



第一章 認識 Java

學習目標

- ✚ Java 的發展歷史
- ✚ Java 虛擬機器 (JVM) 所扮演的角色
- ✚ Java 未來的發展
- ✚ 撰寫第一個 Java 程式





1.1 程式的介紹

Java 於 1995 年誕生，由美國加州的昇陽電腦公司（Sun Microsystems, Inc.）所推出，是一種能夠跨平台（cross platform）使用的程式語言。

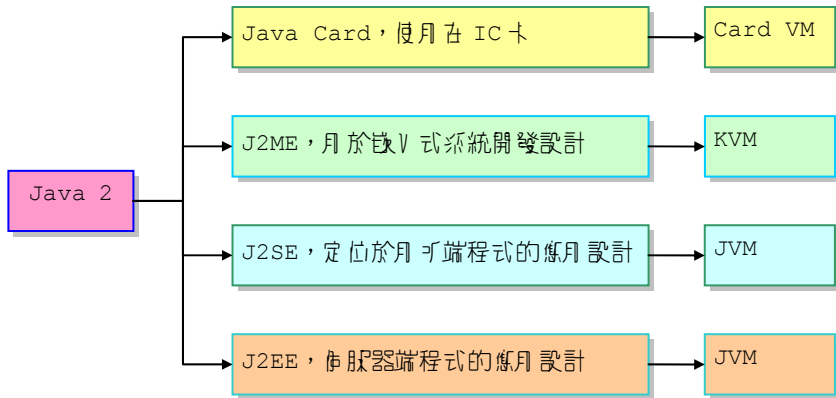
1.1.1 Java 的發展歷史

Oak 程式語言是 Java 語言的前身。後來 SUN 將 Oak 修改而成為 Java，並將這種技術轉移到 Web 網路上。

Web 於 1994 年在 Internet 上開始流行。



下圖為 Java 的四大應用領域：



Java 的架構可分為 4 個部分：「Java Card」、「J2ME」、「J2EE」與「J2SE」。



下表列出 Java 各版本的代號，以及未來版本預計的代號介紹：

版本	代號	中文名稱	發佈日期	備註
JDK 1.1.4	Sparkler	火花	1997/09/12	
JDK 1.1.5	Pumpkin	南瓜	1997/12/03	
JDK 1.1.6	Abigail	阿比蓋爾	1998/04/24	女子名
JDK 1.1.7	Brutus	布魯圖	1998/09/28	古羅馬政治家和將軍名
JDK 1.1.8	Chelsea	切爾西	1999/04/08	城市名
J2SE 1.2	Playground	運動場	1998/12/04	
J2SE 1.2.1	none	無	1999/03/30	
J2SE 1.2.2	Cricket	蟋蟀	1999/07/08	
J2SE 1.3	Kestrel	美洲紅隼	2000/05/08	
J2SE 1.3.1	Ladybird	瓢蟲	2001/05/17	
J2SE 1.4.0	Merlin	灰背隼	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	J2SE 1.5.0
J2SE 6.0	Mustang	野馬	未發佈	J2SE 1.6.0
J2SE 7.0	Dolphin	海豚	未發佈	J2SE 1.7.0



1.1.2 Java 的特點

嬌小且完美的語言

Java 的語法簡潔、名稱，容易撰寫且易於維護。

具物件導向的功能

Java 具有物件導向的特點，同時它是一個全新的語言，因而得以將物件導向的特性發揮到極致。

完美地接納國際網路

Java 所撰寫的程式很容易地在 Netscape 或 IE 裡呈現，也可藉此讓所有能夠上網的電腦都能執行 Java 程式。



通用 的語言

凡是 C/C++ 所能做到的事，Java 也都能做到，而且能以更簡潔的程式碼來呈現。

跨平台的語言

由 Java 語言所撰寫的程式碼，可在不修改程式碼的情況下，便能在不同的作業系統裡執行。

具有豐富的函數庫

Java 語言背後有龐大的函數庫支撐著。這些函數庫包含了繪圖函數庫、圖形使用者介面函數庫（graphical user interface）、網路設計函數庫...等等。



1.2 Java 的虛擬機器

高階語言的執行方式有編譯式與直譯式兩種：

常見的編譯式語言如 C、FORTRAN 與 COBOL 等均是。

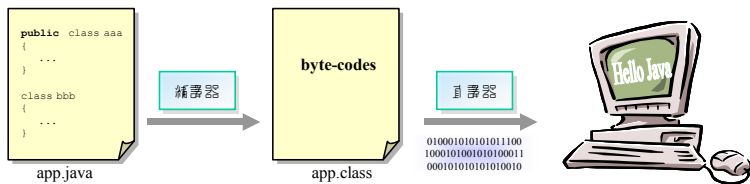
直譯式的語言如 Basic、dBASE 及其他 Script 語言等。



位元碼 (byte-codes)

