

第二章 Java 程式架構與輸入/輸出

2-1 Java 程式架構

程式語言的標準規範可區分為兩大類：**語法 (Syntax)** 與 **語意 (Semantic)**，前者相當於自然語言(如英文)的『**字彙**』，後者類似自然人使用的『**文法**』。吾人使用正確的字彙與文法書寫文章，才能正確的表達意思，讓旁人明瞭文章的內容。當然認識某種程式語言，也必須由字彙及文法開始，本書將一步一步的介紹 Java 程式語法。

2-1-1 Java 程式的組織

從另一個角度而言，編寫文章必須將其內容分為若干個章節，每章節也許會包含多個段落；也就是說，一篇文章(或書本)是由若干個章節與段落所構成。同樣的，一只程式也可能由多個程式模組所構成，而每一程式模組內也可能包含若干個函數程式。圖 2-1 為 Java 原始程式結構的抽象圖，重點說明如下：

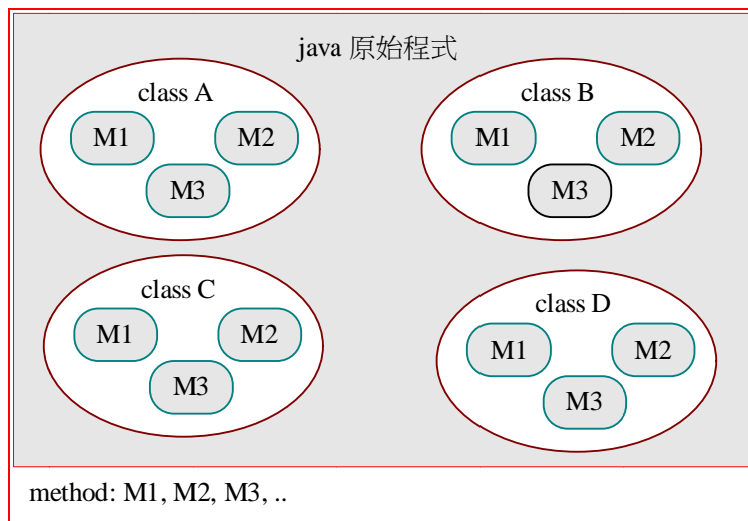


圖 2-1 Java 程式的組織架構

- Java 原始程式的**副檔名**，必須是 java，譬如 **Demo.java**。
- 一個原始程式可能包含若干個『**類別**』(**class**) 區塊 (或稱程式)。當原始程式經由 javac 編譯後，每一類別區塊相對應產生一個『**Bytecode**』檔 (或稱中介程式)，其副檔名為 class，譬如 Demo.class。

- 一個類別程式可能包含若干個『方法成員』(**Method member**) 與『變數成員』(**Variable member**); 類別之間的方法與變數各自獨立，名稱與內容並不衝突。
- 某一類別區塊內包含 **main 方法**者，則稱之為『主類別』。主類別名稱必須與檔案名稱相同，而且大小寫也必須相同。
- 程式是由主類別內的 **main 方法**成員起始執行，再透過它呼叫執行其他方法成員，或產生其他物件。

一只程式到底包含多少個『類別』區塊，一個類別內又包含多少個『方法』或『變數』成員，並沒有一定的規範，而是依照程式設計師自行規劃。某些類別是由套件(如同庫存函數)導入，並不一定需由設計師自己編寫。但無論如何，一只程式內至少必須包含一個類別，而該類別內也至少有一個 **main 方法**。在第六章之前，本書僅利用一個主類別與 **main 方法**來介紹 Java 語法，第六章之後，再介紹主類別內包含多個方法成員；到了物件導向章節再介紹多個類別之間的運作。

2-1-2 Java 程式的架構

圖 2-2 為 Java 程式的基本架構，重點說明如下：

- **單行註解**『`//`』：程式檔案中任何地方出現雙斜線 (`//`)，之後該行所有文字都是註解(跟程式運作無關，純粹是告訴其他程式員或提醒自己，這段程式是做什麼用的)。
- **多行註解**『`/* */`』：由『`/*`』開始到『`*/`』結束，之間可能是同一行或若干行，其中所包含的文字都是註解(非程式部分)。
- **導入套件**『`import java.io.*`』：導入已編寫完成的程式套件。Java 程式套件是以樹狀結構儲存，最頂端樹根為 `java` 再往下延伸，每一節點儲存某一特殊功能的套件(如 `network`、`io`、`util`、)，相關套件的物件再往下節點存放。
- **類別宣告**『`public class 類別名稱 { ...}`』：關鍵字 `class` 表示宣告產生一個類別，後面緊接著『類別名稱』(如 `class Demo`，與宣告整數變數 `int a` 相同)；另外，`public` 表示該類別為『通用』屬性，可任意呼叫執行。類別內容(或稱實體、`body`)由緊接著左大括號與右大括號(`{ ... }`)包起來。主類別名稱必須與檔案名稱相同。

