

程式語言的類別

- ①Machine Language
- ②Assembly Language
- ③High Level Language

語言的轉換方式

- ①Interpreter(直譯式)
- ②Compiler/Assembler(編譯式)
- ③Compiler & Interpreter(混合式)

註:1970~80 年軟體工程方法是使用「結構化程式設計」(structured programming), 1980 年以後開始引入符合人性化的程式設計方法「物件導向程式設計」

Java 起源

由 Sun microsystem 的「James Gosling」所帶領的小組，目的在開發使用在家電控制的程式語言，語言的名稱為「Oak」即 Java 的前身

1995/5 Sun 發表第一個 Java 測試版(1.0 alpha)

1996 Java Development Kit(JDK) 1.0 正式版

1998 Software Development Kit(SDK) 1.2 正式版，此後稱為 Java 2

2000 SDK 1.3

2001 SDK 1.4

2004 JDK 1.5 (JDK5.0)

:

2011 JDK 1.7 (JDK7.0)

2014 JDK 1.8 (JDK8.0)

:

2019 JDK 14

:

2021 JDK 16/17

2022 JDK 18/19

2023 JDK 20

:

:

註: Java SE 8 (LTS) - Oracle 於 2030 年 12 月停止更新

Java SE 11 (LTS) - Oracle 於 2026 年 9 月停止更新

Java SE 17 (LTS) - Oracle 於 2029 年 9 月或之後停止更新

安裝 JDK

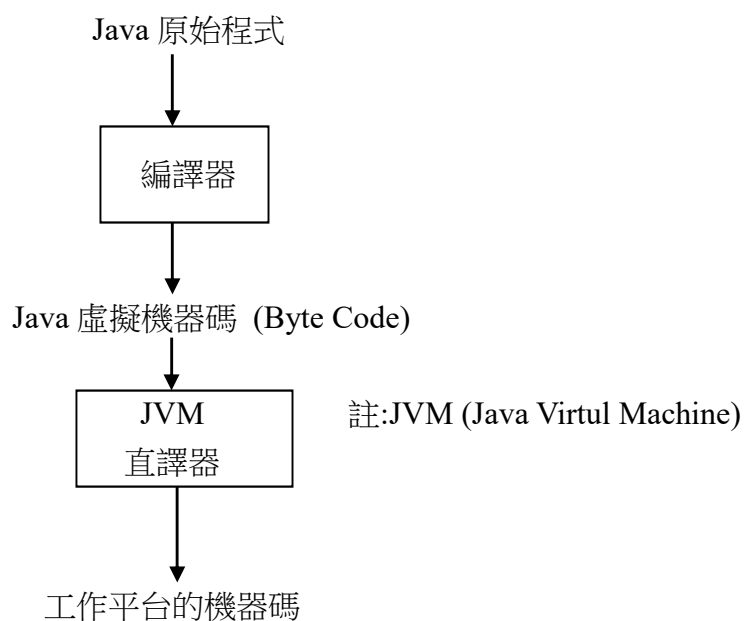
Step1:到 Oracle 官網下載「jdkxxx」

註:建議將 Documentation 一起下載

Step2:安裝

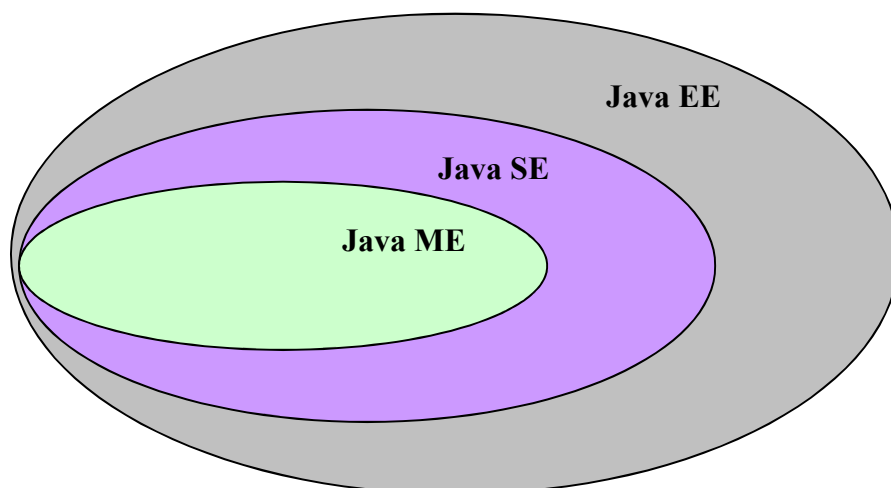
Step3:設定環境變數「PATH」

Java 的執行方式



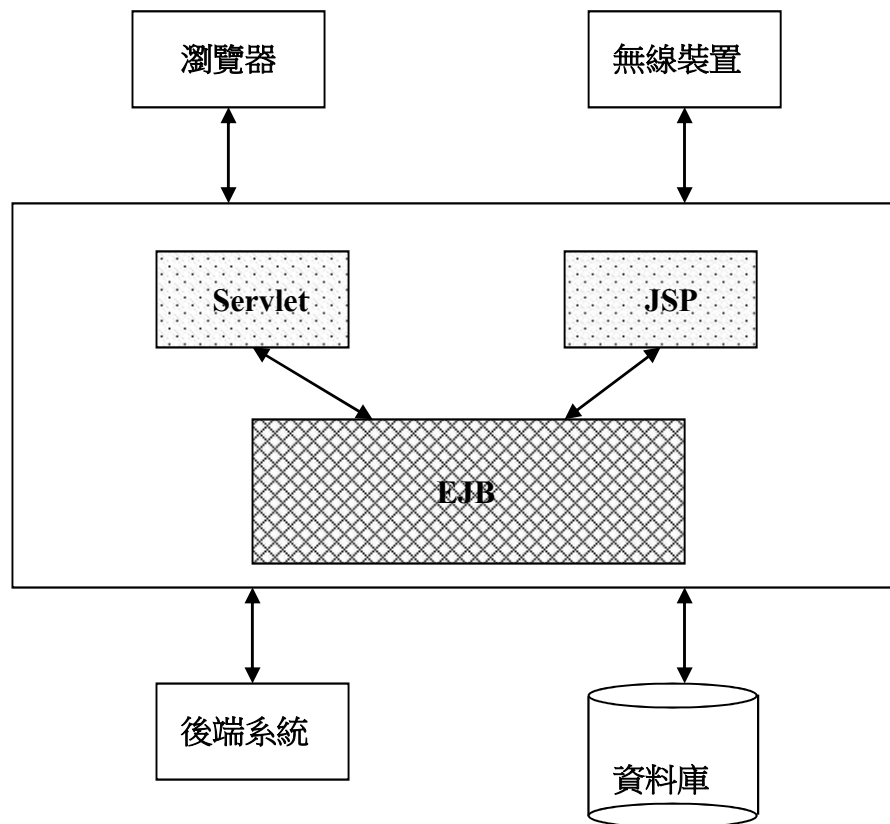
Java 的開發平台

說明:開發平台並非產品而是規格, 在規格之中明訂設計軟體所必須遵循的協議



①JavaEE:Java Platform, Enterprise Edition

- ❶Java 企業版
- ❷整合企業級伺服器端開發平台



- 說明:
- Servlet:是在 Server 端執行的 Java 程式, 用來取代 CGI 的方案
 - JSP:Java Server Pages, 是在 HTML 檔案中內嵌 Java code 產生動態資料
 - EJB:Enterprise JavaBean,定義了伺服器端元件的撰寫方式,提供
Business Rule 與 DataBase 間的溝通,同時也提供更完整的安全、交易、資料庫存取控制

②Java SE:Java Platform, Standard Edition

- ❶Java 標準版, 是 Java 平台的核心
