

# Lab 01

# Hello Java !

Software Studio  
NetDB, CS, NTHU  
Fall, 2013

# Outline

- Introduction
- Say Hello to Java using command-line
- Setting Up Eclipse
- Say Hello to Java using Eclipse
- Today's Mission

# Outline

- Introduction
- Say Hello to Java using command-line
- Setting Up Eclipse
- Say Hello to Java using Eclipse
- Today's Mission

# Lab ?

- A small lecture by TAs first
- Then get your hands dirty !
  - You have to finish it in the class
- Lab slides are available in the course website
- <http://netdb-ss.appspot.com/>

# Grading

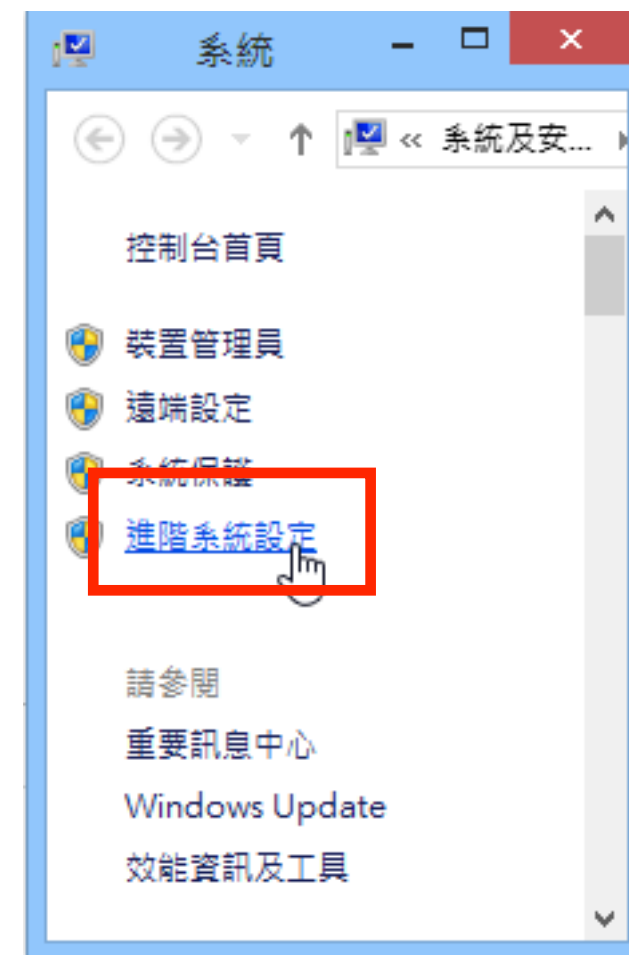
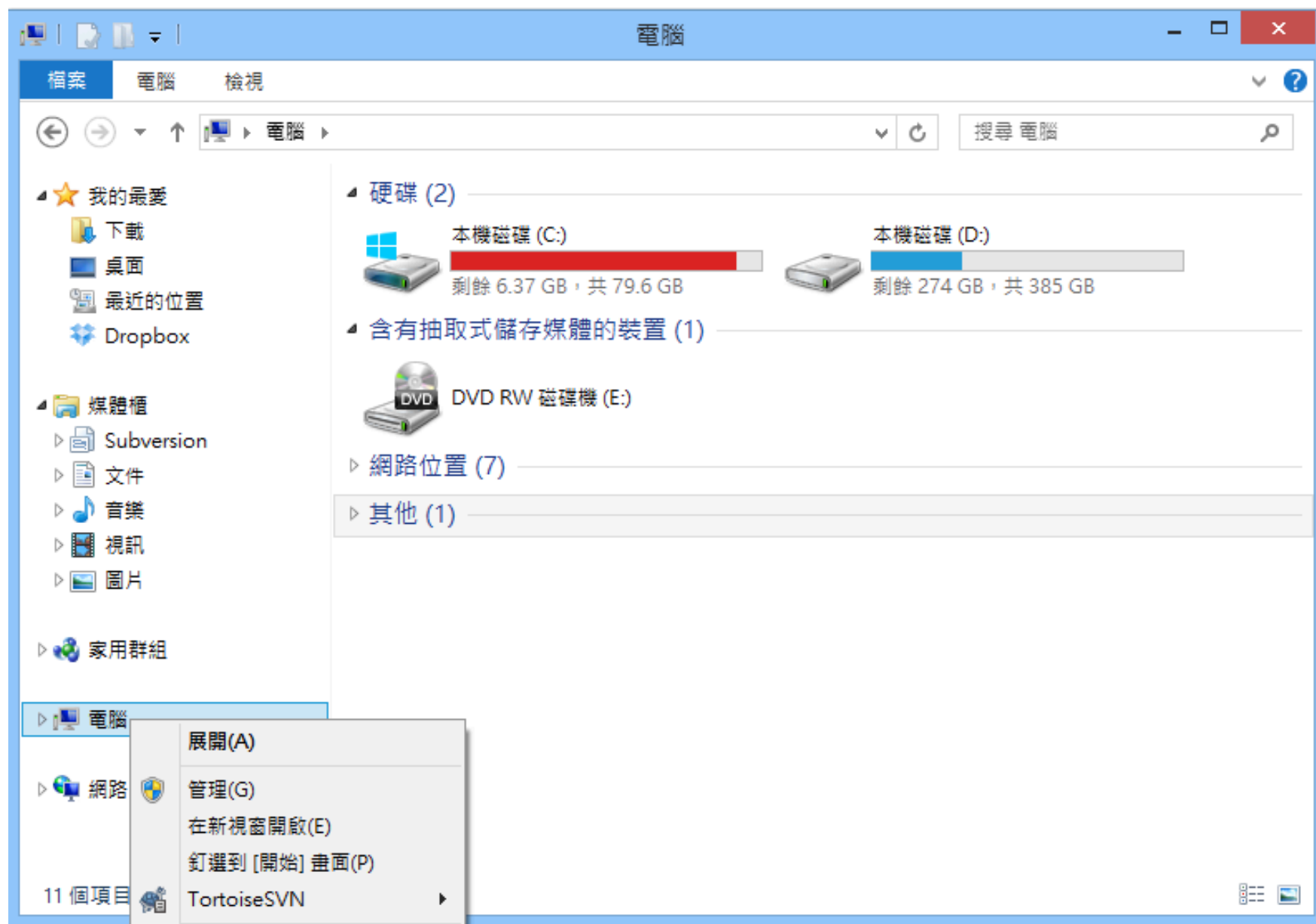
- 2 points for each lab
  - Attendance: 1 point
  - Assignment: 1 point
- Write your student ID on the whiteboard whenever you want to reach TAs

# Outline

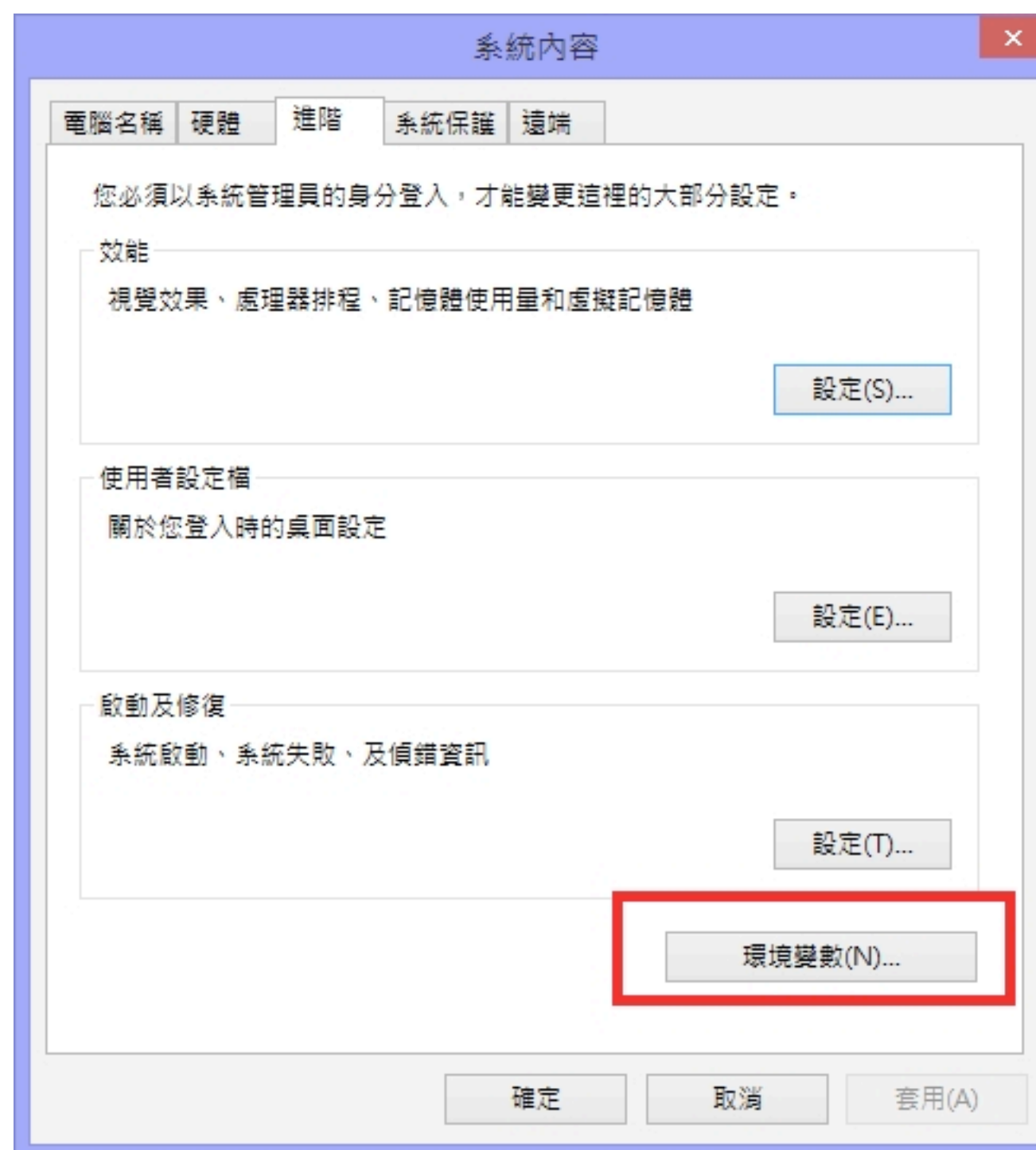
- Introduction
- **Say Hello to Java using command-line**
- Setting Up Eclipse
- Say Hello to Java using Eclipse
- Today's Mission

