Java程式設計入門
流程控制－分支結構

鄭安翔

ansel_cheng@hotmail.com

課程大綱

1) 取得使用者輸入
2) if-else 陳述句
3) switch-case 陳述句

取得使用者輸入

- 標準輸入System.in
  - System 類別的標準輸入成員 in (InputStream型態)
  - 預設資料來源裝置是鍵盤(Keybord)
  - 提供的 read() 方法：一次取得一個位元的資料,資料以int型態傳回

- java.util.Scanner類別
  - JDK1.5新增的類別,可以簡便的取得使用者輸入

    java.util.Scanner sc = new java.util.Scanner(System.in);
  - 使用Scanner所提供的nextXXX()方法取得使用者輸入的各式資料

java.util.Scanner類別

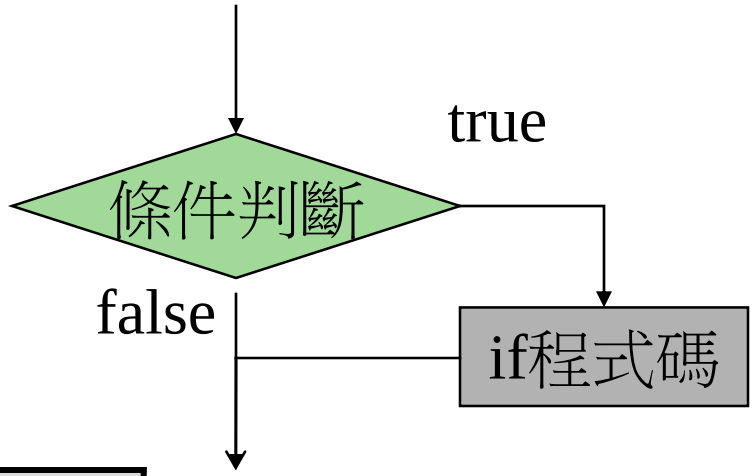
```
01  public class ScannerDemo {
02      public static void main(String args[]) {
03          java.util.Scanner scanner = new java.util.Scanner(System.in);
04
05          System.out.print(“輸入整數:");
06          int input1 = scanner.nextInt();
07
08          System.out.print(“輸入浮點數:");
09          double input2 = scanner.nextDouble();
10
11          System.out.print(“輸入布林值:");
12          boolean input3 = scanner.nextBoolean();
13
14          System.out.print(“輸入字串:");
15          String input4 = scanner.next();
16
17          System.out.println("整數輸入:" + input1);
18          System.out.println(“浮點數輸入:" + input2);
19          System.out.println(“布林值輸入:" + input3);
20          System.out.println(“字串輸入:" + input4);
21      }
22  }
```

課程大綱

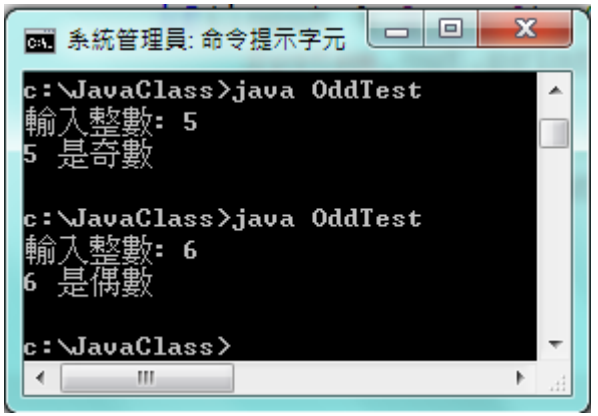
1) 取得使用者輸入
2) **if-else 陳述句**
3) switch-case 陳述句