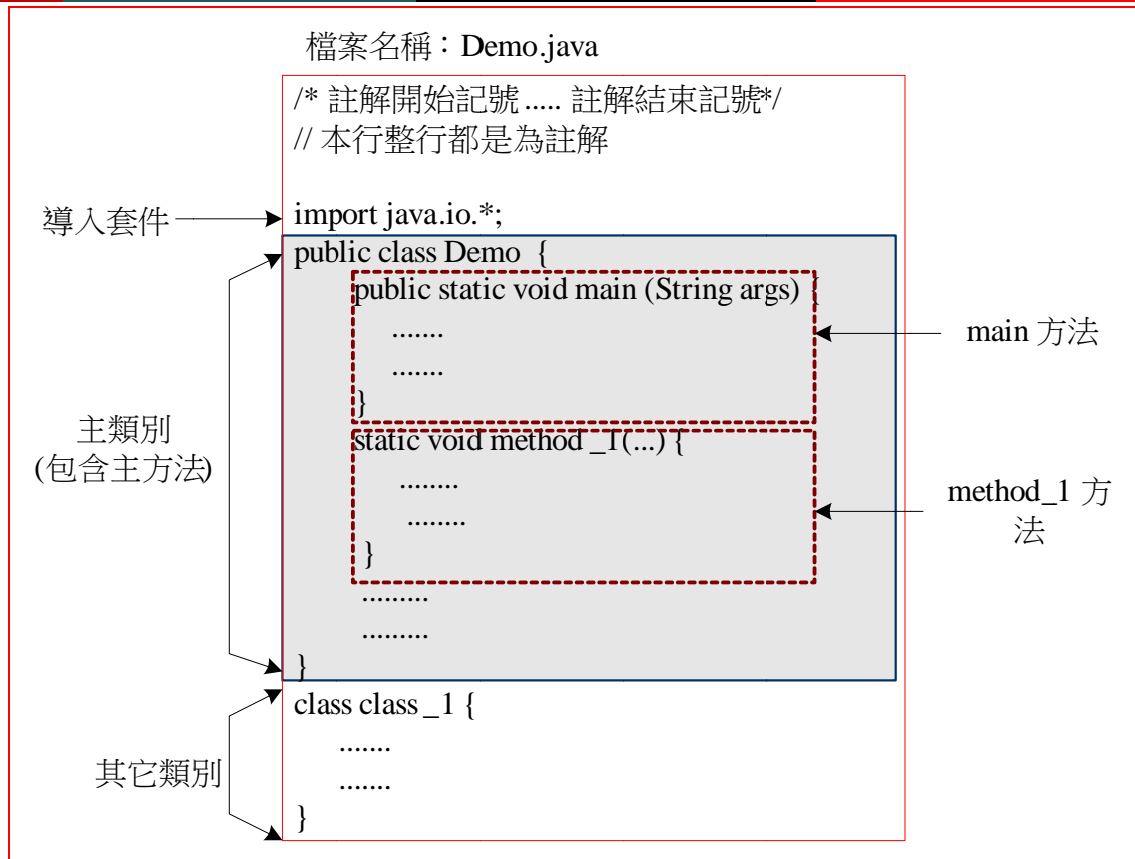


圖 2-2 Java 程式的基本架構

- **主方法宣告『public static void main(String args)』**：表示宣告一個名為 main 的方法，該方法執行後不會傳回任何數值 (void，空)，它是靜態類別 (static)，且屬於公用屬性 (public)，可任意呼叫執行。呼叫 main 方法時是可攜帶 args 引數 (String，字串格式)。
- **程式實體內縮**：基本上 Java 程式內空白一格、多格、一行或多行，意思是相同的。程式員為了方便閱讀程式 (或偵錯程式)，除了將程式實體 (Program body) 利用左右大括號包起來，會將內容內縮若干個字元 (如 6 個空白格)，而這個內縮動作與程式運作無關，純粹是方便「人」閱讀。

2-2 敘述句與敘述區塊

2-2-1 敘述句

程式 (Program) 即是要求電腦執行一連串動作的劇本，程式中每一個『**敘述句**』(**Statement**) 指揮電腦執行一個動作。簡單的說，一套程式 (如劇本) 則是由多個敘述句 (演出項目) 所構成，每一敘述句利用一個**分號 (;)** 表示結束。如圖 2-3 所示。重點說明如下：

- **敘述句內所包含的空格數量不計**，一個空格 (space 鍵)、多個空格、空白一行或空白多行，都表示同樣的意思。
- **每句敘述句利用一個分號 (;) 表示結束**，並不是以行為單位。因此，一行內可以書寫多個敘述句，但之間必須以分號隔離。
- **敘述句可能出現逗點 (,)，表示重複前面『關鍵字』敘述**。如『int a, b;』，表示 a 與 b 都引用『int』關鍵字。

Java 語言敘述	敘述句功能
<pre>public static void main (String args[]) { int a=3, b=4; int sum; sum = a + b; System.out.println("sum = "+sum); }</pre>	<pre>主場演出表 { 宣告及設定 a , b 變數的內容。 宣告變數 sum 。 計算 sum 為 a 與 b 的和。 輸出 sum 變數的內容。 }</pre>

圖 2-3 敘述句功能

2-2-2 敘述區塊

基本上，一個敘述句表達一個工作項目，但如遇到較複雜的工作，也許需要若干個敘述句才能完全表達所要執行的工作細節。如此一來，則需組合若干個敘述句成為一個較大的敘述句，此稱之為『**敘述區塊**』(**Statement block**)。各種程式語言表現『敘述區塊』的方法不盡相同，常見的 C 與 Java 語言都是利用兩個左右大掛號『{}』，包圍起敘述區塊。

其實，任何程式內可能包含著許多大小不一的敘述句。譬如，主方法 (Main) 也是一個敘述句，是由左右大掛號內包含多個敘述句所構成，該區域又稱為『主方法主體』(Main body)。圖 2-4 為一般敘述句的書寫方法。簡單敘述句也許只要一段話就可以表達完成，則不需要敘述區塊，僅利用分號 (;) 表示敘述結束即可。譬如，『如果下雨的話，必須穿雨衣』這句自然語言改用Java程式語言表示就是：

```
if (raining = "yes")
    System.out.println("需穿雨衣");
```

如果較複雜的敘述句，可能需要結合多個敘述句才能完全表達所欲執行的工作細節，則必須利用左右大掛號包起來，成為敘述區塊，但最後不需要分號表示結束。圖 2-4 是 if 敘述句所欲表達的事項，該區域稱之為『**if 主體**』(**if body**)。

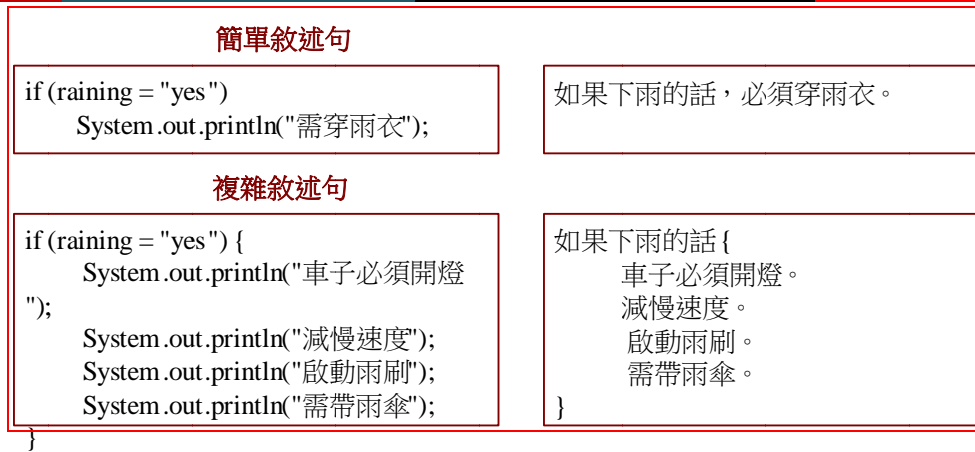


圖 2-4 敘述區塊的功能

2-2-3 多重敘述區塊

由上述的介紹，可瞭解一套程式是由多個敘述句所構成，每一敘述句也可能是結合若干個敘述句。雖然我們可以利用左右大掛號來限制某一敘述區塊的範圍，程式執行時，也大多依照大掛號範圍來運作，但如果程式較複雜或較大時，將很困難觀察出某一敘述句到底是屬於哪一個敘述句所有擁有，也徒增程式除錯 (Debug) 時的困擾。

還好 Java 有一個重要的特性，單一空白鍵、多個空白鍵、單一空白行或多行空白，都是一樣的，所以我們利用這個特性將程式編排漂亮一點，在不影響程式執行的前提下，提高程式的可閱讀性。最簡單的排列方法是，某一敘述的主體都『內縮』某些空白鍵。如圖 2-5 所示，main 敘述句內包含一只 if 敘述句，if 敘述句內包含另一只 for 敘述句，即是敘述句內包含著另一個敘述句的多重敘述區塊。為了方便閱讀，吾人習慣上將某一區塊的所有敘述句內縮固定空白字元，越內部敘述句內縮越多字元，如此看起來比較整齊。