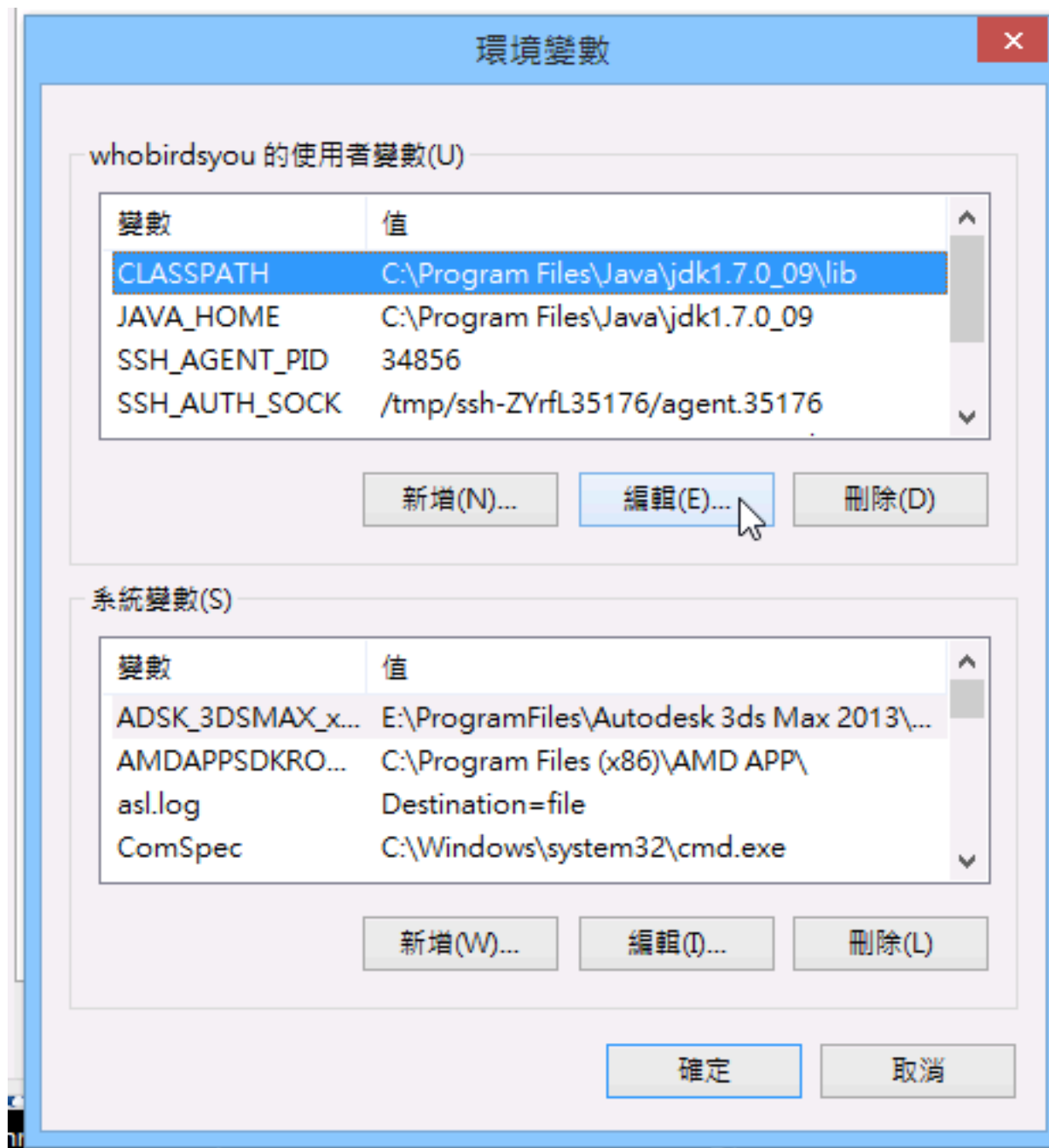
# Running a Java Program

1. Set the path to Java's directory
2. Prepare the .java file
3. Compile the source code
4. Run it !

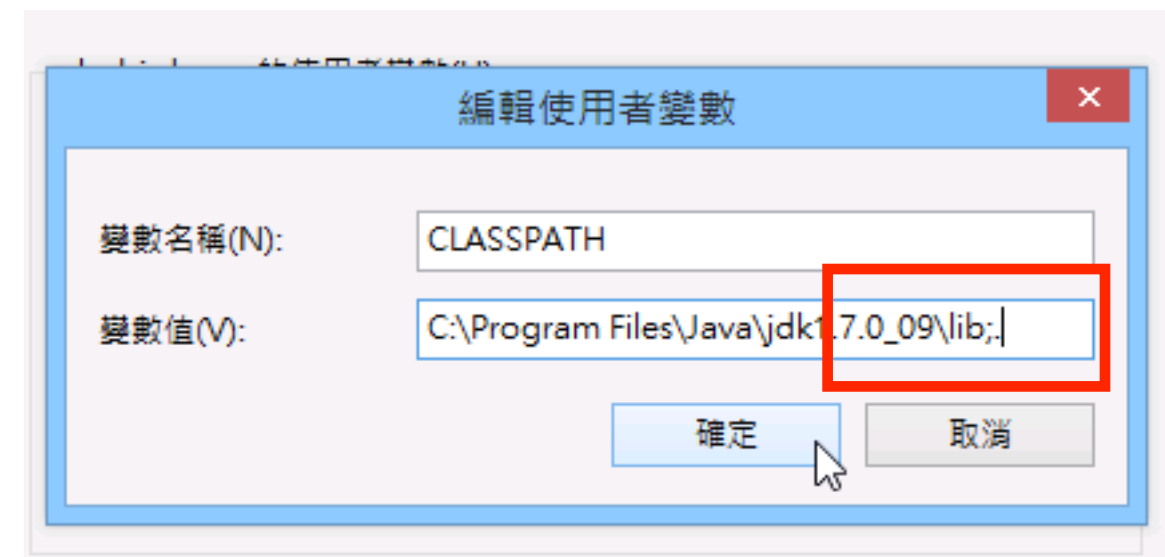




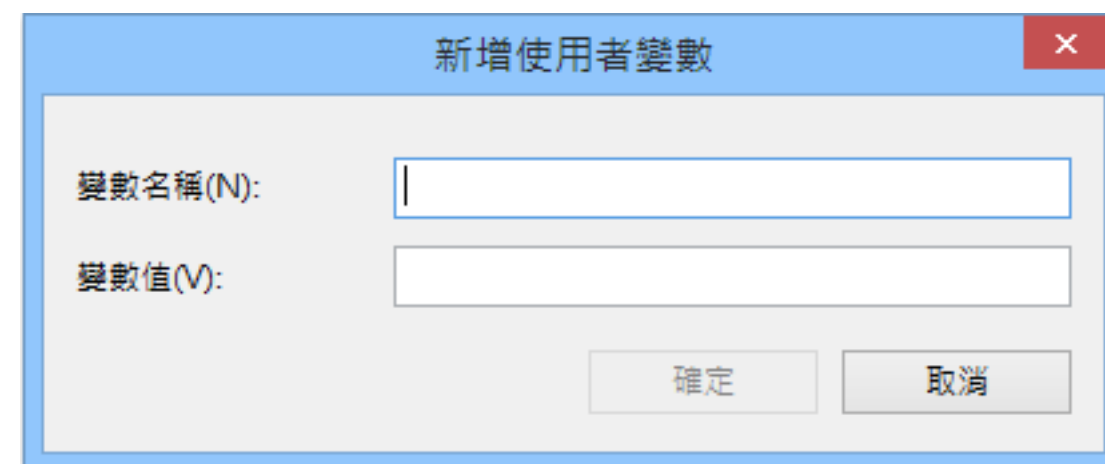
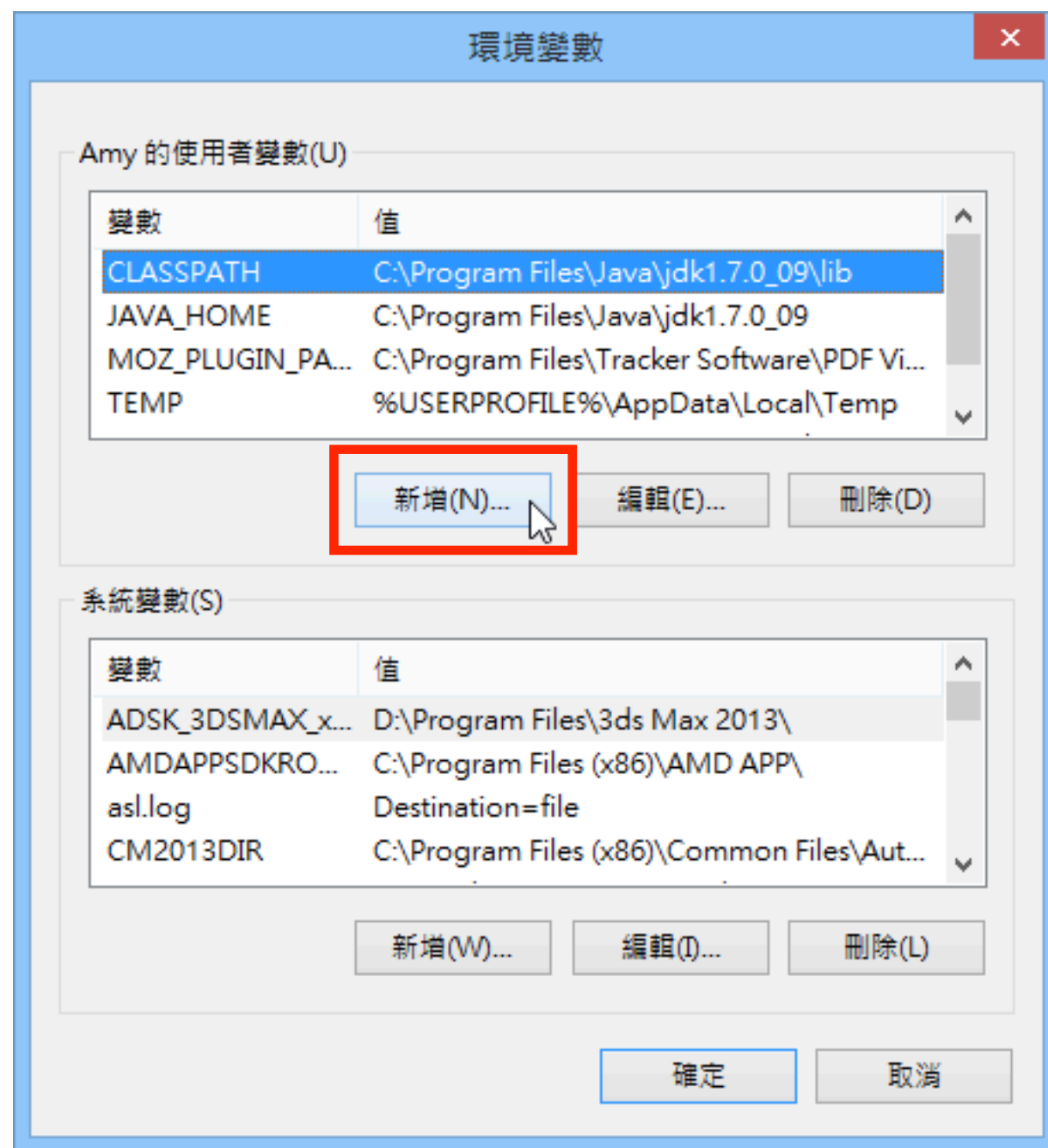