Branching Statements – if

```
if (boolean) {
    statements;
}
```

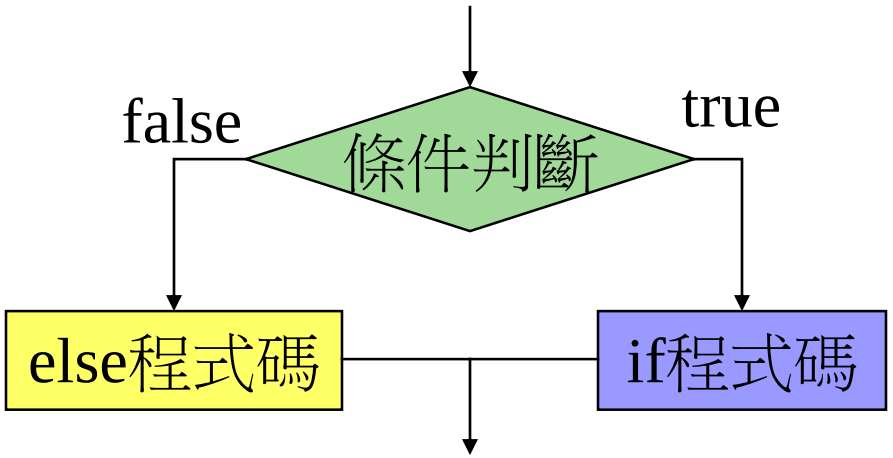



```
01  public class OddTest {
02      public static void main(String[] args) {
03          java.util.Scanner scanner =
04          new java.util.Scanner(System.in);
05
06          System.out.print("輸入整數: ");
07          int input = scanner.nextInt();
08
09          if(input % 2 == 0) //如果餘數為 0
10              System.out.println(input + " 是偶數");
11
12          if(input % 2 != 0) //如果餘數不為 0
13              System.out.println(input + " 是奇數");
14      }
15  }
```

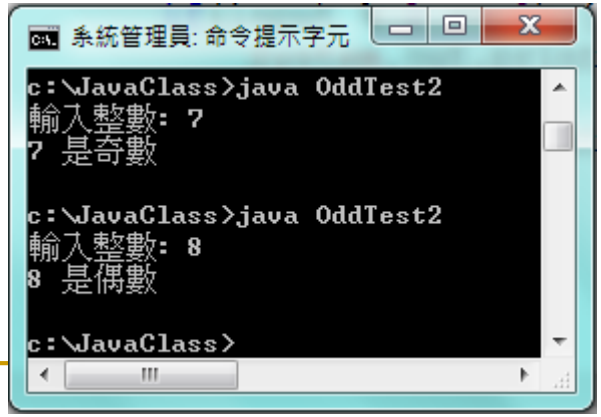


Branching Statements – if / else

```
if (boolean) {
    statements;
} else {
    statements;
}
```

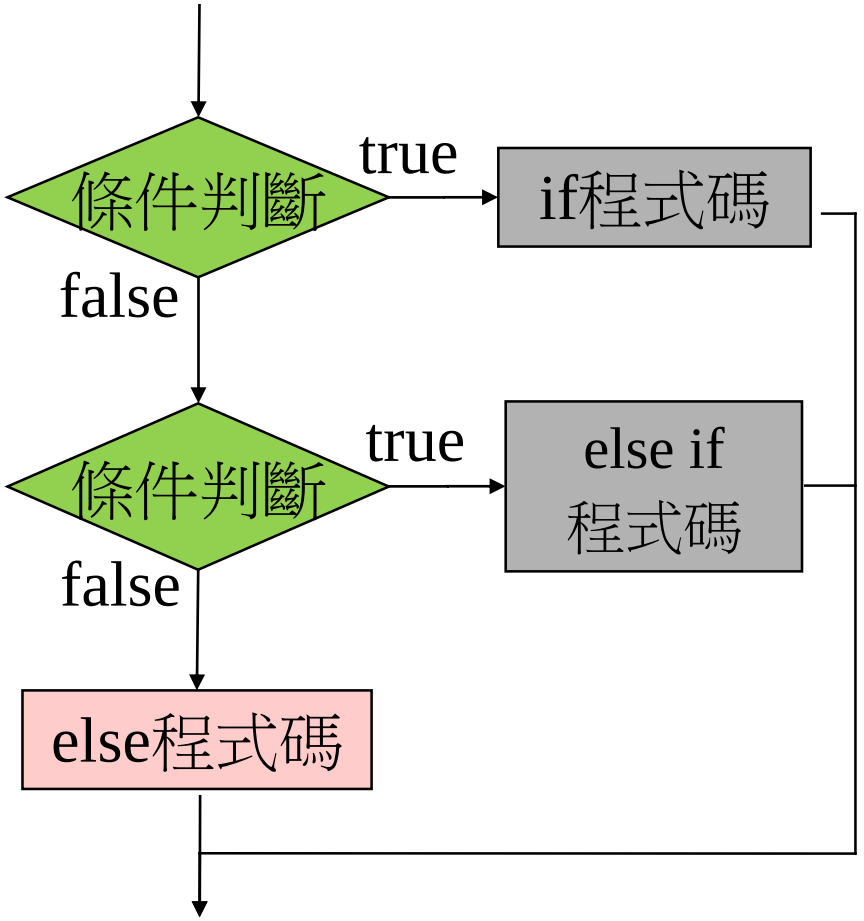



```
01  public class OddTest2 {
02      public static void main(String[] args) {
03          java.util.Scanner scanner =
04          new java.util.Scanner(System.in);
05
06          System.out.print("輸入整數: ");
07          int input = scanner.nextInt();
08
09          if(input % 2 == 0) {
10              System.out.println(input + " 是偶數");
11          } else {
12              System.out.println(input + " 是奇數");
13          }
14      }
15  }
```