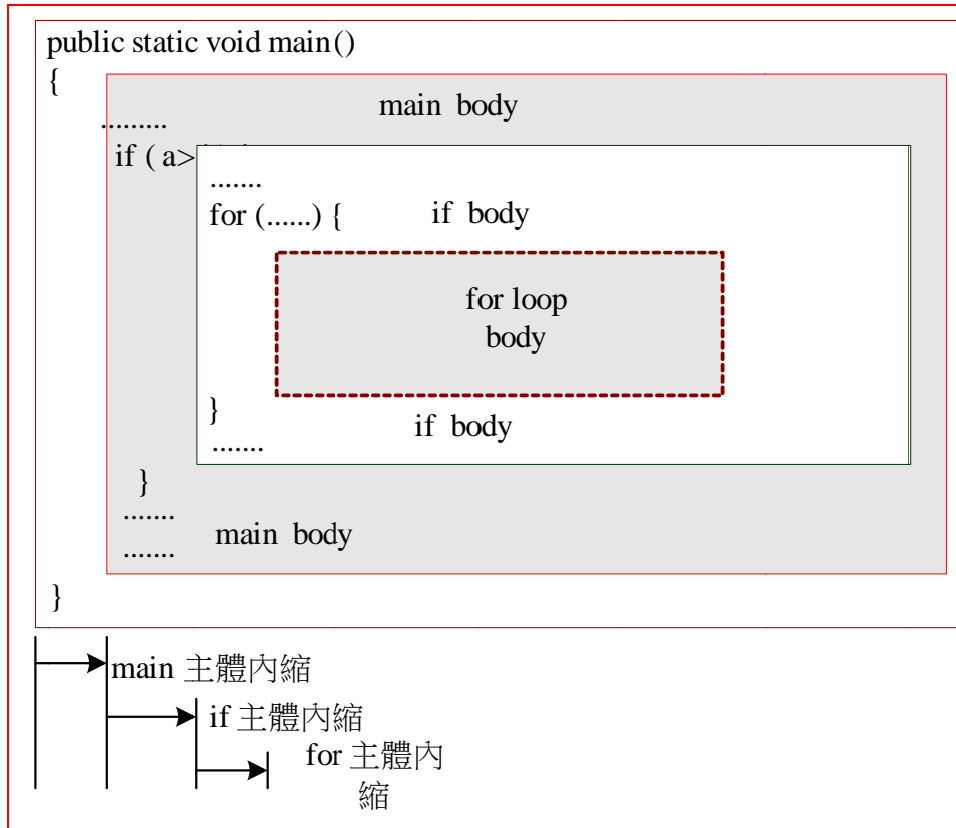


圖 2-5 多重敘述區塊的結構

2-3 變數型態與宣告

2-3-1 何謂變數？

『變數』(**Variables**) 可以想像為某容器的代表名稱，容器內『數值』可隨時改變，因而稱之。一般程式運作時，大多是處理變數之間內容的運算、遷移、複製、集合或排列等等，簡單的說，程式運作是離不開變數處理的。變數之間內容處理的現象，與一般數學演算至為相似，早期『計算機』(Computer) 就是依照這個概念發展而來的。

其實『變數』即是電腦的某一記憶體空間，我們將這空間取一個比較容易有意義的名稱，此即是『變數名稱』(**Variable name**)。另一方面，除了希望記憶體空間能夠充分使用外，也期望不要浪費 CPU 處理時間，因此依照變數可能存放的內容，限制其空間大小。我們則將可能儲存的資料內容，區分為若干個型態，稱之為『資料型態』(**Data type**)；並規劃各種資料型態可能佔用的記憶體空間，也限制了某一資料型態可能表示『數值』(Value) 的範圍。

2-3-2 變數的特性

變數是存放數值的容器，但它的內容僅能被『存放』(或稱指定)與『複製』，並沒有『轉移』或『附加』的功能，也不可能出現『淨空』的現象，這方面與一般真實環境的容器(如杯子)有很大的不同。簡單的說，變數內一定有數值，而此數值僅能被複製到另一個容器，或被覆蓋成另一個數值，並無法將變數的內容移出成淨空狀態。以下我們依照常用的變數處理動作來介紹其特性如何。

(A) 變數的宣告產生

一般程式語言皆會規劃幾種變數的資料型態，不同型態可儲存各種專屬資料。譬如，將某一變數宣告成整數型態(Integer, int)，它僅能存放沒有小數點的資料，雖然有許多不方便，但所占用的記憶體空間小，處理速度較快，某些情況下還是值得的。圖 2-6 為宣告一個整數變數的語法：『int a;』，確實的讀法是『宣告一個名稱為 a 的整數變數』；系統執行該命令時，會由主記憶體內取得一個空間(4 bytes)，並 a 來代表該記憶體空間的位址。

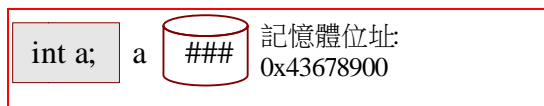


圖 2-6 變數的宣告產生

(B) 指定變數內容

指定變數內容的功能是將某一數值存入變數內，無論原來變數內儲放任何數值，都會覆蓋掉，由新的數值取代。指定內容是變數與一般真實環境的容器之間最大的不同點，一般容器允許加入(或稱附加)數值，譬如，原來內如是 3，吾人再加入 5，使其結果為 8；但變數則不然，它沒有附加的功能，原來內容是 3，如再加入 5，則原來的 3 將被覆蓋，而由 5 取代其內容。圖 2-7 顯示指定變數內容的功能。

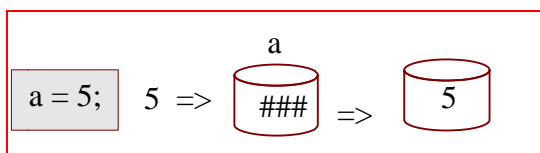


圖 2-7 變數內容的指定

(C) 宣告變數並指定內容

許多情況可以利用一個敘述句，完成宣告變數與指定內容，如『int a=5;』，確實的讀法是：

『宣告一個整數變數，其名稱為 a，並給予內容 5』；此敘述的運作，結合了圖 2-6 (`int a;`) 與 2-7 (`a=5;`) 兩只命令。

(D) 變數內容複製

其實，變數內容是無法移除或附加，僅能被複製或覆蓋。指定變數內容即是覆蓋的意思；但吾人亦可將某一變數的內容複製到另一變數內。圖 2-8 (`int b=a;`) 功能是：『宣告一個名稱為 b 的整數變數，並將變數 a 的內容複製到 b 內』，執行該命令後，變數 a 與 b 的內容都是 5。

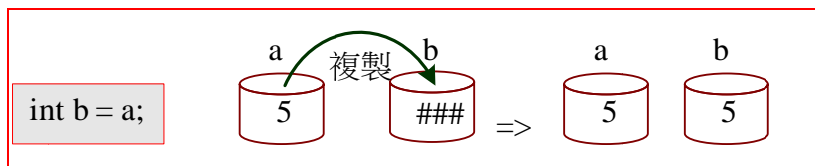


圖 2-8 變數內容的複製

(E) 變數內容的變更

變數內容變更情況如圖 2-9 (`a = a + 10`) 所示，功能是『將 a 變數內容複製一份出來，與 10 做相加運算的結果再存入 a』，如原來 a 的內容是 5 的話，執行該命令後，a 的內容成為 15。

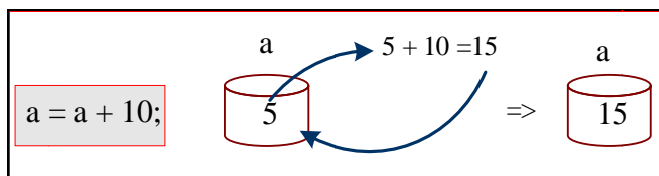


圖 2-9 變數內容的變更

(F) 變數之間的運算

圖 2-10 為兩變數之間的運算程序 (`a = a + b`)，功能是将變數 a 與 b 複製一份出來，兩者相加之後，再將結果回存到 a 內；除了加法之外，其他運算器 (加、減、乘、除、) 皆如此。如果原來 a 內容為 15、b 為 5，運算後 a 的內容被變更為 20，但 b 還保持原來的 5。