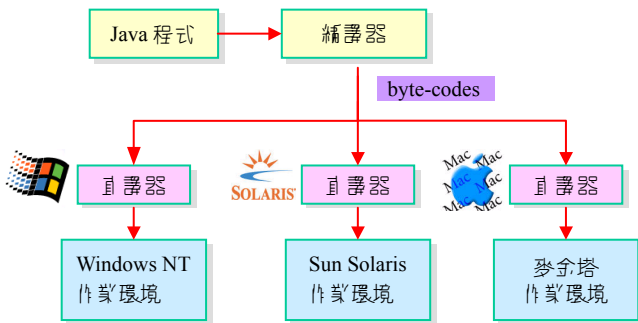
經過編譯的程序，將程式碼轉換與平台無關（platform-independent）的機器碼，稱為「位元碼」（byte-codes）。

下圖說明了 Java 相關的執行流程：





byte-codes 最大的好處是 --可跨越平台來執行：

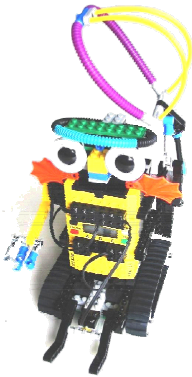




1.3 Java 的未來發展

在學術領域裡，Java 已成為最重要的教學程式語言之一。

許多產品--手機與 PDA，也是使用 Java 來延伸。





1.4 撰寫第一個 Java 程式

Java 程式可分為下列兩種：

- (1) Java 應用程式（Java application）。
- (2) 應用在 www 上的小程序（Java applet）。



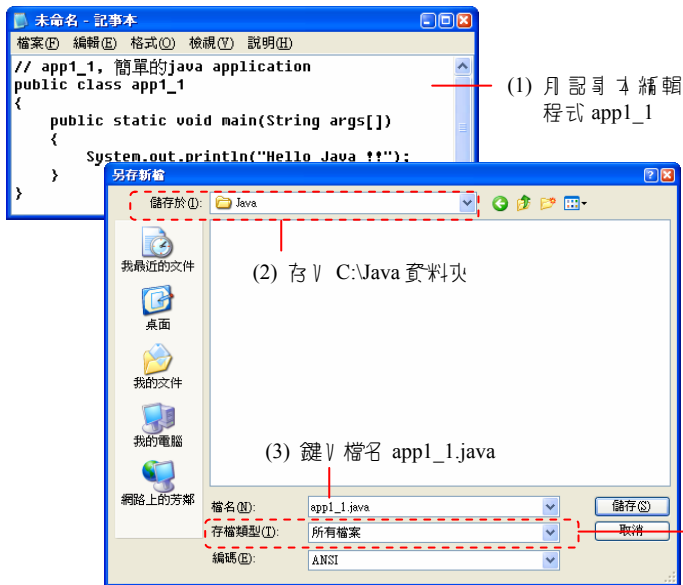
1.4.1 編譯與執行 Java application

下面是 app1_1 的程式碼，app1_1 經編譯和執行後，會在 Dos 視窗上顯示 Hello Java!! 字串：

```
01 // app1_1, 簡單的java application
02 public class app1_1
03 {
04     public static void main(String args[])
05     {
06         System.out.println("Hello Java !!");
07     }
08 }
```



把 app1_1 鍵入 記事本 中，檔名設為 app1_1.java：





app1_1 的執行流程如下圖所示：

```
C:\> 命令提示字元
Microsoft Windows XP [版本 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Administrator>cd c:\Java

C:\Java>javac app1_1.java — 編譯程式

C:\Java>java app1_1 — 直譯程式 — 切換路徑到檔案存放處
Hello Java !!
C:\Java> — 執行結果
```



1.4.2 編譯與執行 Java applet

要執行 Java applet，必須要有 applet 的 byte-codes 與瀏覽器。

App1_2.java 是個簡單的 applet，可在瀏覽器視窗上顯示 "Hello Java!!" 字串：

```
01 // App1_2, Java applet
02 import java.awt.*;
03 import java.applet.*;
04 public class App1_2 extends Applet
05 {
06     public void paint(Graphics g)
07     {
08         g.drawString("Hello Java!!",50,50);
09     }
10 }
```



