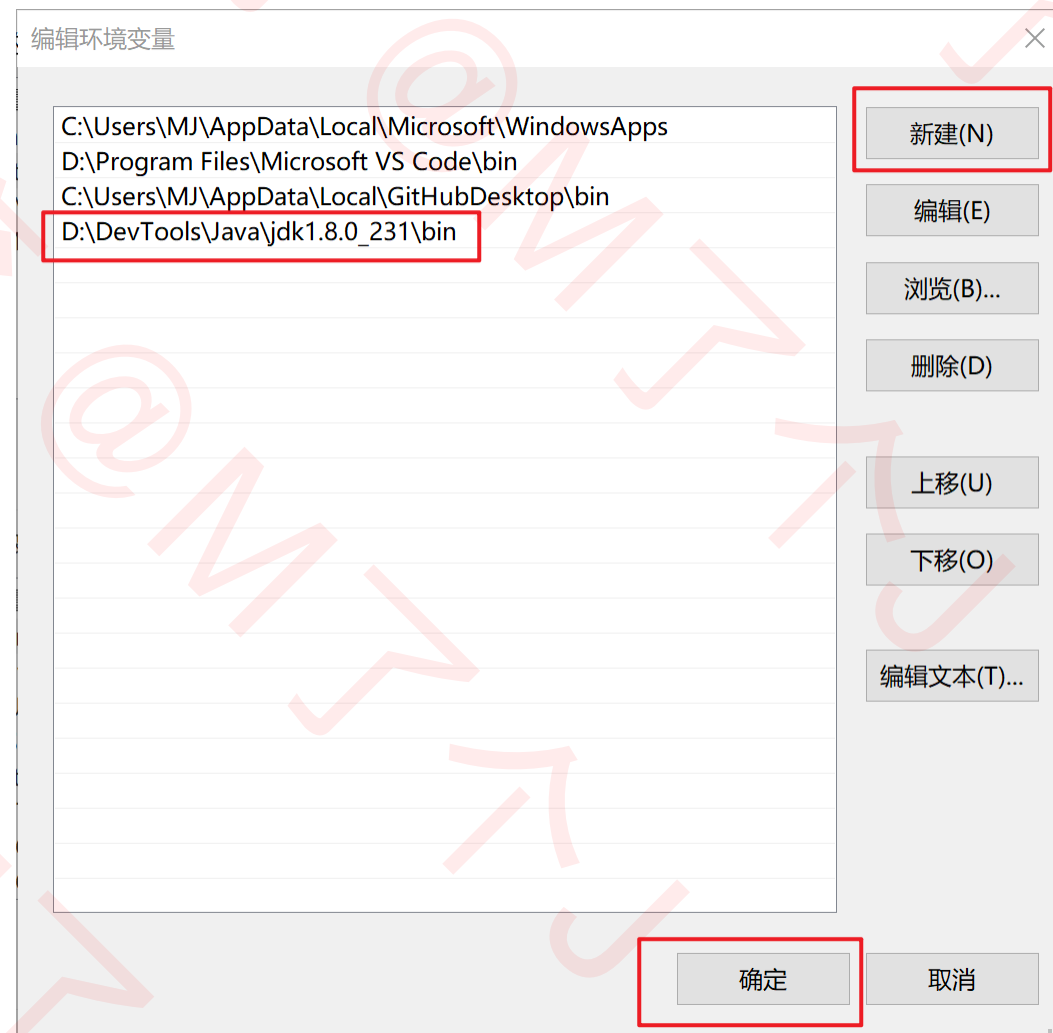
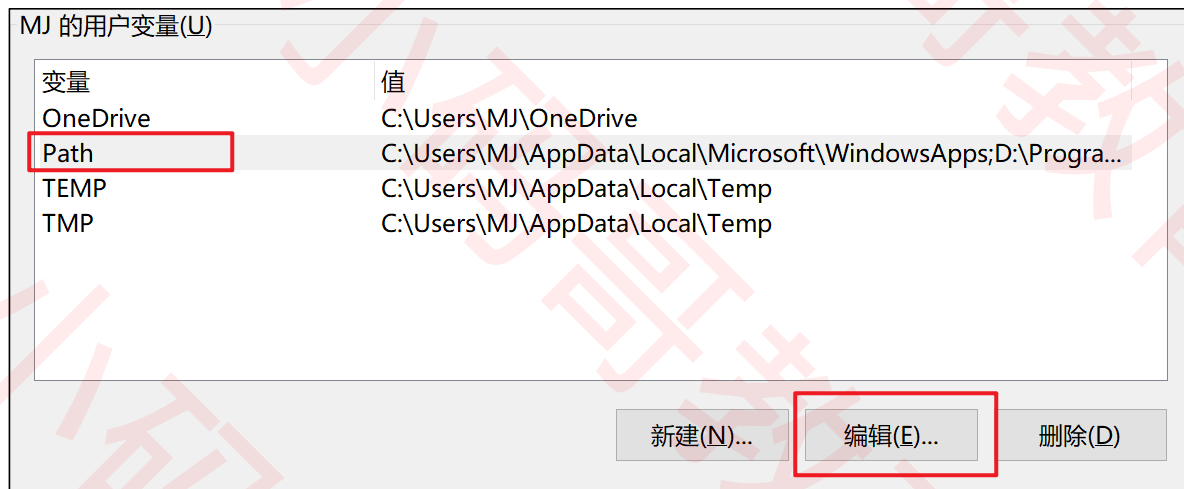
# PATH配置1

■ 右击【电脑\计算机\我的电脑】→【属性】→【高级系统设置】→【环境变量】



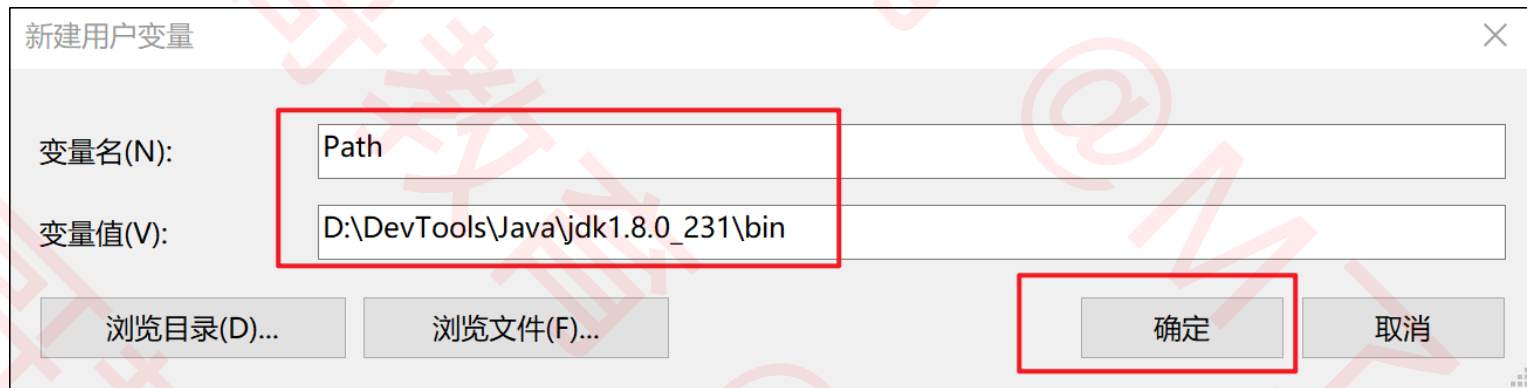
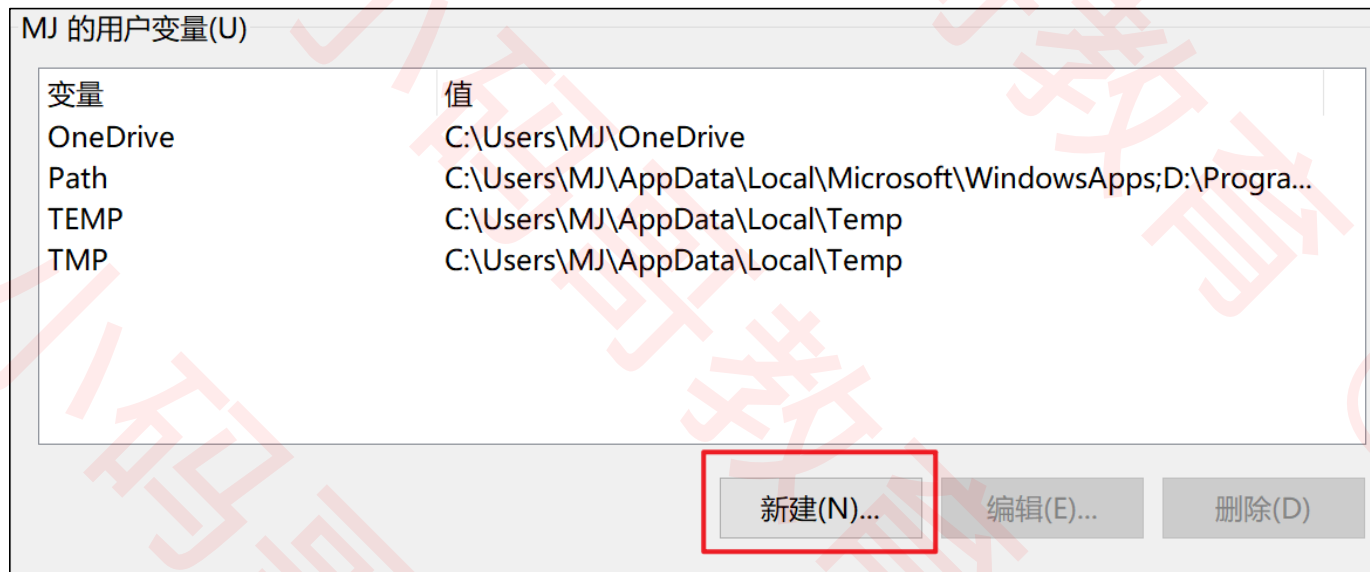
# PATH配置2