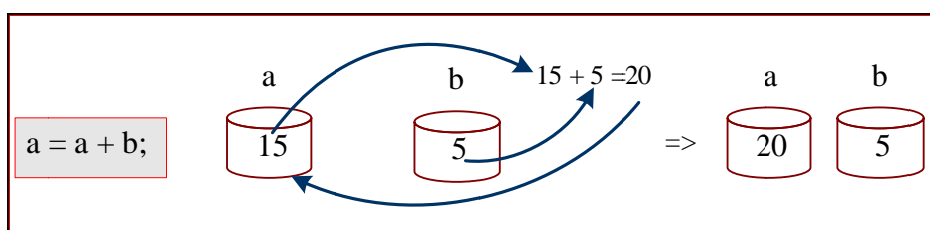


圖 2-10 變數之間的運算

2-3-3 變數的型態

每只變數都會依照他預計儲存的資料格式，給予特定記憶體空間大小；變數的記憶體空間越大，可以儲存的資料內容也就越多。雖然給予變數較大的記憶體空間，可以預備它儲存較複雜的資料，但如果沒有充分使用，就會浪費空間。因此，我們將程式可能儲存的資料分為幾種型態，產生變數時，就依照這些型態來規劃，則稱為『基本資料型態』。簡單的說，它就是產生變數的格式，基本資料型態所占用記憶體空間，與它可以容納資料量的範圍如表 2-1 所示。

表 2-1 Java 基本資料型態

資料型態	長度 (位元數)	範圍
byte	8	-128 ~ 127 ($-2^7 \sim 2^7-1$)
short	16	-32768 ~ 32767 ($-2^{15} \sim 2^{15}-1$)
int	32	-2147483648 ~ 2147483647 ($-2^{31} \sim 2^{31}-1$)
long	64	-9223372036854775808 ~ 9223372036854775807 ($-2^{63} \sim 2^{63}-1$)
float	32	負值： $-3.4028234663852886 \times 10^{38} \sim -1.40129846432481707 \times 10^{-45}$ 正值： $1.40129846432481707 \times 10^{-45} \sim 3.4028234663852886 \times 10^{38}$
double	64	負值： $-1.7976931348623157 \times 10^{308} \sim -4.94065645841246544 \times 10^{-324}$ 正值： $4.94065645841246544 \times 10^{-324} \sim 1.7976931348623157 \times 10^{308}$
char	16	ISO 萬用碼 (Unicode)
boolean		不同平台有不同表示方式
String		字串物件

我們可以將表 2-1 變數型態分類如下：

- **整數 (byte 、 short 、 int 、 long)**：被宣告成此型態的變數，僅能存放沒有小數點的數值，但允許正數或負數。依照所欲儲存數值的大小可以選擇，byte、short、int 或 long。選擇較大者 (如 long) 可存放較多的數值，但佔用記憶體空間較大，CPU 執行速度較慢；相反的，選擇較小執行速度較快，但存放數值可能超載，到底取哪一樣較恰當，程式設計師可要一使用情況來判斷。
- **浮點數 (float 、 double)**：被宣告此型態的變數，可以儲存具有小數點的正或負數值。Float 型

態佔用記憶體空間較小，所能表示範圍較小；Double 具有雙倍記憶體空間，所能表示範圍較大，儲存小數點的位數可以較長，又稱為『雙倍精準度』。

- **字元 (char)**：此類變數可儲存經由 ASCII 碼轉換的單一英文字母（如 'A'、'B'...等），轉換表如附錄 A 所示。其實所儲存內容是正整數，也可經由運算得到另一個字元的 ASCII 碼。
- **字串 (String)**：此類型變數可以存放經由字碼轉換的一連串中英文文字（如：" Good Luck To You " 或 "大家恭喜"），英文係利用 ASCII 碼表示，正體中文大多採用 big-5 碼轉換（各系統也許不同）。
- **布林代數 (boolean)**：此類型變數僅能儲存『真』（True, 1）或『假』（false, 0），大多使用於邏輯運算，或條件判斷式使用。

2-3-4 變數的宣告

我們可利用 Java 某些關鍵字，來宣告產生變數。宣告變數的關鍵字與資料型態名稱相同，如表 2-1 所示（byte、short、int、float、String...等）。基本上，變數名稱是不受限制的，但還是有下列重點：

- 儘量採用有意義的名稱，如 data、selection、cost 等等。
- 不可以有特殊符號，如 +、-、*、\$、#、@、! 等等。
- 不限長短，但有分辨大小寫字母。
- 不可使用 **Java 保留字**（或稱關鍵字）。

變數宣告語法如下：

1	int a;	宣告產生一個整數變數，其名稱為 a。
2	int a, b;	宣告產生兩個整數變數 a 與 b。
3	int count = 10;	宣告整數變數 count，並給予初值 10。
4	short sum;	宣告一個短整數，名稱為 sum。
5	long total=0;	宣告一個長整數變數 total，並設定初值 0。
6	float price;	宣告一個浮點數變數 price。
7	double price;	宣告一個雙倍浮點數 price。

8	char key ='A' ;	宣告一個字元變數 key，並給予初值 'A'。
9	String name = "您好";	宣告字串變數 name，內容為 "您好"。
10	boolean flag = TRUE ;	宣告布林變數 flag，並給以『真』(True · 1)。
11	boolean flag = FALSE ;	宣告布林變數 flag，並給以『假』(False · 0)

2-3-5 保留字彙集

表 2-2 為 Java 保留字彙集，每一個保留字都有其特殊用途 (爾後會陸續介紹到其功能)，不可以作為變數名稱。

表 2-2 Java 保留字彙集

abstract	double	int	strictfp	Do
boolean	else	interface	super	while
break	extends	long	switch	volatile
byte	final	native	synchronized	short
case	finally	new	this	static
catch	float	package	throw	import
char	for	private	throws	instanceof
class	goto	protected	transient	default
const	if	public	try	
continue	implements	return		

2-3-6 範例研討：各種資料型態輸出

(A) 程式功能：Ex2_1.java

列印出各種資料型態的數值，並說明變數的功能為何；輸出格式如下：

我是短整數 short count=10,可當作計數器
 我是整數 int number =20,可當整數運算
 我是長整數 long value=300,可存放較大的整數
 我是浮點 float cost=40.2,可表存放有小數點數值

