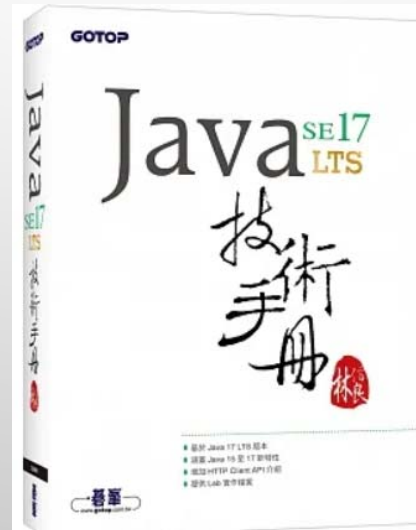


參考書

書名	作者	出版社	出版年度	版次
Java SE 17 技術手冊 ISBN:978-626-324143-5	林信良	碁峰	2022	初版









JAVA平台概論

學習目標

- 簡介JAVA版本遷移
- 認識JAVA SE、JAVA EE、JAVA ME
- 認識JDK規範與實作
- 瞭解JVM、JRE與JDK
- 下載、安裝JDK

[HTTPS://WWW.TIOBE.COM/TIOBE-INDEX/](https://www.tiobe.com/tiobe-index/)

- 流行的程式語言排名(TIOBE 衡量軟體程式品量的組織)

Sep 2022	Sep 2021	Change	Programming Language		Ratings	Change
1	2	▲		Python	15.74%	+4.07%
2	1	▼		C	13.96%	+2.13%
3	3			Java	11.72%	+0.60%
4	4			C++	9.76%	+2.63%
5	5			C#	4.88%	-0.89%
6	6			Visual Basic	4.39%	-0.22%
7	7			JavaScript	2.82%	+0.27%
8	8			Assembly language	2.49%	+0.07%
9	10	▲		SQL	2.01%	+0.21%
10	9	▼		PHP	1.68%	-0.17%

104人力銀行職缺

[HTTPS://WWW.104.COM.TW/](https://www.104.com.tw/)

104人力銀行

c

×

地

☐ 只搜尋職務名稱

相關搜尋：C++、韌體設計工程

職務

公司

更新日期

出勤制度

薪資待遇

經歷要求

找公司：85度C咖啡蛋糕...

PCE TECHNOL...

PCE

全部(19114)

全職(18246)

兼職(763)

高階(35)

104人力銀行

python

×

地

☐ 只搜尋職務名稱

相關搜尋：軟體設計工程師、C

職務

公司

更新日期

出勤制度

薪資待遇

經歷要求

全部(7385)

全職(7090)

兼職(247)

高階(6)

104人力銀行

java

×

地

☐ 只搜尋職務名稱

相關搜尋：軟體設計工程師、J

職務

公司

更新日期

出勤制度

薪資待遇

經歷要求

全部(5919)

全職(5771)

兼職(129)

高階(6)

JAVA不只是語言

- 從1995年至今，已經超過27個年頭
- 不僅是個程式語言，也代表瞭解決問題的平台
(PLATFORM)
- 代表了原廠、各個廠商、社群、開發者與使用者溝通的成果

JAVA誕生

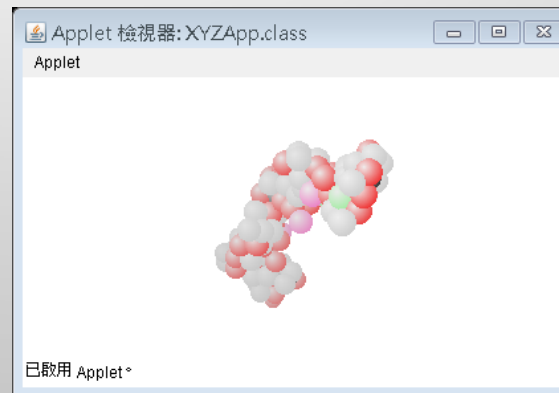
- JAVA最早是SUN公司「綠色專案」（**GREEN PROJECT**）中撰寫**STAR7**應用程式的程式語言，當時名稱不是**JAVA**，而是取名為**OAK**
- **SUN**（**STANFORD UNIVERSITY NETWORK**）是史丹福大學校園網 由斯坦福畢業生從學校出來創辦的公司。
- 後來發現**OAK**名稱已經被註冊了，工程師們邊喝咖啡邊討論著新名稱，最後靈機一動而改名為**JAVA**

JAVA誕生:消費性數位產品

- 一開始是為了消費性數位產品而設計，而這類小型裝置通常有著有限記憶體與運算資源
 - **JAVA**本身會見到許多為了節省資源而作的設計，像是動態載入類別檔案、字串池（**STRING POOL**）等特性

JAVA誕生:全球資訊網

- 全球資訊網（WORLD WIDE WEB）興起，JAVA APPLET成為網頁互動技術代表
- 1995年5月23日，正式將OAK改名為JAVA，JAVA DEVELOPMENT KITS（當時JDK全名）1.0A2版本正式對外發表
- 1996年NETSCAPE NAVIGATOR 2.0也正式支援JAVA，MICROSOFT EXPLORER亦開始支援JAVA



版本演進

- SUN在1998/12/4年發佈JAVA 2 PLATFORM，簡稱J2SE 1.2
- JAVA開發者版本一開始是以JAVA DEVELOPMENT KIT名稱發表，簡稱JDK，而J2SE則是平台名稱，包含了JDK與JAVA程式語言

版本演進

- **JAVA**平台標準版約每兩年為週期推出重大版本更新
- 「**JAVA 2**」這個名稱也從**J2SE 1.2**一直延用至之後各個版本

版本演進

- 2004/9/29發表的JAVA平台標準版的版號不是1.5，而直接跳到5.0，稱為J2SE 5.0
- 2006/12/11發表的JAVA平台標準版，除了版號之外，名稱也有了變化，稱為JAVA PLATFORM, STANDARD EDITION 6，簡稱JAVA SE 6
- JDK6全名則稱為JAVA SE DEVELOPMENT KIT 6，也就是不再像以前JAVA 2帶有"2"這個號碼

江山易主

- 從**JAVA SE 6**之後，**JAVA**開發人員足足等了四年多，才等到新版本的推出 ...
- **2010年ORACLE**宣佈併購**SUN** ...
- **2010年底JCP**（**JAVA COMMUNITY PROCESS**，稍後即會說明這個組織為何）終於通過了**JAVA SE 7**與**JAVA SE 8**的規劃地圖（**ROADMAP**） ...
- **JAVA SE 7**於**2011/7/28**釋出
- 發表**JAVA SE 8**的時間為**2014/03/18**

三大平台

- SUN公司在1999年6月美國舊金山的JAVA ONE大會上，公佈了新的JAVA體系架構
 - J2SE (JAVA 2 PLATFORM, STANDARD EDITION)
PC應用開發
 - J2EE (JAVA 2 PLATFORM, ENTERPRISE EDITION)
企業應用開發
 - J2ME (JAVA 2 PLATFORM, MICRO EDITION)
嵌入式應用開發

三大平台

- JAVA SE 6後JAVA不再帶有“2”這個號碼
 - J2SE → JAVA SE
 - J2EE → JAVA EE
 - J2ME → JAVA ME

JAVA SE



JCP與JSR

- **JAVA**不僅是程式語言，而是標準規範！
- **SUN**公司於**1998**年組成了**JCP**（**JAVA COMMUNITY PROCESS**）
 - 一個開放性國際組織，目的是讓**JAVA**演進由**SUN**非正式地主導，成為全世界數以百計代表成員公開監督的過程

