

CHAPTER 1

- Java平台概論

學習目標

- 簡介Java版本
- 認識Java SE、Java EE、Java ME
- 瞭解JVM、JRE與JDK
- 下載、安裝JDK

Java不只是語言

- 從1995年至今，已經20多個年頭
- 不僅是個程式語言，也代表處理運算解決問題的平台（ Platform ）
- 代表著原廠、各個資訊廠商、社群、開發者與使用者的密切關係。

Java誕生

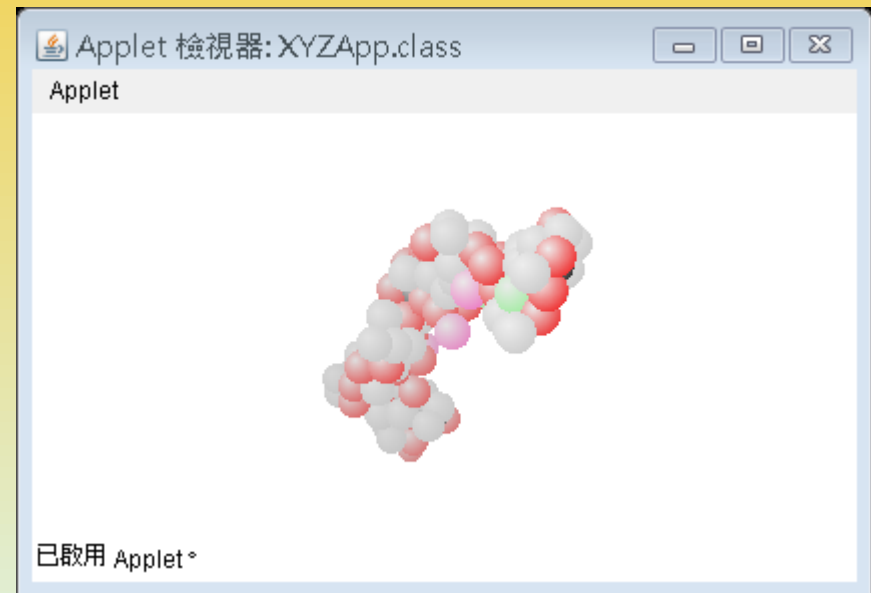
- Java最早是Sun公司「綠色專案」(Green Project) 中撰寫Star7應用程式的程式語言，當時名稱不是Java，而是取名為Oak
- 後來發現Oak名稱已經被註冊了，工程師們邊喝咖啡邊討論著新名稱，最後靈機一動而改名為Java

Java誕生

- 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
- 全球資訊網（World Wide Web）興起，Java Applet成為網頁互動技術代表

Java誕生

- 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	代碼名稱	釋出日期
JDK 1.1.4	Sparkler (煙火)	1997/9/12
JDK 1.1.5	Pumpkin (南瓜)	1997/12/3
JDK 1.1.6	Abigail (聖經故事人物名稱)	1998/4/24
JDK 1.1.7	Brutus (羅馬政治家名稱)	1998/9/28
JDK 1.1.8	Chelsea (足球俱樂部名稱)	1999/4/8
J2SE 1.2	Playground (遊樂場)	1998/12/4
J2SE 1.2.1	無	1999/3/30
J2SE 1.2.2	Cricket (蟋蟀)	1999/7/8
J2SE 1.3	Kestrel (紅隼)	2000/5/8
J2SE 1.3.1	Ladybird (瓢蟲)	2001/5/17
J2SE 1.4.0	Merlin (魔法師名稱)	2002/2/13
J2SE 1.4.1	Hopper (蚱蜢)	2002/9/16
J2SE 1.4.2	Mantis (螳螂)	2003/6/26
J2SE 5.0	Tiger (老虎)	2004/9/29
Java SE 6	Mustang (野馬)	2006/12/11
Java SE 7	Dolphin (海豚)	2011/7/28

- **Oracle** 宣佈合併Sun(昇陽)
- Java8 – Java ...
- 程式語法不斷簡新，程式庫不斷充實

三大設計平台

- **Java SE** (Java 2 Platform, Standard Edition)
- **Java EE** (Java 2 Platform, Enterprise Edition)
- **Java ME** (Java 2 Platform, Micro Edition)