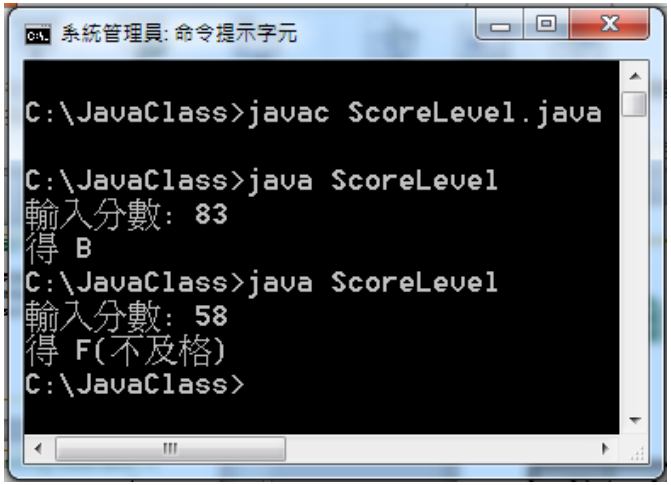
Branching Statements– if/else if/else

```
if (boolean) {
  statements;
} else if (boolean) {
  statements;
} else {
  statements;
}
```



Branching Statements– if/else if/else

```
01  public class ScoreLevel {
02      public static void main(String[] args) {
03          java.util.Scanner scanner =
04          new java.util.Scanner(System.in);
05
06          System.out.print("輸入分數: ");
07          int score = scanner.nextInt();
08
09          if(score >= 90)
10              System.out.print("得 A ");
11          else if(score >= 80 && score < 90)
12              System.out.print("得 B ");
13          else if(score >= 70 && score < 80)
14              System.out.print("得 C ");
15          else if(score >= 60 && score < 70)
16              System.out.print("得 D ");
17          else
18              System.out.print("得 F(不及格)");
19      }
20  }
```



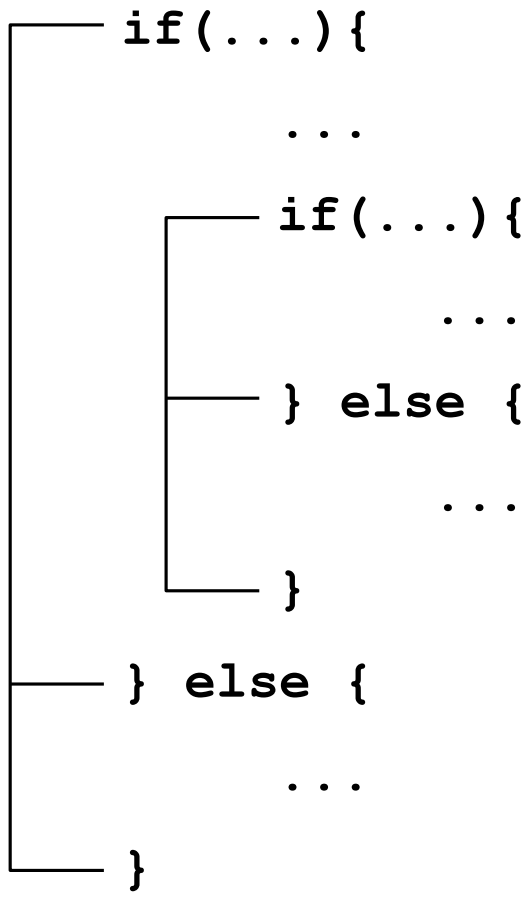
```
01 int money = 100;
02 if(money = 100)
03     System.out.println("等於 100");
04 }
05 else {
06     System.out.println("不等於 100");
07 }
```