JCP與JSR

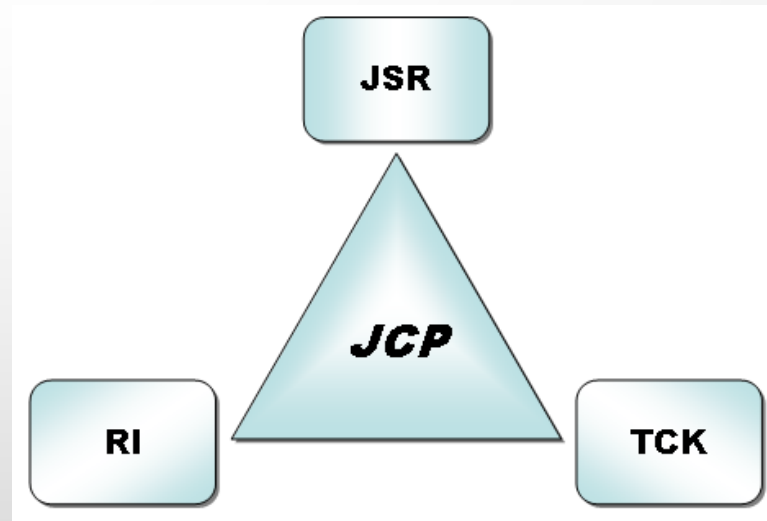
- 任何想要提議加入**JAVA**的功能或特性，必須以**JSR**（**JAVA SPECIFICATION REQUESTS**）正式文件的方式提交
- **JSR**必須經過**JCP**執行委員會（**EXECUTIVE COMMITTEE**）投票通過，方可成為最終標準文件
- 有興趣的廠商或組織可以根據**JSR**實現產品

JCP與JSR

- 若**JSR**成為最終文件後，必須根據**JSR**實作出免費且開發原始碼的參考實現，稱為**RI**（**REFERENCE IMPLEMENTATION**）
- 提供**TCK**（**TECHNOLOGY COMPATIBILITY KIT**）作為技術相容測試工具箱
 - 方便其它想根據**JSR**實現產品的廠商或組織參考與測試相容性

JCP與JSR

- 無論是**JAVA SE**、**JAVA EE**或**JAVA ME**，都是業界共同訂製的標準
- **JSR**作為正式標準規範文件，不同的技術解決方案標準規範會給予一個編號
- 在**JSR**規範的標準之下，各廠商可以各自實作成品



JCP與JSR

- **JAVA SE 8**的主要規範是在**JSR 337**文件之中，而**JAVA SE**平台中的特定技術，則再規範於特定的**JSR**文件之中...
- **[HTTP://JCP.ORG/EN/JSR/DETAIL?ID=337](http://jcp.org/en/jsr/detail?id=337)**

ORACLE JDK與OPENJDK

- 在過去，SUN JDK實現，也就是被ORACLE收購之後的ORACLE JDK實現，就是JDK的參考實作
- 有興趣的廠商或組織也可以根據JSR自行實現產品
- 只有通過TCK 相容性測試的實作，才可以使用JAVA這個商標

ORACLE JDK與OPENJDK

- 2006年的JAVAONE大會上，SUN宣告對JAVA開放原始碼，從JDK7 B10 開始有了OPENJDK
- 2009年4月15日正式發佈 OPENJDK
- ORACLE時代發佈的JDK7正式版本，指定了OPENJDK7為官方參考實作

ORACLE JDK7與OPENJDK7

- SUN JDK採JRL，而 OPENJDK7採GPL（帶有GPL LINKING EXCEPTION的修正版本）
- 前者原始碼可用於個人研究使用，但禁止任何商業用途，後者則允許商業上的使用
- OPENJDK7必須拿掉許多在兩個授權間有衝突的程式碼，也不包括一些部署（DEPLOYMENT）工具以及軟體套件等

ORACLE JDK7與OPENJDK7

- 如果執行**JAVA -VERSION**，原始的OPENJDK7顯示的會是**OPENJDK VERSION**字樣，而不是**JAVA VERSION**字樣
- RED HAT於2007年發起了**ICEDTEA**計畫，致力於修補OPENJDK7使之完備，並通過了JCK相容測試
- 如果使用**ICEDTEA**修補過後的OPENJDK7，執行**JAVA -VERSION**，就會顯示**JAVA VERSION**字樣

OPEN JDK7與OPENJDK6

- OPENJDK6並不是SUN JDK6的分支，而是將OPENJDK7中JDK7的特性拿掉，使之符合JDK6的規範
- OPENJDK6實際上是OPENJDK7的分支，OPENJDK6可以通過JCK相容測試

OPEN JDK7與OPENJDK6

- ORACLE在2013年2月時宣佈JDK6 UPDATE 43時，宣佈這是最後一個免費更新版本
- RED HAT於 2013年3月時宣布持有 OPENJDK6 領導權，以能持續對OPENJDK6發現的臭蟲與安全問題進行修補

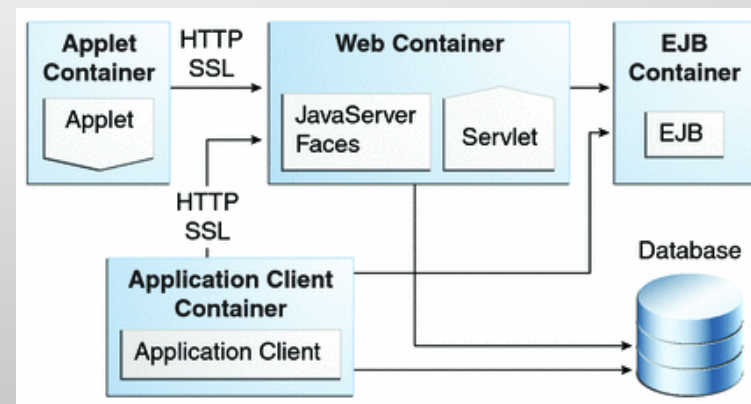
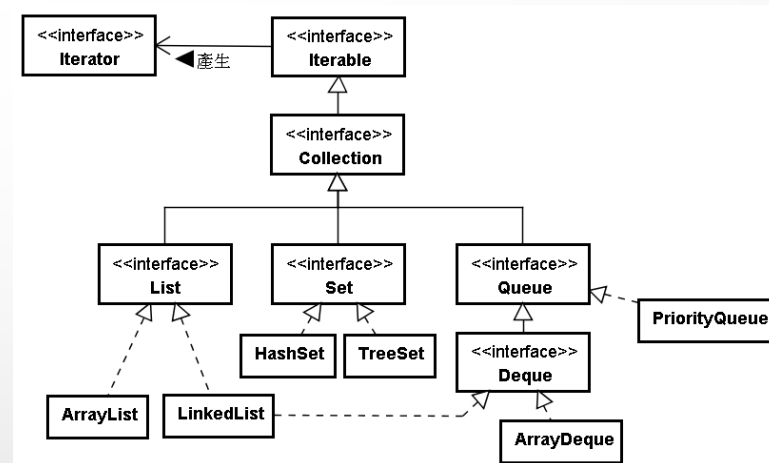
版本	代碼名稱	釋出日期
JDK 1.1.4	Sparkler (煙火)	1997/09/12
JDK 1.1.5	Pumpkin (南瓜)	1997/12/03
JDK 1.1.6	Abigail (聖經故事人物名稱)	1998/04/24
JDK 1.1.7	Brutus (羅馬政治家名稱)	1998/09/28
JDK 1.1.8	Chelsea (足球俱樂部名稱)	1999/04/08
J2SE 1.2	Playground (遊樂場)	1998/12/04
J2SE 1.2.1	無	1999/03/30
J2SE 1.2.2	Cricket (蟋蟀)	1999/07/08
J2SE 1.3	Kestrel (紅隼)	2000/05/08
J2SE 1.3.1	Ladybird (瓢蟲)	2001/05/17
J2SE 1.4.0	Merlin (魔法師名稱)	2002/02/13
J2SE 1.4.1	Hopper (蚱蜢)	2002/09/16
J2SE 1.4.2	Mantis (螳螂)	2003/06/26
J2SE 5.0	Tiger (老虎)	2004/09/29
Java SE 6	Mustang (野馬)	2006/12/11
Java SE 7	Dolphin (海豚)	2011/07/28
Java SE 8	無	2014/03/18