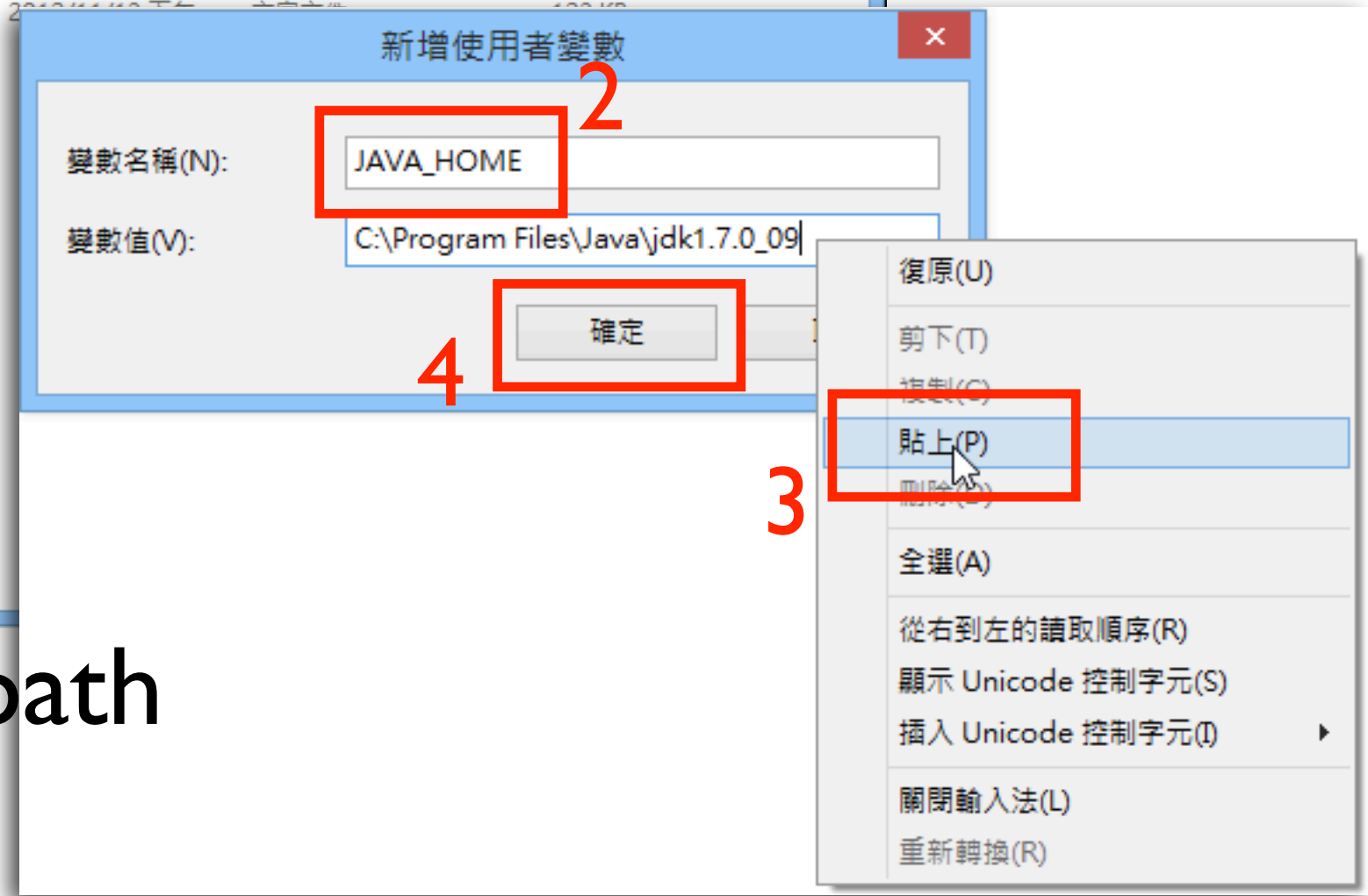
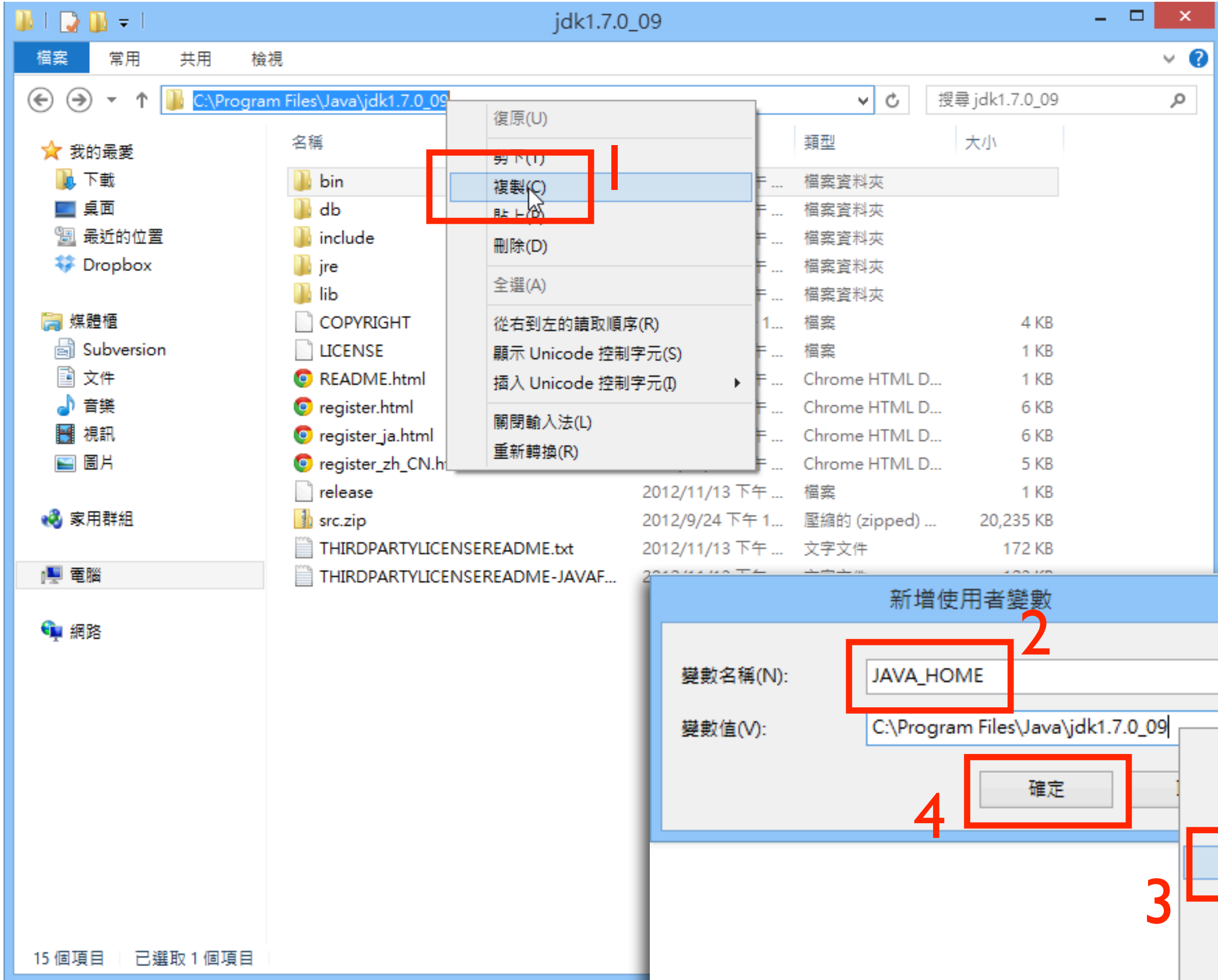


1 Edit the CLASSPATH

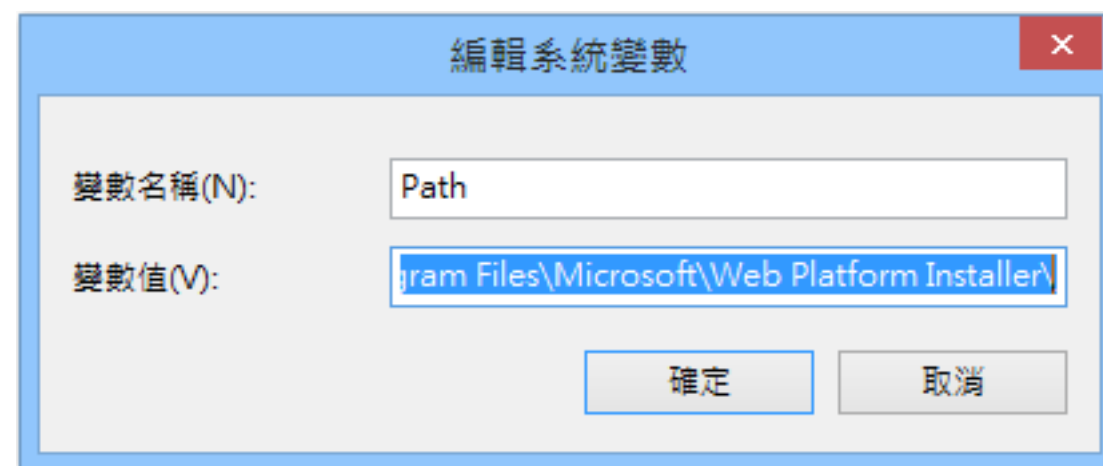
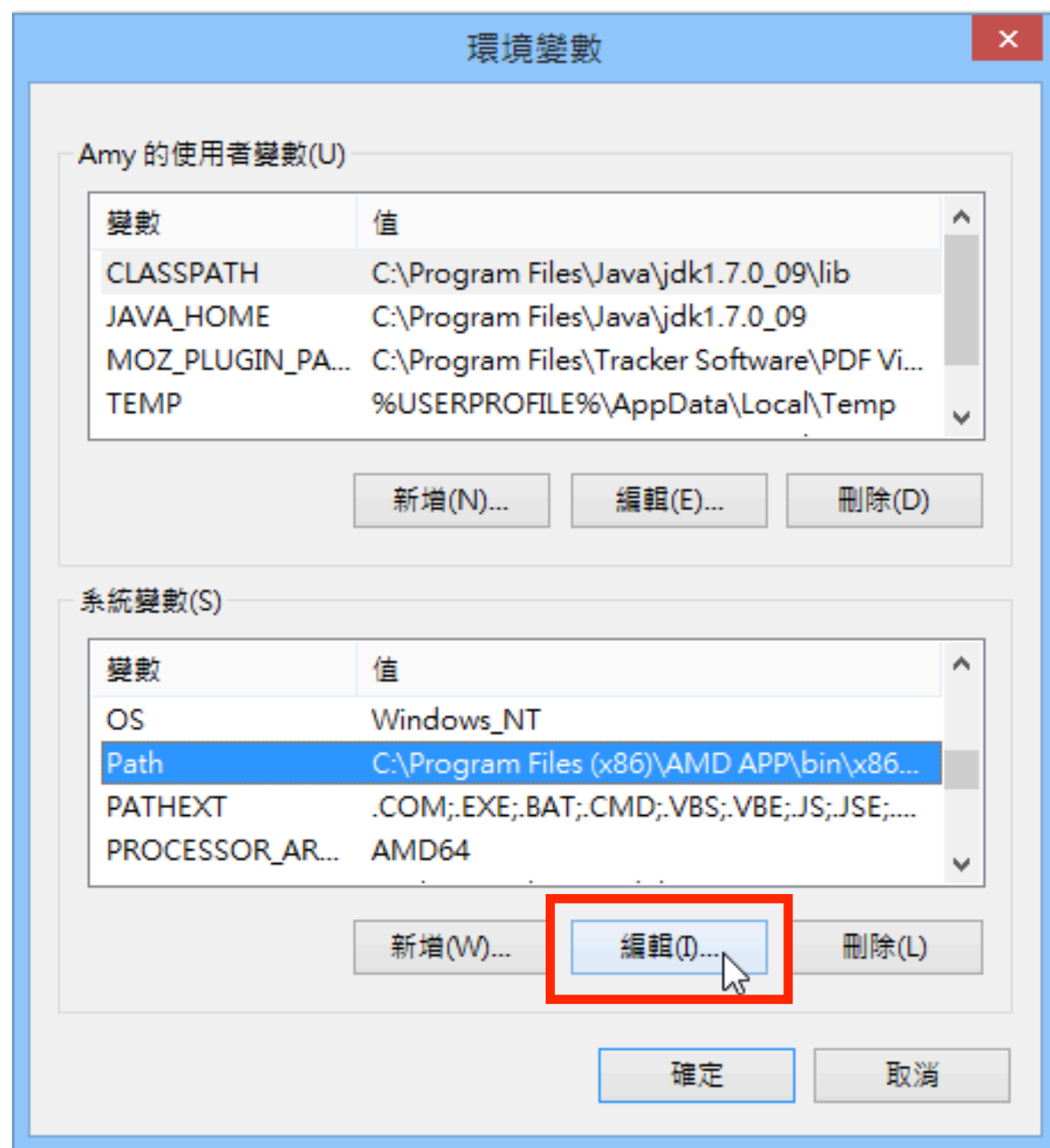


