



第七章

內建類別函數的應用

本章學習目標



1. 讓讀者瞭解內建函數的種類及各種應用。
2. 說明使用者自定函數與內建函數的差異。



本章內容

7-1 字串類別(String)

7-2 轉換類別(Convert)

7-3 數值類別(Math)

7-4 亂數類別(Random)

7-5 日期類別(DateTime)

7-6 習題

前言



函數是另一種特殊形式的副程式，這種副程式有的存在於JAVA的語言系統裡，稱為「內建類別函數庫」，我們曾經使用過的Math.random()等等，就是JAVA的內建類別函數庫，另一種函數就像副程式一樣可以由我們自己定義的，稱為「自定函數」。一般而言，善加利用函數，可以節省軟體開發的時間與成本，並且可以讓程式容易修改與維護。

在JAVA中，提供一系列的「內建類別函數庫」來供設計者使用，大致上可以分為字串類別、轉換類別、數值類別及日期類別。



7-1 字串類別(String)

在我們撰寫程式時，時常需要字串函數來取出字串中的某一部份為子字串。例如我們要判斷使用者輸入的身份證字號，來顯示他們的性別，因此，我們就必須要利用chartAt(1)方法來取得身份證字號的第二個字元，如果是「1」則為男生，如果是「2」則為女生。其常見的字串類別(String)之方法如表7-1所示：

表7-1字串類別(String)表

方法	功能說明
7-1.1 charAt()方法	取得字串指定的字元
7-1.2 length 方法	取得字串的總字元數
7-1.3 compareTo() 方法	比較兩個字串的 ASCII 碼的大小
7-1.4 contains() 方法	判斷 Str1 字串是否有包括指定的 Str2 字串
7-1.5 endsWith() 方法	判斷 Str1 字串的結尾是否為 Str2 字串
7-1.6 equals() 方法	比較 Str1 字串與 Str2 字串是否相同
7-1.7 indexOf() 方法	取得第一個符合指定字串的索引位置
7-1.8 Insert() 方法	在指定的字串位置中插入指定的字串
7-1.9 LastIndexOf() 方法	取得反向尋找字串的最先出現位置
7-1.10 Replace() 方法	將字串中某些特定字串取代成另一字串
7-1.11 Split() 方法	將字串分割成字串陣列
7-1.12 StartsWith() 方法	字串開頭比對
7-1.13 Substring() 方法	取得字串中的子字串
7-1.14 ToLower() 方法	字母轉成小寫
7-1.15 ToUpper() 方法	字母轉成大寫
7-1.16 Trim() 方法	刪除字串中的空白字元



7-1.1 charAt 方法

【語法】char c=str.charAt(索引)

【說明】取出字串中某一字元。

【注意】指定位置之索引是從0開始。

【實例】取出身份證字號的第2個字元

程式檔案名稱	ch7_1_1.java
01	public class ch7_1_1
02	{
03	public static void main(String[] args)
04	{
05	String str="A123456789";
06	char c=str.charAt(1);
07	System.out.println("身份證：" + str + " 的第2個字元是：" + c);
08	}
09	}

【執行結果】

身份證：A123456789 的第2個字元是：1



7-1.2 length方法

【語法】`len=str.length();`

【說明】計算字串的總字元數

【實例1】計算身份證字號的內容字數

程式檔案名稱		ch7_1_2.java
01	<code>public class ch7_1_2</code>	

```
01 public class ch7_1_2
02 {
03     public static void main(String[] args)
04     {
05         String str="A123456789";
06         int len=str.length();
07         System.out.println("你的身份證是：" + len + "位數");
08     }
09 }
```

【執行結果】

你的身份證是：10位數

【實例2】請設計一個檢查身份證字號的字數之程式，如果非10個字，則顯示ID錯誤。

ALETHEIA
UNIVERSITY

程式檔案名稱	ch7_1_2A.java
01	<pre>public class ch7_1_2A { public static void main(String[] args) { String str="A12345"; int len=str.length(); if(len!=10) System.out.println("你的ID有誤！"); } }</pre>

【執行結果】

你的ID有誤！



7-1.3 compareTo() 方法

【語法】`str1.compareTo(str2);`

【說明】比較str1與str2兩個字串的ASCII碼的大小，其比較結果有三種情況：

- ① str1 > str2時，則傳回1
- ② str1 = str2時，則傳回0
- ③ str1 < str2時，則傳回-1

【實例】比較Str1與Str2兩個字串是否相同。

程式檔案名稱	ch7_1_3.java
01	<pre>public class ch7_1_3 { public static void main(String[] args) { String str1="A12345"; String str2="a12345"; int Result; Result =str1.compareTo(str2); if (Result == 0) System.out.println(str1 + "與" + str2 + "相同！"); else System.out.println(str1 + "與" + str2 + "不相同！"); } }</pre>