■ 【用户变量】→【Path】→【编辑】→【新建】→【确定】



# PATH配置3

■ 如果【Path】是不存在的，【新建】→【输入Path和JDK的bin目录】→【确定】



- 如果Eclipse可以成功打开，看到了启动界面，说明PATH配置成功



# PATH配置 (Windows 7、XP)

- JDK的bin目录跟Path中已经存在的路径之间用英文的分号 ; 隔开

编辑用户变量

变量名(N): Path

变量值(V): C:\Users\MJ\AppData\Local\GitHubDesktop\bin;D:\DevTools\Java\jdk1.8.0\_231\bin;

浏览目录(D)... 浏览文件(F)... 确定 取消

# PATH配置 - 补充

- 以后可能会经常在环境变量中用到JDK目录，可以考虑将JDK目录定义为**JAVA\_HOME**，方便重复利用

MJ 的用户变量(U)

变量	值
JAVA_HOME	D:\DevTools\Java\jdk1.8.0_231
OneDrive	C:\Users\MJ\OneDrive
Path	C:\Users\MJ\AppData\Local\Microsoft\Windows
TEMP	C:\Users\MJ\AppData\Local\Temp
TMP	C:\Users\MJ\AppData\Local\Temp

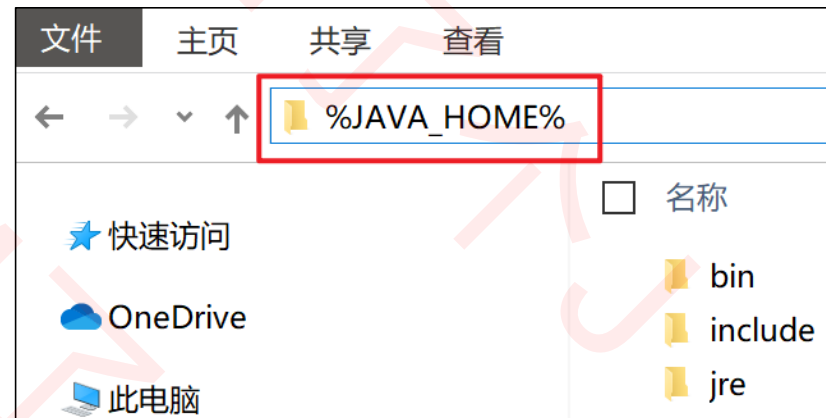
- 在Path中就可以使用**%JAVA\_HOME%**引用JDK目录

编辑环境变量

C:\Users\MJ\AppData\Local\Microsoft\WindowsApps
D:\Program Files\Microsoft VS Code\bin
C:\Users\MJ\AppData\Local\GitHubDesktop\bin
%JAVA_HOME%\bin

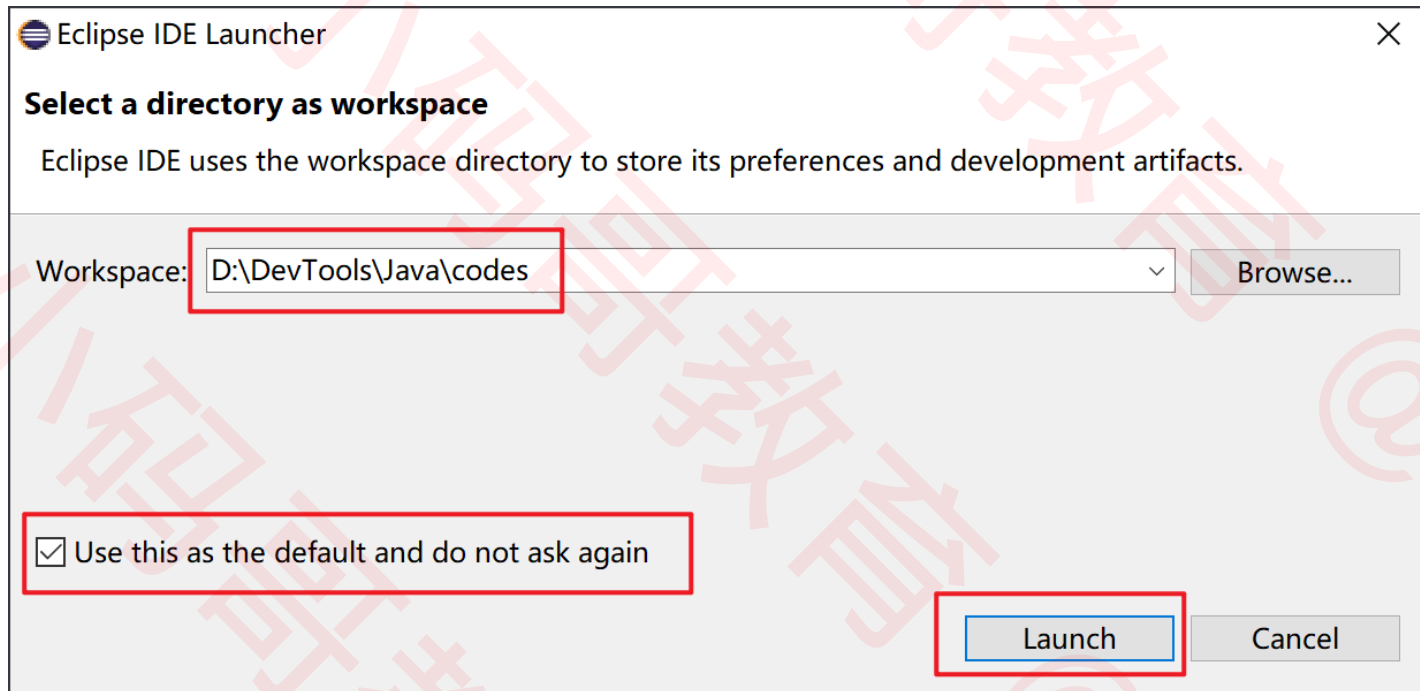
- 甚至可以直接在文件浏览器中输入**%JAVA\_HOME%**

- 敲回车后会自动跳转到JDK目录



# 设置工作空间

- 第一次成功启动Eclipse的时候，需要选择一个目录作为工作空间，用以存放以后开发的Java项目



- 勾选【Use this as the default and do not ask again】之后
- 下次启动Eclipse时就会使用上次选择的目录作为工作空间，不会再次弹框询问