Java SE 9	2017 年 9 月	OpenJDK 於 2018 年 3 月停止更新	不適用
Java SE 10	2018 年 3 月	OpenJDK 於 2018 年 9 月停止更新	不適用
Java SE 11 (LTS)	2018 年 9 月	AdoptOpenJDK 於 2024 年 10 月或之後停止更新 Amazon Corretto 於 2027 年 9 月或之後停止更新	2026 年 9 月
Java SE 12	2019 年 3 月	OpenJDK 於 2019 年 9 月停止更新	不適用
Java SE 13	2019 年 9 月	OpenJDK 於 2020 年 3 月停止更新	不適用
Java SE 14	2020 年 3 月	OpenJDK 於 2020 年 9 月停止更新	不適用
Java SE 15	2020 年 9 月	OpenJDK 於 2021 年 3 月停止更新	不適用
Java SE 16	2021 年 3 月	OpenJDK 於 2021 年 9 月停止更新	不適用
Java SE 17 (LTS)	2021 年 9 月	Azul 於 2029 年 9 月停止更新 Microsoft 於 2027 年 9 月停止更新	2029 年 9 月
Java SE 18	2022 年 3 月	OpenJDK 於 2022 年 9 月停止更新	不適用

格式： ■ 舊版本 ■ 舊版本，仍被支援 ■ 目前版本 ■ 未來版本

建議的學習路徑

- 深入瞭解JVM/JRE/JDK
- 理解封裝、繼承、多型
- 掌握常用JAVA SE API架構
- 學習容器觀念
- 研究開放原始碼專案
- 學習設計模式與重構
- 熟悉相關開發工具

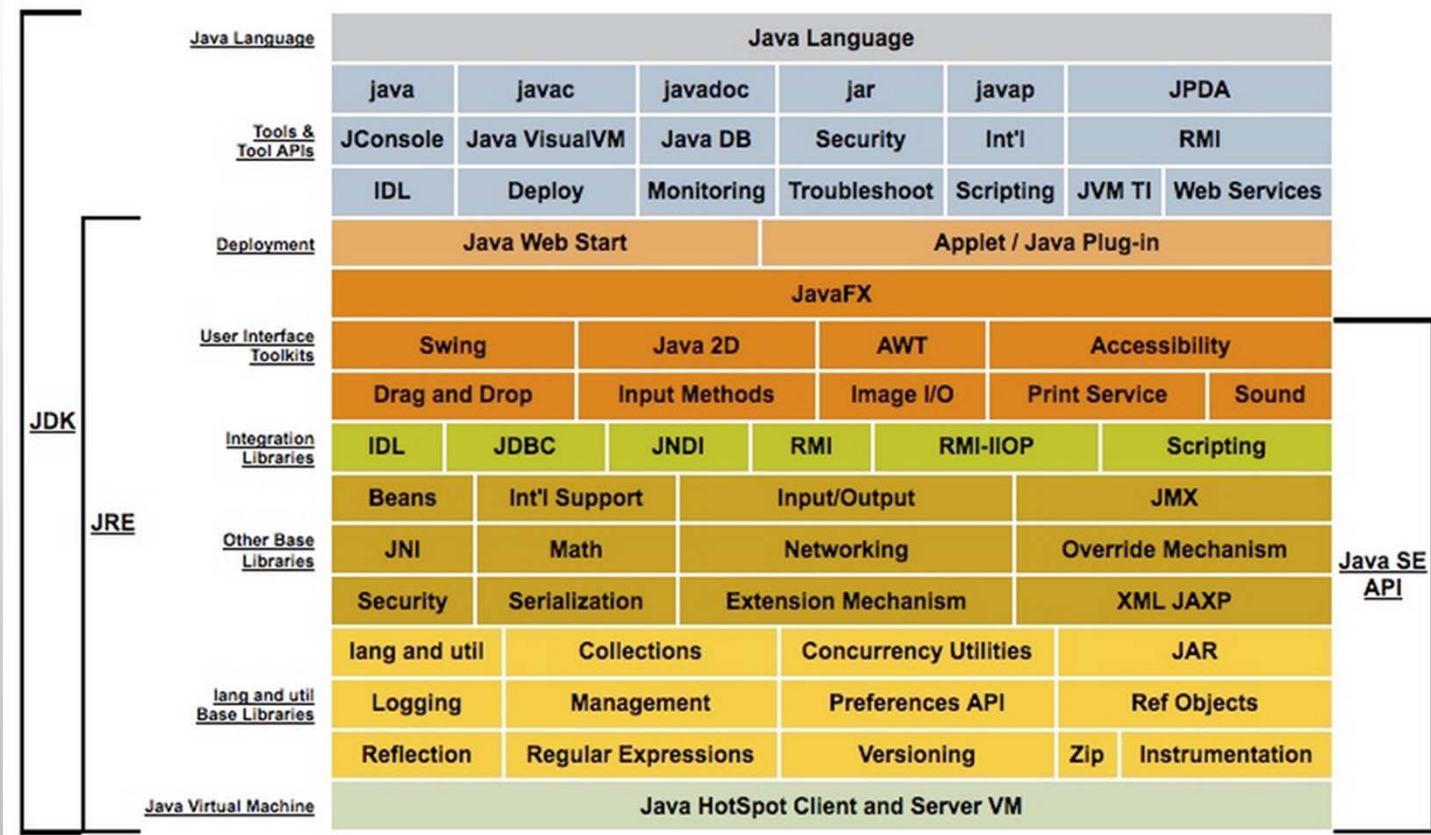


JDK / JRE

- JRE (Java Runtime Environment)
用來執行 Java 程式
- JDK (Java Development Kit)
開發、測試或執行 Java 程式、包含 JRE

JRE/JDK

Jave SE Platform at a Glance



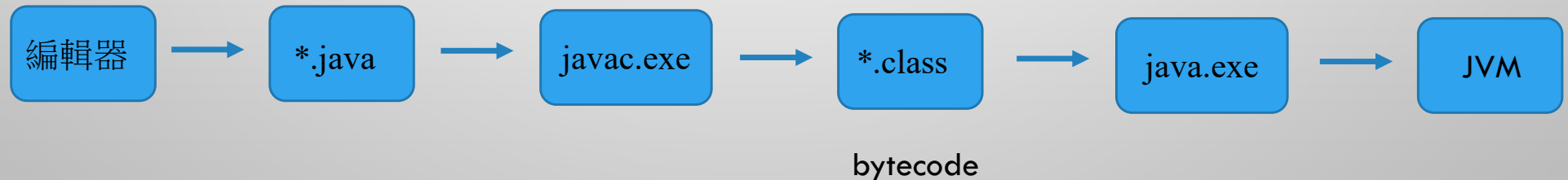
JRE/JDK TOOLS

- 編輯器 (editor)
寫 Java 程式
- 編譯器 (compiler)
把 Java 程式轉成 Java bytecode，也就是 class 檔
- 解譯器 (Interpreter)
執行 Java 的 class 檔
- 除錯器 (debugger)
找出程式的問題

JRE/JDK TOOLS

路徑在 JDK 的 \bin 裡面：

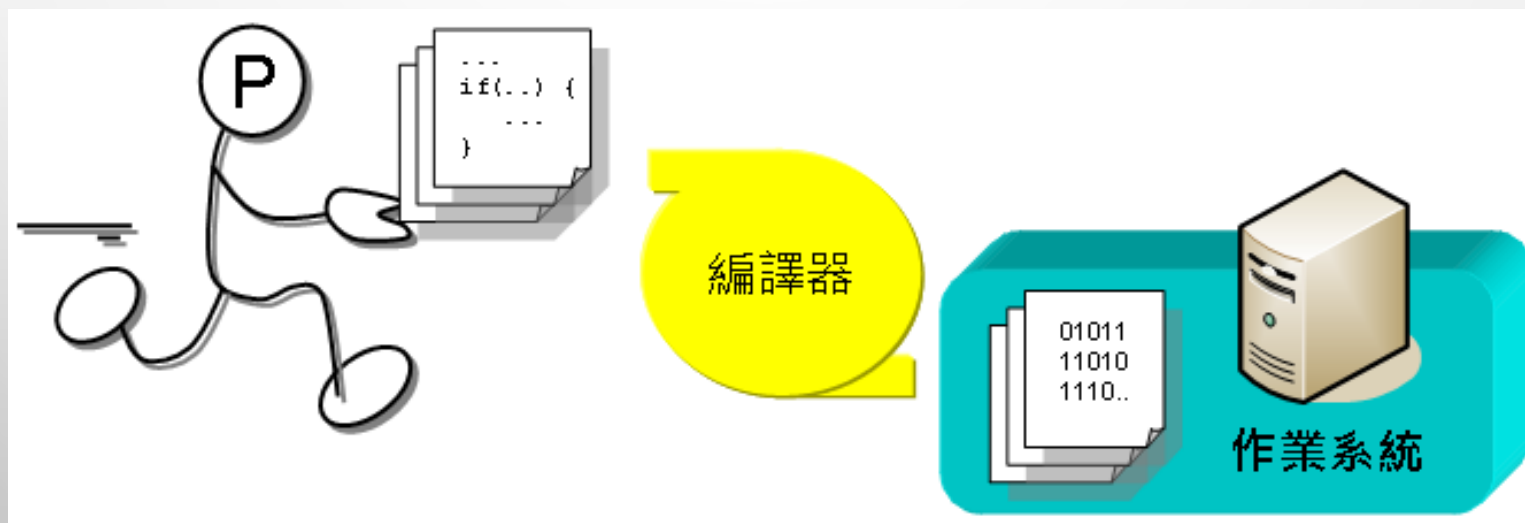
- javac.exe : Java compiler (編譯器)
- java.exe : Java interpreter (解譯器)
- javap : Class file disassembler (反組譯)