【執行結果】

A12345與a12345不相同！

7-1.4 contains() 方法



【語法】`str1.contains(str2);`

【說明】判斷A字串是否有包括指定的B字串，其比較結果有兩種情況：

- ① str1包括 str2時，則傳回true
- ② str1不包括 str2時，則傳回false

【實例】判斷str1字串是否有包括指定的str2字串

程式檔案名稱 ch7_1_4.java

```
01 public class ch7_1_4
02 {
03     public static void main(String[] args)
04     {
05         String str1 = "12345";
06         String str2 = "123";
07         boolean Result;
08         Result = str1.contains(str2);
09         if (Result ==true )
10             System.out.println(str1 + "包括" +str2);
11         else
12             System.out.println(str1 + "不包括" +str2);
13     }
14 }
```

【執行結果】

12345 包括 123

7-1.5 endsWith() 方法



【語法】`str1.endsWith(str2);`

【說明】判斷`str1`字串的結尾是否為`str2`字串，其比較結果有兩種情況：

- ① `str1`結尾是 `str2`時，則傳回`true`
- ② `str1`結尾不是 `str2`時，則傳回`false`

【實例】判斷str1字串的結尾是否為str2字串

程式檔案名稱	ch7_1_5.java
--------	--------------

```
01 public class ch7_1_5
02 {
03     public static void main(String[] args)
04     {
05         String str1 = "12345";
06         String str2 = "345";
07         boolean Result;
08         Result = str1.endsWith(str2);
09         if (Result ==true )
10             System.out.println(str1 + "結尾是" +str2);
11         else
12             System.out.println(str1 + "結尾不是" +str2);
13     }
14 }
```

【執行結果】

12345結尾是345

7-1.6 equals() 方法



【語法】`str1.equals(str2);`

【說明】比較str1字串與str2字串是否相同，其比較結果有兩種情況：

- ① str1與str2兩字串相同時，則傳回true
- ② str1與str2兩字串不相同時，則傳回false

【實例】比較str1字串與str2字串是否相同

程式檔案名稱	ch7_1_6.java
01	<pre>public class ch7_1_6 { public static void main(String[] args) { String str1 = "12345"; String str2 = "12345"; boolean Result; Result = str1.equals(str2); if (Result ==true) System.out.println(str1 + "與" + str2 + "兩字串相同！"); else System.out.println(str1 + "與" + str2 + "兩字串不相同！"); } }</pre>

【執行結果】

12345與12345兩字串相同！

7-1.7 indexOf() 方法



【語法】`str1.indexOf(str2);`

【說明】取得第一個符合指定字串的索引位置，亦即從`str1`字串的左邊開始往右邊第1個字元找起，找出`str2`首次出現的位置，若找不到，則傳回-1。

【實例】請設計一個可以在摘要內容中尋找指定的關鍵字。

程式檔案名稱	ch7_1_7.java
--------	--------------

```
01 public class ch7_1_7
02 {
03     public static void main(String[] args)
04     {
05         String str1 = "利用物件的方法可以節省軟體開發的時間與成本。";
06         String str2 = "時間與成本";
07         int Result;
08         Result = str1.indexOf(str2);
09         if (Result != -1)
10             System.out.println("摘要中有：" + str2 + " 關鍵字");
11         else
12             System.out.println("摘要中沒有：" + str2 + " 關鍵字");
13     }
14 }
```

【執行結果】

摘要中有：時間與成本 關鍵字

7-1.8 lastIndexOf() 方法



【語法】`str1.lastIndexOf(str2);`

【說明】反向尋找字串的最先出現位置，亦即從`str1`字串的右邊開始往左邊第1個字元找起，找出`str2`首次出現的位置，若找不到，則傳回-1。

【實例】請設計一個可以在摘要內容中尋找指定的關鍵字。

程式檔案名稱	ch7_1_8.java
	<pre>01 public class ch7_1_8 02 { 03 public static void main(String[] args) 04 { 05 String str1 = "數位學習是目前最新的學習型態。"; 06 String str2 = "學習"; 07 int Result; 08 Result = str1.lastIndexOf(str2); 09 if (Result != -1) 10 System.out.println("摘要中有：" + str2 + " 關鍵字"); 11 else 12 System.out.println("摘要中沒有：" + str2 + " 關鍵字"); 13 } 14 }</pre>

【執行結果】

```
摘要中有：學習 關鍵字
```

7-1.9 replace() 方法



【語法】`str1.replace(str2,str3);`

【說明】將字串中某些特定字串取代成另一字串，亦即將`str1`字串中的
 `str2`字串取代成為`str3`字串，若`str3`字串為空字串，則`str1`字
 串中的`str2`字串將被刪除。

【實例】請計計一個可以取代內容區中的某些特定字串。

程式檔案名稱	ch7_1_9.java
01	<pre>public class ch7_1_9</pre>

02 {

03 public static void main(String[] args)