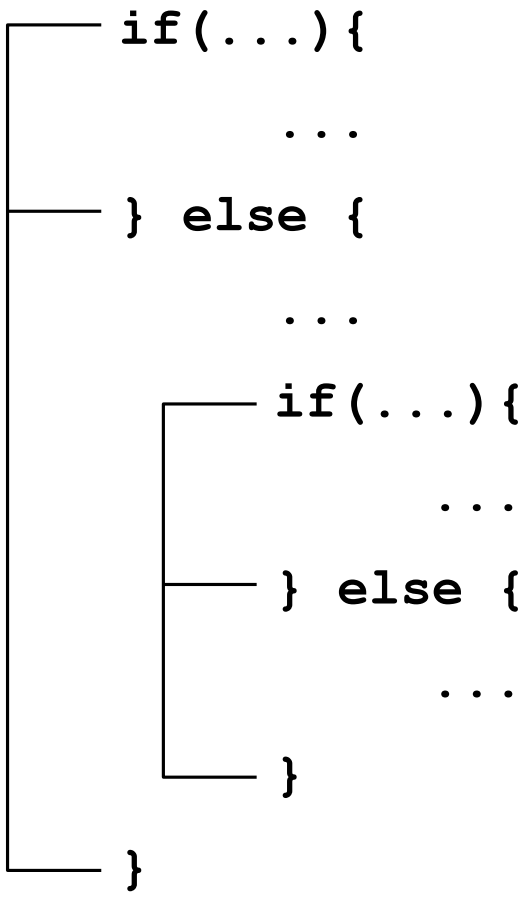
執行結果：編譯錯誤

```
01 boolean b = false;
02 if(b = false) {
03     System.out.println("false");
04 }
05 else {
06     System.out.println("true");
07 }
```