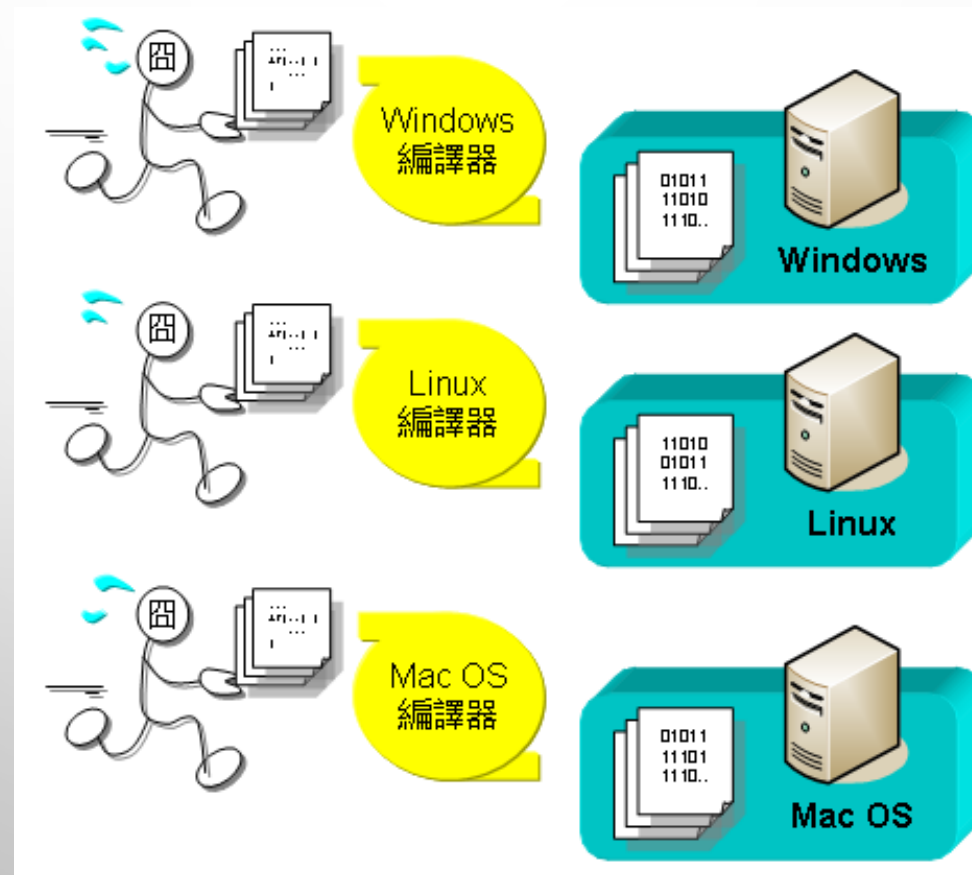
什麼是JVM？

- **JAVA**程式都會**COMPILE**(編譯)成**BYTE CODE** (位元組碼)，而不是 **MACHINE-DEPENDENT CODE** (要在特定機器才能跑的程式)**JAVA BYTE CODE**執行時透過各機器上的 **JAVA VIRTUAL MACHINE (JVM)** 來解讀.
- 對**JAVA**程式而言，只認識一種作業系統，這個系統叫**JVM**，位元碼檔案（副檔名為**.CLASS**的檔案）就是**JVM**的可執行檔。