04 {

05 String str1 = "數位學習是目前最新的學習型態。";

06 String str2 = "數位";

07 String str3 = "網路";

08 String Result;

09 Result = str1.replace(str2,str3);

10 System.out.println("結果：" + Result);

11 }

12 }

【執行結果】

結果：網路學習是目前最新的學習型態。

7-1.10 startsWith() 方法



【語法】`str1.startsWith(str2);`

【說明】字串開頭比對，其比較結果有兩種情況：

- ① str1的開頭是 str2時，則傳回true
- ② str1的開頭不是 str2時，則傳回false

【實例】

程式檔案名稱	ch7_1_10.java
01	<pre>public class ch7_1_10</pre>
02	{
03	public static void main(String[] args)
04	{
05	String str1 = "A12345";
06	String str2 = "A123";
07	boolean Result;
08	Result = str1.startsWith(str2);
09	if (Result == true)
10	System.out.println(str1 + "的開頭是" + str2);
11	else
12	System.out.println(str1 + "的開頭不是" + str2);
13	}
14	}

【執行結果】

A12345的開頭是A123

7-1.11 substring() 方法



【語法】`str1.substring(m,n);`

【說明】取得字串中的子字串，其中m代表Str1的起始索引值，n-1代表結束索引值。

【注意】`substring()`方法已經取代了VB2008中的Left, Mid及Right三個字串函數。

【實例1】

程式檔案名稱	ch7_1_11.java
01	<pre>public class ch7_1_11 { public static void main(String[] args) { String str = "I love Java!"; int m=2; int n=11; String Result; Result = str.substring(m,n); System.out.println("結果：" + Result); } }</pre>

【執行結果】

結果：love Java

【實例2】請設計數字三角形。

程式檔案名稱	ch7_1_11A.java
--------	----------------

```
01 public class ch7_1_11A
02 {
03     public static void main(String[] args)
04     {
05         String str = "12345";
06         int i;
07         for(i = 0;i<=str.length()-1;i++)
08             System.out.println(str.substring(0, i+1));
09     }
10 }
```

【執行結果】

1 12 123 1234 12345

【實例3】請設計一個可以判斷身份證字號是屬於男生或女生的程式。

程式檔案名稱	ch7_1_11B.java
--------	----------------

```
01 public class ch7_1_11B
02 {
03     public static void main(String[] args)
04     {
05         String str = "A123456789";
06         String Result;
07         Result= str.substring(1,2); //起始索引從開始
08         if (Result.equals("1"))
09             System.out.println("男生");
10         else
11             System.out.println("女生");
12     }
13 }
```

【執行結果】

男生



【實例4】迴文判斷(利用字串連結)

【說明】

- ① 【輸入範例】12321
- ② 【輸出報表】12321 是迴文
- ③ 【執行結果】

【實例】

程式檔案名稱	ch7_1_11C.java
	<pre>01 public class ch7_1_11C 02 { 03 public static void main(String[] args) 04 { 05 String D1, D2="", D3; 06 int i; 07 D1 = "12321"; 08 D3 = D1; 09 for(i =D1.length()-1; i>=0; i--) 10 D2 = D2 + D1.substring(i,i+1); //代表一次取一個字元 11 if (D2.equals(D3)) 12 System.out.println(D3 + " 是迴文."); 13 else 14 System.out.println(D3 + " 不是迴文."); 15 } 16 }</pre>

【執行結果】

12321 是迴文.

【實例5】請設計一個可以計算字串中某些字出現的次數。

程式檔案名稱	ch7_1_11D.java
01	<code>package ch7_1_11D;</code>
02	<code>public class ch7_1_11D</code>
03	<code>{</code>
04	<code> public static void main(String[] args)</code>
05	<code> {</code>
06	<code> String str= "Computer And Information";</code>
07	<code> int i, t=0;</code>
08	<code> String Result;</code>
09	<code> for(i = 0;i<=str.length()-1;i++)</code>
10	<code> {</code>
11	<code> Result=str.substring(i,i+1);</code>
12	<code> if(Result.equals("m"))</code>
13	<code> t+=1;</code>
14	<code> }</code>
15	<code> System.out.println("在Computer And Information 字串中共有：" + t + "</code>
16	<code> 個m字元");</code>
17	<code> }</code>
18	<code>}</code>

【執行結果】

在Computer And Information 字串中共有：2個m字元

7-1.12 toLowerCase() 方法



【語法】str1.toLowerCase();

【說明】字母轉成小寫

【實例】請將大寫「I LOVE JAVA」轉換小寫「i love java」

程式檔案名稱	ch7_1_12.java
01	public class ch7_1_12
02	{
03	public static void main(String[] args)
04	{
05	String str= "I LOVE JAVA";
06	String Result;
07	Result=str.toLowerCase();
08	System.out.println(str + "轉成小寫為" + Result);
09	}
10	}

【執行結果】

I LOVE JAVA轉成小寫為i love java

真理大學

資訊工程學系

Page : 33

7-1.13 ToUpper() 方法



【語法】Str1. ToLower();

【說明】將小寫轉換成大寫。

【實例】請設計一個將小寫身份證字號轉換成大寫。

程式檔案名稱	ch7_1_13.java
01	<code>public class ch7_1_13</code>
02	<code>{</code>
03	<code>public static void main(String[] args)</code>
04	<code>{</code>
05	<code>String str= "a123456789";</code>
06	<code>String Result;</code>
07	<code>Result=str.toUpperCase();</code>
08	<code>System.out.println(str + "轉成大寫為" + Result);</code>
09	<code>}</code>
10	<code>}</code>