- ❷主要用於開發桌面應用程式
- ❸提供 Java Applet 標準類別

③Java ME:Java Platform, Micro Edition

- ❶Java 微型版
- ❷消費性電子商品與內嵌式裝置的開發平台
如, 行動電話、PDA、汽車導航、Internet 影像電話、TV-box
- ❸用來設計在有限運算能力、網路連接(通常是無線)以及圖像式使用者介面功能的裝置上

撰寫 Java 程式

①以小寫為主, 區分大小寫,每個敘述必須以「;」結尾, 一列中可以有多個敘述, 當敘述太長時可分成若干列而不用加上連接符號, 如

```
BufferedReader input = new BufferedReader(  
                                new FileReader( new File("data.txt")) );
```

②每一個 Java 程式檔中只能有一個 class 的存取權限設成「**public**」, 而且必須以此 class 的名稱做為**主檔名**

③class 的內容為屬性的宣告與方法, 除屬性外 class 與方法必須以「{ }」標示其定義的區域

④必須有一個 class 的內容有 main 方法, main 方法是程式執行時的進入點, main 方法的語法如下:

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

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

⑤程式的首列使用「**package**」來指定 class 檔儲存的路徑, 其次以「**import**」叫用 JDK 所提供的標準類別或自訂類別, 而後是 class 主體

⑥程式中可以適度加入一些說明, 以方便邏輯的瞭解, 方式有:

❶ `//`:單列式的註解

❷ `/* */`:多列式的註解

❸ `/** */`:內容會被加以系統說明文件的多列式註解

```
public class One  
{  
    :  
    :  
}  
  
class Two  
{  
    public static void main(String [ ] args) ← 程式進入點  
    {  
        :  
        :  
    }  
}
```

Commandline 如何編譯與執行一

編譯命令

javac 原始程式的檔名

執行命令

java class 檔的主檔名

package 的用法

- ①語法: **package** 路徑;
- ②必須在程式的首列

import 的用法

- ①語法: **import** 套件路徑.類別名稱;
- ②用途:在執行時將套件路徑下的類別匯入
- ③類別名稱可以用「*」取代

JDK 的套件

- ①java.lang
- ②java.io
- ③java.awt
 - javax.swing
- ④java.awt.event
 - javax.swing.event
- ⑤java.util
- ⑥java.sql
- ⑦javax.sql
- ⑧java.nio.file