執行結果：true

巢狀結構

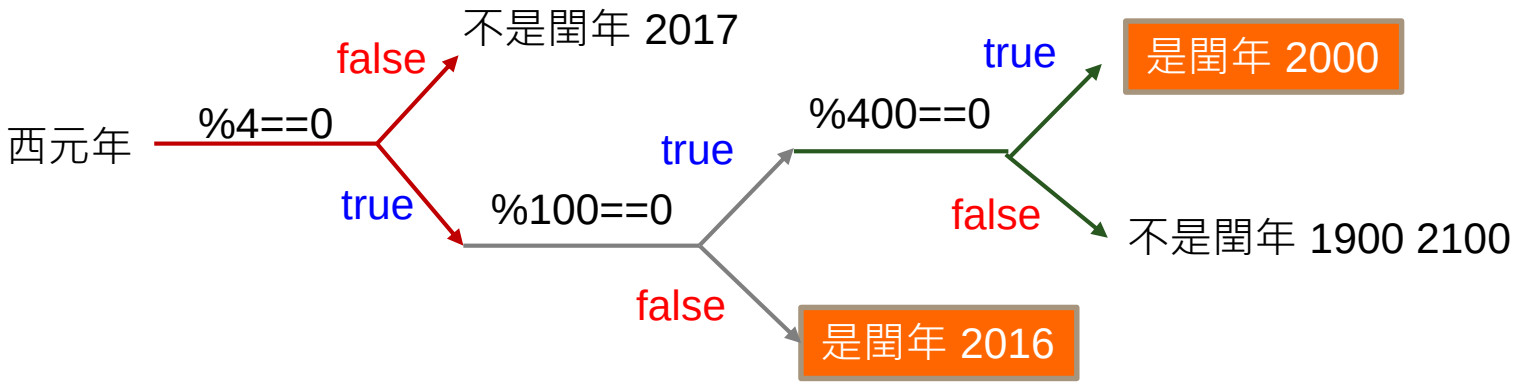




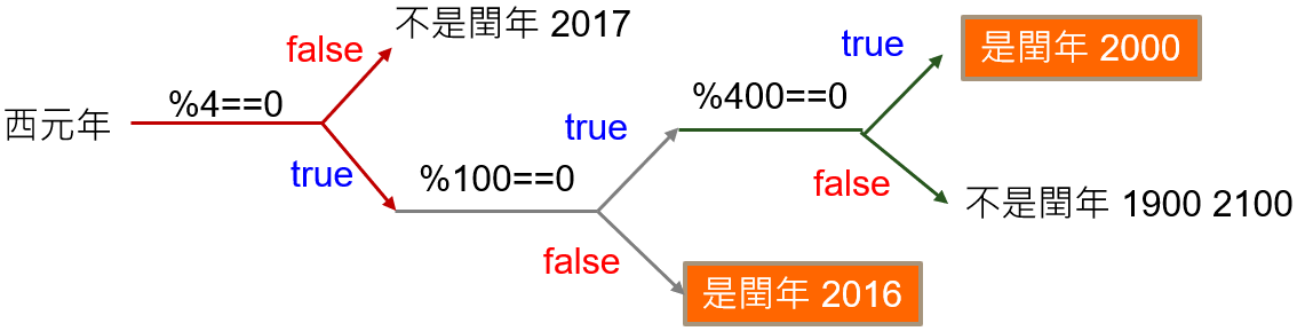
何者為閏年？

- 閏年條件
  - 西元年是 4 的倍數，但不是 100 的倍數
  - 西元年是 400 的倍數

1900 / 2000 / 2016 / 2017 / 2100



何者為閏年？




```
System.out.print("請輸入西元年：");
int year = scanner.nextInt();
if(year%4==0){
    if(year%100==0){
        if(year%400==0) {
            System.out.println(year +"年是閏年",);
        } else {
            System.out.println(year +"年不是閏年",);
        }
    } else {
        System.out.println(year +"年是閏年",);
    }
} else {
    System.out.println(year+"年不是閏年");
}
```

if – else 配對

- 當程式碼只有一行時，括號可省略
- else 配對時，應由前面的先配對
- else 先和最靠近自己的 if 配對
- 若最靠近的 if 已經配對了，則找次靠近者

```
int a = 0, i = 1, j = -1, k = 2;
if ( i >0 )
        if (j > 0)
                if (k > 0)
                        a = 100;
                else
                        a = 200;
        else
                a = 300;
System.out.println("a= " + a);
```

課程大綱

1) 取得使用者輸入
2) if-else 陳述句
3) **switch-case 陳述句**

Branching Statements - Switch

```
switch (variable) {
        case literal_value:
                statements;
                [break;]
        case literal_value:
                statements;
                [break;]
        [default:]
                statements;
}
```