【執行結果】

a123456789轉成大寫為A123456789

真理大學

資訊工程學系

Page : 34

7-2 轉換類別



在第二章所介紹System.in(標準輸入)，其功能就是不管輸入的資料是文字或是數值，Java都自動轉成字串。所以，在程式碼中欲對所輸入的資料做數值計算時，必須先透過轉換的方法，來將文字資料轉成數值方能做各種計算。

7-2.1 字串轉換成數值資料的方法



將字串類別型態轉換成為其他的資料型態。其常用的方法如下：

字串資料可以轉換成數值資料型態的方法如下：

轉換方法	說明
Short.parseShort(str)	將字串資料轉換成「短整數型態(Short)」資料。
Integer.parseInt(str)	將字串資料轉換成「整數型態(Integer)」資料。
Long.parseLong(str)	將字串資料轉換成「長整數型態(Long)」資料。
Float.parseFloat(str)	將字串資料轉換成「單精度型態(Float)」資料。
Double.parseDouble(str)	將字串資料轉換成「倍精度型態(Double)」資料。

★ 字串資料轉換成「整數型態(Integer)」資料

【語法】`Integer.parseInt(str)`

【例子】`int num= Integer.parseInt(str);`

【實例】計算學生的兩科科目之平均成績



程式檔案名稱	ch7_2_1.java
	<pre>01 import java.util.Scanner; //載入Scanner類別套件 02 public class ch7_2_1 03 { 04 public static void main(String[] args) 05 { //輸入 06 Scanner inData=new Scanner(System.in); 07 System.out.println("請輸入姓名、國文成績及英文成績："); 08 System.out.print("請輸入學生姓名："); 09 String stuName=inData.next(); //學生姓名 10 System.out.print("請輸入國文成績："); 11 String chiScore=inData.next(); //國文成績 12 System.out.print("請輸入英文成績："); 13 String engScore=inData.next(); //英文成績 14 //處理 15 int chi=Integer.parseInt(chiScore); 16 int eng=Integer.parseInt(engScore); 17 int average=(chi+eng)/2; 18 //輸出 19 System.out.println();//換行 20 System.out.println("姓名：" + stuName); 21 System.out.println("國文：" + chi); 22 System.out.println("英文：" + eng); 23 System.out.println("平均：" + average); 24 } 25 }</pre>



【執行結果】

請輸入姓名、國文成績及英文成績：

請輸入學生姓名：李安

請輸入國文成績：**80**

請輸入英文成績：**90**

姓名：李安

國文：**80**

英文：**90**

平均：**85**

7-2.2 數值轉換成字串資料的方法

在Java語言中，除了可以將使用者所輸入的字串資料轉換成數值資料之外，它也提供讓使用者將數值資料換成字串資料的方法。

★ 數值資料轉換成「字串型態(String)」資料

【語法】`str=String.valueOf(數值資料)`

【實例】請將成績60的數值資料轉換「字串資料」。

程式檔案名稱	ch7_2_2.java
01	<code>public class ch7_2_2</code>
02	<code>{</code>
03	<code>public static void main(String[] args)</code>
04	<code>{</code>
05	<code> int score=60;</code>
06	<code> String str=String.valueOf(score); //str="60"</code>
07	<code> System.out.println("數字" + str + " 已被轉換成字串資料!");</code>
08	<code>}</code>
09	<code>}</code>

【執行結果】



7-3 數值類別(Math)

System.Math命名空間中的Math類別，提供了很多和數學有關的函數。在我們利用程式來處理數值性的資料時，如果能夠善加利用JAVA內建的數值函數時，則可以節省許多時間。例如欲求某一個數值的絕對值時，如果利用if結構來撰寫也可以達到此目標，但必須要撰寫較多的程式，但是，如果利用Math.abs (x);函數就可以輕易取得絕對值。其數值函數列表如下：

函數名稱	功能說明
Math.abs (x);	取 x 的絕對值
Math.sin (x); Math.cos(x) Math.tan (x)	正三角函數
Math.asin(x) Math.acos(x) Math.atan(x)	反三角函數
Math.exp (x);	取 e 的 x 次方(約為 2.71828)
Math.floor (x);	取 $\leq X$ 的最大整數值
Math.log (x);	取 x 以 e 為底數的對數值
Math.log10 (x);	取 x 以 10 為底數的對數值
Math.max(x,y);	取 x 與 y 較大者
Math.min(x,y);	取 x 與 y 較小者
Math.pow(x,y);	取 x 的 y 次方
Math.round(x);	可依照指定所求出 X 的小數點位數並四捨五入
Math.sqrt(x);	取 x 的平方根
Math.ceil(x);	取 $\geq x$ 的最小整數值
Math.PI;	取得圓周率π 值
Math.E;	取得指數 e 值(約為 2.71828)



7-3.1 Math.abs(x) 取絕對值

【語法】Math.abs(x)