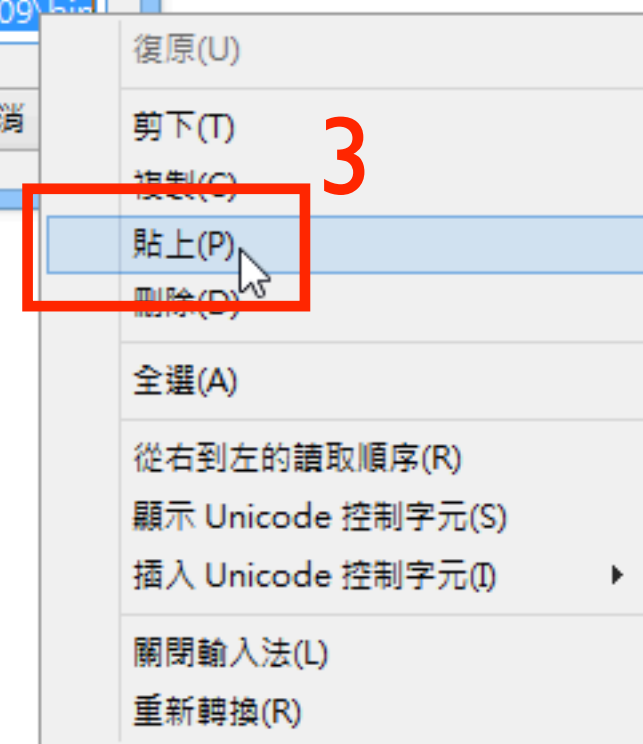
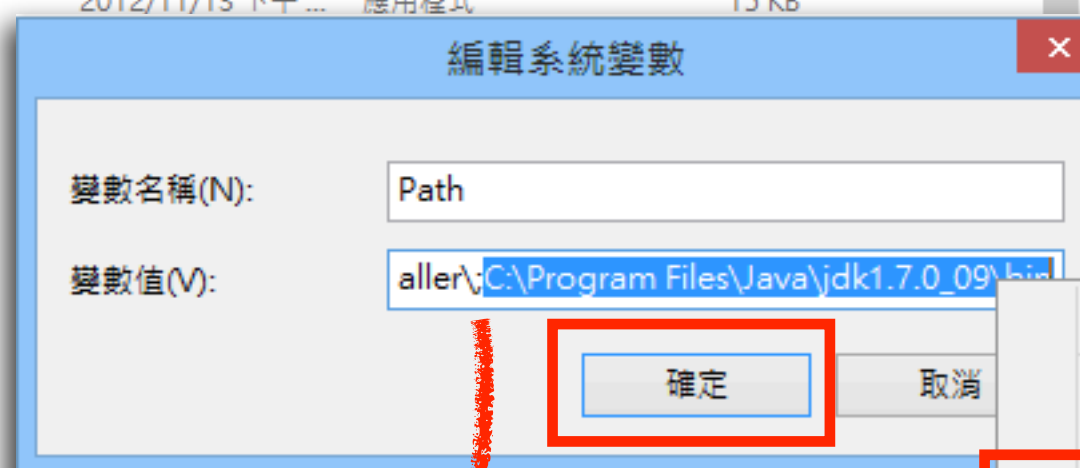
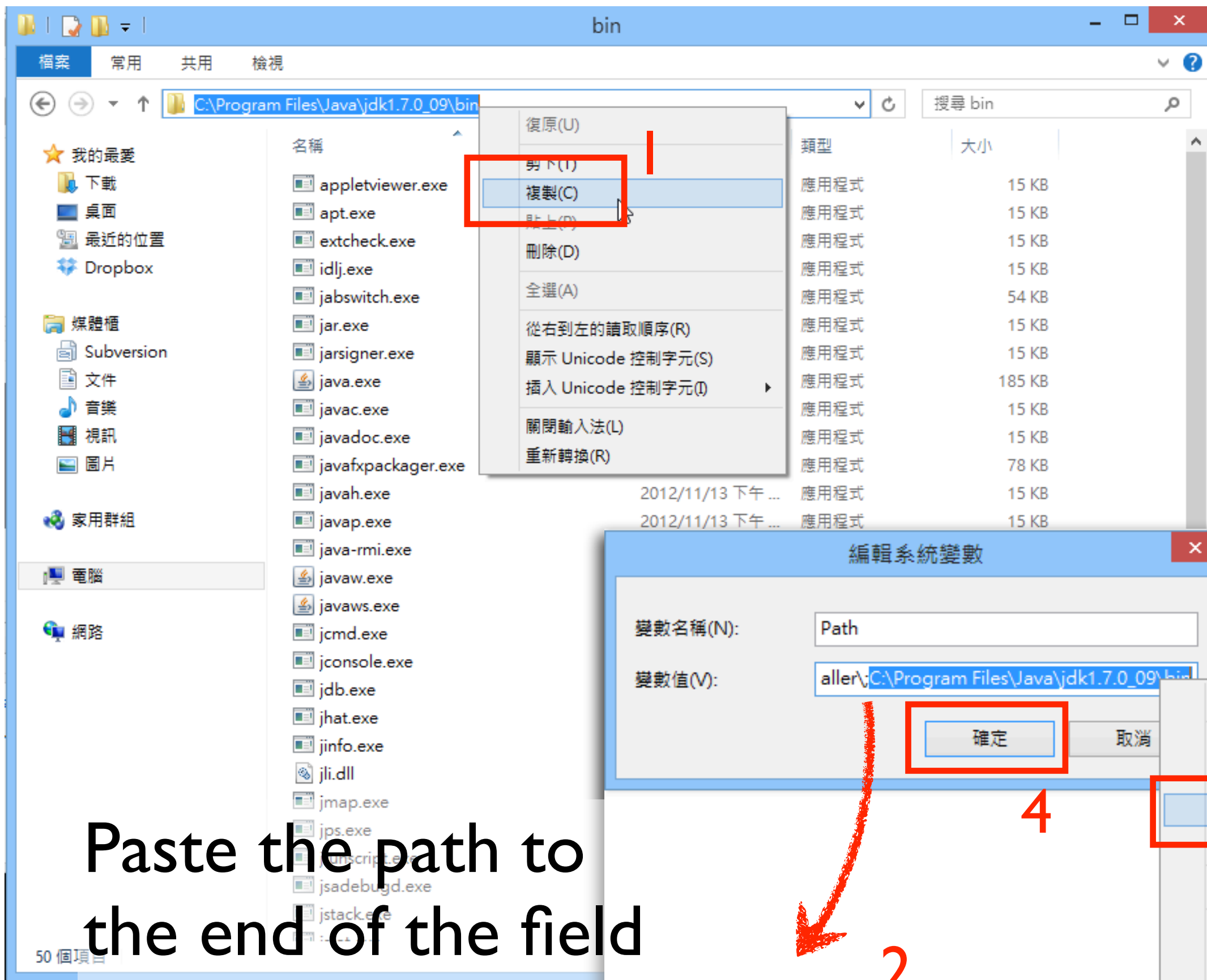
2 Add ';' to the value





Paste the Java jdk path  
to the field





Paste the path to  
the end of the field

(Don't forget to add a **semicolon**  
before you paste the path)

# Check if Java is Ready

Go to command-line and type:

```
javac -version
```

```
java -version
```

```
C:\Users\Keith\Desktop\java>javac -version
javac 1.7.0_09

C:\Users\Keith\Desktop\java>java -version
java version "1.7.0_09"
Java(TM) SE Runtime Environment (build 1.7.0_09-b05)
Java HotSpot(TM) Client VM (build 23.5-b02, mixed mode, sharing)
```

```
public class HelloJava {  
    public static void main(String args[]){  
        System.out.println("Hello Java !");  
    }  
}
```

Create a file named HelloJava.java



# Compile and Run

```
javac HelloJava.java
```

```
java HelloJava
```

```
C:\Users\Keith\Desktop\java>javac HelloJava.java  
  
C:\Users\Keith\Desktop\java>java HelloJava  
Hello Java!  
  
C:\Users\Keith\Desktop\java>
```

# Outline


- Introduction
- Say Hello to Java using command-line
- **Setting Up Eclipse**
- Say Hello to Java using Eclipse
- Today's Mission

# Java




<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

[Java Embedded](#)
[JavaFX](#)
[Java DB](#)
[Web Tier](#)
[Java Card](#)
[Java TV](#)
[New to Java](#)
[Community](#)
[Java Magazine](#)



**Java Platform (JDK) 7u40**



**JDK 7u40 & NetBeans 7.3.1**

**Java Platform, Standard Edition**

**Java SE 7u40**  
 This release includes several new features like Java Mission Control, Deployment Rule Set, support for Retina display on Mac, and Hard Float ABI support on Linux ARM v7. It also includes bug fixes and enhancements.  
[Learn more](#)

**Which Java package do I need?**

- **JDK:** (Java Development Kit). For Java Developers. Includes a complete JRE plus tools for developing, debugging, and monitoring Java applications.
- **Server JRE:** (Server Java Runtime Environment) For deploying Java applications on servers. Includes tools for JVM monitoring and tools commonly required for server applications, but does not include browser integration (the Java plug-in), auto-update, nor an installer. [Learn more](#)
- **JRE:** (Java Runtime Environment). Covers most end-users needs. Contains everything required to run Java applications on your system.

**JDK**

[DOWNLOAD](#)

[JDK 7 Docs](#)

**Server JRE**

[DOWNLOAD](#)

[Server JRE 7 Docs](#)

**JRE**


[DOWNLOAD](#)

[JRE 7 Docs](#)

[NetBeans IDE](#)
[Java Mission Control](#)

**Java Resources**


















[Java APIs](#)
[Technical Articles](#)
[Demos and Videos](#)
[Forums](#)
[Java Magazine](#)
[Java.net](#)
[Developer Training](#)
[Tutorials](#)
[Java.com](#)