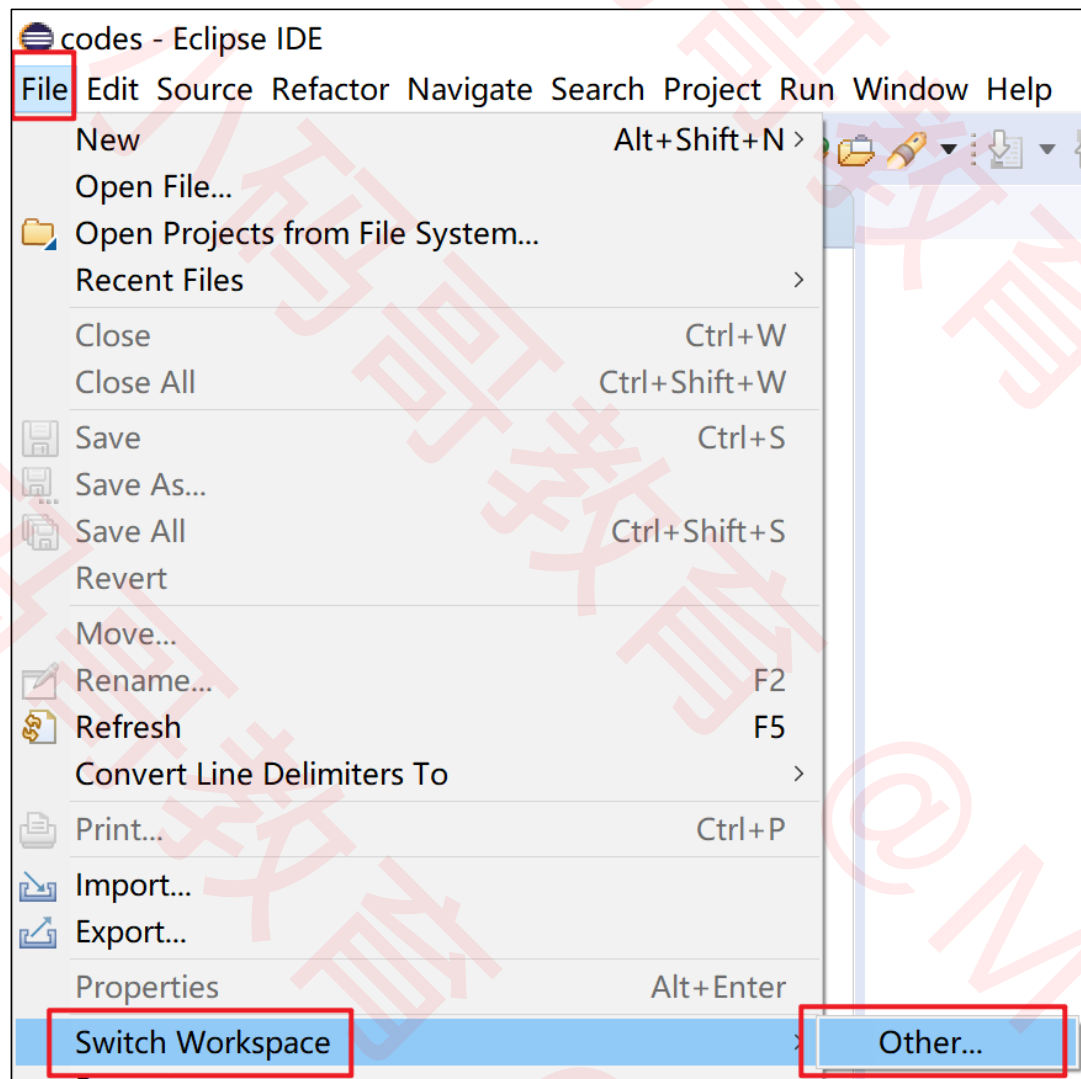


- Eclipse启动完毕后，默认会显示一个欢迎界面，建议永远关闭它
- 去除勾选【Always show Welcome at start up】



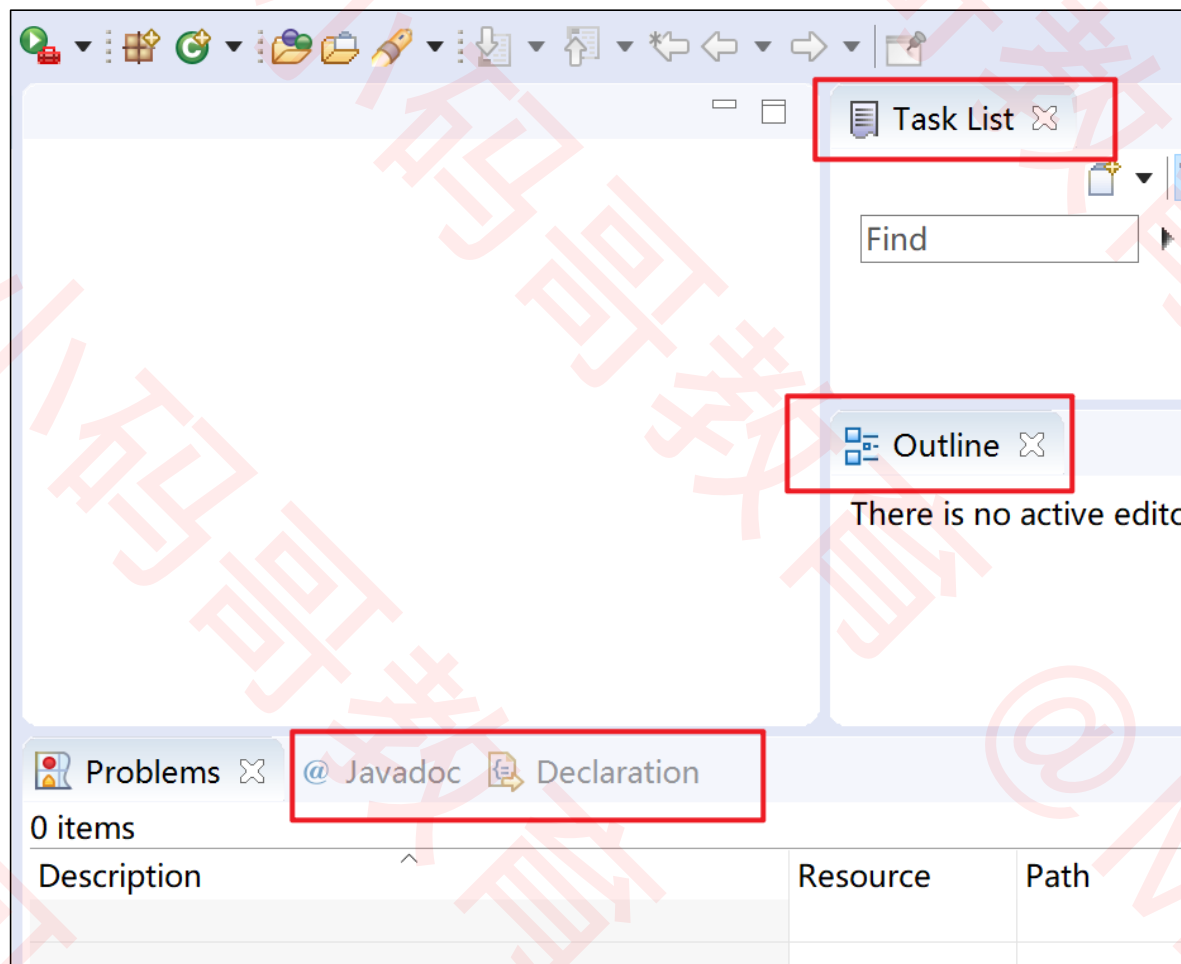
- 点击【Workbench】即可进入主界面

# 切换工作空间



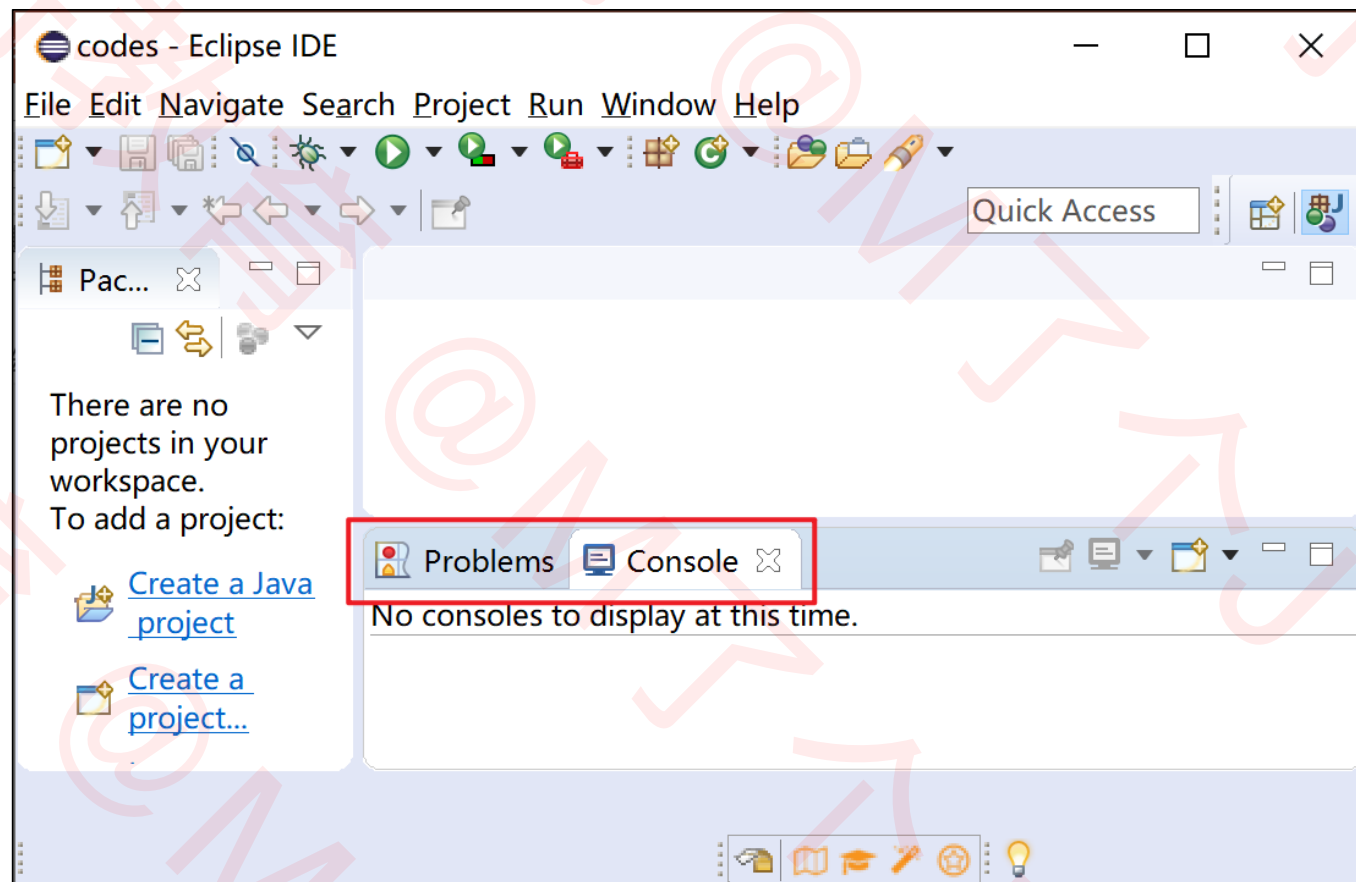