【說明】取x的絕對值

【舉例】

程式檔案名稱		ch7_3_1.java
01	<code>public class ch7_3_1</code>	
02	{	
03	<code>public static void main(String[] args)</code>	
04	{	
05	<code>System.out.println("印出：" + Math.abs(100));</code>	//印出：100
06	<code>System.out.println("印出：" + Math.abs(-100));</code>	//印出：100
07	<code>System.out.println("印出：" + Math.abs(-3.14));</code>	//印出：3.14
08	<code>System.out.println("印出：" + Math.abs(0.11));</code>	//印出：0.11
09	<code>System.out.println("印出：" + Math.abs(0));</code>	//印出：0
10	}	
11	}	

【執行結果】

印出：100

印出：100

印出：3.14

印出：0.11

印出：0

【實例】請設計一個以星號堆疊成K字型的圖

【解答】

程式檔案名稱	ch7_3_1A.java
01	<pre>public class ch7_3_1A { public static void main(String[] args) { int i, j; for(i = -4;i<=4;i++) { for(j = 1;j<=Math.abs(i) + 1;j++) System.out.print("*"); //連續列印 System.out.println(); //換行 } } }</pre>

【執行結果】

```
*****
****
 ***
 **
 *
 **
 ***
 ****
*****
```

7-3.2 Math.sin()、Math.cos()、 Math.tan() 正三角函數



【語法1】Sin_value= Math.sin(X) //傳回X數值的正弦值

【語法2】Cos_value= Math.cos(X) //傳回X數值的餘弦值

【語法3】Tan_value= Math.tan(X) //傳回X數值的正切值

【說明】將角度轉成徑度的計算公式如下所示：

徑度 = 角度 * PI / 180 ,

其中PI是指圓週率3.14159265358979

【舉例】

程式檔案名稱	ch7_3_2.java
01	public class ch7_3_2
02	{
03	public static void main(String[] args)
04	{
05	double PI= 3.14159265358979F;
06	System.out.println ("Sin(30)="+ Math.sin (30 * PI / 180));
07	System.out.println ("Cos(30)="+ Math.cos (30 * PI / 180));
08	System.out.println ("Tan(45)="+ Math.tan (45 * PI / 180));
09	}
10	}

【執行結果】

```
Sin(30)=0.5000000126183913
Cos(30)=0.8660253964992068
Tan(45)=1.000000043711391
```

7-3.3 Math.asin()、Math.acos()、 Math.atan() 反三角函數



【語法1】`Sin_value= Math.asin(X)` //傳回X數值的反正弦值

【語法2】`Cos_value= Math.acos(X)` //傳回X數值的反餘弦值

【語法3】`Tan_value= Math.atan(X)` //傳回X數值的反正切值

【說明】將角度轉成徑度的計算公式如下所示：

徑度 = 角度 * PI / 180 ,

其中PI是指圓週率3.14159265358979

【舉例】

程式檔案名稱	ch7_3_3.java
01	public class ch7_3_3
02	{
03	public static void main(String[] args)
04	{
05	double PI= 3.14159265358979;
06	System. out .println ("Sin(30)=" + Math.asin(30 * PI / 180));
07	System. out .println("Cos(30)=" + Math.acos(30 * PI / 180));
08	System. out .println("Tan(45)=" + Math.atan(45 * PI / 180));
09	}
10	}

【執行結果】

```
Sin(30)=0.5510695830994458
Cos(30)=1.019726743695451
Tan(45)=0.6657737500283534
```



7-3.4 Math.exp (x) 指數函數

【語法】`Exp_value=Math.exp(x)` //傳回e的x次方，也就是 e^x

【說明】e的值是2.71828182845905

當 $x > 709.782712893$ 時，將會產生溢位。因為超出
Double(雙精準度)的表示範圍。

【舉例】

程式檔案名稱	ch7_3_4.java
01	<pre>public class ch7_3_4</pre>

```
01 public class ch7_3_4
02 {
03     public static void main(String[] args)
04     {
05         System.out.println("Exp(1)=" + Math.exp(1));
06         System.out.println("Exp(2)=" + Math.exp(2));
07     }
08 }
```

【執行結果】

```
Exp(1)=2.718281828459045
Exp(2)=7.38905609893065
```

Ps:程式中的Math.exp(2)在數學上是寫成 e^2

7-3.5 Math.floor() 取 $\leq x$ 的最大整數值



【語法】Math.floor(x);

【說明】取 $\leq x$ 的最大整數值

【舉例】

程式檔案名稱	ch7_3_5.java
01	public class ch7_3_5
02	{
03	public static void main(String[] args)
04	{
05	System.out.println("Math.floor(99.9)=" + Math.floor(99.9));
06	System.out.println("Math.floor(-99.9)=" + Math.floor(-99.9));
07	System.out.println("Math.floor(1.99)=" + Math.floor(1.99));
08	}
09	}

【執行結果】

```
Math.floor(99.9)=99.0
Math.floor(-99.9)=-100.0
Math.floor(1.99)=1.0
```