我是雙倍浮點 double weight=500.34 ,可存較精密的數值

我是字元 char key=A ,可做字元比較或測試

我是布林 boolean flag=true ,可做事件發生的旗標

我是字串 String name=Tien-Shou Nien ,可當文字串列使用

(B) 程式範例：

```
01 public class Ex2_1 {
02     public static void main(String[] args) {
03         short count=10;
04         System.out.println("我是短整數 short count="+count+" ,可當作計數器");
05         int number=20;
06         System.out.println("我是整數 int number =" +number+" ,可當整數運算");
07         long value=300;
08         System.out.println("我是長整數 long value="+value+" ,可存放較大的整數");
09         float cost=40.2F;
10         double weight=500.34;
11         System.out.println("我是浮點 float cost="+cost+" ,可表存放有小數點數值");
12         char key='A';
13         System.out.println("我是雙倍浮點 double weight="+weight+" ,可存較精密的數值");
14         boolean flag=true;
15         System.out.println("我是字元 char key="+key+" ,可做字元比較或測試");
16         System.out.println("我是布林 boolean flag="+flag+" ,可做事件發生的旗標");
17         String name="Tien-Shou Nien";
18         System.out.println("我是字串 String name="+name+" ,可當文字串列使用");
19     }
20 }
21 }
22 }
23 }
24 }
```

(C) 程式重點說明：

- 第 5 行：『short count=10;』。宣告一個名稱為 count 的短整數，並給予初值 10。
- 第 6 行：『System.out.println("我是短.... "+count+" ,可當作計.... ");』。功能是首先印出一串文字 (我是短整數...)，再接著印出 count 變數的內容，最後接著印出下一段文字 (, 可當作計....)。
- 第 15 行：『float cost=40.2F;』。宣告一個浮點變數 cost，並給予初值 40.2F，後面必須增加一個『F』表示浮點數的意思，否則會被認為是雙倍浮點數。

- 第 21 行：『char key='A';』。宣告一個字元變數 key，並給予初值 A，數值僅能一個字元，而且必須用單引號 ('A') 包起來。
- 第 24 行：『boolean flag=true;』。宣告一個布林變數 flag，並將其設定為『真』(true, 1)。如設定為『否』，則 flag=false；但許多情況是利用 1 與 0 來表示真或否。
- 第 27 行：『String name="Tien-Shou Nien";』。宣告一個字串變數 name，並給予初值，內容必須利用雙引號包起來 ("Tien-Shou Nien")，但雙引號內允許空格。

2-4 標準螢幕輸出

如沒有涉及特殊輸入/輸出設備 (如視窗介面)，而僅以文字模式由螢幕上顯示的話，Java 提供兩種簡單的輸出方法，這對初學者很有幫助，以下將說明之。

2-4-1 簡單輸出方法 – println/print

對於沒有特殊格式的簡單螢幕輸出，可使用下列兩種輸出方法：(屬於 java.lang 套件)

- **System.out.println()**：螢幕標準行輸出。每次輸出一行文字到螢幕上 (print line, 列印後跳到下一行)。
- **System.out.print()**：螢幕標準文字輸出。輸出文字到螢幕上，但不換行。

Java 編譯器會自動導入 java.lang 套件，因此不需要在程式內導入該套件；值得注意的是，Java 來自 Sun Unix 系統，會分辨大小寫，因此 system 不可以取代 System，兩者表示不相同的意義。又 print 與 println 兩者之間，只差列印後是否換行，其語法如下：

```
System.out.println("文字輸出 1" + Var_1 + "文字輸出 2" + Var_2 +, ... );
```

左右小括號 ((...)) 為輸出內容 (或稱為『引數』)，括號內兩個雙引號包起來 ("...") 表示文字內容直接輸出，加號 (+) 表示緊接著輸出的意思。println 並不辨識變數型態，而直接將變數內容 (如 v1 與 v2) 輸出。則該敘述表示，首先印出『文字輸出 1』字樣，緊接著印出 Var_1 變數的內容，再輸出『文字輸出 2』樣式，再輸出 Var_2 變數內容，依此類推。

2-4-2 範例研討：印出三角形圖形

(A) 程式功能：Ex2_2.java

請編寫一程式由螢幕輸出下列圖形：

```

*
***
*****
*****
*****
*****
*****
*****
*****
*****

```

(B) 製作技巧研討：

吾人可利用 `println()` 函數依序列印星狀 (*) 符號；共計利用 6 個 `println()` 函數分別印出 1 到 18 個星號。

(C) 程式範例：

```

01 public class Ex2_2 {
02     public static void main(String[] args) {
03         System.out.println("    *");
04         System.out.println("    ***");
05         System.out.println("    *****");
06         System.out.println("    *****");
07         System.out.println("    *****");
08         System.out.println("    *****");
09         System.out.println("    *****");
10     }
}

```

(D) 程式重點說明：

- 此程式僅一個類別 (Ex2_2)，其中也僅包含一個 `main` 方法；檔案名稱必須與類別名稱相同，則檔案名為 `Ex2_2.java`。
- 此程式編譯後會得到一個中介碼 (Bytecode)，為 `Ex2_2.class`。

2-4-3 範例研討：印出產品資料

(A) 程式功能：Ex2_3.java

假設電腦內儲存了兩樣產品的名稱、數量、單價，請編寫一程式將其內容及分別總和印出，如下：

```
酥餅20個50元/個  合計 = 1000元  
脆笛酥100個30元/個  合計 = 3000元
```

(B) 製作技巧分析：

假設電腦裡儲存酥餅與脆笛酥兩樣產品的數量與單價，分別利用字串變數(String)name1 與 name2 儲存產生名稱，再利用整數變數(int)price1 與 price2 分別存放兩產品單價，利用 num1 與 num2 兩整數變數存放產品數量。接著再利用 println 與 print 函數輸出相關變數內容。

(C) 程式範例：

```
01 public class Ex2_3 {  
02     public static void main(String[] args) {  
03         int price1, price2, num1, num2;           //宣告整數變數  
04         String name1, name2;                     // 宣告字串變數  
05         name1 = "酥餅"; price1 = 50; num1 = 20;  // 指定便述內容  
06         name2 = "脆笛酥"; price2 = 30; num2 = 100;  
07         System.out.print(name1+num1+"個"+price1+"元/個  合計 =");  
08         System.out.print(price1 * num1 + "元");  
09         System.out.print("\n");                 // 換行  
10         System.out.print(name2+num2+"個"+price2+"元/個  合計 =");  
11         System.out.println(price2 * num2 + "元"); // 列印含換行  
12     }  
13 }  
14 }  
15 }  
16 }  
17 }
```

(D) 程式重點分析：

- 第 10 行：利用 print 函數印出酥餅的資料。首先印出 name1 變數的內容(酥餅)、緊接著(+)、再印出 num1 內容(20)、再緊接著印出『個』(+'個')、再接著印 price1 內容(20)，最後連結印出『元/個 合計 =』，但沒有換行。
- 第 11 行：計算並印出酥餅的總額(price1 * num1)，最後連結印出『元』，也沒有換行。

- 第 12 行：利用 `print("\n")` 輸出一個換行 (`\n`) 的控制字元。
- 第 15 行：輸出脆笛酥的總額並換行 (`println()`)。

2-4-4 自我挑戰：印出員工識別證

(A) 程式功能：PM2_1.java

請編寫一程式，製作員工識別證，如下圖所示：

```
*****
*   新世代課程研討會   *
*                       *
*   Java 程式設計組     *
*                       *
*   主持人：粘添壽     *
*   翻轉教學教材發展委員會 *
*****
```

(B) 製作技巧提示：

吾人可利用 `println` 函數，依序印出每一行的內容，提示如下：

```
System.out.println("*****");
System.out.println("*   新世代課程研討會   *");
System.out.println("*                       *");
System.out.println("*   Java 程式設計組     *");
System.out.println("*                       *");
System.out.println("*   主持人：粘添壽     *");
System.out.println("*   翻轉教學教材發展委員會 *");
System.out.println("*****");
```