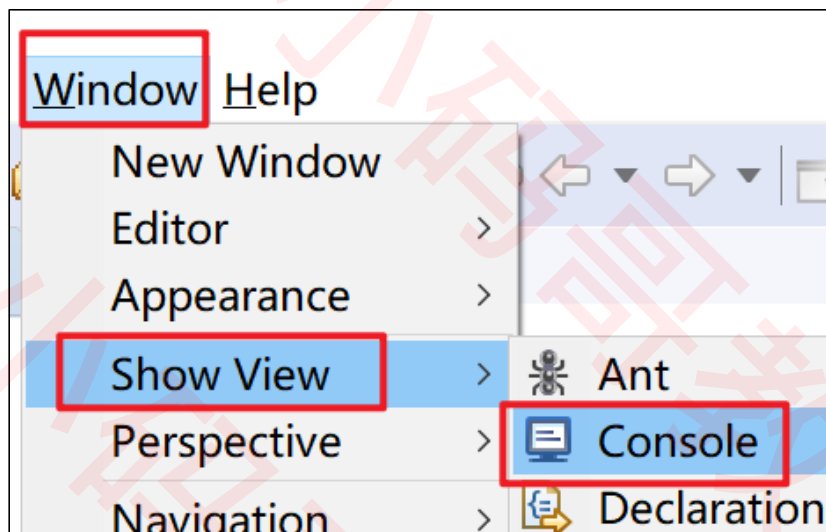
# 关闭一些前期用不上的窗口

- 关闭【Javadoc】、【Declaration】、【Task List】、【Outline】



# 打开Console窗口

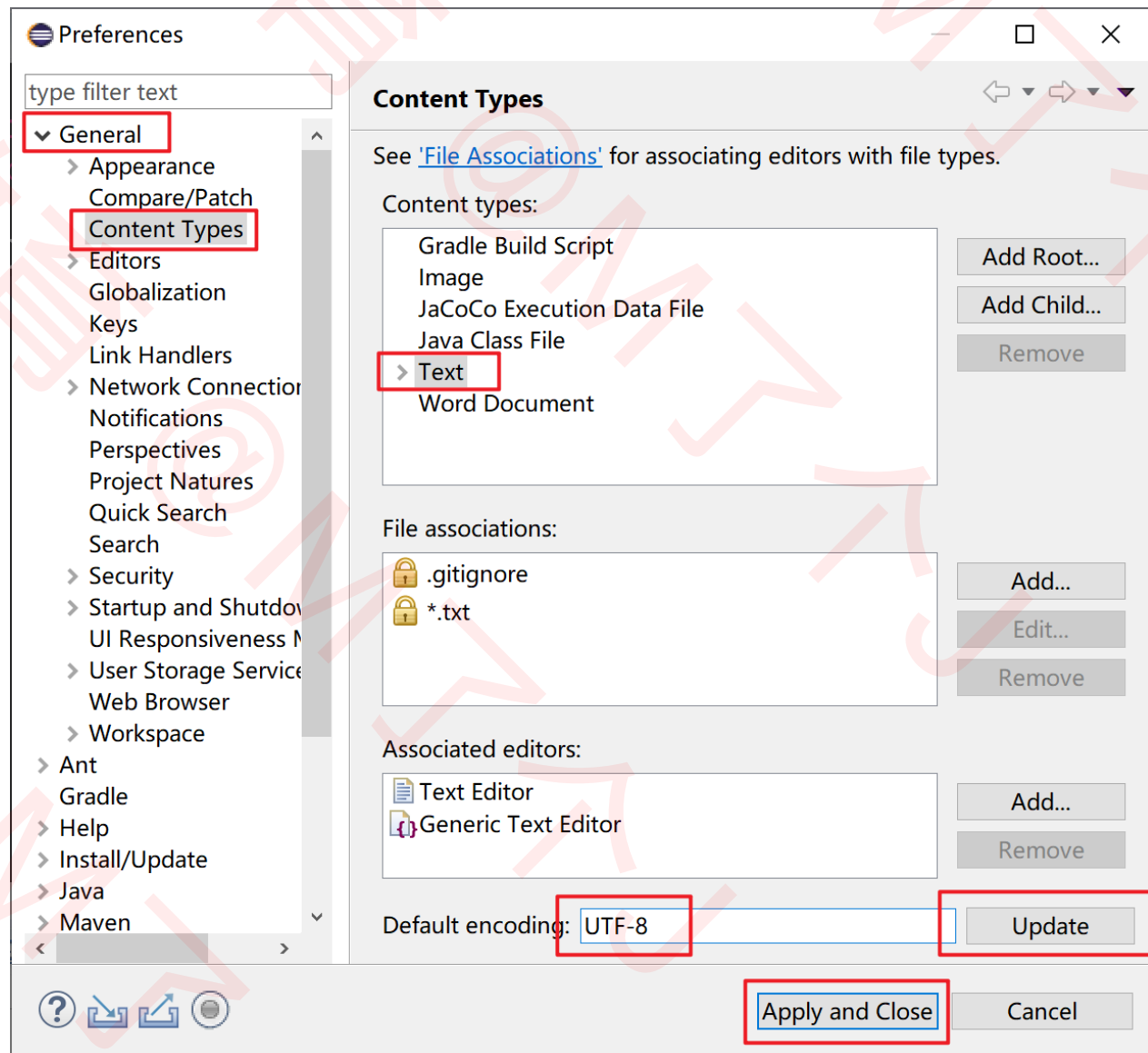
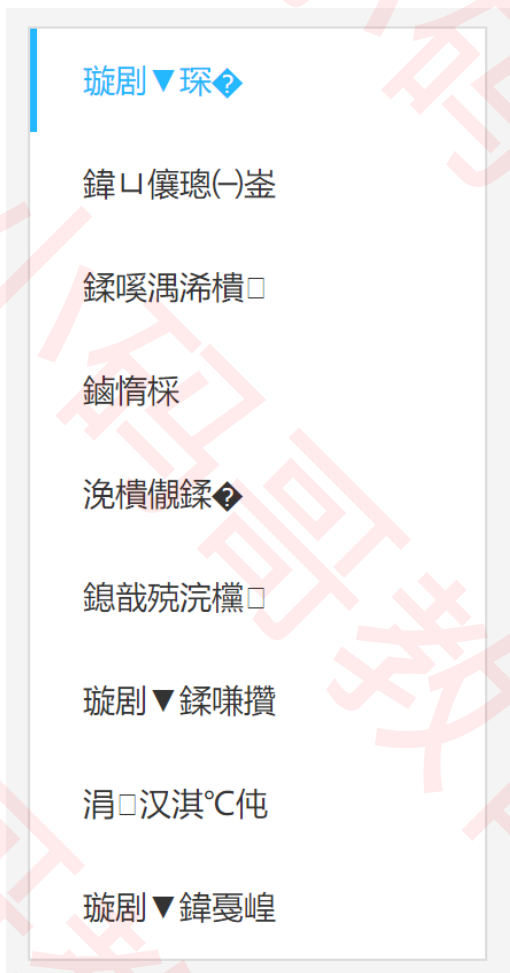
- 【Window】→【Show View】→【Console】