7-3.6 Math.log() 取對數值

【語法】`Math.log(x);`

【說明】取x以e為底數的對數值，而Log與Exp互為反函數

$$\rightarrow \text{Exp}(\text{Log}(x)) = \text{Log}(\text{Exp}(x))$$

在數學上的 $e^x = Y$ 則 $\text{Log} e Y = x$ ，也就是說：

如果 $Y = \text{Exp}(x)$ 時，則 $x = \text{Log}(Y)$

【舉例】

程式檔案名稱	ch7_3_6.java
01	<code>public class ch7_3_6</code>
02	<code>{</code>
03	<code>public static void main(String[] args)</code>
04	<code>{</code>
05	<code> System.out.println("Exp(Log(100))=" + Math.exp(Math.Log(100)));</code>
06	<code>}</code>
07	<code>}</code>

【執行結果】

`Exp(Log(100))=100.00000000000004`

7-3.7 Math.Log10() 以10為底數取對數值



【語法】`Math.Log10(x);`

【說明】取x以10為底數的對數值。

【舉例】

程式檔案名稱	ch7_3_7.java
01	<code>public class ch7_3_7</code>
02	<code>{</code>
03	<code>public static void main(String[] args)</code>
04	<code>{</code>
05	<code>System.out.println("Exp(Log10(100))=" + Math.exp(Math.Log10(100)));</code>
06	<code>}</code>
07	<code>}</code>

【執行結果】

Exp(Log10(100))=7.38905609893065



7-3.8 Math.max() 取較大者

【語法】`Math.max(x,y);`

【說明】取x與y較大者。

【舉例】

程式檔案名稱	ch7_3_8.java
01	<code>public class ch7_3_8</code>
02	<code>{</code>
03	<code> public static void main(String[] args)</code>
04	<code>{</code>
05	<code> System.out.println("Max(10,20)=" + Math.max(10,20));</code>
06	<code>}</code>
07	<code>}</code>

【執行結果】

Max(10,20)=20



7-3.9 Math.min() 取較小者

【語法】`Math.min(x,y);`

【說明】取x與y較小者。

【舉例】

程式檔案名稱	ch7_3_9.java
01	<code>public class ch7_3_9</code>
02	<code>{</code>
03	<code>public static void main(String[] args)</code>
04	<code>{</code>
05	<code> System.out.println("Min(10,20)=" + Math.min(10,20));</code>
06	<code>}</code>
07	<code>}</code>

【執行結果】

Min(10,20)=10



7-3.10 Math. pow() 取次方

【語法】`Math.pow(x,y);`

【說明】取x的y次方。

【舉例】

程式檔案名稱	<code>ch7_3_10.java</code>
01	<code>public class ch7_3_10</code>
02	<code>{</code>
03	<code> public static void main(String[] args)</code>
04	<code>{</code>
05	<code> System.out.println("Pow(2,10)="+ Math.pow(2, 10));</code>
06	<code>}</code>
07	<code>}</code>

【執行結果】

`Pow(2,10)=1024.0`

7-3.11 Math.round() 取四捨五入



【語法】`Math.round(x);`

【說明】1. 可依照指定所求出X的小數點位數並四捨五入，如果X數值
小數點右邊第一位是大於5時，則X數值的整數部份會加1
2. 在程式語言中，X數值的四捨五入是指 $X > 5$ ，如果 $X = 5$ ，
則會被捨去。如下面的例子所示：

【舉例】

程式檔案名稱	ch7_3_11.java
--------	---------------

```
01 public class ch7_3_11
02 {
03     public static void main(String[] args)
04     {
05         System.out.println ("Round(100.15)=" + Math.round(100.15)) ;
06         System.out.println ("Round(1.51)=" + Math.round(1.51));
07     }
08 }
```

【執行結果】

Round(100.15)=100 Round(1.51)=2



7-3.12 Math.sqrt() 取平方根

【語法】`Math.sqrt(x);`

【說明】取x的平方根，x必須 ≥ 0 ，否則程式會產生錯誤。程式中的
`Math.sqrt(x)`是數學上的 $X^{1/2}$

【舉例】

程式檔案名稱	ch7_3_12.java
01	<code>public class ch7_3_12</code>
02	<code>{</code>
03	<code>public static void main(String[] args)</code>
04	<code>{</code>
05	<code> System.out.println("Sqrt(2)="+ Math.sqrt(2));</code>
06	<code> System.out.println("Sqrt(4)="+ Math.sqrt(4));</code>
07	<code> System.out.println("Sqrt(8)="+ Math.sqrt(8));</code>
08	<code>}</code>
09	<code>}</code>

【執行結果】

```
Sqrt(2)=1.4142135623730951
Sqrt(4)=2.0
Sqrt(8)=2.8284271247461903
```

7-3.13 Math.ceil() 取 $\geq x$ 的最小整數值



【語法】Math.ceil(x);

【說明】取 $\geq x$ 的最小整數值

【舉例】

程式檔案名稱	ch7_3_13.java
01	public class ch7_3_13
02	{
03	public static void main(String[] args)
04	{
05	System.out.println("Ceil(99.9)=" + Math.ceil(99.9));
06	System.out.println("Ceil(-99.9)=" + Math.ceil(-99.9));
07	System.out.println("Ceil(1.99)=" + Math.ceil(1.99));
08	}