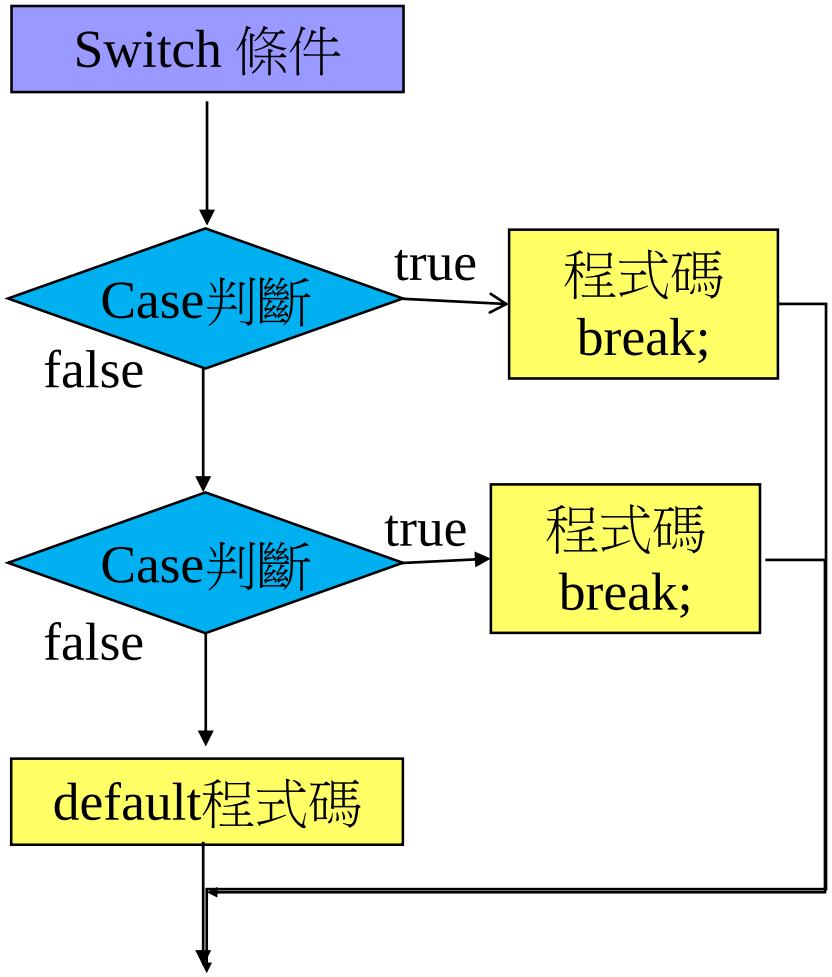
switch

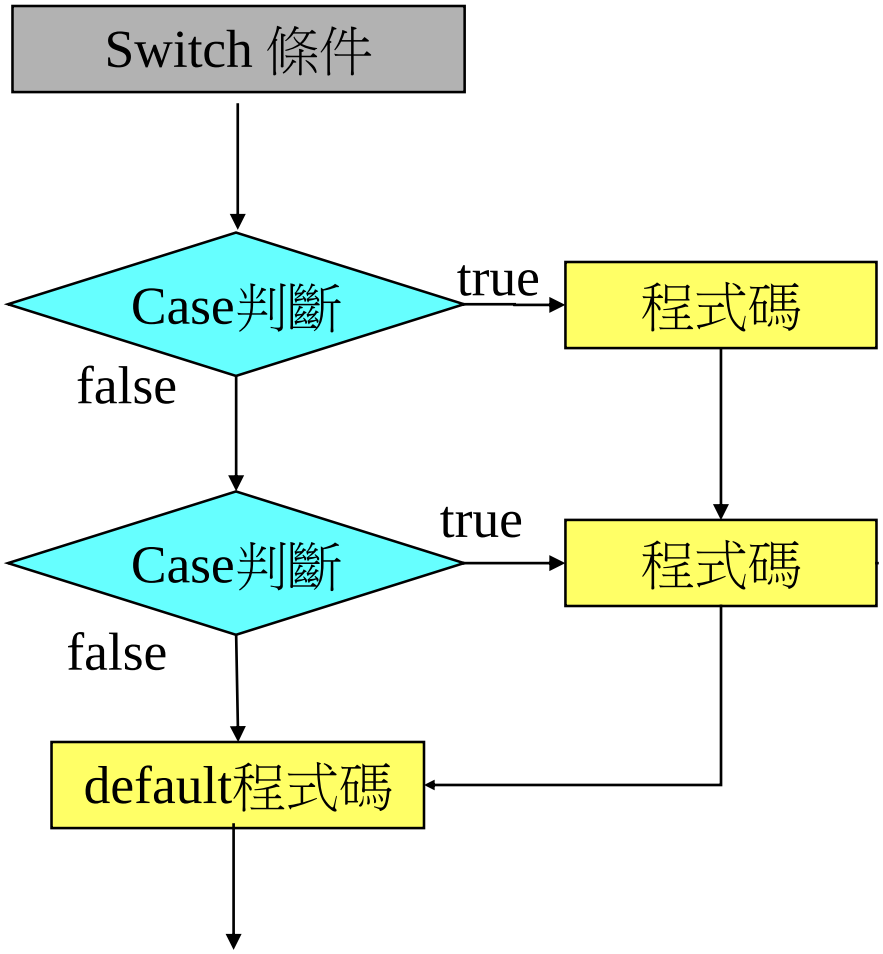
- 分支結構
  - 針對單一變數
  - 等於條件判斷
- 比對鍵值
  - byte, short, char, int, enum, String
- 運作特性
  - case 和 default 視為標籤，其順序沒有限制。
  - default case ：排除所有比對子列舉值之外的其他值
  - case 後面只能接**常數**或是**常數的運算式**，不能包含變數