2-5 格式化輸出

2-5-1 格式化輸出方法 - `printf`

許多情況下，我們希望列印出來的資料能排列整齊，如此不但較為美觀，也較容易閱讀，這

就必須仰賴格式化輸出方法：System.out.printf()。其實，printf 與 C 語言的 printf 非常類似，列印時需分辨數值的資料型態，再依此設定列印格式，語法如下：

```
System.out.printf("列印文字與格式", 變數_1, 變數_2, ...);
```

左右括號內引數包含兩種資料，一者由兩個雙引號包起來 ("...") 的是文字直接輸出，以及列印變數的資料型態與指定格式；另一者為一連串所欲輸出的變數。變數的列印格式由 % 作為標頭記號，每一個 % 表示一個變數內容；也就是說，有幾個變數輸出，就需要同樣多個 % 表示每一變數的輸出型態，較常用的輸出型態如表 2-3 所示。

表 2-3 printf 列印格式

列印格式	輸出敘述	列印格式	輸出敘述
%c	字元	%o	8 進位整數
%d	十進位整數	%x	16 進位整數
%5d	5 個位置整數	\n	跳行
%s	字串	\t	跳 Tab 格
%f	浮點數列印	\'	印出單引號
%5.2f	5 位數及 2 小數點	\"	印出雙引號
%.2f	浮點數及 2 小數點	\\	印出反斜線

2-5-2 範例研討：列印格式範例

(A) 程式功能：Ex2_4

圖 2-11 為 printf 的範例，列印順序如下，將變數 car 內容以字串格式 (%s) 印出，接著再印出『定價是』(注意，空白格也要印出)，再將變數 price 內容以整數格式 (%d) 印出，並印出『萬元』，最後將游標跳到下一行 (\n)。

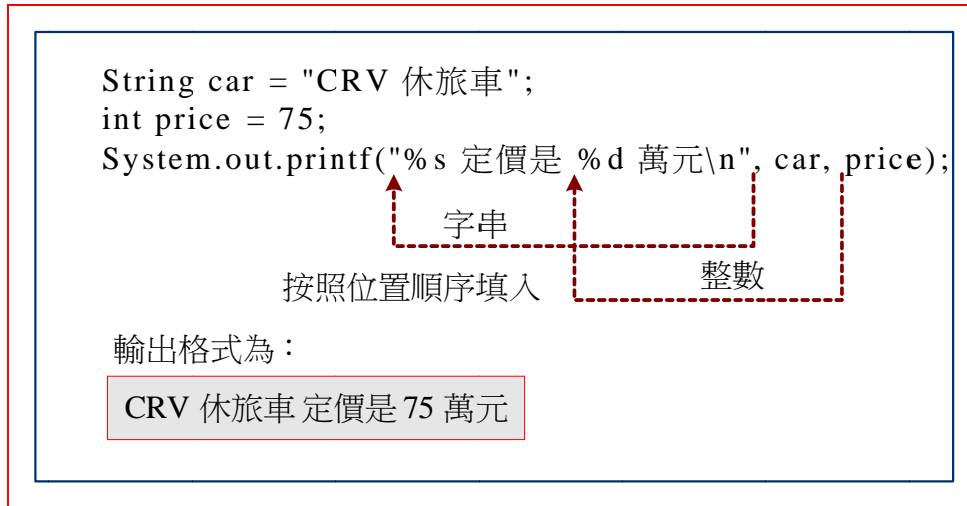


圖 2-11 printf 列印範例

(B) 程式範例：

```
01 public class Ex2_4 {
02     public static void main(String[] args) {
03         String car = "CRV 休旅車";
04         int price = 75;
05         System.out.printf("%s 定價是 %d 萬元\n", car, price);
06     }
07 }
08 }
```

2-5-3 範例研討：不同資料格式輸出**(A) 程式功能：Ex2_5.java**

請製作一只程式，它會要求使用者輸入一個整數，再以 10 進位、8 進位、16 進位與浮點數格式印出該數值的內容，如下所示：

```
請輸入一個整數 =>32
10 進位表示 = 32
16 進位表示 = 20
8 進位表示 = 40
浮點數表示 = 32.00
```

(B) 製作技巧分析

由鍵盤 (System.in) 讀入一個整數，並將其儲存於 value 變數內，再 10 進位 (%d)、8 進

位 (%o)、16 進位 (%x) 與浮點數 (%f) 格式分別印出。

(C) 程式範例：

```

01 // Ex2_5.java
02 import java.util.*;
03 public class Ex2_5 {
04     public static void main (String[] args) {
05         Scanner keyin = new Scanner(System.in);
06         int value;
07         System.out.printf("請輸入一個整數 =>");
08         value = keyin.nextInt();
09         System.out.printf("\t 10 進位表示 = %d\n", value);
10         System.out.printf("\t 16 進位表示 = %x\n", value);
11         System.out.printf("\t 8 進位表示 = %o\n", value);
12         System.out.printf("\t 浮點數表示 = %.2f\n", (float)value);
13     }
14 }
15
16
17

```

2-5-4 自我挑戰：印出漂亮的購物清單

(A) 程式功能：PM2_2.java

如同 Ex2_2.java 範例，但我們希望印出比較漂亮的格式，如下：

品名	數量	單價	小計
酥餅	50	20	1000
脆笛酥	30	100	3000
總計 = 4000			

(B) 製作技巧提示：

如同範例 Ex2_3.java(請複製 Ex2_3.java 來修改)，吾人取用一些變數儲存相關資料 (name1 = "酥餅"、 price1 = 50、 num1 = 20、 name2 = "脆笛酥"、 price2 = 30、 num2 = 100)，再宣告 sum1 (=price1 * num1) 與 sum2 (=price2 * num2) 變數作為兩產品的總額。最後利用格式化輸出 printf 印出相關資料，程式片段如下：

```

01      ....
02      int sum1 = price1 * num1;
03      int sum2 = price2 * num2;
04      System.out.printf("品名\t數量\t單價\t小計\n");
05      System.out.printf("=====\n");
06      System.out.printf("%s\t%d\t%d\t%d\n", name1, price1, num1, sum1);
07      System.out.printf("%s\t%d\t%d\t%d\n", name2, price2, num2, sum2);
08      System.out.printf("=====\n");
09      System.out.printf("總計 = %d\n", sum1+sum2);
10
11      ....
12

```

2-6 標準鍵盤輸入

2-6-1 標準輸入套件 - java.io

雖然 Java 提供有多種輸入/輸出套件，但我們為了方便介紹 Java 語言的操作技巧，我們還是利用最簡單的螢幕輸出與鍵盤輸入。Java 沿用 Unix 系統的定義，將螢幕指定為『標準輸出』與『標準錯誤顯示』；鍵盤為『標準輸入』。前者為『系統輸出』(System.out)，後者為『系統輸入』(System.in)。java.io 套件內組裝有系統輸入/輸出方法，但螢幕輸出 (printf()) 較常使用，因此將 System.out 類別延伸套用於 java.lang 套件內，因此呼叫 System.out.* 方法不用再導入 java.io 套件 (因 java.lang 套件會被自動導入，但必須書寫 System.out.printf())。