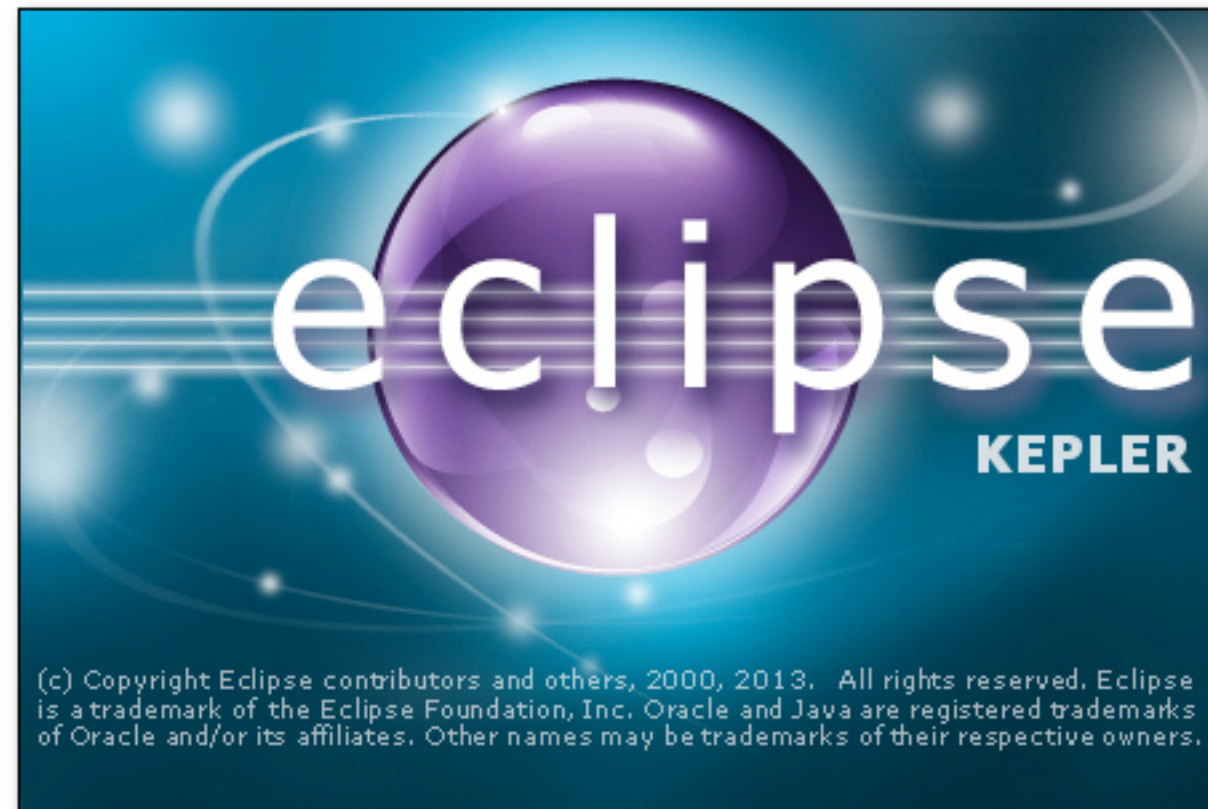
### Java SE Development Kit 7u40

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

☐ Accept License Agreement ☒ Decline License Agreement

Product / File Description	File Size	Download
Linux ARM v6/v7 VFP Hard Float ABI	67.62 MB	 <a href="#">jdk-7u40-linux-arm-vfp-hflt.tar.gz</a>
Linux ARM v6/v7 VFP Soft Float ABI	67.62 MB	 <a href="#">jdk-7u40-linux-arm-vfp-sflt.tar.gz</a>
Linux x86	115.55 MB	 <a href="#">jdk-7u40-linux-i586.rpm</a>
Linux x86	132.83 MB	 <a href="#">jdk-7u40-linux-i586.tar.gz</a>
Linux x64	116.83 MB	 <a href="#">jdk-7u40-linux-x64.rpm</a>
Linux x64	131.63 MB	 <a href="#">jdk-7u40-linux-x64.tar.gz</a>
Mac OS X x64	183.35 MB	 <a href="#">jdk-7u40-macosx-x64.dmg</a>
Solaris x86 (SVR4 package)	139.84 MB	 <a href="#">jdk-7u40-solaris-i586.tar.Z</a>
Solaris x86	95.29 MB	 <a href="#">jdk-7u40-solaris-i586.tar.gz</a>
Solaris x64 (SVR4 package)	24.43 MB	 <a href="#">jdk-7u40-solaris-x64.tar.Z</a>
Solaris x64	16.17 MB	 <a href="#">jdk-7u40-solaris-x64.tar.gz</a>
Solaris SPARC (SVR4 package)	139.06 MB	 <a href="#">jdk-7u40-solaris-sparc.tar.Z</a>
Solaris SPARC	98.07 MB	 <a href="#">jdk-7u40-solaris-sparc.tar.gz</a>
Solaris SPARC 64-bit (SVR4 package)	23.74 MB	 <a href="#">jdk-7u40-solaris-sparcv9.tar.Z</a>
Solaris SPARC 64-bit	18.18 MB	 <a href="#">jdk-7u40-solaris-sparcv9.tar.gz</a>
Windows x86	123.46 MB	 <a href="#">jdk-7u40-windows-i586.exe</a>
Windows x64	125.25 MB	 <a href="#">jdk-7u40-windows-x64.exe</a>

# Eclipse





# Let's Get Eclipse





eclipse



Google 搜尋

好手氣

If you are too lazy, click this [link](#).



[網頁](#)[圖片](#)[地圖](#)[更多 ▾](#)[搜尋工具](#)

約有 153,000,000 項結果 (搜尋時間：0.14 秒)

### [Eclipse Downloads](#)

[www.eclipse.org/downloads/](http://www.eclipse.org/downloads/) ▾ 翻譯這個網頁

超過 10 筆 - Package Solutions.

Eclipse IDE for Java EE Developers 246 MB Downloaded 1,118,422 Times

Eclipse IDE for Java Developers 151 MB Downloaded 500,096 Times

[Developer Builds - Eclipse Classic 4.2.2 - Eclipse IDE for Java EE ... - Mac OS X](#)

### [Eclipse - The Eclipse Foundation open source community website.](#)

[www.eclipse.org/](http://www.eclipse.org/) ▾ 翻譯這個網頁

A project aiming to provide a universal toolset for development. Open Source IDE, mostly provided in Java, but the development language is independent and ...

### [Eclipse簡介與教學 - 計中首頁](#)

[www.cc.ntu.edu.tw/chinese/epaper/0020/20120320\\_2003.html](http://www.cc.ntu.edu.tw/chinese/epaper/0020/20120320_2003.html) ▾

2012/3/20 - 首頁 > 專題報導. 專題報導. Eclipse簡介與教學. 作者：張仁峰/ 臺灣大學工程科學及海洋工程研究所研究生. 在開發軟體過程中，選擇好的開發工具 ...

### [Eclipse - 维基百科，自由的百科全书](#)

[zh.wikipedia.org/zh-tw/Eclipse](http://zh.wikipedia.org/zh-tw/Eclipse) ▾ 轉為繁體網頁

Eclipse是著名的跨平台开源集成开发环境（IDE）。最初主要用来Java语言开发，目前亦

# Eclipse Downloads

Packages Developer Builds

Follow @EclipseFdn

4.5 萬

Eclipse Kepler (4.3) Packages for Windows



## Eclipse Standard 4.3, 198 MB

Downloaded 2,040,178 Times [Other Downloads](#)

The Eclipse Platform, and all the tools needed to develop and debug it: Java and Plug-in Development Tooling, Git and CVS...



Windows 32 Bit

Windows 64 Bit

## Package Solutions

Filter Packages



## Eclipse IDE for Java EE Developers, 246 MB

Downloaded 1,118,422 Times

Tools for Java developers creating Java EE and Web applications, including a Java IDE, tools for Java EE, JPA, JSF, Mylyn...



Windows 32 Bit

Windows 64 Bit



## Eclipse IDE for Java Developers, 151 MB

Downloaded 500,096 Times

The essential tools for any Java developer, including a Java IDE, a CVS client, Git client, XML Editor, Mylyn, Maven integration...



Windows 32 Bit

Windows 64 Bit



## Eclipse IDE for C/C++ Developers, 140 MB

Downloaded 297,585 Times

An IDE for C/C++ developers with Mylyn integration.



Windows 32 Bit

Windows 64 Bit

## Eclipse IDE for Java and Report Developers, 277 MB



Windows 32 Bit

## Related Links

- [Compare & Combine Packages](#)
- [Eclipse Indigo \(3.7\)](#)
- [Eclipse Juno \(4.2\)](#)
- [Install Guide](#)
- [Documentation](#)
- [Updating Eclipse](#)
- [Forums](#)
- [Older Versions](#)

## Hint:

You will need a [Java runtime environment \(JRE\)](#) to use Eclipse (Java SE 6 or greater is recommended). All downloads are provided under the terms and conditions of the [Eclipse Foundation Software User Agreement](#) unless otherwise specified.

Downloads Home >>

- Source code
- More Packages

## Give Back to Eclipse

\$5

\$15

\$25

Donate \$35 or more and  
Become a Friend of Eclipse!

PayPal

The Eclipse Foundation is a not-for-profit organization, not a charitable organization, so we are unable to provide charitable tax receipts.

Amounts shown are in US dollars.

## Eclipse downloads - mirror selection

All downloads are provided under the terms and conditions of the [Eclipse Foundation Software User Agreement](#) unless otherwise specified.

Download eclipse-standard-kepler-R-win32.zip from:



[\[Taiwan\] Providence University \(http\)](#)

Checksums: [\[MD5\]](#) [\[SHA1\]](#) [BitTorrent](#)

...or pick a mirror site below.

## Get It Faster Here



### Obeo

Download the latest Eclipse easily and discover how to create your OWN modeling solutions.



### SpringSource

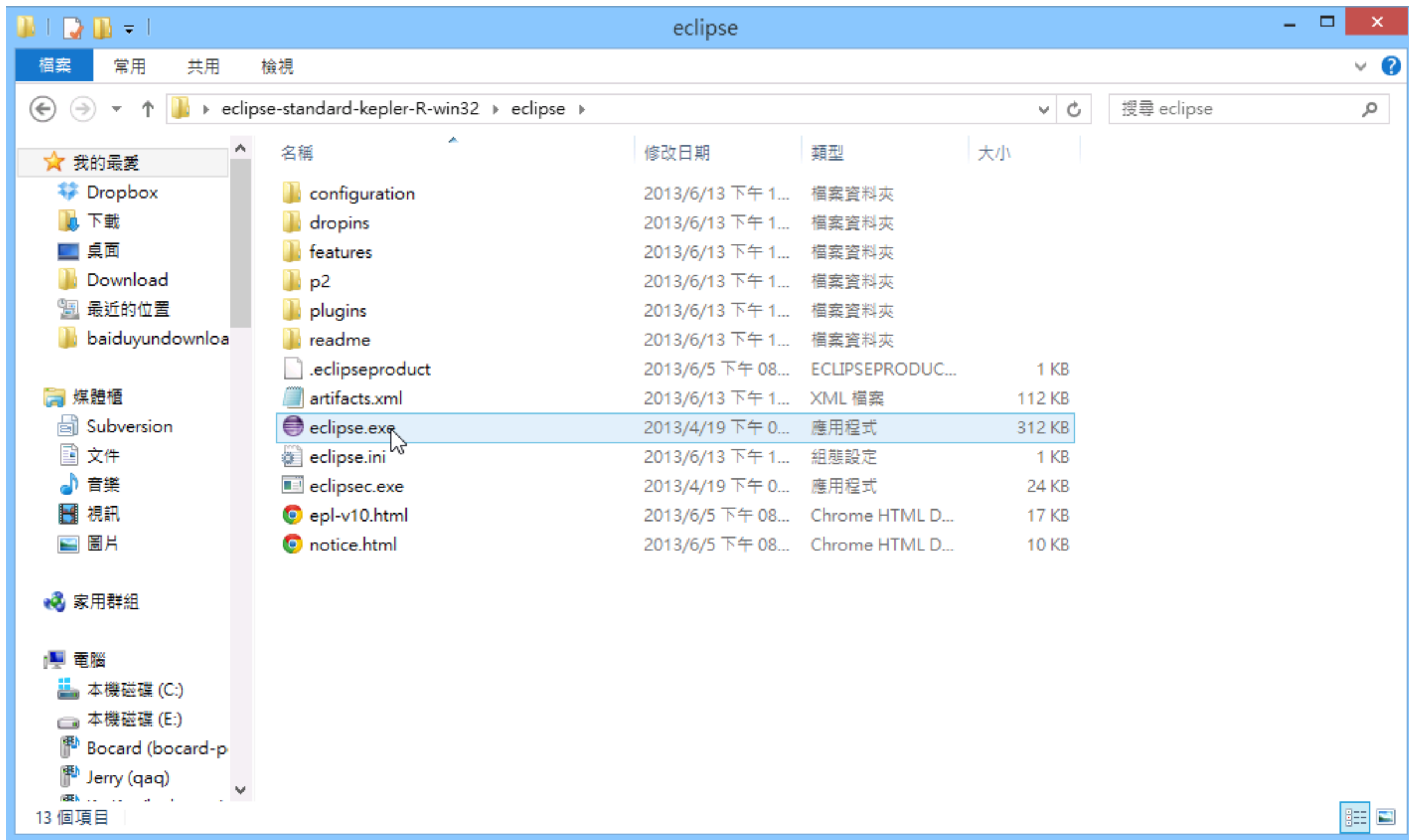
Rapid downloads of Eclipse packages. Free downloads of SpringSource Tool Suite for Spring, AspectJ, Groovy and Grails.