switch

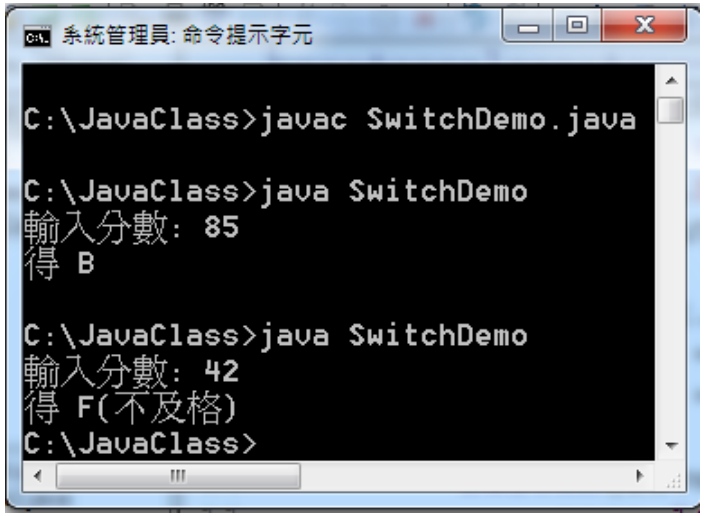
- 運作特性
  - 比對時，比對參數與比對子會轉換成 int 來進行比對
    - String 使用equals()方法比對
  - default 敘述，不一定要放在switch 結構最後
    - 如果不放在最後的話，需要加上 break 敘述
  - break 敘述省略，程式執行會將下一個比對內容一併執行

Switch


Switch 條件
Case判斷
true
程式碼
break;
false
Case判斷
true
程式碼
break;
false
default程式碼


Switch 條件
Case判斷
true
程式碼
false
Case判斷
true
程式碼
false
default程式碼

```
01  public class SwitchDemo {
02      public static void main(String[] args) {
03              java.util.Scanner scanner =
04              new java.util.Scanner(System.in);
05              System.out.print("輸入分數: ");
06              int score = scanner.nextInt();
07              int level = (score/10);
08
09              switch(level) {
10                  case 10:
11                  case  9:
12                          System.out.println("得 A ");
13                          break;
14                  case  8:
15                          System.out.println("得 B ");
16                          break;
17                  case  7:
18                          System.out.println("得 C ");
19                          break;
20                  case  6:
21                          System.out.println("得 D");
22                          break;
23                  default:
24                          System.out.print("得 F(不及格)");
25              }
26      }
27  }
```



```
01 int x = 1;
02 switch(x) {
03   case 1:
04     System.out.print("A");
05     break;
06   case 2:
07     System.out.print("B");
08 }
```

執行結果：A

```
01 int x = 1;
02 switch(x) {
03   case 1:
04     System.out.print("A");
05   case 2:
06     System.out.print("B");
07 }
```

執行結果：AB

```
01 char x = 'A';
02 switch(x) {
03   case 'A':
04     System.out.print("A");
05     break;
06   case 'B':
07     System.out.print("B");
08 }
```

執行結果：A

```
01 char x = 'A';
02 switch(x) {
03   case 65:
04     System.out.print("A");
05     break;
06   case 66:
07     System.out.print("B");
08 }
```

ASCII(65) = A

執行結果：A

```
01 char x = 'A';
02 char valueA = 'A';
03 switch(x) {
04   case valueA:
05     System.out.print("A");
06     break;
07   case 'B':
08     System.out.print("B");
09 }
```