套件 java.io 內包含了許多 IO 輸入/輸出『類別方法』(爾後再介紹)，如螢幕、鍵盤、檔案、週邊裝置等等的輸入與輸出類別，其中也製作了多不同存取格式。針對鍵盤輸入還是利用 BufferedReader 類別最為合適，並以 readLine() 方法使用最普遍；但經由 readLine() 方法讀取得到僅是『字串』格式，還需要經由其他類別方法轉換所需的資料型態，較常使用轉換方法如表 2-4 所示 (假設輸入物件為 keyin，讀取方法為 keyin.readLine())。

表 2-4 BufferedReader 類別的輸入方法

輸入方法及格式轉換	功能敘述
keyin.readLine()	讀入一行字串
Integer.parseInt(keyin.readLine())	將讀入字串轉換成整數
Long.parseLong(keyin.readLine())	將讀入字串轉換成長整數

Float.parseFloat(keyin.readLine())	將讀入字串轉換成浮點數
Double.parseDouble(keyin.readLine())	將讀入字串轉換成雙倍浮點數

圖 2-12 為導入 java.io 套件的程式架構，重點說明如下：



圖 2-12 導入 java.io 套件

- 『import java.io.*;』敘述：表示導入 java.io 套件內所有類別。
- 『throws IOException』敘述：引用輸出/輸入類別時，可能會出現異常狀態 (Exception)，必須指定異常訊息『擲往』(throws)到哪一個類別 (IOException) 上。
- 『BufferedReader keyin = new BufferedReader ()』敘述：宣告產生一個 BufferedReader 物件變數，其名稱為 keyin (如同 int a 之功能)；接著，利用 new 方法產生一個 BufferedReader 物件並填入 keyin 變數內 (如同 int a = 5;)。
- 『new InputStreamReader(System.in)』敘述：引用『系統輸入』(System.in，即是鍵盤)物件，產生一個串列輸入讀取器的物件。
- 『keyin.readLine()』敘述：引用輸入物件 keyin 內的 readLine() 方法。

2-6-2 範例研討：捐款收據印出

(A) 程式功能：Ex2_6.java

請幫真自在遊民收容所製作捐款收據，功能是系統允許輸入捐款人姓名與金額，之後印出收據，如下所示：

請輸入大德先生/小姐姓名 =>張有名

```

請輸入捐款金額 =>50000
列印收據如下:

****    真自在遊民收容所    捐款收據    ****

        感謝 張有名 先生/小姐大德贊助
                捐款 50000 元整

****    四海之內皆兄弟 順祝 平安快樂    ****

```

(B) 製作技巧分析：

首先產生鍵盤輸入裝置的物件方法 (keyin)，再宣告兩個變數，一個儲存姓名的字串變數 (String name)；另一個儲存金額的整數變數 (int value)。由鍵盤輸入姓名與金額後，再利用 printf 函數印出收據，重點是收據格式應該如何設計較雅觀。

(C) 程式範例

```

01 // Ex2_6.java
02 import java.io.*;
03 public class Ex2_6 {
04     public static void main(String args[]) throws IOException {
05         BufferedReader keyin = new BufferedReader(new
06             InputStreamReader(System.in));
07         int value;
08         String name;
09         System.out.printf("請輸入大德先生/小姐姓名 =>");
10         name = keyin.readLine();
11         System.out.printf("請輸入捐款金額 =>");
12         value = Integer.parseInt(keyin.readLine());
13         System.out.printf("列印收據如下: \n\n");
14         System.out.printf("****    真自在遊民收容所    捐款收據    ****\n");
15         System.out.printf("\n");
16         System.out.printf("        感謝 %s 先生/小姐大德贊助\n", name);
17         System.out.printf("                捐款 %d 元整\n", value);
18         System.out.printf("****    四海之內皆兄弟 順祝 平安快樂    ****\n");
19     }
20 }
21
22
23
24

```

(D) 程式重點說明：

- 第 6~7 行：宣告產生一個鍵盤輸入物件，名稱為 `keyin`。
- 第 11 行：`name = keyin.readLine()`。執行 `keyin` 物件的 `readLine()` 方法，功能使讀取一串文字，並存放於 `name` 變數內。
- 第 13 行：`value = Integer.parseInt(keyin.readLine())`。將由鍵盤輸入的字串，轉換成整數資料，並存入 `value` 變數內。
- 第 15~19 行：製作收據的樣式。

2-6-3 自我挑戰：簡單加法程式

(A) 程式功能：PM2_3.java

請編寫一程式，程式會要求使用者輸入兩的整數，再計算兩數的和，並輸出計算結果其執行結果。程式執行範例如下：

```
請輸入第一個整數 =>45
請輸入第二個整數 =>65
45 + 65 = 110
```

(B) 製作技巧提示：

首先宣告鍵盤輸入物件 (`keyin`)，再由鍵盤輸入兩個整數 (`value1`、`value2`)，計算並輸出兩者相加 (`sum`) 的結果，程式提示如下：

```
01 .....
02         int value1, value2, sum;
03         System.out.print("請輸入第一個整數 =>");
04         value1 = Integer.parseInt(keyin.readLine());
05         System.out.print("請輸入第二個整數 =>");
06         value2 = Integer.parseInt(keyin.readLine());
07         sum = value1 + value2;
08         System.out.printf("%d + %d = %d\n", value1, value2, sum);
09 .....
.....
```

2-6-4 自我挑戰：加油站計費

(A) 程式功能：PM2_4.java

請製作一只加油站計算客戶油資，程式功能會顯示目前每公升單價多少，輸入客戶所加多少公升，再計算並顯示費用多寡；操作介面如下：(總額小數點不計)

```
目前 95 無鉛汽油每公升 28.50 元
請輸入加油數量 =>30
應收金額 = 855 元
```

(B) 製作技巧提示：

雖然執行本程式時，汽油單價是不變的，但為了此程式能應付爾後油價變更時，程式修改越少越好，因此利用一個常數變數 (final) 儲存單價；日後油價變動僅修改該變數的內容即可。吾人可利用 Scanner 套件讀入所加油的數量，再計算與輸出應收金額多寡；另外，本系統要求不要計算小數點，輸出總金額時，不顯示小數點 (%.0f) 即可 (小數點的資料還是在，只是不顯示而已)，程式提示如下：

```
01 .....
02     final double price=28.5;           // 變數內容不可更改
03     double number, total;
04     System.out.printf("目前 95 無鉛汽油每公升 %.2f 元\n", price);
05
06     System.out.printf("請輸入加油數量 =>");
07     number = Double.parseDouble(keyin.readLine());
08     total = number * price;
09     System.out.printf("應收金額 = %.0f 元\n", total); // 捨棄小數點
.....
```

2-7 掃描鍵盤輸入

利用 BufferedReader 所產生物件 (如 keyin) 的讀取輸入方法是 readLine()，readLine() 功能是一次讀取一行資料；也就是說，無論使用者由鍵盤鍵入多少字元，敲入『Enter』鍵之後，一次將所有字元讀入。如果，吾人希望由鍵盤上一次輸入多個資料，再敲入『Enter』鍵，系統再分別讀入，可以另一個掃描輸入 Scanner) 套件才可以。

2-7-1 掃描輸入套件 – Scanner

Scanner 是屬於 java.util 套件的類別，不需指定異常狀態處理 (如 throws IOException)；導入套件與宣告語法如下：

- **套件名稱**：java.util.Scanner;
- **物件宣告產生**：Scanner in = new Scanner(System.in);