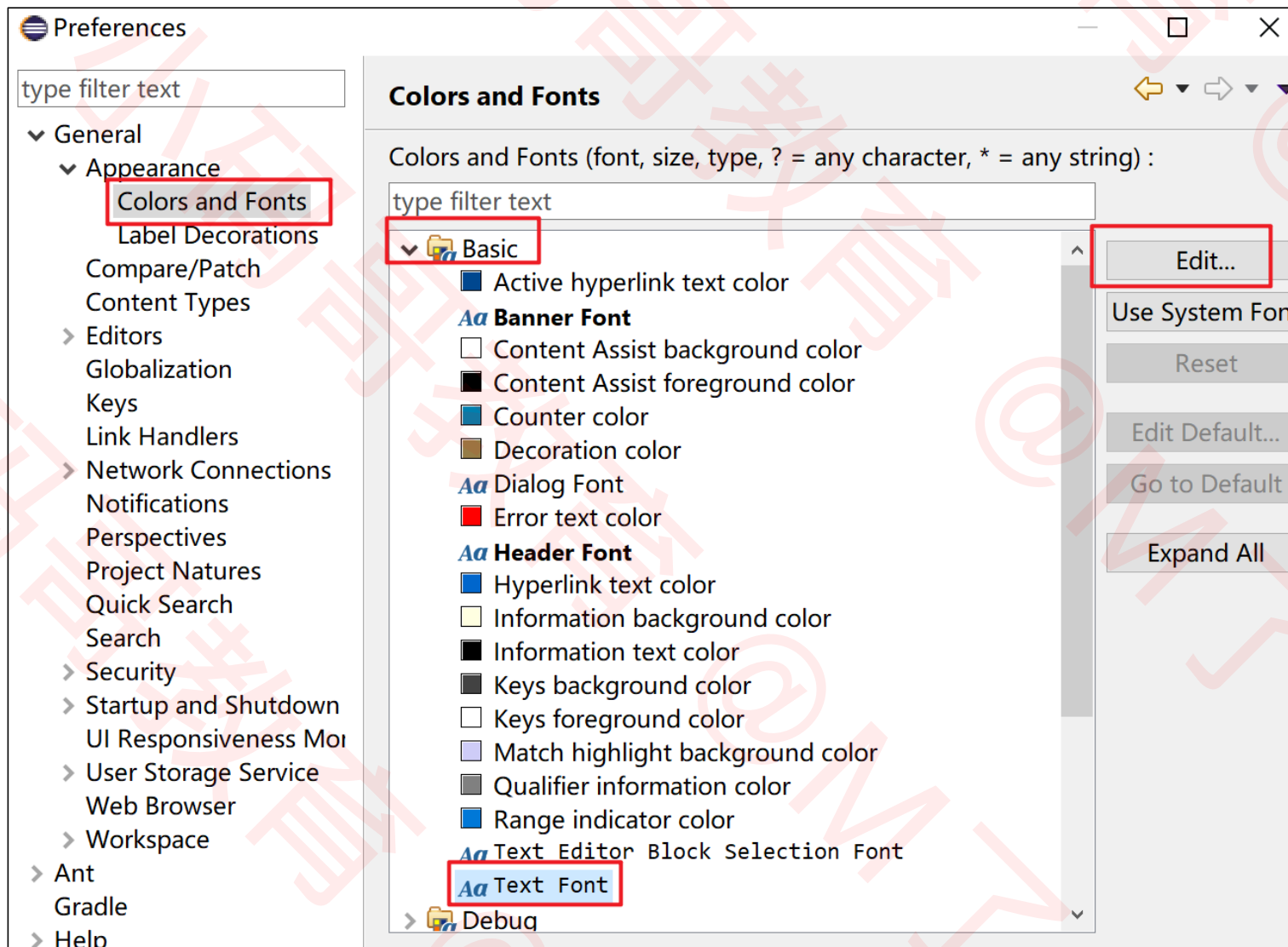


- 最后只留下【Problems】、【Console】窗口即可

# 设置文件编码为UTF-8

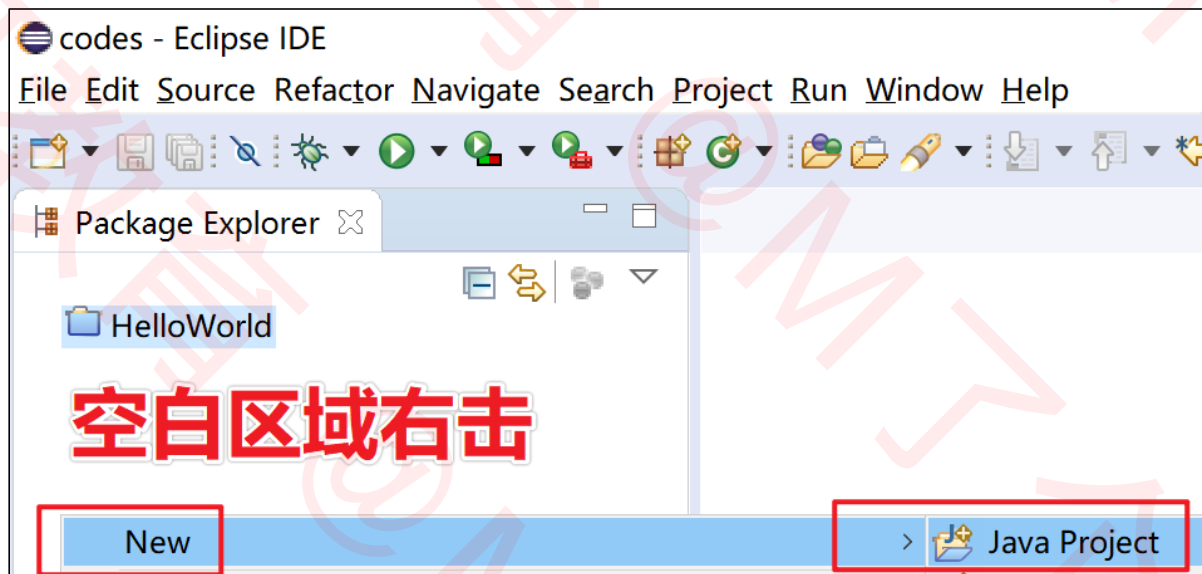
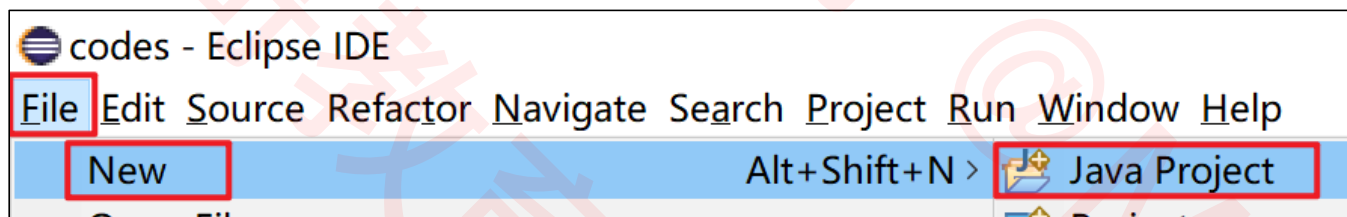
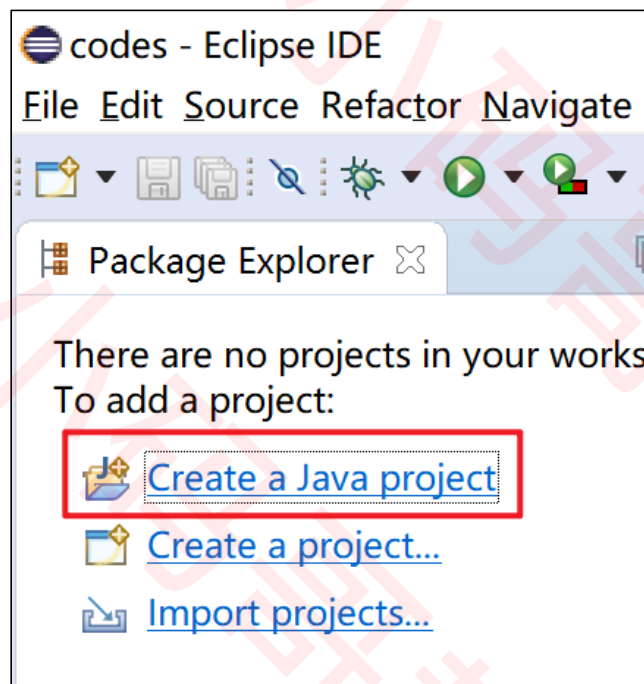
- 文件编码如果设置错误可能会导致文字显示乱码
- 建议使用UTF-8编码，几乎涵盖了世界上所有的文字





# HelloWorld – 新建项目

- 在学习新的编程语言时，我们习惯把第一个项目叫做【HelloWorld】：写一个程序向世界问好





# HelloWorld – 输入项目名称

New Java Project

**Create a Java Project**

Create a Java project in the workspace or in an external location.

Project name: HelloWorld