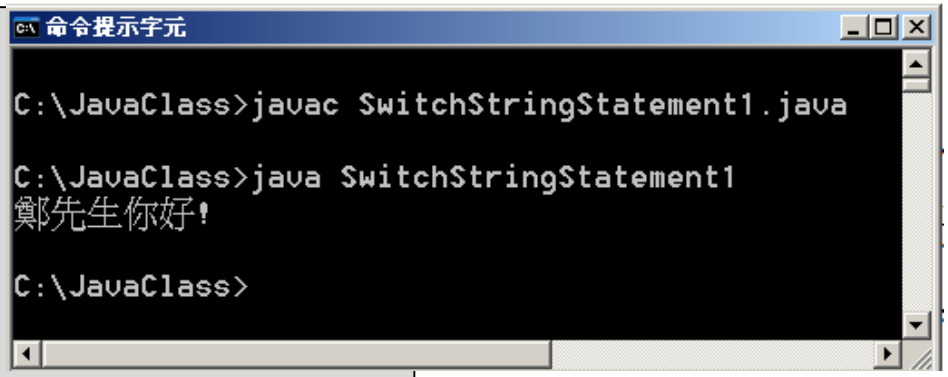
編譯錯誤！

```
01 char x = 'A';
02 final char valueA = 'A';
03 switch(x) {
04   case valueA:
05     System.out.print("A");
06     break;
07   case 'B':
08     System.out.print("B");
09 }
```

執行結果：A

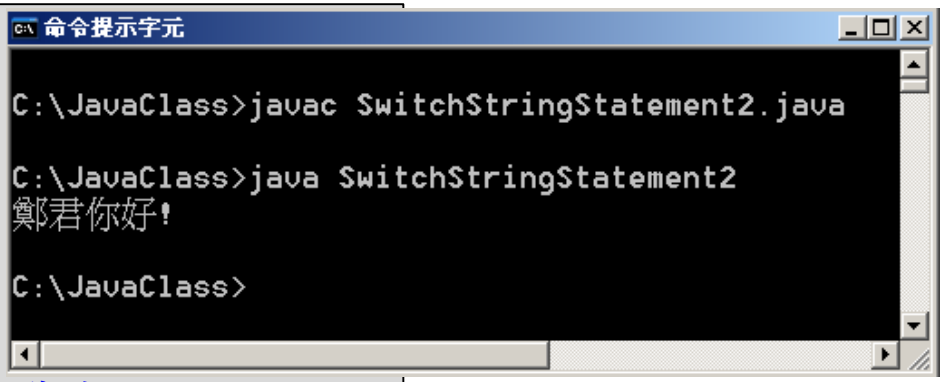
Java 7：Switch結構使用字串鍵值

```
public class SwitchStringStatement 1{
    public static void main(String args[]){
        String gender = “Male";
        String lastName = "鄭";
        String nameFormat;
        switch (gender){
            case “Male":
                nameFormat = lastName + "先生";
                break;
            case “Female":
                nameFormat = lastName + "小姐" ;
                break;
            default:
                nameFormat = lastName + "君" ;
        }
        System.out.println(nameFormat + "你好!" );
    }
}
```



Java 7：Switch結構使用字串鍵值

```
public class SwitchStringStatement 2{
    public static void main(String args[]){
        String gender = “male";
        String lastName = "鄭";
        String nameFormat;
        switch (gender){
            case “Male":
                nameFormat = lastName + "先生";
                break;
            case “Female":
                nameFormat = lastName + "小姐" ;
                break;
            default:
                nameFormat = lastName + "君" ;
        }
        System.out.println(nameFormat + "你好!" );
    }
}
```



Java 7：Switch結構使用字串鍵值

```
public class SwitchStringStatement 3{
    public static void main(String args[]){
        String gender = null;
        String lastName = "鄭";
        String nameFormat;
        switch (gender){
            case “Male":
                nameForm
                break;
            case “Female"
                nameForm
                break;
            default:
                nameFormat = lastName + "君" ;
        }
        System.out.println(nameFormat + "你好!" );
    }
}
```