【執行結果】

```
Ceil(99.9)=100.0
Ceil(-99.9)=-99.0
Ceil(1.99)=2.0
```

7-3.14 Math.PI 取圓周率



【語法】Math.PI;

【說明】取得圓周率π值

程式檔案名稱	ch7_3_14.java
01	public class ch7_3_14
02	{
03	public static void main(String[] args)
04	{
05	System.out.println("PI=" + Math.PI);
06	}
07	}

【執行結果】

PI=3.141592653589793



7-3.15 Math.E 取指數

【語法】Math.E;

【說明】取得指數e值(約為2.71828)

程式檔案名稱	ch7_3_15.java
01	public class ch7_3_15
02	{
03	public static void main(String[] args)
04	{
05	System.out.println("e=" + Math.E);
06	}
07	}

【執行結果】

```
e=2.718281828459045
```



7-4 亂數類別(Random)

Math.random()亂數類別會產生一個大於或等於0，但小於1的數值，即 $0 \leq \text{Math.random()} < 1$ 。如果我們要取得某一特定範圍的亂數時，我們可以套用以下的公式：

語法： $\text{Math.random()} * (\text{上限} - \text{下限} + 1) + \text{下限};$

例如：如果我們拿投擲骰子的每一個點當作1到6的亂數，其上限值為6；下限值為1，則我們可以套用上面的公式得：

$\text{Math.random()} * (6 - 1 + 1) + 1;$

或簡化為： $\text{Math.random()} * 6 + 1;$

【實例1】從1~100之中來自動產生10個亂數值

程式檔案名稱	ch7-4A.java
01	<pre>public class ch7_4A { public static void main(String[] args) { int[] A = new int[11]; int i; System.out.println("產生10個亂數值(範圍：1~6)："); //處理 for (i = 1; i <= 10; i++) { A[i] =(int)(Math.random()*6)+1; ; //產生1~6的整數亂數值 System.out.print(A[i] + " "); } } }</pre>

【執行結果】

產生10個亂數值(範圍：1~6)：

4 2 4 4 1 2 4 4 1 6

【實例2】統計投擲100次骰子各點的次數



程式檔案名稱	
	ch7_4B.java

```
01 public class ch7_4B
02 {
03     public static void main(String[] args)
04     {
05         String TempStr;
06         int[] A = new int[7];
07         int i, j, x;
08         System.out.println("投擲骰子100次，各點出現的次數：");
09         //處理
10         for (i = 1; i <= 100; i++)
11         {
12             x =(int)(Math.random()*6)+1;    ; //產生1~6的整數亂數值
13             A[x]=A[x]+1;
14         }
15         for(j = 1;j<=6;j++)
16             System.out.println(j + "點" + " " + A[j] + "次");
17     }
18 }
```

【執行結果】

投擲骰子100次，各點出現的次數：

1點 16次

2點 18次

3點 17次

4點 17次

5點 19次

6點 13次



7-5 日期類別(DateTime)

在我們利用程式來處理日期與時間性的資料時，如果能夠善加利用JAVA內建的日期類別時，則可以節省許多時間。例如欲設計一個網路預約上課系統時，而其開放時間只有星期一到五，其餘時間不開放。

想要利用取得系統的日期或時間時，首先，宣告及建立Date物件，再透過Calendar類別來完成。

步驟一：匯入「`import java.util.*;`」

步驟二：宣告及建立Date物件

【語法】`Date dt = new Date();`

【實例1】取得今天的日期與時間

程式檔案名稱	ch7_5A.java
--------	-------------

```
01 import java.util.*;  
02 public class ch7_5A  
03 {  
04     public static void main(String[] args)  
05     {  
06         Date dt = new Date();  
07         System.out.println("今天的日期：" + dt);  
08     }  
09 }
```

【執行結果】

```
今天的日期：Thu Sep 03 10:38:07 CST 2015
```

說明：利用Date()物件所取得的內容為：星期、月、日、時間、地區及西元

由於利用Date()物件所取得的內容相當的多，但是，我們的需求可能只須要某些資料即可，因此，我們可以透過Calendar類別來完成。

【實例2】傳回今年的西元(Calendar.YEAR)

程式檔案名稱	ch7_5B.java
01	<code>import java.util.*;</code>
02	<code>public class ch7_5B</code>
03	<code>{</code>
04	<code> public static void main(String[] args)</code>
05	<code> {</code>
06	<code> Calendar cal = Calendar.getInstance();</code>
07	<code> int dt;</code>
08	<code> dt=cal.get(Calendar.YEAR);</code>
09	<code> System.out.println("今年的西元：" + dt);</code>
10	<code> }</code>
11	<code>}</code>

【執行結果】

今年的西元：2015

【實例3】傳回今年的月份(Calendar.MONTH)

程式檔案名稱	ch7_5C.java
--------	-------------

```
01 import java.util.*;  
02 public class ch7_5C  
03 {  
04     public static void main(String[] args)  
05     {  
06         Calendar cal = Calendar.getInstance();  
07         int dt;  
08         dt=cal.get(Calendar.MONTH);  
09         System.out.println("今年的月份：" + (dt+1));  
10     }  
11 }
```

【執行結果】

今年的月份：9

注意：系統取得的月份，如果是0時代表1月，1代表2月，...，以此類推。所以，取得的月份必須要再加「1」，才符合我們的需求。

【實例4】傳回今年的月份之日期(Calendar.DAY_OF_MONTH)

程式檔案名稱	ch7_5D.java
--------	-------------

```
01 import java.util.*;  
02 public class ch7_5D  
03 {  
04     public static void main(String[] args)  
05     {  
06         Calendar cal = Calendar.getInstance();  
07         int dt;  
08         dt=cal.get(Calendar.DAY_OF_MONTH);  
09         System.out.println("今年的月份之日期：" + dt);  
10     }  
11 }
```

【執行結果】

今年的月份之日期：3

【實例5】傳回現在的時間<小時>(Calendar.HOUR_OF_DAY)

程式檔案名稱	ch7_5E.java
--------	-------------

```
01 import java.util.*;  
02 public class ch7_5E  
03 {  
04     public static void main(String[] args)  
05     {  
06         Calendar cal = Calendar.getInstance();  
07         int dt;  
08         dt=cal.get(Calendar.HOUR_OF_DAY);  
09         System.out.println("現在的時間：" + dt);  
10     }  
11 }
```

【執行結果】

現在的時間：10

【實例6】傳回現在的時間<分> (Calendar.MINUTE)

程式檔案名稱	ch7_5F.java
01	<pre>import java.util.*;</pre>

```
02 public class ch7_5F
03 {
04     public static void main(String[] args)
05     {
06         Calendar cal = Calendar.getInstance();
07         int dt;
08         dt=cal.get(Calendar.MINUTE);
09         System.out.println("現在的時間<分>：" + dt);
10     }
11 }
```

【執行結果】

現在的時間<分>：54

【實例7】傳回現在的時間<秒> (Calendar.SECOND)

程式檔案名稱	ch7_5G.java
--------	-------------

```
01 import java.util.*;  
02 public class ch7_5G  
03 {  
04     public static void main(String[] args)  
05     {  
06         Calendar cal = Calendar.getInstance();  
07         int dt;  
08         dt=cal.get(Calendar.SECOND);  
09         System.out.println("現在的時間<秒>：" + dt);  
10     }  
11 }
```

【執行結果】

現在的時間<秒>：1

【實例8】傳回現在的時間<毫秒> (Calendar.MILLISECOND)

程式檔案名稱

ch7_5H.java

```
01 import java.util.*;
02 public class ch7_5H
03 {
04     public static void main(String[] args)
05     {
06         Calendar cal = Calendar.getInstance();
07         int dt;
08         dt=cal.get(Calendar.MILLISECOND);
09         System.out.println("現在的時間<毫秒>：" + dt);
10     }
11 }
```

【執行結果】

現在的時間<毫秒>：543

課後評量



1. 請設計一支程式，可以將10進位轉換成2進位

例如 $1 : 22_{(10)} = 10110_{(2)}$

【演算過程】

2	22	0
2	11	1
2	5	1
2	2	0
2	1	1
0		

2.【題目】直角三角形列印

【說明】請從表單中讀取一個數字，列印從1開始直到該數字為止之直角三角形。

【輸入範例】7

【輸出報表】

請輸入一個數字

7

1

12

123

1234

12345

123456

1234567

3.【題目】質數計算

【說明】請從表單中讀取一個數字，若此數字是質數則印出此數字及“是質數”，若不是則印出此數字及“不是質數”

【輸入範例】12

【輸出報表】12不是質數

是質數	不是質數
請輸入一個數字 13 13 是質數	請輸入一個數字 12 12不是質數

【解析】

以 for 迴圈將 N 分別除以 1 到 N，並計算整除的個數，若個數為 2 (1 及本身)，則此數為質數。

4.【題目】體質指數BMI

【說明】請從表單中讀取兩個數字(分別為身高與體重)，而體質指數
BMI(Body Mass Index)是常用在評估人體肥胖程度的一種指
標，其計算公式為體重除以身高的平方：
$$BMI = \text{體重(公斤)} / (\text{身高} * \text{身高}) \text{ (公尺}^2)$$

一般而言，正常的體重其BMI範圍=20~25。請設計一個程
式，輸入3組身高與體重後，將BMI質最小者印出並判斷是否
在正常範圍內(BMI之計算身高以公尺，體重以公斤計算，計
算至個位數，小數點後第一位數以四捨五入計算)

【輸入範例】

#1	176	45
----	-----	----



【輸出報表】BMI值=14.5，不正常。

正常	不正常
請輸入身高 172	請輸入身高 165
請輸入體重 66	請輸入體重 75
BMI 值：22 是正常	BMI值：28 不正常