☒ Use default location

Location: D:\DevTools\Java\codes\HelloWorld [Browse...](#)

JRE

☒ Use an execution environment JRE: JavaSE-1.8

☐ Use a project specific JRE: jdk1.8.0\_231

☐ Use default JRE 'jdk1.8.0\_231' and workspace compiler preferences [Configure JREs...](#)

Project layout

☐ Use project folder as root for sources and class files

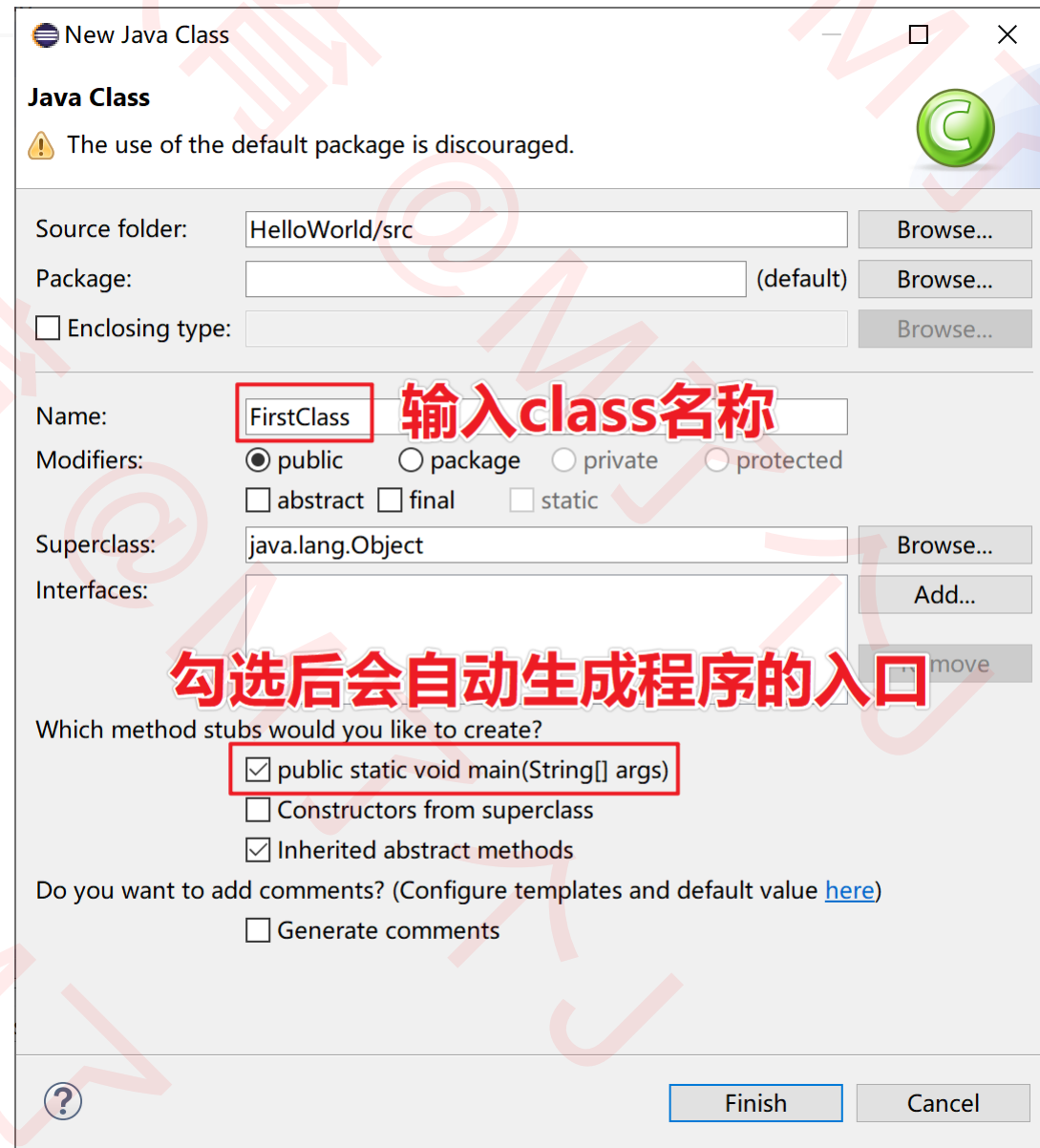
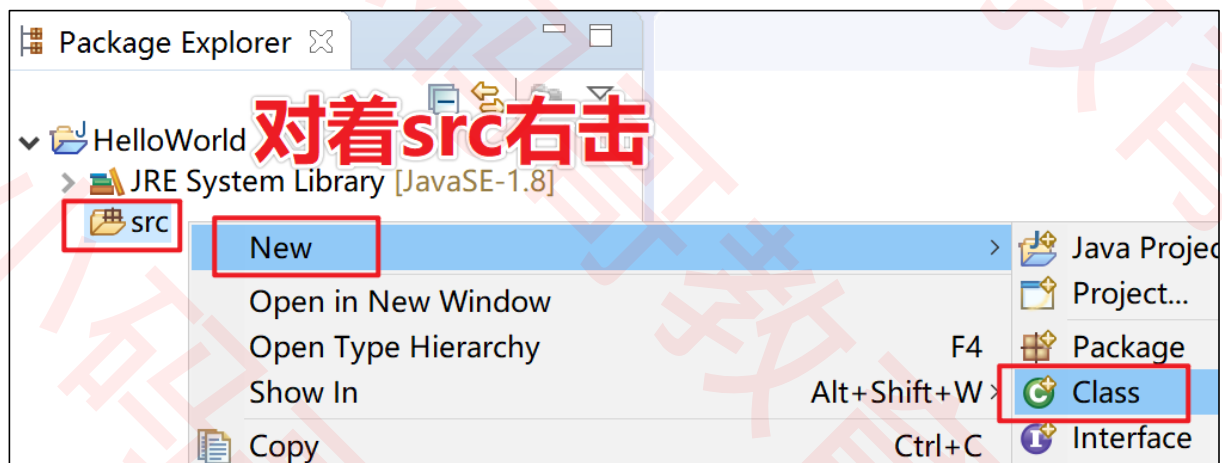
☒ Create separate folders for sources and class files [Configure default...](#)