圖 2-13 為引用 Scanner 套件的範例，吾人在連續輸入三筆資料，之間利用空白格 (Space) 隔離，可利用三個讀取方法 (next()、nextInt() 與 nextFloat()) j 分別讀取這三筆資料。表 2-5 為 Scanner 常用的類別方法。

```
import java.util.Scanner;
public class Ex2_4 {
    public static void main(String args[]) {
        Scanner keyin = new Scanner(System.in);
        System.out.printf("請輸入客戶資料 =>");
        String name = keyin.next();
        int age = keyin.nextInt();
        float cost = keyin.nextFloat();
        .....
    }
}
```

請輸入客戶資料 => 張大名 50 480.5

in.next() in.nextInt() in.nextFloat()

圖 2-13 Scanner 套件的運作程序

表 2-5 Scanner 類別的輸入方法 (物件名稱為 keyin)

輸入方法	功能敘述
keyin.nextLine()	讀入一行字串
keyin.next()	讀入一個字串
keyin.nextInt()	讀入一個整數
keyin.nextFloat()	讀入一個浮點數
keyin.nextDouble()	讀入一個雙倍浮點數
keyin.useDelimiter(“;”)	設定分隔符號，如分號 (;)

值得注意的是，上述各種讀取方法大多已指定資料型態，如果讀取方法與輸入資料的型態不符合的話，將會出現執行錯誤 (Run-time error)。另外，useDelimiter() 方法是設定輸入行的分隔記號，即是當每一行包含多項資料時，資料項之間的分隔記號為何。較常見的分隔記號有：空格 (space，一個或多個皆相同)、分號 (;) 或冒號 (:)；如果沒有特殊指定大多

採用空白鍵 (第八章有範例說明)。

2-7-2 範例研討：登錄客戶資料

(A) 程式功能：Ex2_7.java

請製作一套登錄客戶資料的雛形軟體，系統要求輸入客戶資料後，再印出輸入結果，觀察是否正確，期望操作格式如下所示：

```
請輸入客戶資料(姓名 年齡 消費金額) =>張大銘 22 45000  
客戶資料如下:  
  
客戶姓名: 張大銘  
客戶年齡: 22  
消費金額: 45000.00
```

(B) 製作技巧分析：

本系統要求每一筆資料必須連續輸入，一直到敲入『Enter』鍵後結束，必須採用 Scanner 類別才可達成。輸入資料的順序是姓名 (字串, String)、年齡 (整數, int)、最後是消費金額 (浮點數, float)，讀取時，資料格式也不可弄混 (next()、nextInt()、nextFloat())。

(C) 程式範例：

```
01 // Ex2_7.java
02
03 import java.util.Scanner;
04 public class Ex2_7 {
05     public static void main(String args[]) {
06         Scanner keyin = new Scanner(System.in);
07         System.out.print("請輸入客戶資料(姓名 年齡 消費金額) =>");
08         String name = keyin.next();
09         int age = keyin.nextInt();
10         float costs = keyin.nextFloat();
11
12
13         System.out.printf("客戶資料如下: \n\n");
14         System.out.printf("客戶姓名: %s\n", name);
15         System.out.printf("客戶年齡: %d\n", age);
16         System.out.printf("消費金額: %.2f\n", costs);
17
18     }
19 }
```

2-7-3 自我挑戰：製作單位名片

(A) 程式功能：PM2_5.java

請製作某一單位的名片製作軟體，系統要求使用者輸入名片擁有人的工作單位與、姓名與電話，則印出名片格式如下：

```

請輸入職位名稱 =>程式設計小組
請輸入姓名 => 張大得
請輸入電話 =>(07) 74312134 ext 231
名片列印如下：
-----
台灣遊戲軟體發展協會
程式設計小組

          張大得

電話: (07) 74312134 ext 231
地址: 高雄縣烏松鄉澄清路 840 號
-----

```

(B) 製作技巧提示：

首先宣告鍵盤輸入物件 (keyin) 和三個字串變數 (String); 再由鍵盤輸入名片主人的姓名 (name, 張大名)、職位 (position, 程式設計工作小組) 與電話 (tel)。接著再設計並印出名片格式，程式提示如下：

```

01 .....
02         System.out.printf("名片列印如下：\n");
03         System.out.printf("-----\n");
04         System.out.printf("台灣遊戲軟體發展協會          \n");
05         System.out.printf("%s\n\n", position);
06         System.out.printf("\t%s\n\n", name);
07         System.out.printf("電話: %s\n", tel);
08
09         System.out.printf("地址: 高雄縣烏松鄉澄清路 840 號\n");
10         System.out.printf("-----\n");
.....

```

2-8 專題研討

2-8-1 範例研討：計算圓的參數

(A) 系統功能：Ex2_8.java

請製作一個可以了解有關圓參數的系統，只要輸入圓半徑 (radius)，則系統會輸出圓面積、圓週長與圓體積等相關參數，其執行結果如下：

```
圓參數計算工具
請輸入圓的半徑 =>12.5
圓半徑 = 12.50 的相關參數如下：
    圓周長 = 78.50
    圓面積 = 490.63
    圓體積 = 4599.61
```

(B) 程式製作技巧：

首先我們必須查出有關圓參數的計算方法：圓面積 ($area = \text{Pi} * \text{radius}^2$)、圓週長 ($\text{circum} = 2 * \text{Pi} * \text{radius}$)與圓體積($\text{volume} = 3/4 * \text{radius}^3$ ，其中請將 Pi 設定為固定變數 3.14 (`final double Pi = 3.14`)。

(C) 程式範例


```

01 //Ex2_8.java
02 import java.util.*;
03 public class Ex2_8 {
04     public static void main(String[] args) {
05         Scanner keyin = new Scanner(System.in);
06         final double Pi = 3.14;
07         double radius, area, circum, volume;
08         System.out.printf("圓參數計算工具\n");
09         System.out.printf("請輸入圓的半徑 =>");
10
11         radius = keyin.nextDouble();
12         area = Pi * radius * radius;
13         circum = 2 * Pi * radius;
14         volume = 3.0/4.0 * Pi * radius * radius * radius;
15         System.out.printf("圓半徑 = %.2f 的相關參數如下 :\n", radius);
16         System.out.printf("\t圓周長 = %.2f\n", circum);
17         System.out.printf("\t圓面積 = %.2f\n", area);
18         System.out.printf("\t圓體積 = %.2f\n", volume);
19     }
20 }

```

2-8-2 自我挑戰：計算數學方程式

(A) 系統功能：PM2_6.java

吾人需要一套數學方程式的計算工具，假設 $y_1 = x^2 + 1$ 、 $y_2 = 4x^2 + 3x + 2$ ，系統只要輸入 x 的數值則系統則輸出 y_1 與 y_2 相對的數值。期望結果如下：

```

方程式計算工具：
請輸入變數 x 的數值 =>23.5
變數 x = 23.50 計算如下：
    4*x*x + 3*x + 1 = 2281.50
    x*x + 1 = 553.25

```

(B) 製作提示：

```

01 .....
02         System.out.printf("請輸入變數 x 的數值 =>");
03         x = keyin.nextDouble();
04         y1 = 4*x*x + 3*x + 2;
05         y2 = x*x + 1;
06         ....

```