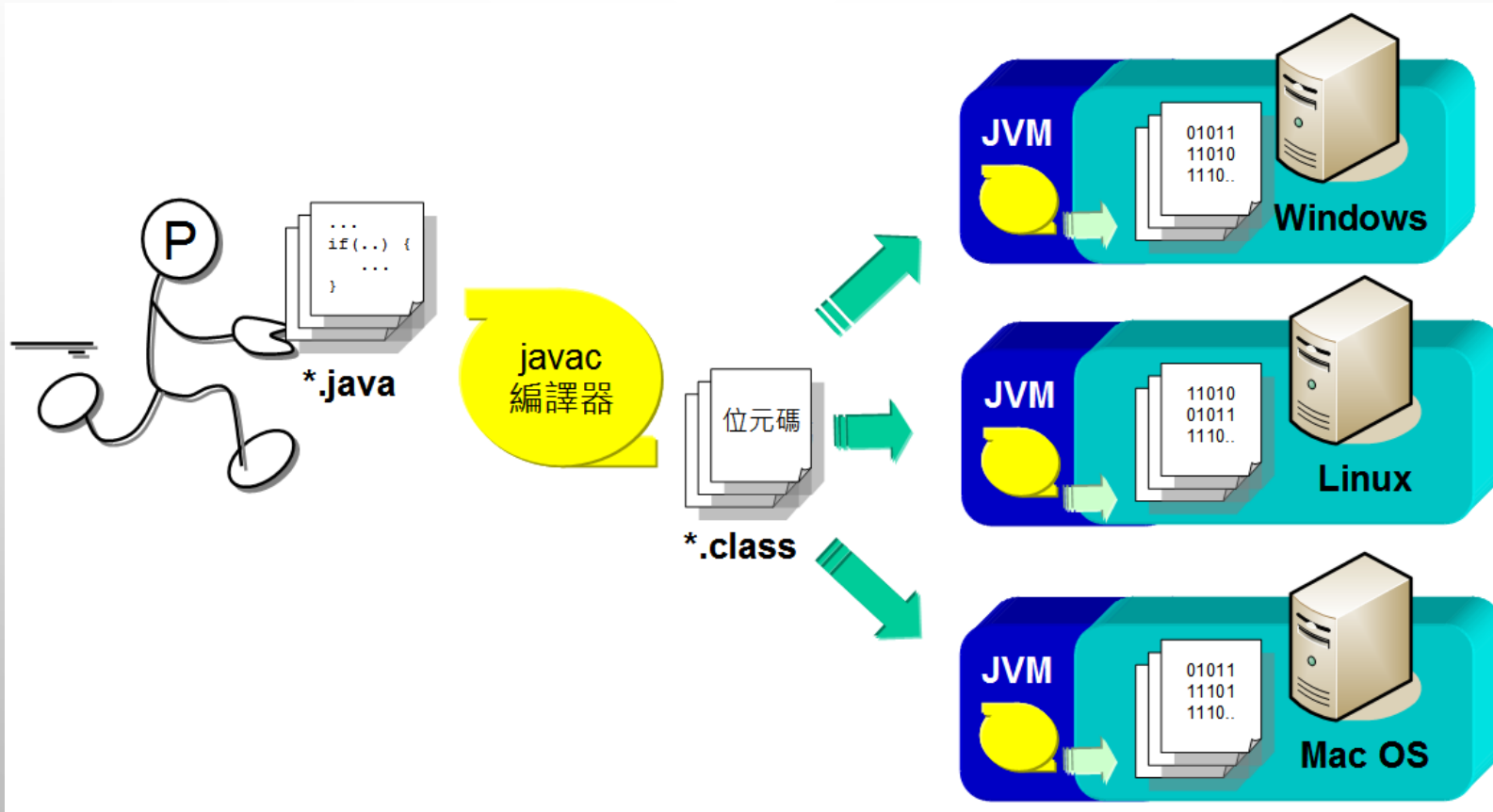
一般編譯過程



各平台編譯過程



JAVA編譯過程



常見的JAVA語言

- **Java application**

單獨跑在電腦PC上的Java程式。

例如整個Borland Jbuilder本身就是用Java寫的。

- **Java Applet**

主要嵌在Web page網頁上的Java程式。

基於安全性考量，Java Applet能執行的功能會比較受限制。

- **Java script**

在瀏覽器Client端所執行的Java語法的Script程式,可提供網頁互動功能,由 Netscape, Inc. 開發。

- **Java Server Pages (JSP)**

在伺服器Server端所執行的Java語法的Script程式。

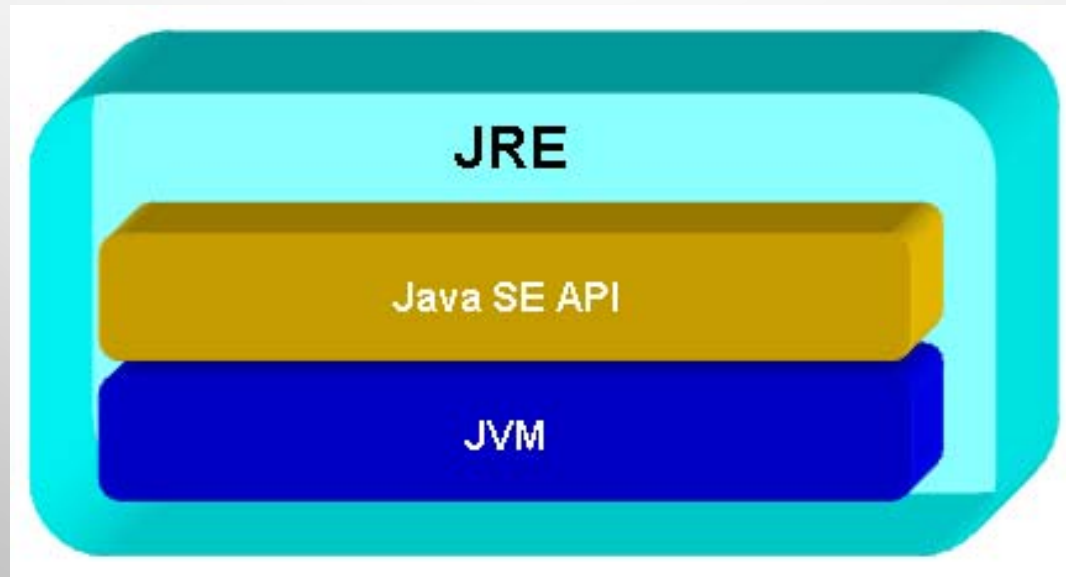
標準API

```
System.out.println("Hello World");
```

- **SYSTEM**、**OUT**、**PRINTLN**這些名稱，都是標準中規範的名稱
- 有人根據標準撰寫出**SYSTEM.JAVA**，編譯為**SYSTEM.CLASS**
- 你才能在撰寫第一個**JAVA**程式時，使用**SYSTEM**類別（**CLASS**）上**OUT**物件（**OBJECT**）的**PRINTLN()**方法（**METHOD**），引用如**SYSTEM**這些標準API

區分JRE與JDK

- JAVA RUNTIME ENVIRONMENT就是JAVA執行環境，簡稱JRE，包括了JAVA SE API與JVM



區分JRE與JDK

- 像編譯器這樣的工具程式是由誰提供？答案就是JDK，全名為是JAVA DEVELOPMENT KIT！

區分JRE與JDK

- JDK包括了JAVA程式語言、工具程式與JRE
- JRE則包括了部署技術、JAVA SE API與JVM
- 撰寫JAVA程式才需要JDK，如果你的程式只是想讓朋友執行？那他只要裝JRE就可以了

安裝JDK步驟

1.下載JDK

2.安裝JDK

3.設定JAVA_HOME環境變數

4.設定Path環境變數

5.確認安裝狀態

下載JDK

<https://www.oracle.com/java/technologies/downloads/#jdk17-windows>

Java 18 Java 17

Java SE Development Kit 17.0.4.1 downloads

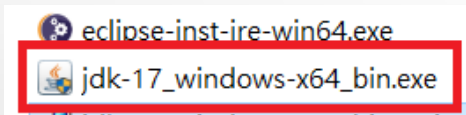
Thank you for downloading this release of the Java™ Platform, Standard Edition Development Kit (JDK™). The JDK is a development environment for building applications and co programming language.

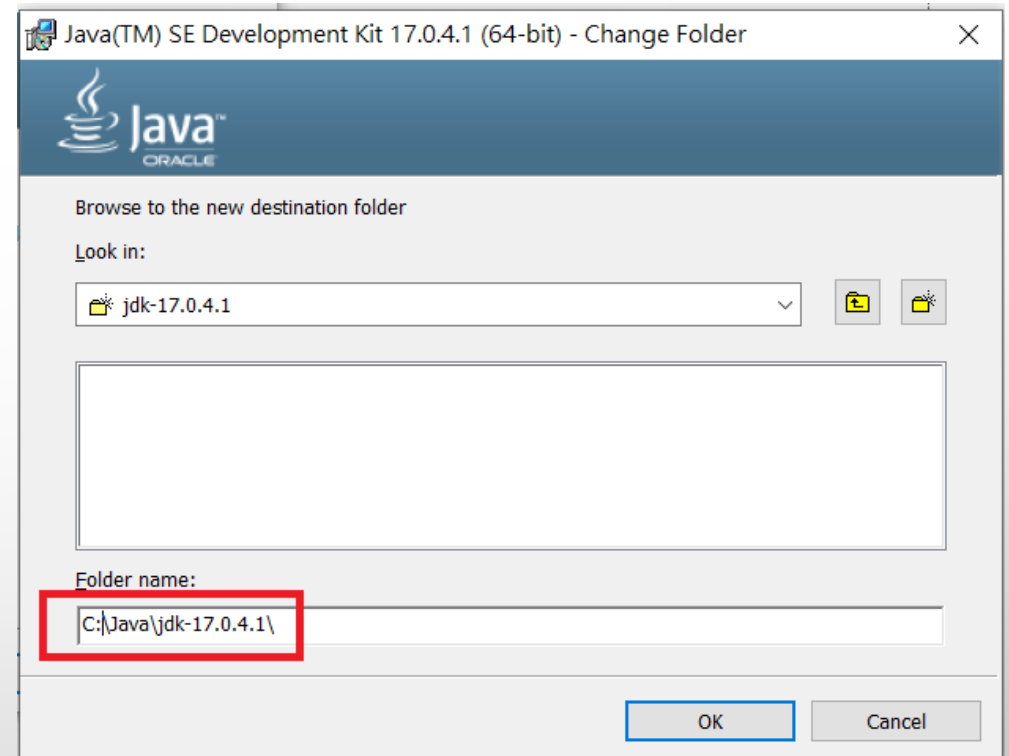
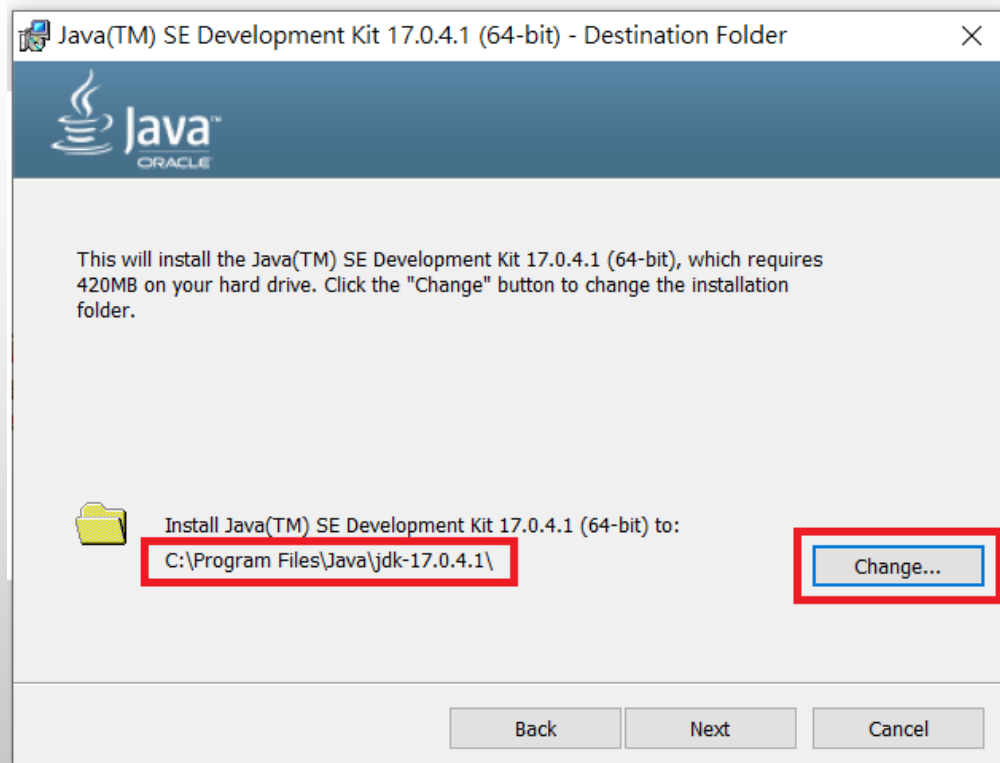
The JDK includes tools for developing and testing programs written in the Java programming language and running on the Java platform.