Working sets

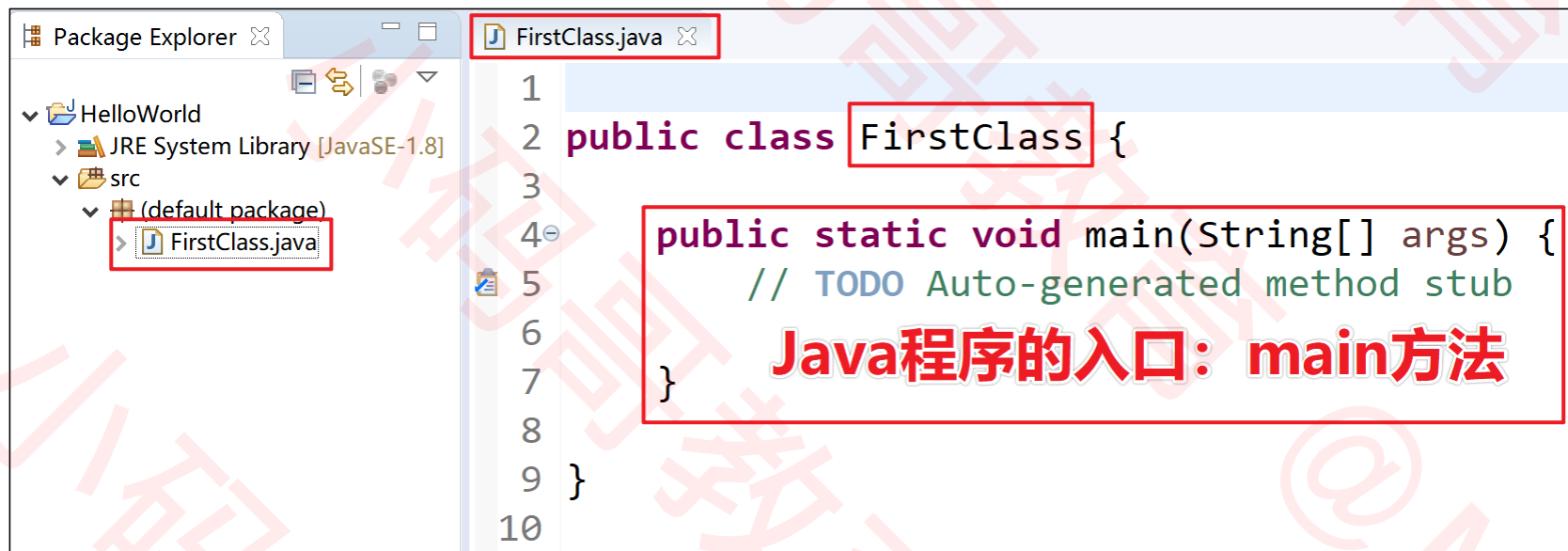
[?](#) [< Back](#) [Next >](#) [Finish](#) [Cancel](#)

# HelloWorld – 新建一个class

- Java程序是以class为基础的，必须要有一个class
- class名称可以自定义，不一定要跟项目名称一致



# HelloWorld – 程序的入口



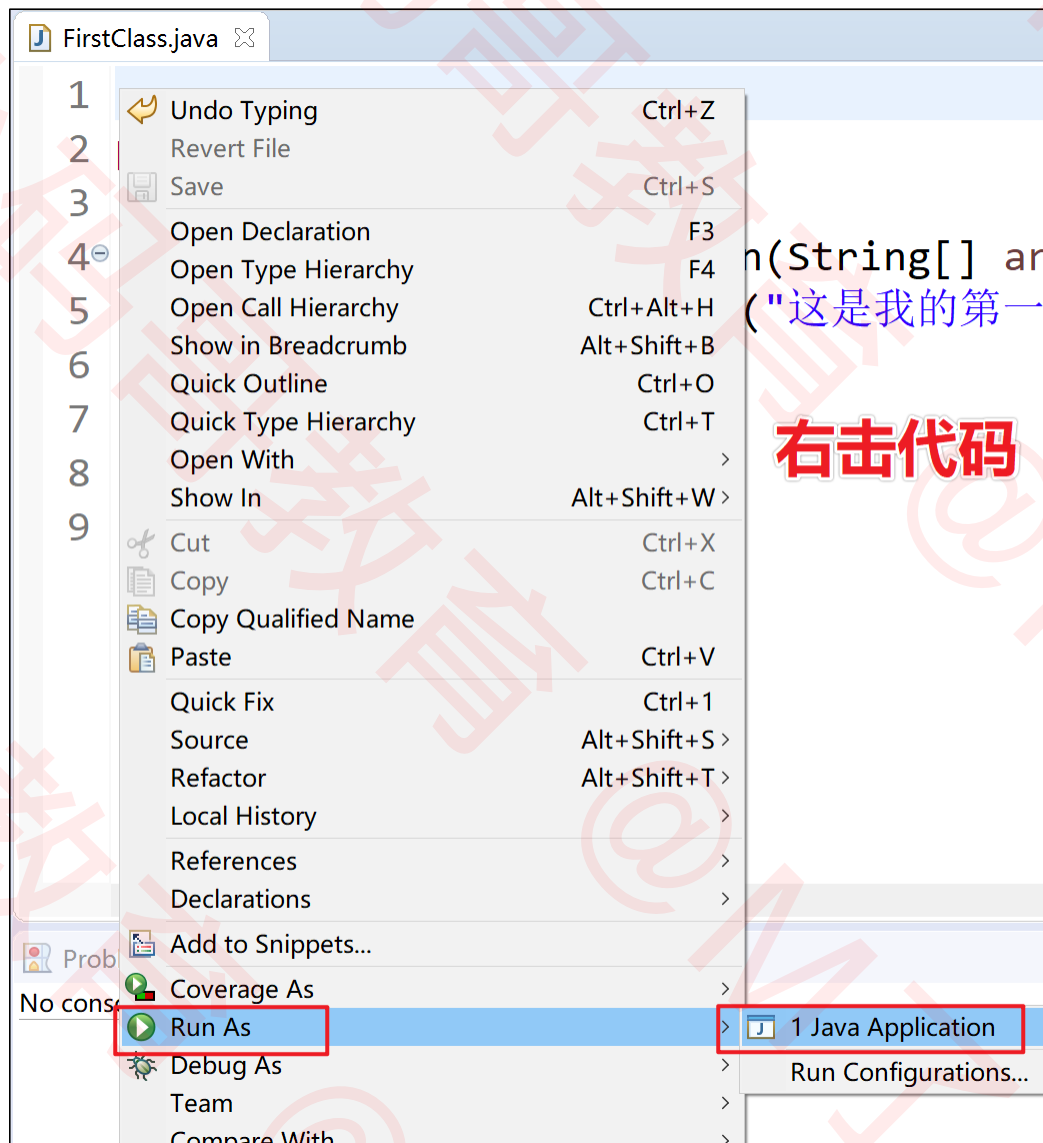
- Eclipse自动生成了一个FirstClass.java文件（Java源代码文件的扩展名是java）
- 也自动在FirstClass.java文件中生成了一些代码
- 第4~7行的代码叫做：main方法，是Java程序的入口
- ✓ Java程序运行之后会自动开始执行main方法中的代码