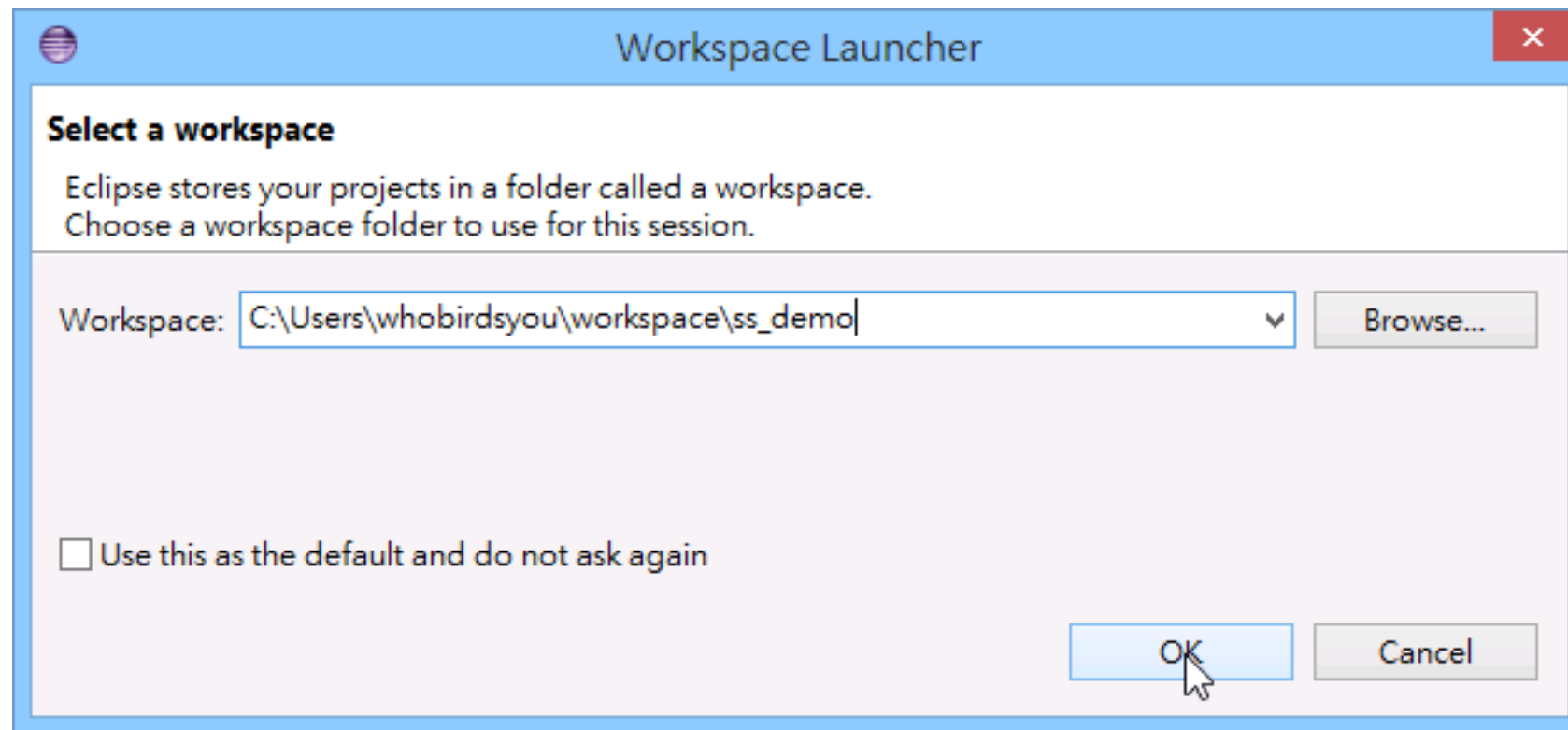
## Other options for this file

- [HTTP mirrors only \(xml\)](#)
- [FTP mirrors only \(xml\)](#)
- [All mirrors \(xml\)](#)
- [Direct link to file](#) (download starts immediately from best mirror)

## Related Links

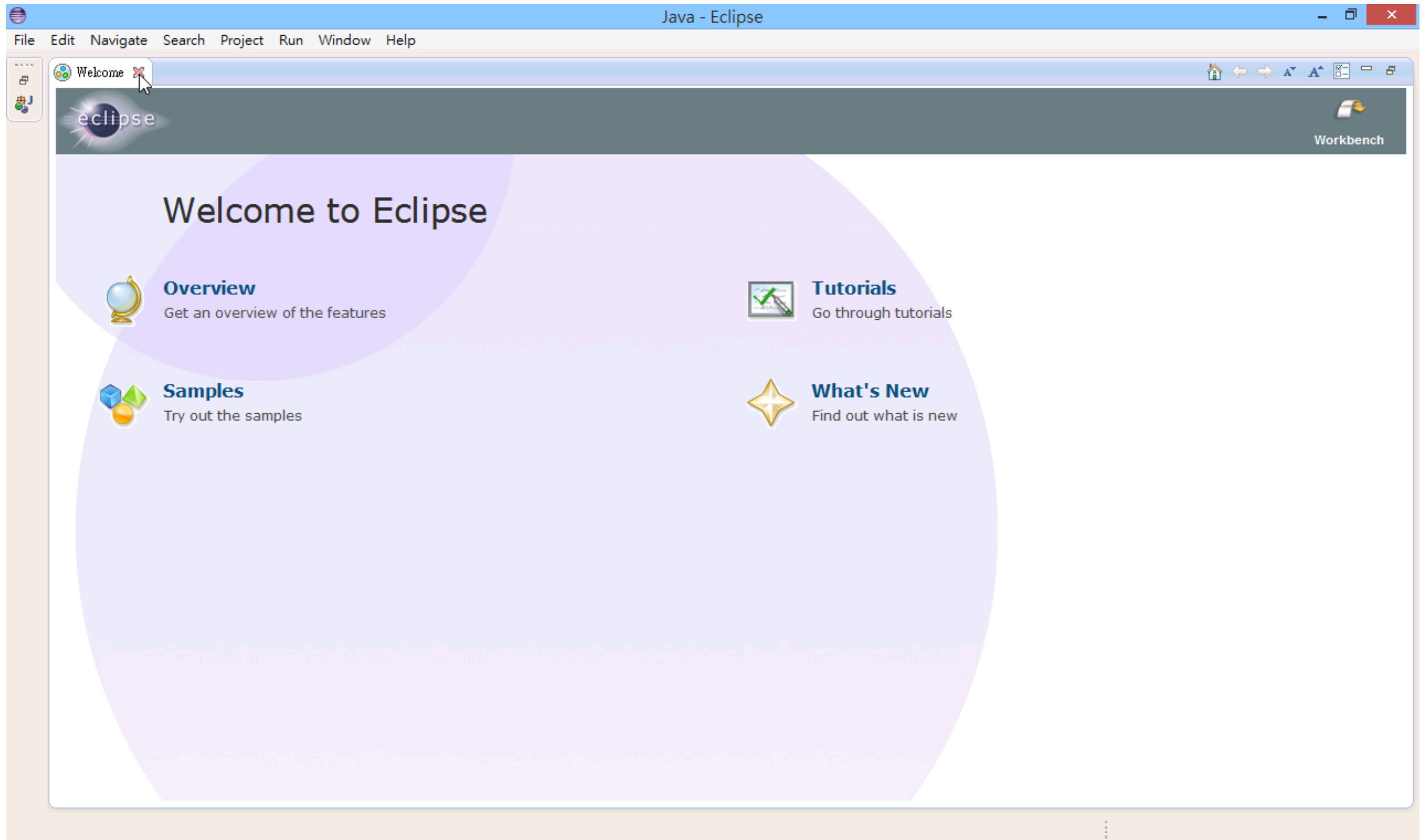
- [Friends of Eclipse](#)
- [Becoming a mirror site](#)
- [Updating and installing Eclipse components](#)
- [Eclipse newsgroups](#)

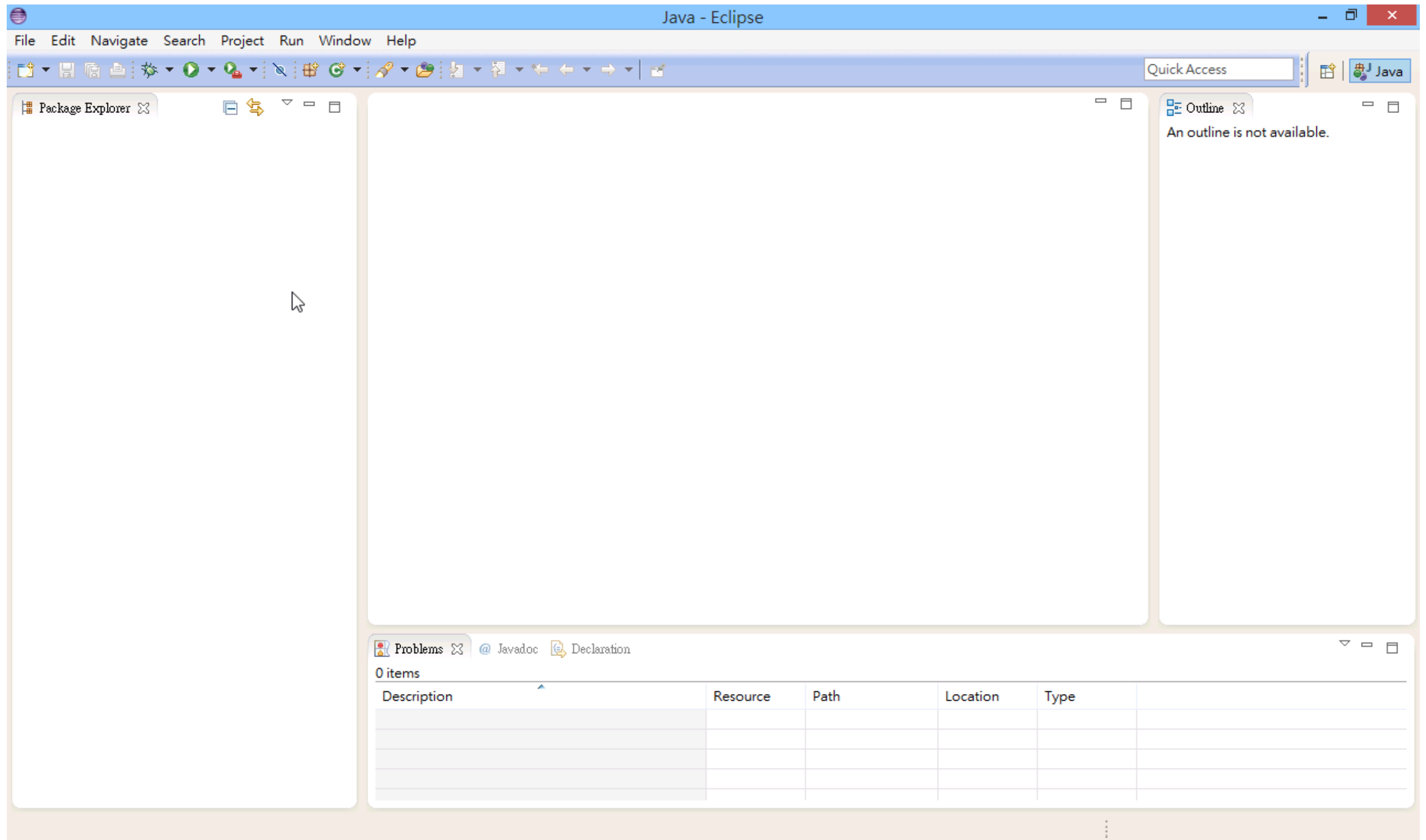




Workspace is the folder  
where you save your project files.