Linux macOS Windows

Product/file description	File size	Download
x64 Compressed Archive	171.81 MB	https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.zip (sha256 🔗)
x64 Installer	152.78 MB	https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.exe (sha256 🔗)
x64 MSI Installer	151.66 MB	https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.msi (sha256 🔗)

安裝JDK

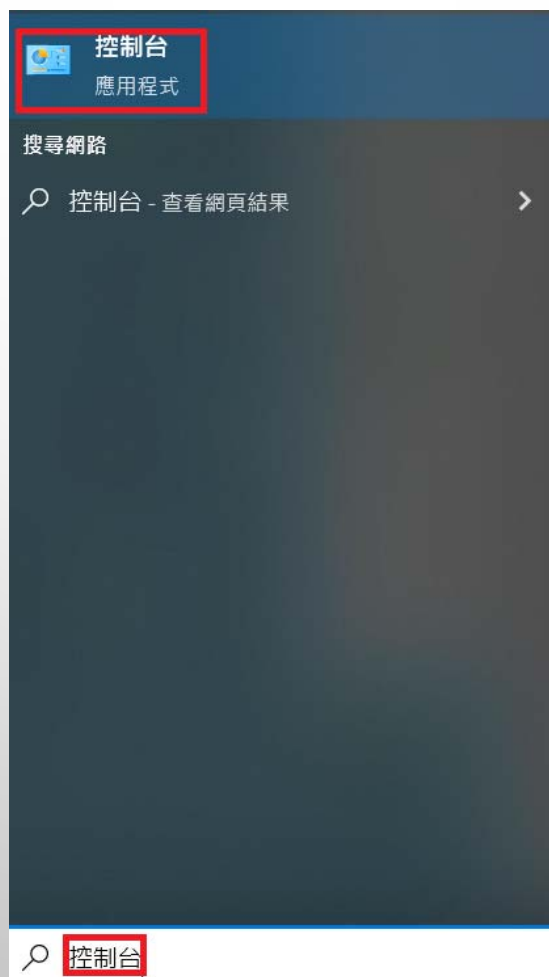




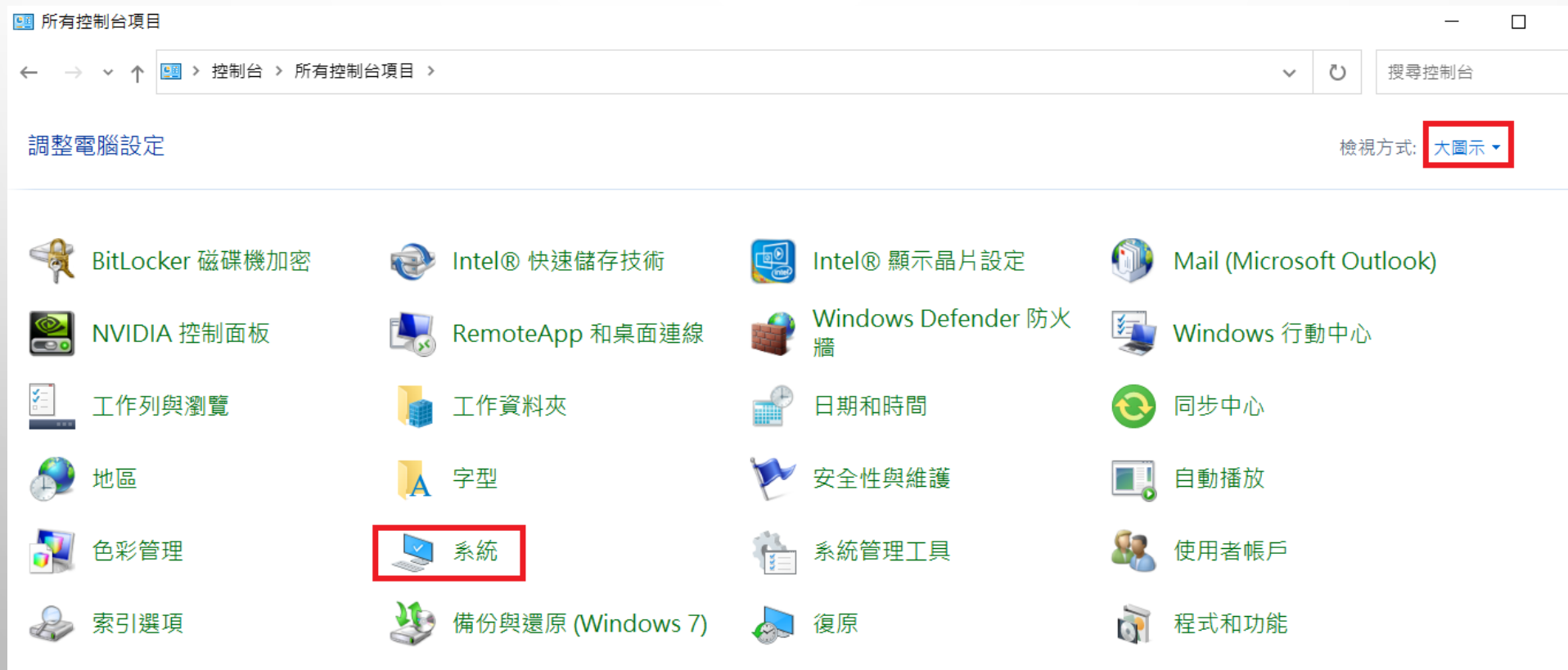




設定JAVA_HOME環境變數



設定JAVA_HOME環境變數



設定JAVA_HOME環境變數



設定JAVA_HOME環境變數

系統內容

×

電腦名稱 硬體 **進階** 系統保護 遠端

您必須以系統管理員的身分登入，才能變更這裡的大部分設定。

效能
視覺效果、處理器排程、記憶體使用量和虛擬記憶體
設定(S)...

使用者設定檔
關於您登入時的桌面設定
設定(E)...

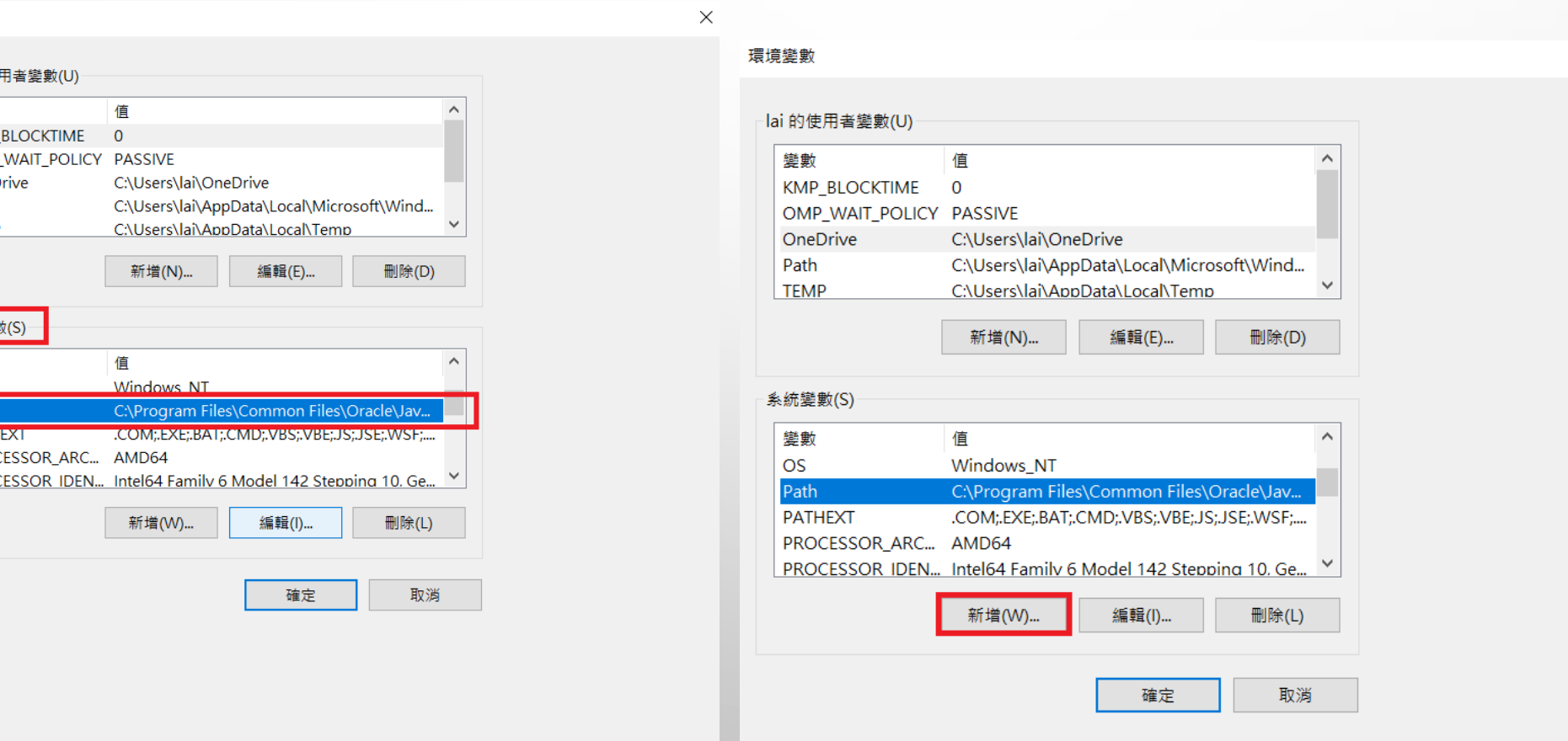
啟動及修復
系統啟動、系統失敗、及偵錯資訊
設定(T)...

環境變數(N)...

確定 取消 套用(A)

設定JAVA_HOME環境變數

系統變數優先使用者變數



設定JAVA_HOME環境變數

新增系統變數

變數名稱(N): JAVA_HOME

變數值(V): C:\java\jdk-17.0.4.1

瀏覽目錄(D)... 瀏覽檔案(F)... 確定 取消

安裝JDK的路徑

設定PATH環境變數

環境變數

The screenshot shows the 'Environment Variables' dialog box in Windows. It is divided into two sections: 'User variables for lai' and 'System variables'. In the 'System variables' section, the 'Path' variable is selected and highlighted with a red rectangle. The 'Edit...' button for this variable is also highlighted with a red rectangle. The 'Path' variable's value is 'C:\Program Files\Common Files\Oracle\Jav...'. Below the 'Path' variable, the 'PATH' variable is also visible with its value '.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;...'. At the bottom of the dialog, the 'OK' button is highlighted with a blue rectangle.

lai 的使用者變數(U)

變數	值
KMP_BLOCKTIME	0
OMP_WAIT_POLICY	PASSIVE
OneDrive	C:\Users\lai\OneDrive
Path	C:\Users\lai\AppData\Local\Microsoft\Wind...
TEMP	C:\Users\lai\AppData\Local\Temp

新增(N)... 編輯(E)... 刪除(D)

系統變數(S)

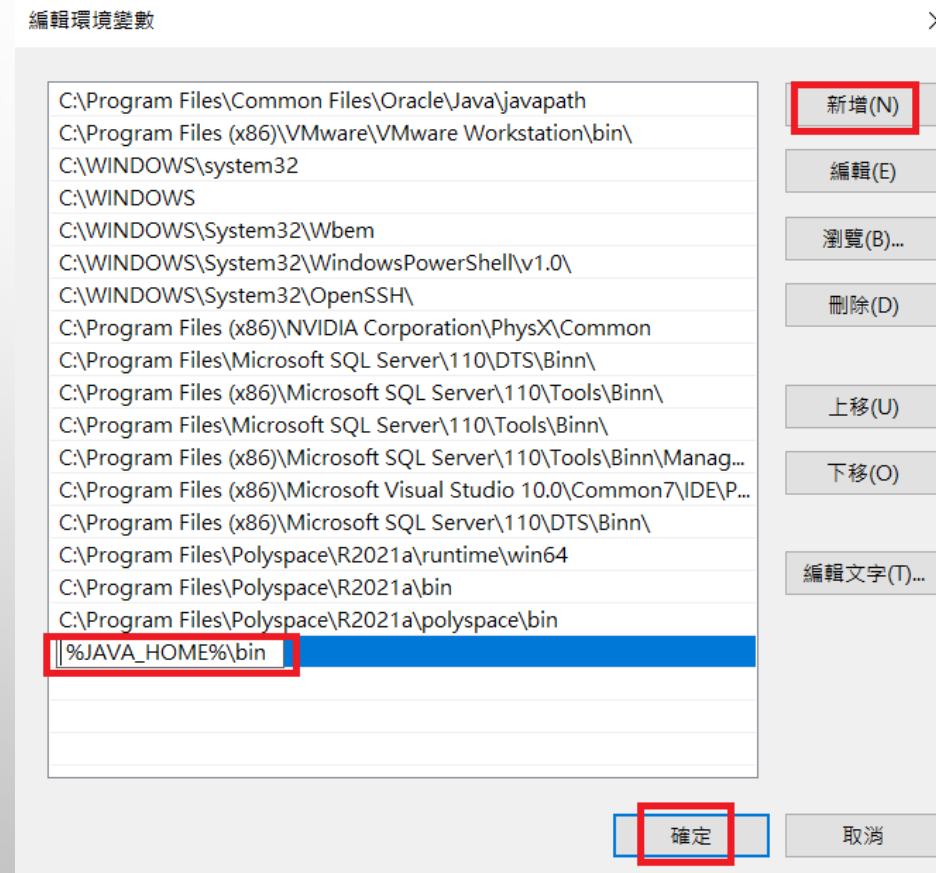
變數	值
NUMBER_OF_PRO...	8
OS	Windows NT
Path	C:\Program Files\Common Files\Oracle\Jav...
PATH	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;...
PROCESSOR ARC...	AMD64

新增(W)... 編輯(I)... 刪除(L)

確定 取消

設定PATH環境變數

增加 %JAVA_HOME%\bin



設定PATH環境變數

置頂設定為最優先



確認安裝狀態

開啟命令列



確認安裝狀態

輸入 `java -version`

命令提示字元

```
Microsoft Windows [版本 10.0.19044.1889]  
(c) Microsoft Corporation. 著作權所有，並保留一切權利。  
  
C:\Users\lai>java -version  
java version "17.0.4.1" 2022-08-18 LTS  
Java(TM) SE Runtime Environment (build 17.0.4.1+1-LTS-2)  
Java HotSpot(TM) 64-Bit Server VM (build 17.0.4.1+1-LTS-2, mixed mode, sharing)  
  
C:\Users\lai>
```

確認安裝狀態

輸入 javac

命令提示字元

```
Microsoft Windows [版本 10.0.19044.1889]  
(c) Microsoft Corporation. 著作權所有，並保留一切權利。  
C:\Users\lai>javac  
Usage: javac <options> <source files>  
where possible options include:  
  @<filename>           Read options and filenames from file  
  -Akey[=value]          Options to pass to annotation processors  
  --add-modules <module>(,<module>)*  
                        Root modules to resolve in addition to the initial modules, or all modules  
                        on the module path if <module> is ALL-MODULE-PATH.  
  --boot-class-path <path>, -bootclasspath <path>  
                        Override location of bootstrap class files  
  --class-path <path>, -classpath <path>, -cp <path>  
                        Specify where to find user class files and annotation processors  
  -d <directory>        Specify where to place generated class files  
  -deprecation           Output source locations where deprecated APIs are used
```

編輯程式工具

- 一般文字文件編輯
記事本
- 進階記事本
Ultraedit, Notepad++
- 整合發展環境**IDE(Integrated Development Environment)**
Eclipse, NetBeans 文件管理 編譯 除錯

7 Best Java IDE For Developers in 2022

<https://www.geeksforgeeks.org/7-best-java-ide-for-developers-in-2022/>

1. Eclipse

貢獻了大約 48% 的市場份額，**免費**集成開發環境，用戶評分為 4.8，用戶滿意度為 92%，允許開發人員自定義他們自己的功能。它支持 100 多種編程語言，包括 Groovy、JavaScript、C++、Python 等等。

2. NetBeans

用戶評分為 4.1 分（滿分 5 分），用戶滿意度高達 82%。它在 Windows、Linux、macOS 和 Solaris 上運行，可為從產品設計到部署的整個軟件開發生命週期增加價值。

3. BlueJ

它是一個**免費**的 Java IDE，也是最好的 Java Web 開發 IDE，由 John Rosenberg 和 Michael Kolling 於 1999 年創立，用戶滿意度為 4.2，用戶滿意度高達 80%。它最初是為教育和培訓目的而開發的。

4. JDeveloper

JDeveloper 是 Oracle Corporation 提供的**免費**集成開發環境。用戶評分為 4 分和 5 分，用戶滿意度高達 80%。它為 XML、Java、SQL、PL/SQL、JavaScript、HTML、BPEL 和 PHP 的開發提供了廣泛的功能。

5. IntelliJ IDEA

佔市場份額的 33% 以上。它具有版本控制系統、框架和對不同語言的支持等工具。用戶評分高達 4.3，用戶滿意度高達 89%。

6. JCreator

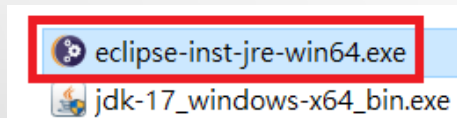
JCreator 是由Xinox Software 開發的頂級Java IDE。它的界面與 Microsoft Visual Studio 非常相似。它完全是用 C++ 編寫的，因此與其他 Java IDE 相比，它非常高效和快速。

7. DrJava

DrJava 是一個極其輕量級的 Java IDE**免費**集成開發環境，由萊斯大學的 JavaPLT 小組設計、開發和維護。它還提供自動縮進、自動完成、註釋以及通過插件與 Eclipse 完美集成。

安裝 ECLIPSE

<https://www.eclipse.org/downloads/>



★ DONATE

The folder is specified as 'C:\java\java-2022-06' but 'C:\java\java-2022-06' will be used.



Eclipse IDE for Java Developers

[details](#)

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

Java 11+ VM

C:\java\jdk-17.0.4.1



Installation Folder

C:\java\java-2022-06



create start menu entry



create desktop shortcut

↓ INSTALL

← BACK



Eclipse IDE for Java Developers

[details](#)

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

Java 11+ VM

C:\Java\jdk-17.0.4.1



Installation Folder

C:\java\java-2022-06



☒ create start menu entry

☒ create desktop shortcut

▶ LAUNCH

show readme file

open in system explorer

keep installer

◀ BACK

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 'jdk-17.0.4.1' 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

☐ Add project to working sets

Working sets:

Module

☒ Create module-info.java file

New module-info.java

Create module-info.java

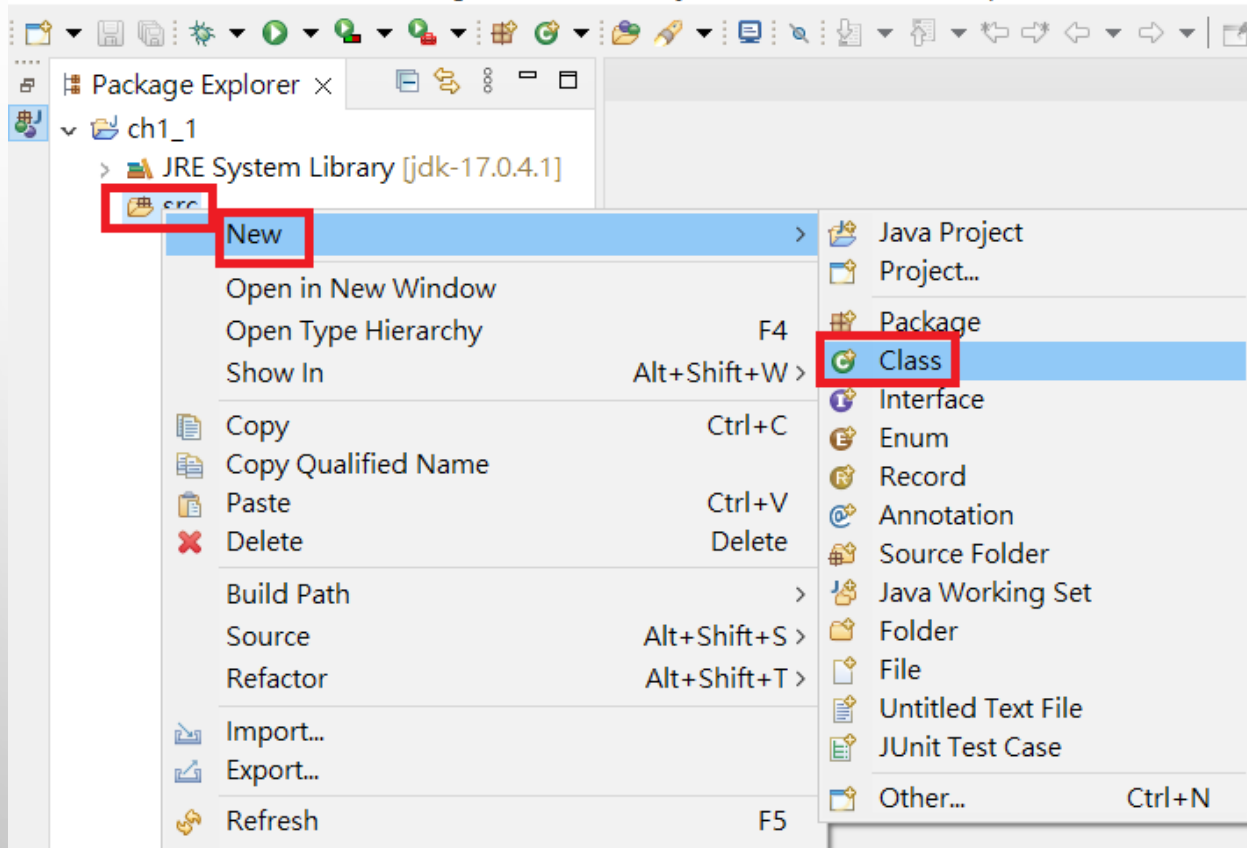
Create a new module-info.java file.

Module name:

☒ Generate comments (configure templates and default value [here](#))

ch1_1 - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help



New Java Class

Java Class

⚠ Type name is discouraged. By convention, Java type names usually start with an uppercase letter



Source folder:

Package:

☐ Enclosing type:

Name:

Modifiers: ☒ public ☐ package ☐ private ☐ protected
☐ abstract ☐ final ☐ static
☒ none ☐ sealed ☐ non-sealed ☐ final

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



Finish

Cancel

HELLO WORLD

```
package ch1_1;
```

```
public class hello {
```

```
    public static void main(String[] args) {
```

```
        System.out.printf("Hello world!");
```

```
        int a=1;
```

```
        a=a+1;
```

```
        a=a+1;
```

```
        a=a+1;
```

```
        a=a+1;
```

```
    }
```

```
}
```