# HelloWorld – 编写代码

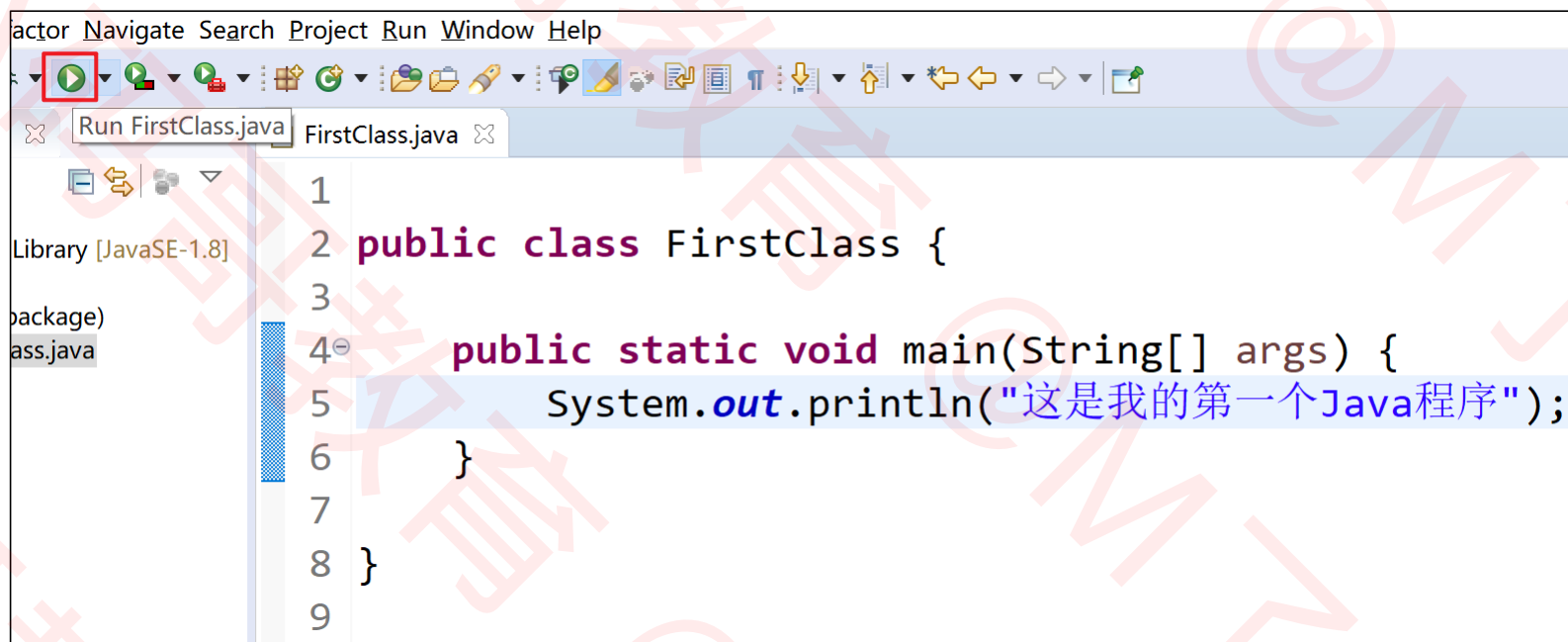
- 接下来在main方法中编写一段代码，在屏幕上输出一段话：这是我的第一个Java程序

```
FirstClass.java
1
2 public class FirstClass {
3
4     public static void main(String[] args) {
5         System.out.println("这是我的第一个Java程序");
6     }
7
8 }
9
```

# HelloWorld – 运行程序1



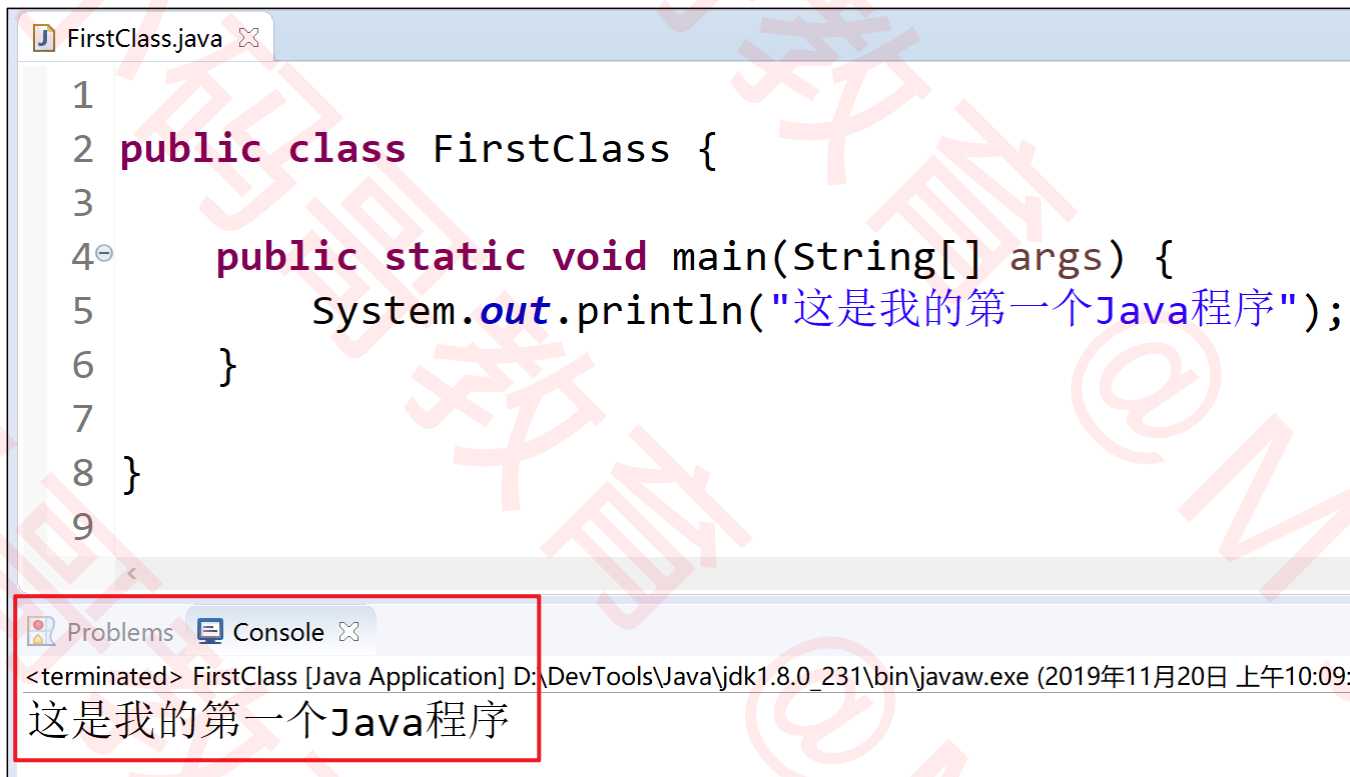
# HelloWorld – 运行程序2



```
actor Navigate Search Project Run Window Help
Run FirstClass.java FirstClass.java
Library [JavaSE-1.8]
package)
ass.java
1
2 public class FirstClass {
3
4     public static void main(String[] args) {
5         System.out.println("这是我的第一个Java程序");
6     }
7
8 }
9
```

# HelloWorld – 运行结果

- 运行之后，可以在Console窗口中看到一段文字：这是我的第一个Java程序



```
1
2 public class FirstClass {
3
4     public static void main(String[] args) {
5         System.out.println("这是我的第一个Java程序");
6     }
7
8 }
9
```

Problems Console

<terminated> FirstClass [Java Application] D:\DevTools\Java\jdk1.8.0\_231\bin\javaw.exe (2019年11月20日 上午10:09:)

这是我的第一个Java程序