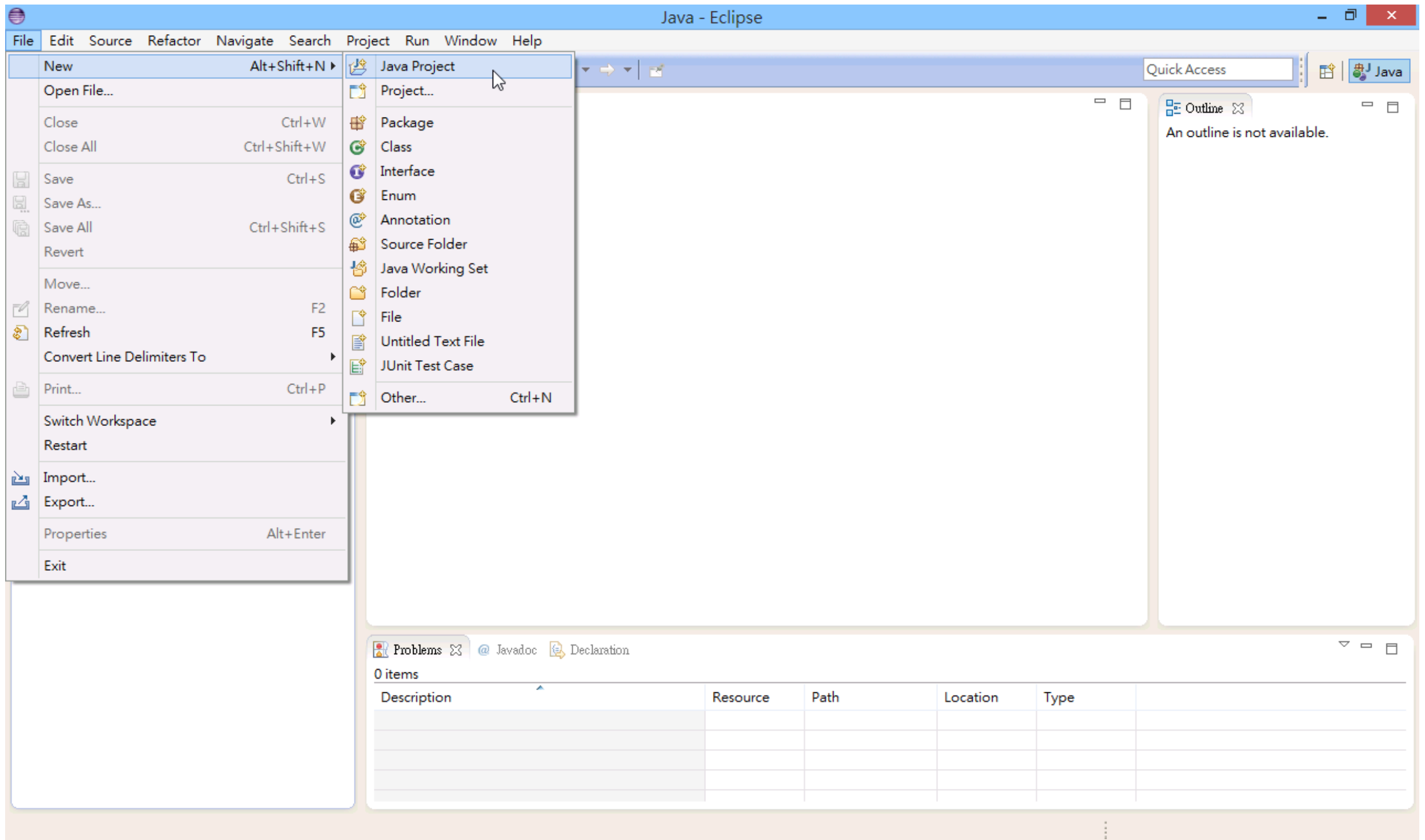


# Outline

- Introduction
- Say Hello to Java using command-line
- Setting Up Eclipse
- **Say Hello to Java using Eclipse**
- Today's Mission



Create you first project !



File > New > Java Project

**New Java Project**

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

Project name:

☒ Use default location  
Location:

**JRE**

☒ Use an execution environment JRE:

☐ Use a project specific JRE:

☐ Use default JRE (currently 'jre7') [Configure JREs...](#)

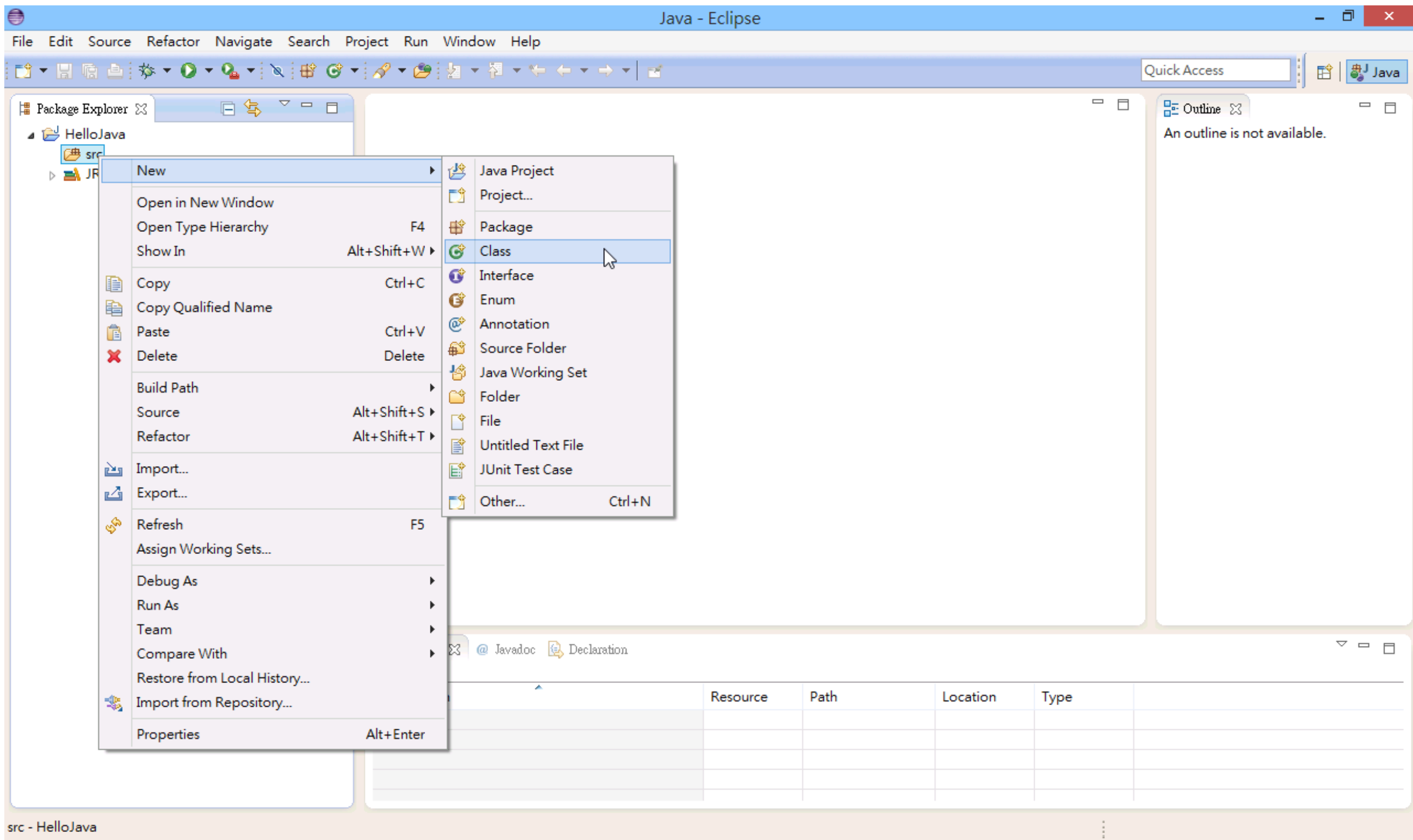
**Project layout**

☐ Use project folder as root for sources and class files

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

**Working sets**


☐ Add project to working sets  
Working sets:



Right click on the src folder > New > Class

**New Java Class**

**Java Class**

 The use of the default package is discouraged.

Source folder:

Package:

☐ Enclosing type:

---

Name:

Modifiers: ☒ public ☐ default ☐ private ☐ protected  
☐ abstract ☐ final ☐ static

Superclass:

Interfaces:

Which method stubs would you like to create?

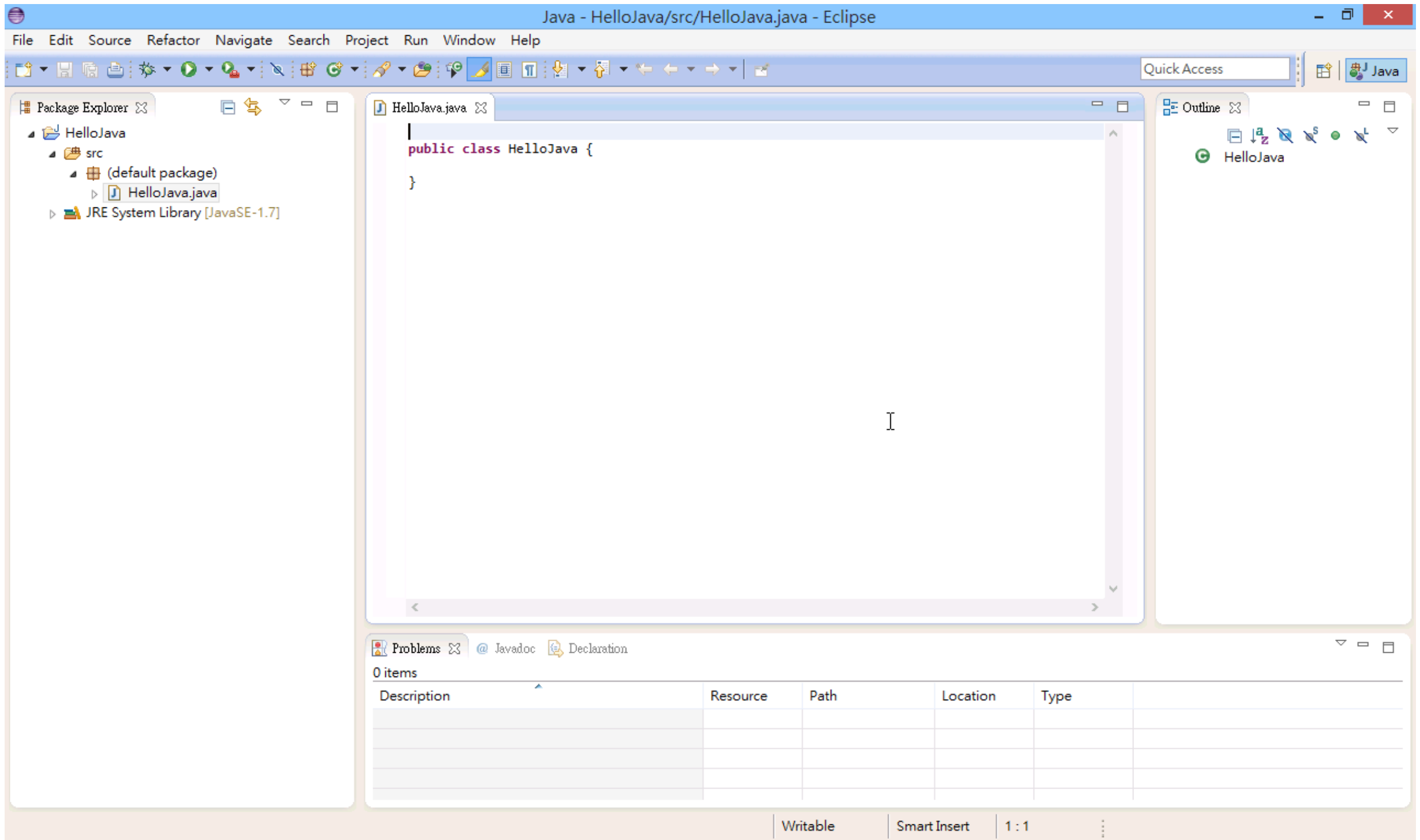
☐ public static void main(String[] args)