請用記事本編輯 App1_2.java :

```
App1_2.java - 記事本
檔案(F) 編輯(E) 格式(O) 檢視(V) 說明(H)

import java.awt.*;
import java.applet.*;
public class App1_2 extends Applet
{
    public void paint(Graphics g)
    {
        g.drawString("Hello Java!",50,50);
    }
}
```

本檔是以大寫 A 開頭的檔名代表 applet 程式，因此必須把檔名存成 App1_2.java，而非 app1_2.java。



編譯 Java applet

編譯 App1_2.java :

```
c:\Java> javac App1_2.java
```

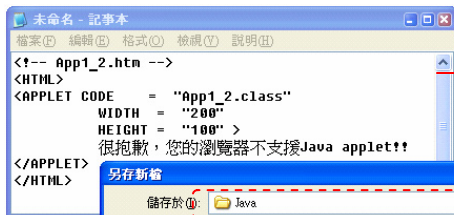
準備好 HTML 檔案

下面是一個典型的 html 範例檔案：

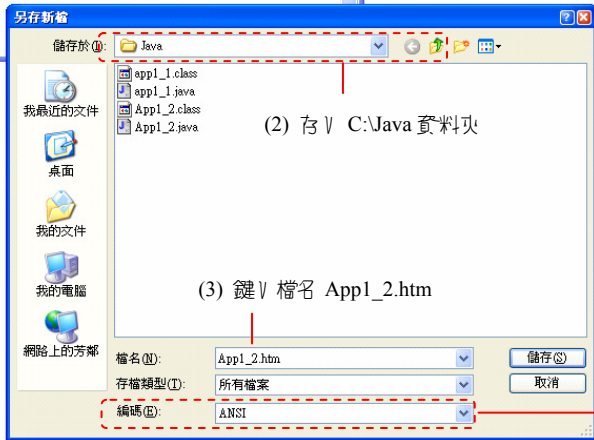
```
01 <!-- App1_2.htm -->
02 <HTML>
03 <APPLET CODE    = "App1_2.class"
04             WIDTH = "200"
05             HEIGHT = "100" >
06             很抱歉，您的瀏覽器不支援 Java applet!!
07 </APPLET>
08 </HTML>
```



將 html 的語法編輯好，並將它存檔為 App1_2.htm：



(1) 用記事本編輯
htm 檔



(2) 存 V C:\Java 資料夾

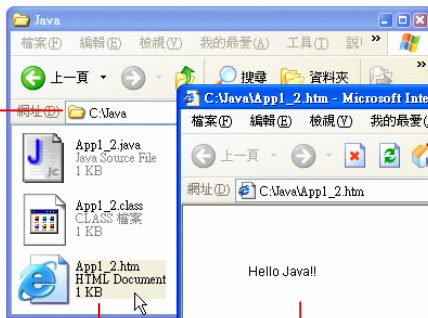
(3) 鍵 V 檔名 App1_2.htm

(4) 存檔類型請選擇
「所有檔案」

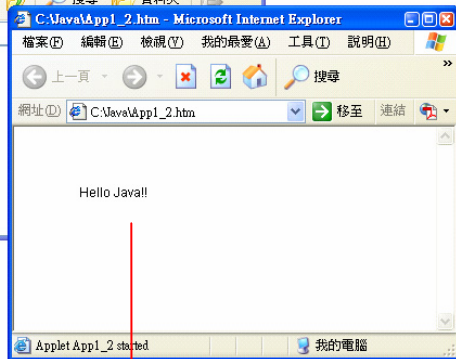


執行 Java applet

(1) 找到 C:\Java
資料夾



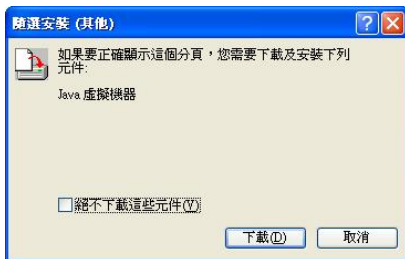
(2) 開啟 App1_2.htm



(3) applet 的執行結果



如果使用 Windows XP 搭配 IE 6.0，且從未下載過 JVM（Java 虛擬機器），則 XP 會跳出一視窗，要您下載它：





利用 applet 檢視器執行 Java applet

```
C:\Java> appletviewer App1_2.htm
```

按下 Enter 鍵就會看到「Applet 檢視器」視窗與執行結果：





-The End-