前端：指的是眾多瀏覽器端技術——HTML、CSS、Javascript，以及JavaFX技術

後端：開發人員有更堅實的Java(JSP)、資料庫（SQL）訊息傳遞和Web服務（SOAP/REST）技能。

Java SE

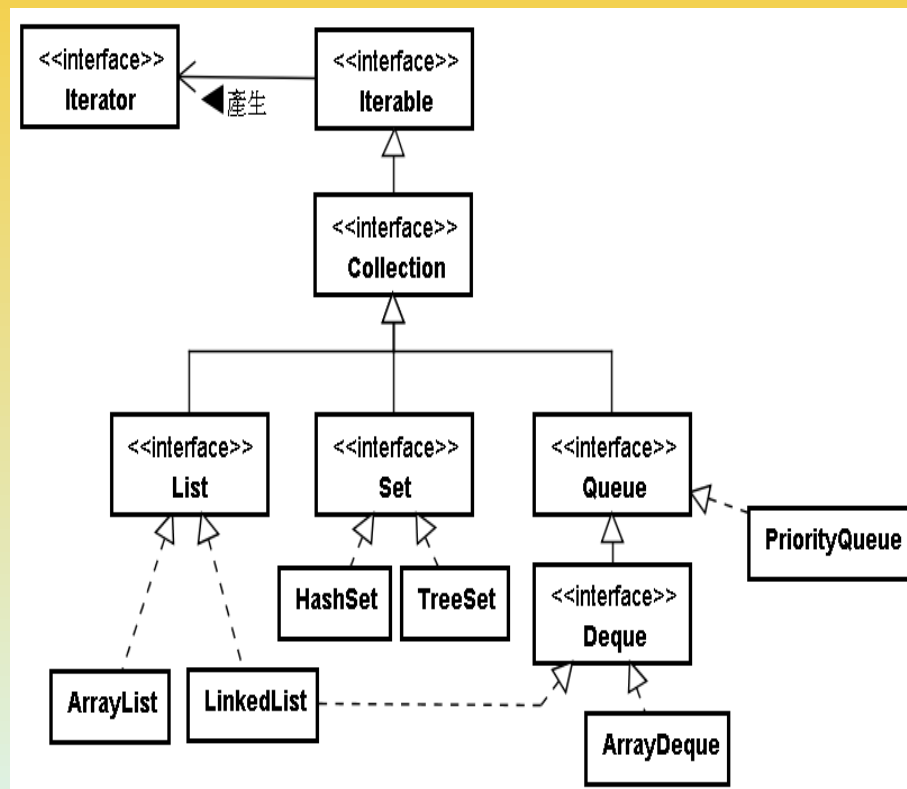


JCP

- 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

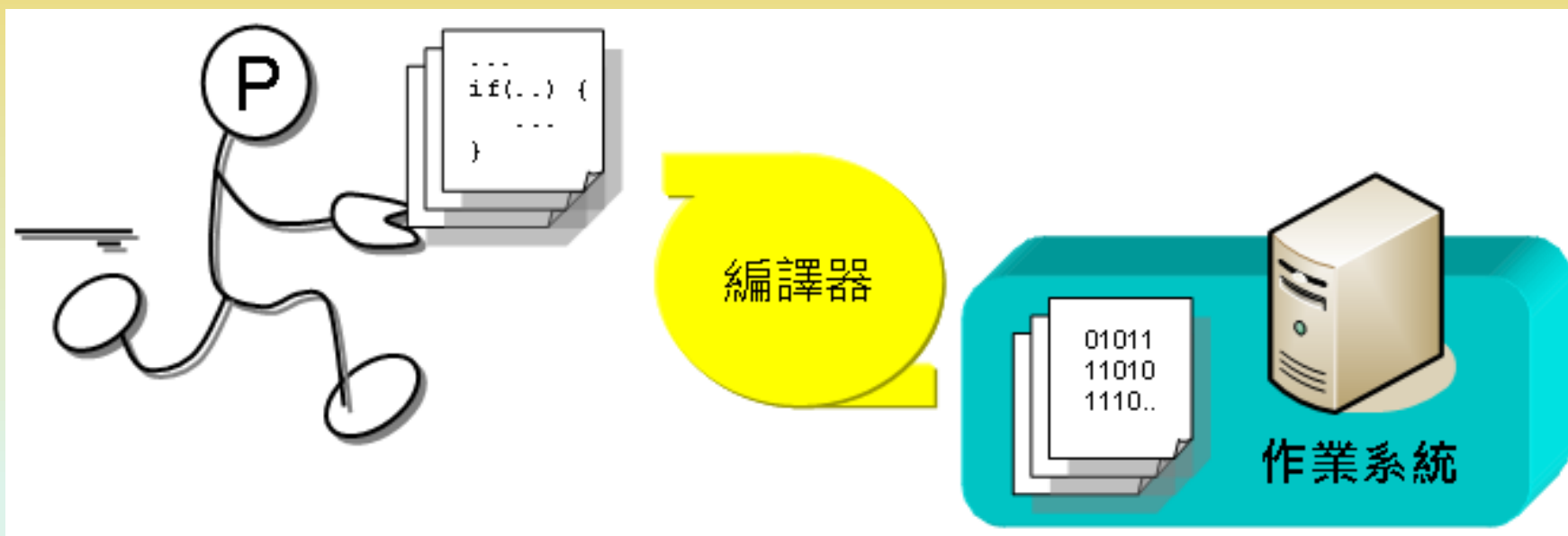
學習目標概況

- 瞭解JVM/JRE/JDK
- 瞭解封裝、繼承、多型
- 掌握常用Java SE API
- 學習集合容器
- 學習設計模式
- 熟悉IDE開發工具

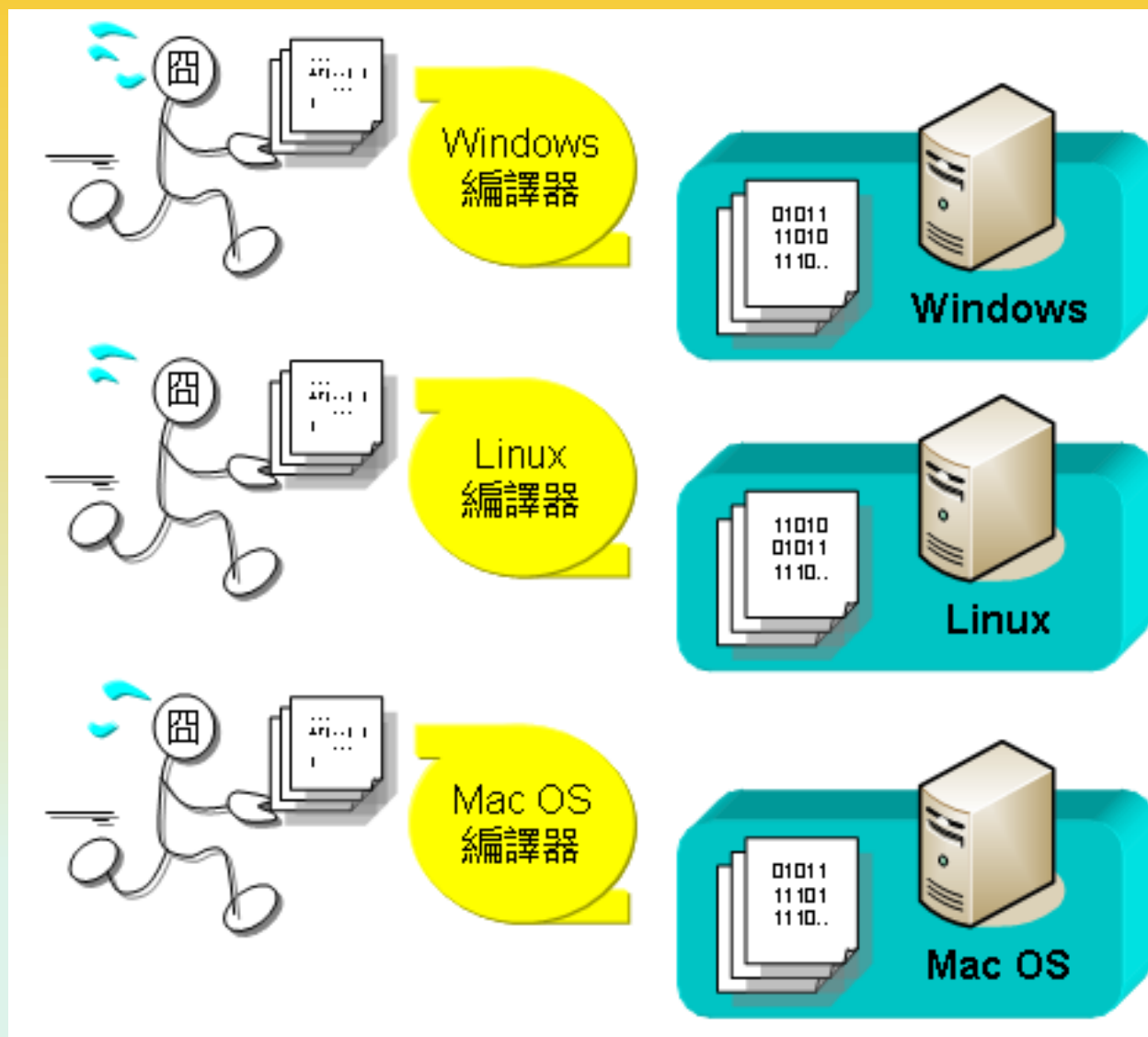


什麼是JVM？

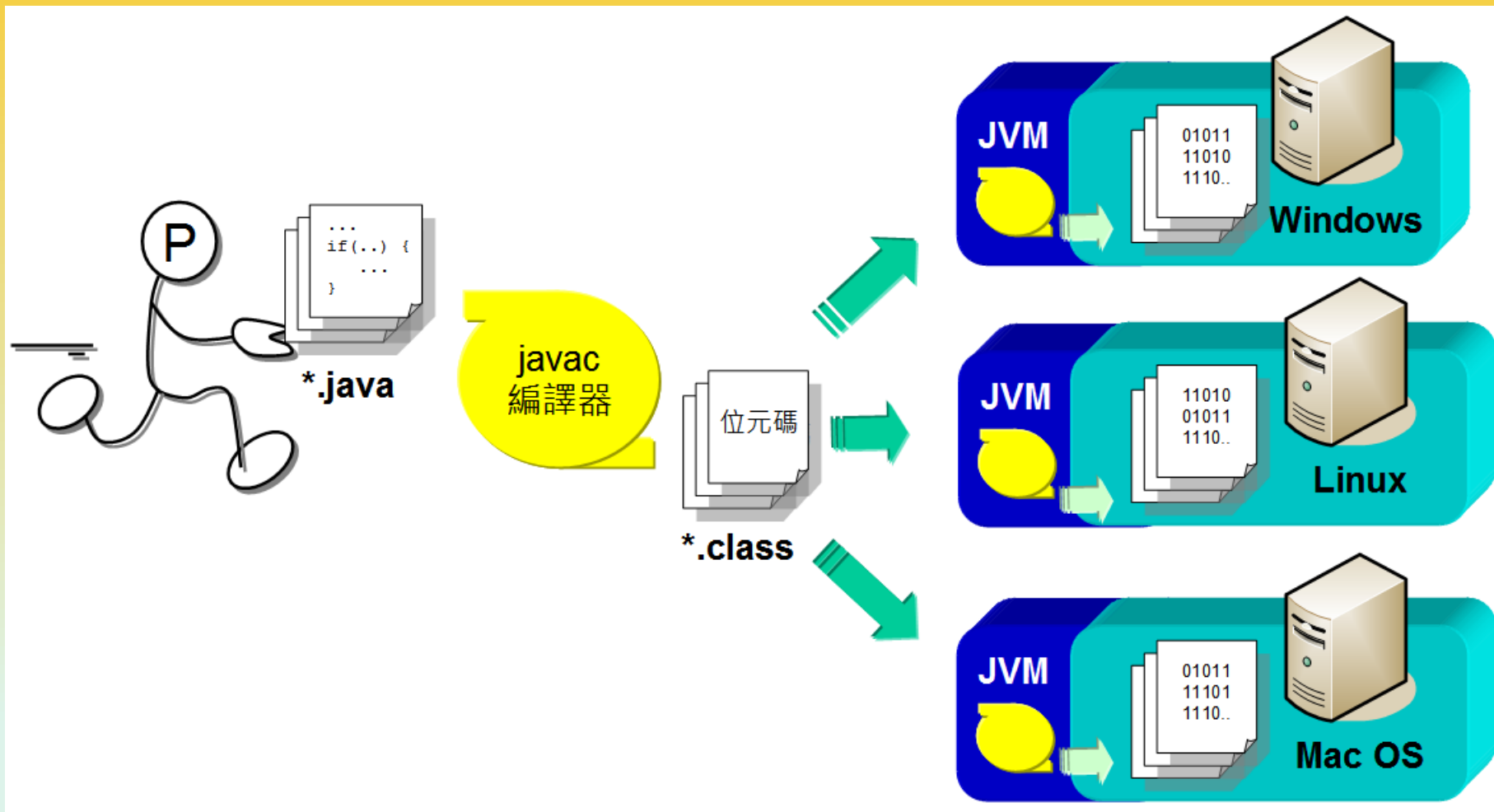
- JVM讓Java可以跨平台，但是跨平台是怎麼一回事？



什麼是JVM？



什麼是JVM？



什麼是JVM？

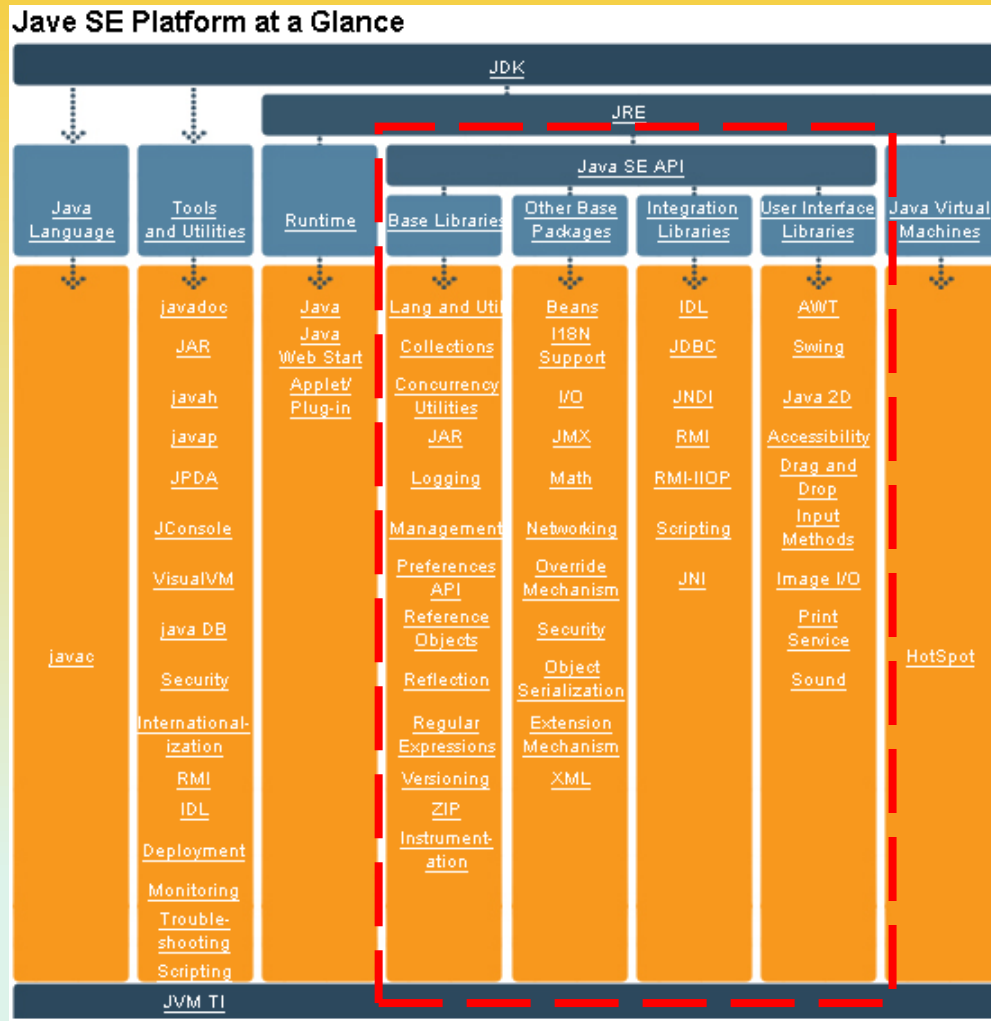
- 對Java程式而言，只認識一種作業系統，這個系統叫JVM，位元碼檔案（副檔名為.class的檔案）就是JVM的可執行檔。
 - Java程式理想上，並不用理會真正執行於哪個平台，只要知道如何執行於JVM就可以了
 - JVM實際上就相當於Java程式的作業系統，JVM就負責了Java程式的各種資源管理

區分JRE與JDK

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

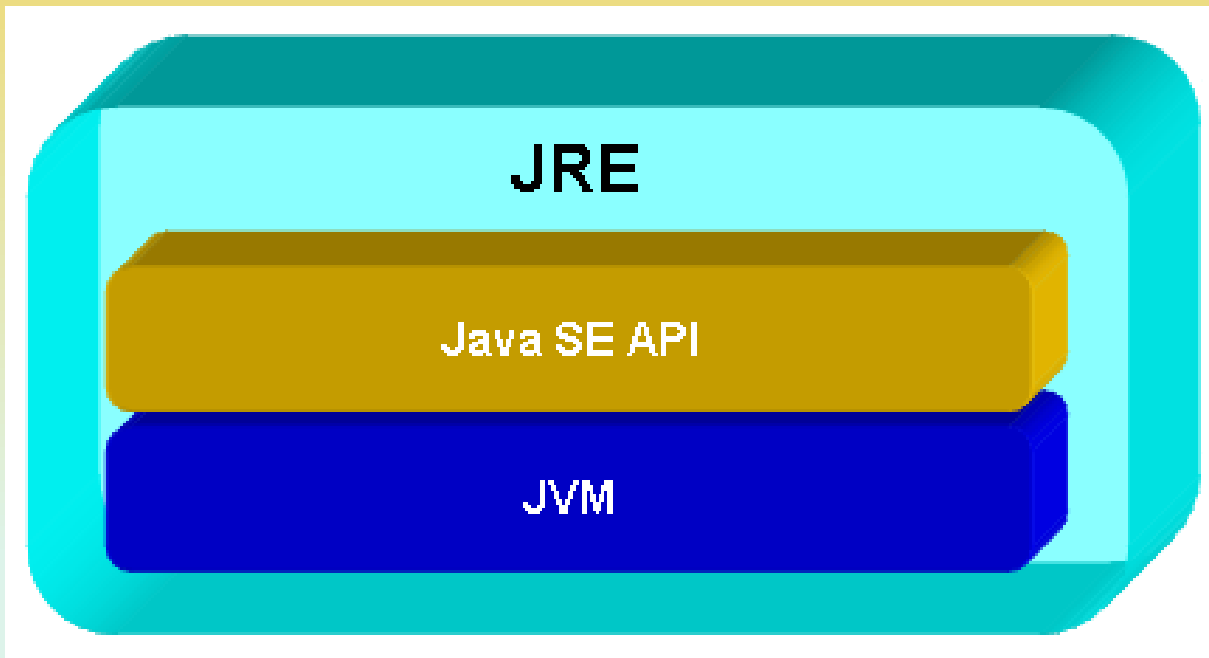
- System、out、println這些名稱，都是標準程式中規範的名稱
- 在撰寫一個Java程式時，使用System類別（Class）的out物件（Object）的println（）方法（Method）

區分JRE與JDK



JRE

- Java Runtime Environment就是Java執行環境，簡稱JRE，包括了Java SE API與JVM



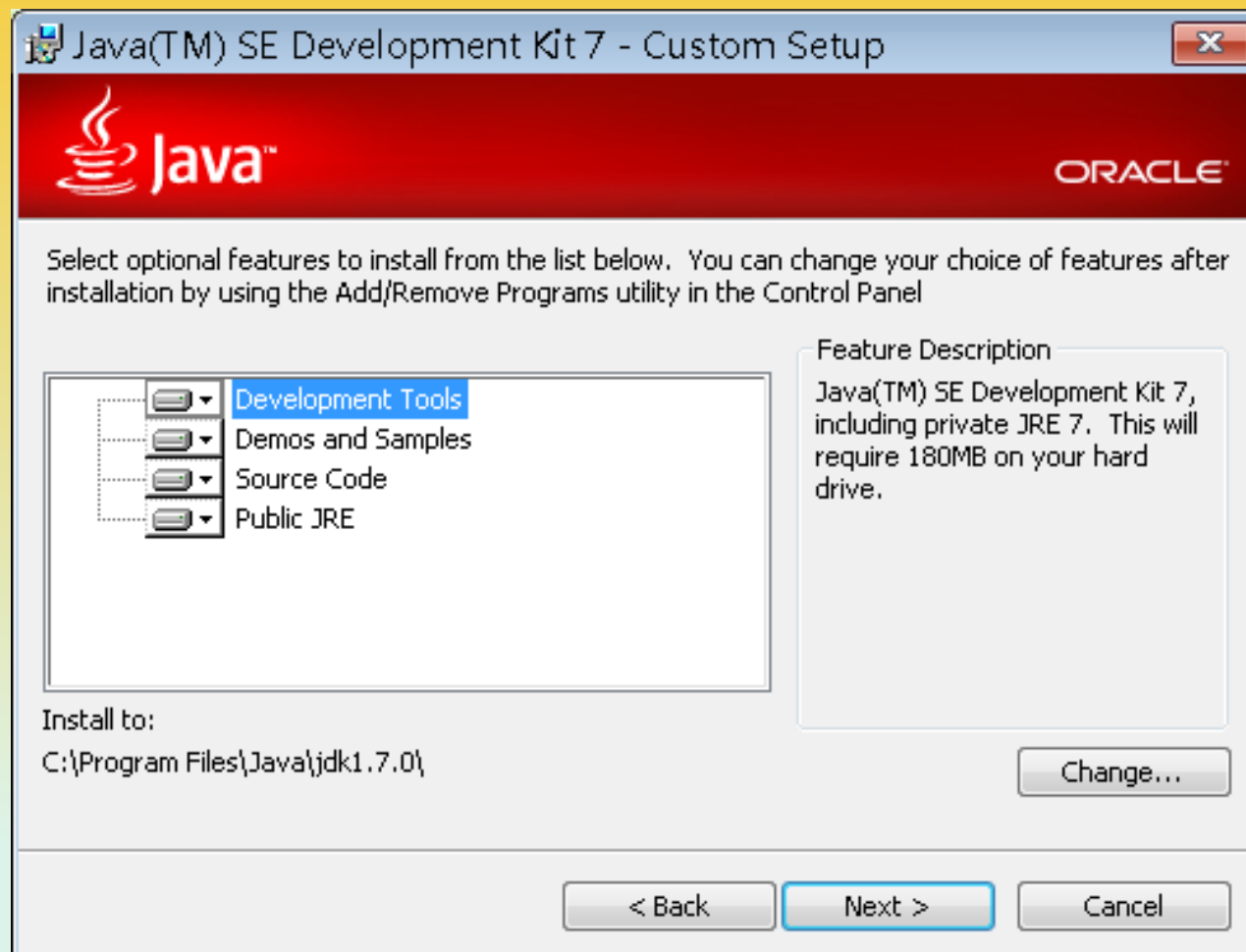
區分JRE與JDK

- 像編譯器這樣的工具程式是由誰提供？答案就是JDK，全名為~~Java Developer Killer~~！呃！不對！是Java Development Kit！

區分JRE與JDK

- 撰寫Java程式時才需要**JDK**，如果你的程式只是想讓使用者的電腦執行～
那使用者只要裝**JRE**就可以了。

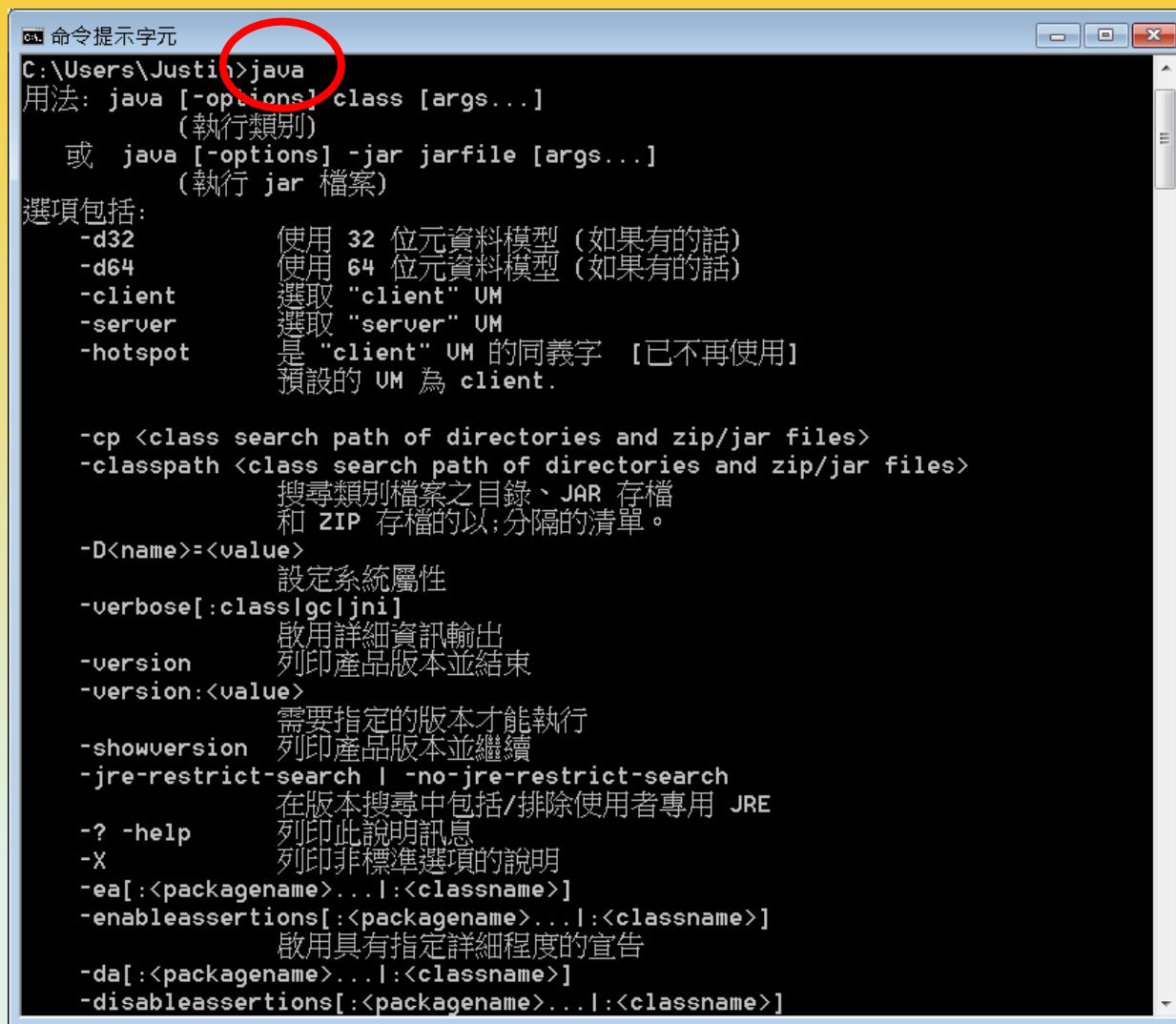
下載、安裝JDK



下載、安裝JDK

1. 下載安裝 JDK (開發套件API等)
 - www.oracle.com 甲骨文官網
2. 下載安裝 Netbean IDE (整合式開發軟體)
3. 若需要在 console 環境(文字命令模式)
Compile Java 原始碼，必須設定JDK bin path
(參考下個單元 CH02)

電腦是否安裝JDK?



```
命令提示字元
C:\Users\Justin>java
用法: java [-options] class [args...]
       (執行類別)
或 java [-options] -jar jarfile [args...]
       (執行 jar 檔案)

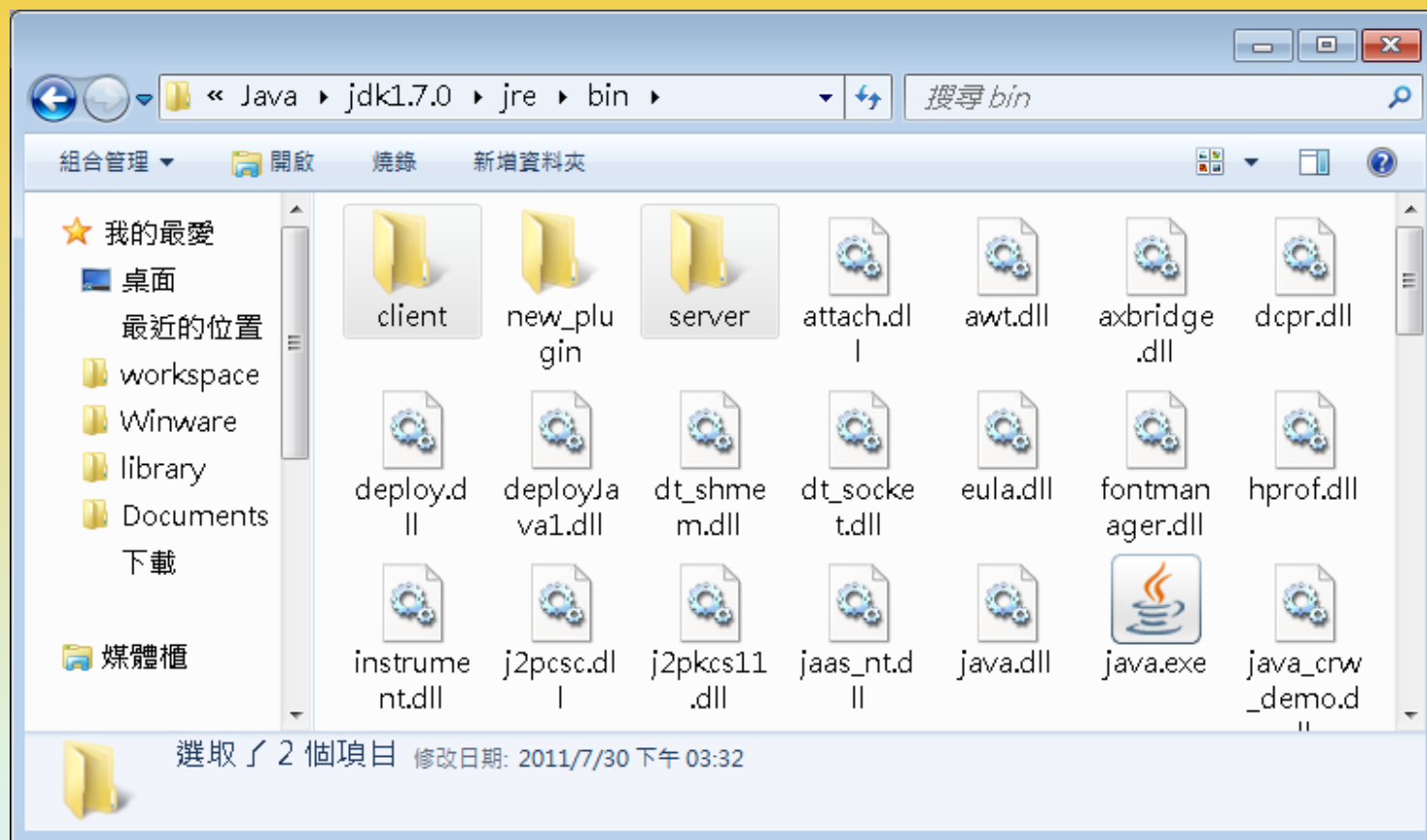
選項包括:
  -d32          使用 32 位元資料模型 (如果有的話)
  -d64          使用 64 位元資料模型 (如果有的話)
  -client       選取 "client" VM
  -server       選取 "server" VM
  -hotspot      是 "client" VM 的同義字 [已不再使用]
                預設的 VM 為 client.

  -cp <class search path of directories and zip/jar files>
  -classpath <class search path of directories and zip/jar files>
                搜尋類別檔案之目錄、JAR 存檔
                和 ZIP 存檔的以;分隔的清單。
  -D<name>=<value>
                設定系統屬性
  -verbose[:class|gc|jni]
                啟用詳細資訊輸出
  -version      列印產品版本並結束
  -version:<value>
                需要指定的版本才能執行
  -showversion  列印產品版本並繼續
  -jre-restrict-search | -no-jre-restrict-search
                在版本搜尋中包括/排除使用者專用 JRE
  -? -help      列印此說明訊息
  -X            列印非標準選項的說明
  -ea[:<packagename>...[:<classname>]]
  -enableassertions[:<packagename>...[:<classname>]]
                啟用具有指定詳細程度的宣告
  -da[:<packagename>...[:<classname>]]
  -disableassertions[:<packagename>...[:<classname>]]
```

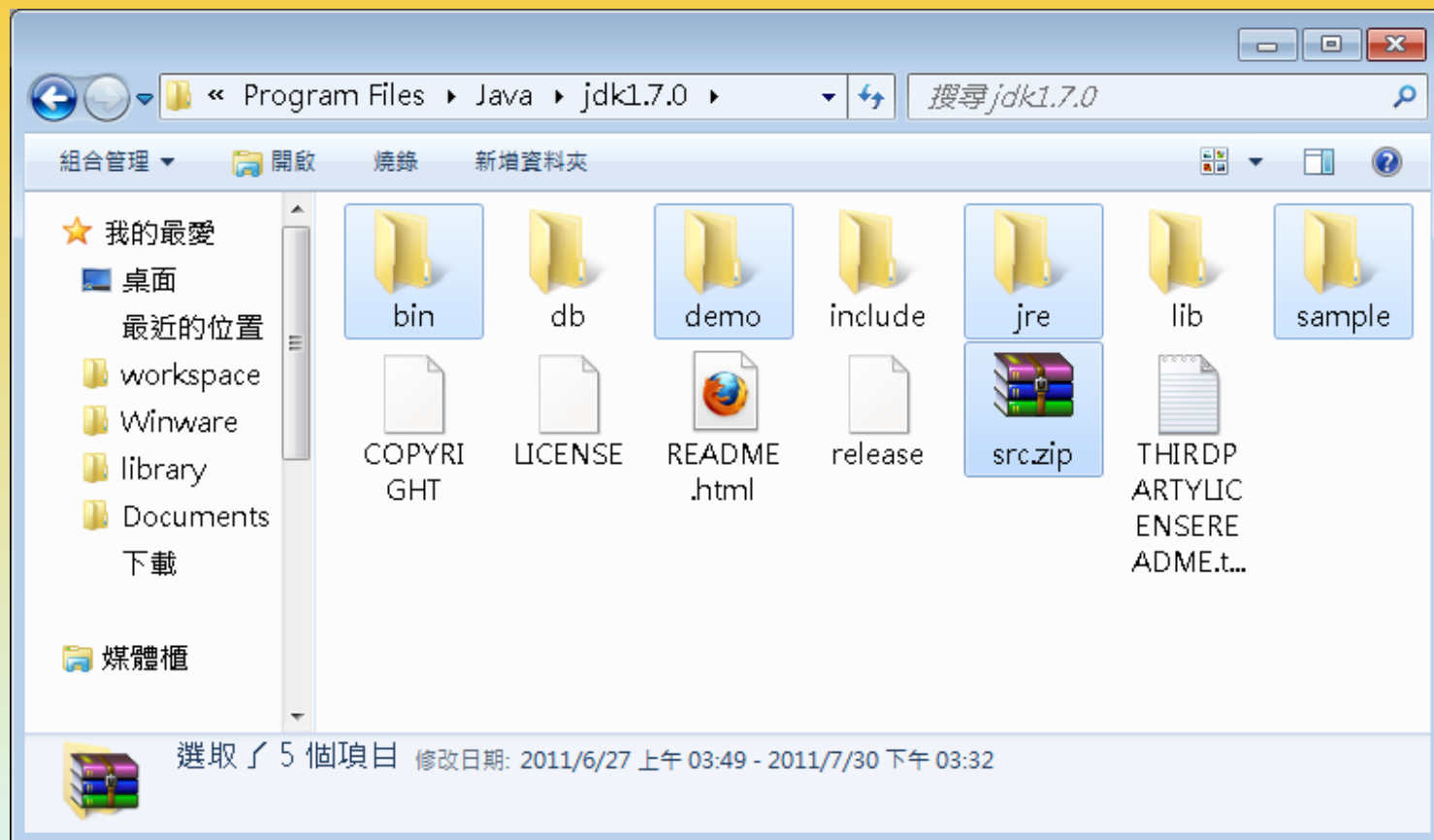
認識JDK安裝內容

- 預設JDK與Public JRE各安裝至
「C:\Program Files\Java\jdk1.8.0\」
及
「C:\Program Files\Java\jre8\」

JDK安裝內容\bin\ (工具程式)



認識JDK安裝內容



認識JDK安裝內容

- 那麼Java SE API編譯好的.class檔放在哪呢？
 - 無論是Private JRE或Public JRE的資料夾中，都會有個lib目錄，其中會有個**rt.jar**檔案，JAR (Java Archive) 檔案是zip壓縮格式，可以使用解壓縮軟體開啟，你就會看到許多編譯好的.class檔案

Java SE