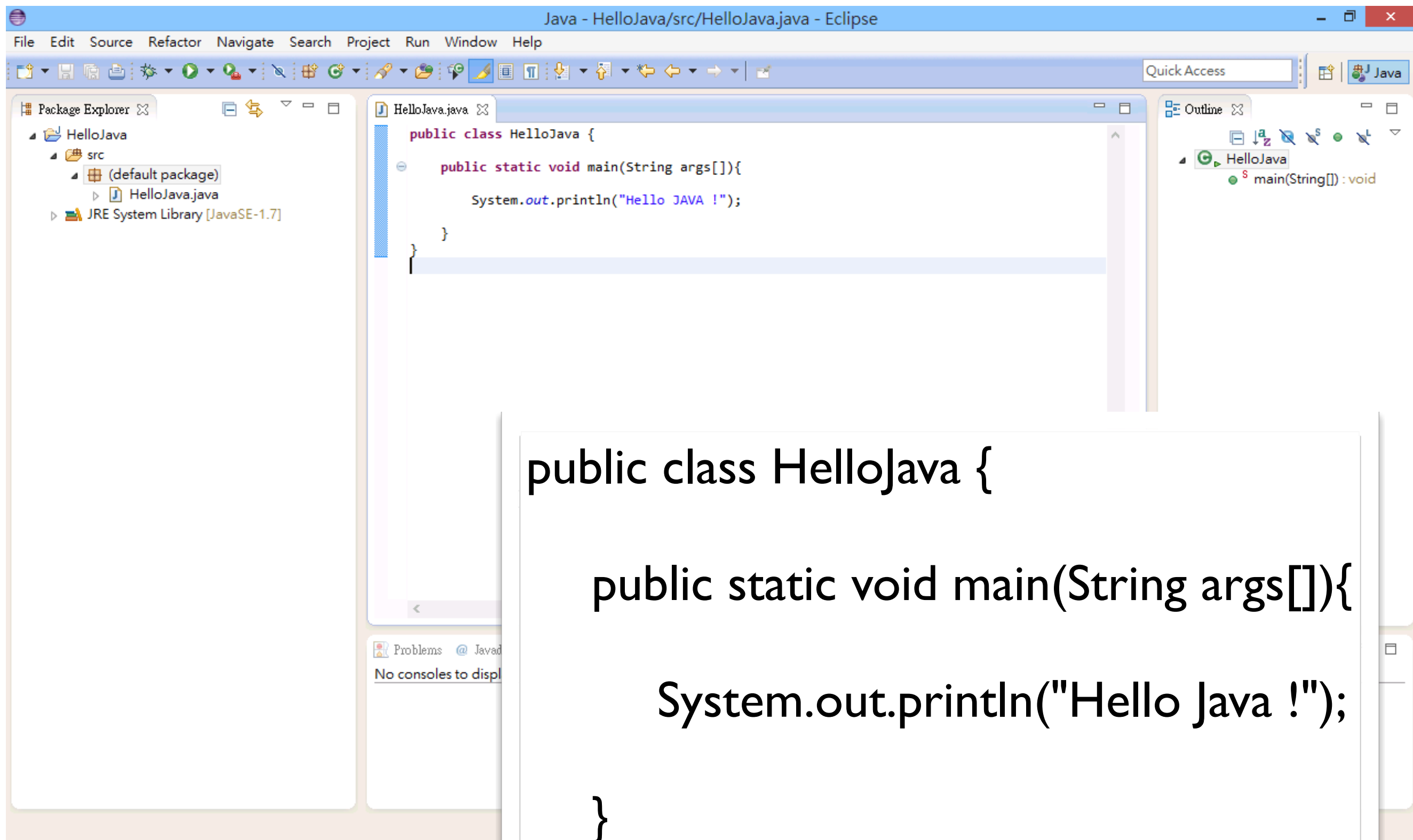
☐ Constructors from superclass

☒ Inherited abstract methods

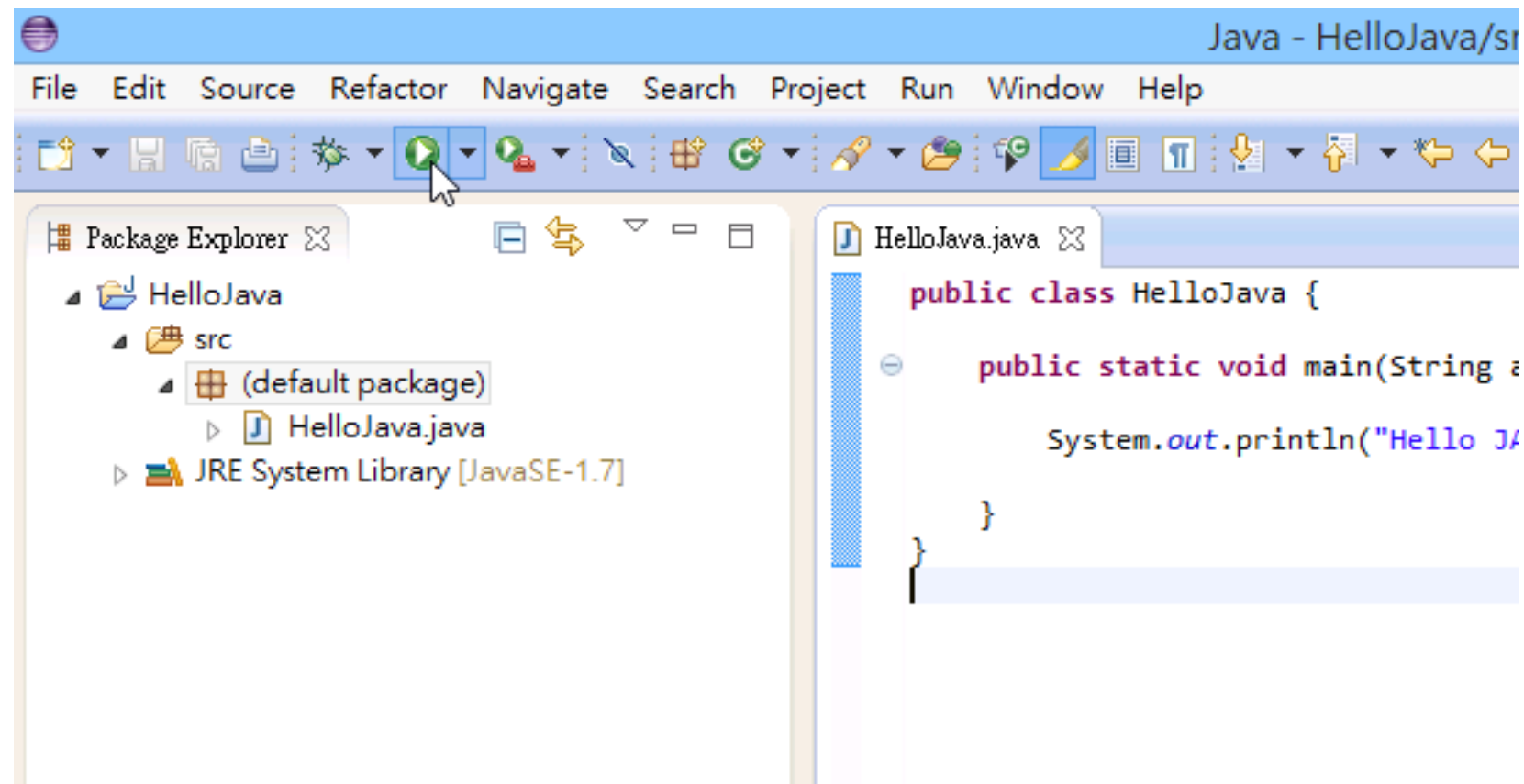
Do you want to add comments? (Configure templates and default value [here](#))

☐ Generate comments



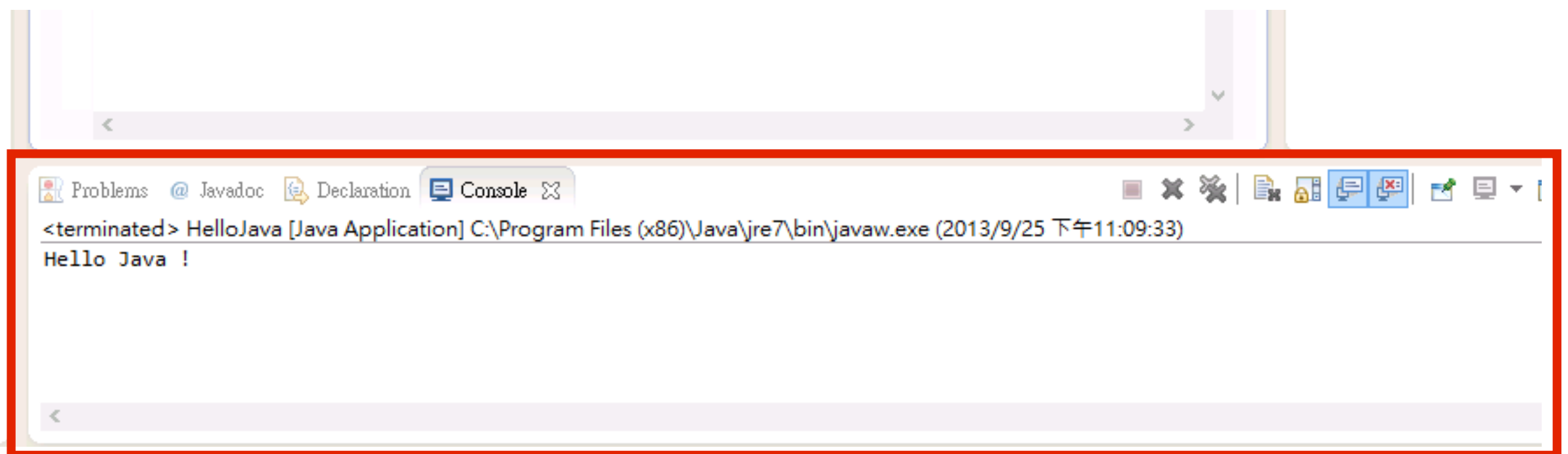


```
public class HelloJava {  
  
    public static void main(String args[]){  
  
        System.out.println("Hello Java !");  
  
    }  
}
```



Or use the hot key: **Ctrl + F11**





# Outline

- Introduction
- Say Hello to Java using command-line
- Setting Up Eclipse
- Say Hello to Java using Eclipse
- **Today's Mission**

# Today's Mission

- Create a Student class
  - You can set a name for him
  - Whenever you greet him, he will say "Hello! My name is xxx !"

# Hints (I)

- You may want to create two classes
  - One containing the main method
  - One implementing the Student class

# Hints (2)

- In the Student class
  - You may need to create a String field to store the student's name

```
private String name;
```

# Hints (3)

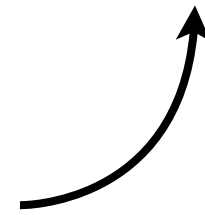
- In the Student class
  - You may need to create a method so that you can set the student's name
  - You may need to create a method so that you can greet the student

# Hints (4)

- You can print a String on the screen like this

```
System.out.println( ) ;
```

*put your String here*



# Hints (5)

- You can combine two String using '+'

```
String s1 = "Hi";  
String s2 = s1 + " there.";   
System.out.println(s2);
```

 *it will output 'Hi there.'*



# Hints (6)

- In the main method
  - You may need to **new** a student
  - You may need to set student's name and greet him

If you did everything right, the output may like:

