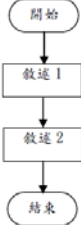
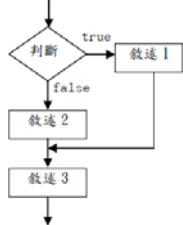
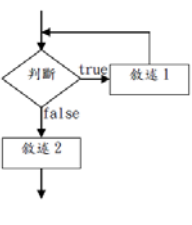


第五章 選擇性敘述與迴圈

- 5.1 認識程式的結構設計
- 5.2 學習選擇性敘述與各種迴圈的用法
- 5.3 學習多重選擇敘述的用法

1

類型	循序結構 (Sequence Structure)	選擇結構 (Selection Structure)	重複結構 (Iteration Structure)
說明	循序結構是程式預設的執行方式，程式是一個程式敘述接著一個敘述依序的執行	選擇結構是一種條件控制敘述，它可以分為單一選擇、二選一或多選一三種，程式執行的順序是依照關係判斷運算式的條件，來決定執行哪一個程式區塊的程式碼	重複結構是一種迴圈控制，它是重複執行一個程式區塊的程式碼，提供一個結束條件結束迴圈的執行，依據結束條件測試的位置不同分為兩種： 1. 前測式重複結構 2. 後測式重複結構
語法		<ul style="list-style-type: none"> if...else switch...case 	<ul style="list-style-type: none"> for while do...while
流程圖	 <pre> graph TD Start([開始]) --> S1[敘述 1] S1 --> S2[敘述 2] S2 --> End([結束]) </pre>	 <pre> graph TD Entry(()) --> Jud{判斷} Jud -- true --> S1[敘述 1] Jud -- false --> S2[敘述 2] S1 --> S3[敘述 3] S2 --> S3 S3 --> Exit(()) </pre>	 <pre> graph TD Entry(()) --> Jud{判斷} Jud -- true --> S1[敘述 1] Jud -- false --> S2[敘述 2] S1 --> Entry S2 --> Entry </pre>

2

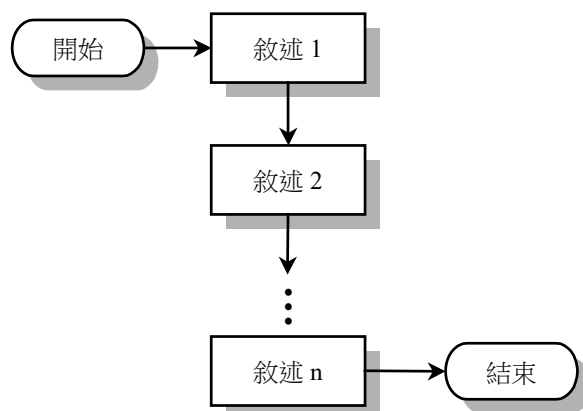
運算式、運算元與運算子

- 程式的結構包含有下面三種：
 - 循序性結構（sequence structure）
 - 選擇性結構（selection structure）
 - 重複性結構（iteration structure）

3

循序性結構

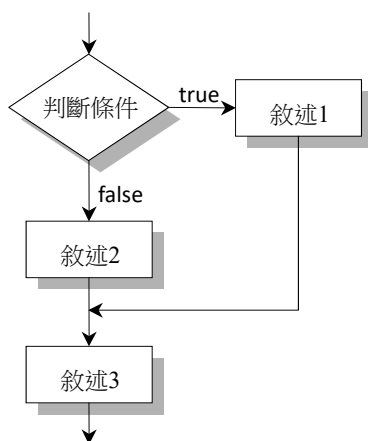
- 採上至下（top to down）的敘述執行



4

選擇性結構

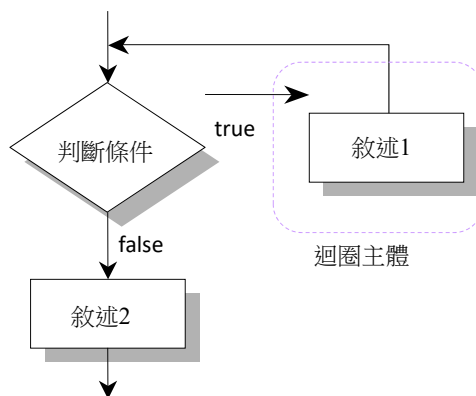
- 根據條件的成立與否，再決定要執行哪些敘述



5

重複性結構

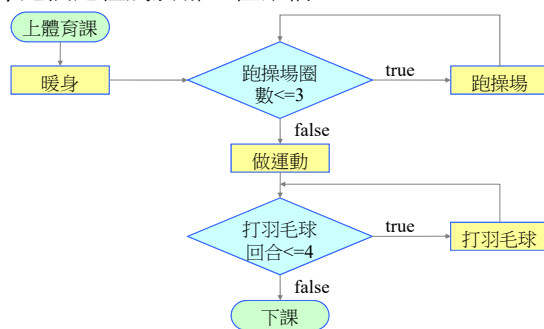
- 根據判斷條件的成立與否，決定程式段落的執行次數
- 重複性結構有for、while及do while三種迴圈



6

5.1 課堂練習

- 學生小惠將她某日上體育課的情形記錄下來，一開始先暖身，接著跑操場3圈、做運動、打羽毛球4回合，羽毛球打完後即下課。試回答下列的問題：
 - (a) 試繪製出適合的流程圖。
 - (b) 這個過程屬於哪一種結構？

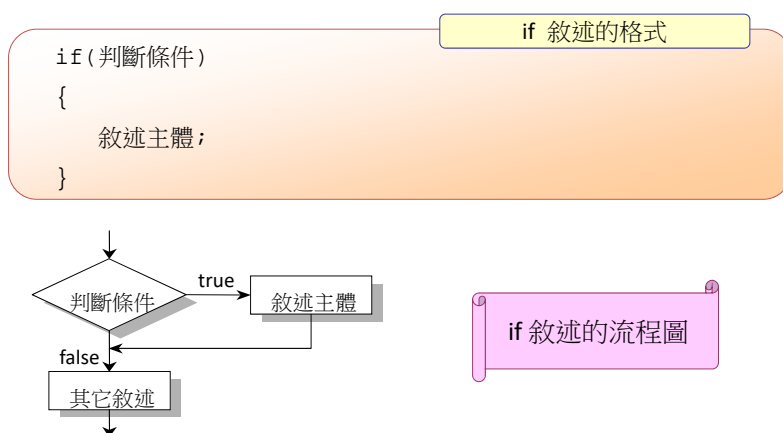


7

5.2 我的程式會轉彎--選擇性敘述

if 敘述

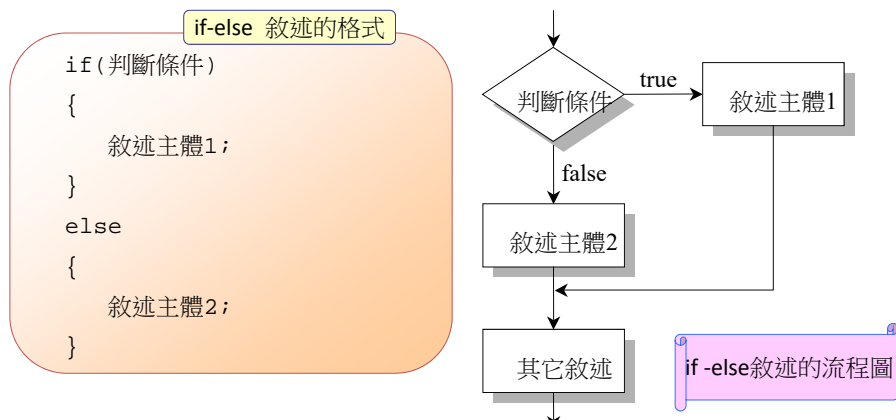
- 根據判斷的結果來執行不同的敘述



8

if-else 敘述 (1/2)

- if-else敘述的格式與流程圖如下：



9

if-else 敘述 (2/2)

- 下面的範例可用來判斷變數a是奇數或是偶數

```

01 // app5_1,if-else敘述
02 public class app5_1
03 {
04     public static void main(String args[])
05     {
06         int a=15;
07
08         if (a%2==0) // 如果可被2整除
09             System.out.println(a+" is an even number");// 印出a為偶數
10         else
11             System.out.println(a+" is an odd number");// 印出a為奇數
12     }
13 }

```

```

/* app5_1 OUTPUT---
15 is an odd number
-----*/

```

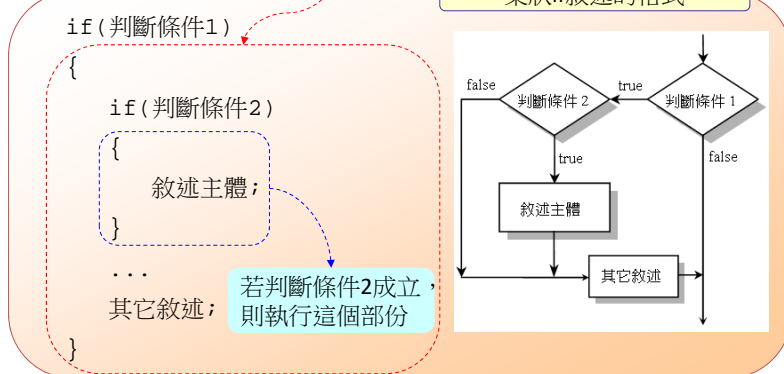
10

巢狀 if 敘述

- if 敘述中又包含其它 if 敘述時，稱為巢狀 if 敘述 (nested if)

若判斷條件1成立，
則執行這個部份

巢狀if敘述的格式



11

5.2 課堂練習

- ex5_2_1.java
 - 由鍵盤輸入一個大於0的整數
 - 被5與6整除，則yes!

*以if-else撰寫

```

/* output-----
Input an integer:28
No!
-----*/
  
```

12

5.2 課堂練習

- ex5_2_2.java
 - 由鍵盤輸入一個大於0的整數
 - 被5與6整除，則yes!

*以巢狀的if來撰寫

```

if(true)
{
    if(true)
    {
        .....
    }
    else
    {
        .....
    }
}
else
{
    .....
}

```

```

/* output-----
Input an integer:28
No!
-----*/

```

13

補充

if –else if-else if.....else

```

if(條件判斷1)
{敘述主體1};
else if(條件判斷2)
{敘述主體2};
else if(條件判斷3)
{敘述主體3};
....
else //捕捉其他條件
{敘述主體n}

```

14

5.2 課堂練習

ex5_2_3.java

- 程式設計課本每本500元，書商給予購書折扣，買一本9折，買兩本8折，買三本以上7折，請計算班級買幾本書籍時應該支付廠商多少錢？

設變數： 單價、本數、總計
變數型態： int? int ? float?

```
if (書為1本)
{9折}
else if(書為2本)
{8折}
else
{7折}
```

書本價格與購買本數為:500 2
總書價為800.0

15

ex5_2_3.java 講解1

```
import java.util.*;
public class ex5_2_3
{
    public static void main(String args[])
    {
        Scanner sca=new Scanner(System.in);
        int item;
        int price;
        float count;
        System.out.print("書本價格與購買本數為:");
        price=sca.nextInt();
        item=sca.nextInt();

        if(item==1)
            count=price*item*0.9;
        else if(item==2)
            count=price*item*0.8;
        else
            count=price*item*0.7;
        System.out.println("總書價為"+count);
    }
}
```

Type mismatch: cannot convert from double to float
????? Why?

16

ex5_2_3.java 講解2

```
import java.util.*;
public class ex5_2_3
{
    public static void main(String args[])
    {
        Scanner sca=new Scanner(System.in);
        int item;
        int price;
        float count;
        System.out.print("書本價格與購買本數為:");
        price=sca.nextInt();
        item=sca.nextInt();

        if(item==1)
            count=price*item*0.9f;
        else if(item==2)
            count=price*item*0.8f;
        else
            count=price*item*0.7f;
        System.out.println("總書價為"+count);
    }
}
```

書本價格與購買本數為:500 2
總書價為800.0

17

5.2 課堂練習

• ex5_2_4.java

假設某加油站的工讀生每個月打工的薪資，可以依照下列方式計算：

60個小時之內，基本時薪為每小時100元

61~75個小時，以基本時薪的1.5倍計算

76個小時以後以基本時薪的2.5倍計算

例如，如果工作時數為80小時，則薪資為

$60 \times 100 + 15 \times 100 \times 1.5 + 5 \times 100 \times 2.5 = 9500$ 元。

試撰寫一程式，設定某工讀生該月的工作時數為93小時，然後計算實領的薪資。

```
/* output-----
工作時數為: 93
實領薪資為 12750.0
-----*/
```

18

條件運算子 (1/2)

- 條件運算子的說明：

條件運算子	意義
?:	根據條件的成立與否，來決定結果為?或:後的運算式

- ?: 的格式：

?: 的敘述格式

傳回值 = 判斷條件 ? 運算式1 : 運算式2;

語法：變數=(條件判斷) ? (運算式1) : (運算式2);

功能：依照條件判斷的成立與否，分別執行運算式1或運算式2，並將結果回傳給=左邊的變數。

注意：有分號

19

補充

運算結果
等於 true

result = **condition** ? value1 : value2

運算結果
等於 false

20

補充

//範例 (Top.java) : 三元運算子

```

001 public class Top {
002
003     public static void main(String[] args){
004
005         int a = 300 ;
006         int b = 200 ;
007         int c = 500 ;
008
009         boolean abCompare = a>b ;
010         String resultabc = abCompare ? "a>b" : "a<b" ;
011         String resultacc = a>c ? "a>c" : "a<c" ;
012
013         System.out.println(resultabc) ;
014         System.out.println(resultacc) ;
015     }
016 }

```

abCompare = true

a>b : true結果

a>b : false結果

a>c= false

21

```

import java.util.*;
public class e
{
    public static void main(String args[])
    {
        Scanner sca=new Scanner(System.in);
        int a,b,c;
        System.out.print("請輸入a b c 三個整數:");
        a=sca.nextInt();
        b=sca.nextInt();
        c=sca.nextInt();

        boolean abcompare=a>b; //true
        String relultabc=abcompare ? "a>b" : "a<b";
        String relultacc=a>c ? "a>c" : "a<c";//a>c是false

        System.out.println(relultabc);
        System.out.println(relultacc);
    }
}

```

請輸入a b c 三個整數 300 200 500
a>b
a<c

22

補充

//範例 (TopVar.java) : 三元運算子

```

001 public class TopVar {
002     public static void main(String[] args){
003
004         int a = 300 ;
005         int b = 200 ;
006
007         boolean abCompare = a>b ;
008         String trueString = "a 大於 b" ;
009         String falseString = "a 小於 b" ;
010
011         String resultabc =
012             abCompare ? trueString : falseString ;
012         System.out.println(resultabc);
013     }
014 }

```

abCompare=true

true 結果

a 大於 b

23

5.2 我的程式會轉彎--選擇性敘述

條件運算子 (1/2)

- 下面的程式可找出二數之間較大的數：

```

01 // app5_2,條件運算子?:的使用
02 public class app5_2
03 {
04     public static void main(String args[])
05     {
06         int a=8,b=3,max;
07
08         max=(a>b)?a:b; // a>b時,max=a,否則max=b
09
10         System.out.println("a="+a+", b="+b);
11         System.out.println(max+"是較大的數");
12     }
13 }

```

```

if (a>b)
    max=a;
else
    max=b;

```

```

/* app5_2 OUTPUT----
a=8, b=3
8是較大的數
-----*/

```

24

5.2 課堂練習

- ex5_2_5.java
- 將ex5_2_1.java改成條件運算子來撰寫
 - ex5_2_1.java
 - 由鍵盤輸入數字
 - 被5與6整除，則yes!

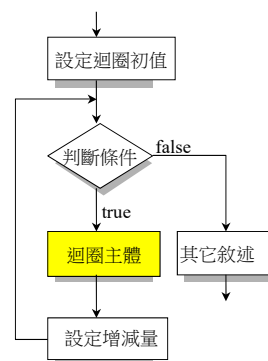
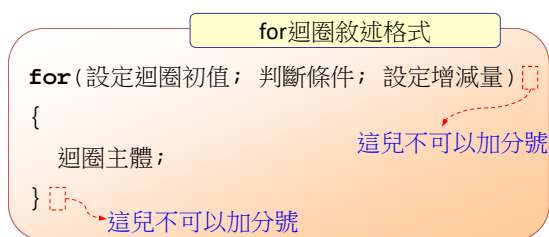
```
/* output-----
Input an integer:28
No!
-----*/
```

25

5.3 迴圈

for迴圈 (1/2)

- for迴圈的格式及執行流程：



1. 第一次進入for迴圈時，設定迴圈控制變數的起使值。
2. 根據判斷條件的內容，檢查是否要繼續執行迴圈，當條件判斷為真(true)，繼續執行迴圈的主體；條件判斷為假(false)，則跳出迴圈執行其它敘述。
3. 執行完迴圈主體內的敘述後，迴圈控制變數會根據增減量的設定，更改迴圈控制變數的值，再回到步驟2重新判斷是否繼續執行迴圈。

26

補充

範例：1加到10計算功能

由上
往下
逐行
執行
重複
執行
加法
運算
10次

//範例 (OneToTen.java) : 1加到10計算功能

```

001 public class OneToTen {
002
003     public static void main(String[] args){
004
005         int i = 0 ;
006         i += 1 ;
007         i += 2 ;
008         i += 3 ;
009         i += 4 ;
010         i += 5 ;
011         i += 6 ;
012         i += 7 ;
013         i += 8 ;
014         i += 9 ;
015         i += 10 ;
016
017         System.out.println("1加到10 : " + i ) ;
018     }
019 }

```

i=1
i=3
i=4
i=10
i=15
i=21
i=28
i=36
i=45
i=55

1加到10 : 55

27

5.3 迴圈

for迴圈 (2/2)

- 下面的程式利用for迴圈計算1+2+...+10：

```

/* app5_3 OUTPUT--
1+2+...+10=55
-----*/

```

```

01 // app5_3,for迴圈
02 public class app5_3
03 {
04     public static void main(String args[])
05     {
06         int i,sum=0;
07
08         for(i=1;i<=10;i++)
09             sum+=i; // 計算sum=sum+i
10         System.out.println("1+2+...+10="+sum); // 印出結果
11     }
12 }

```

i	i<=10	sum=sum+i
1	y	1
2	y	1+2
3	y	1+2+3
4	y	1+2+3+4
...		
10	y	1+2+3+...+9+10
11	No	跳出迴圈

28

for迴圈裡的區域變數

- 迴圈裡宣告的變數是區域變數（local variable），跳出迴圈，這個變數便不能再使用
- for迴圈裡的區域變數使用範例：

```

01 // app5_4, 區域變數
02 public class app5_4
03 {
04     public static void main(String args[])
05     {
06         int sum=0;
07         for(int i=1;i<=5;i++) // 在迴圈內宣告變數i
08         {
09             sum=sum+i;
10             System.out.println("i="+i+", sum="+sum);
11         }
12     }
13 }
14 }

```

```

/* app5_4 OUTPUT---
i=1, sum=1
i=2, sum=3
i=3, sum=6
i=4, sum=10
i=5, sum=15
-----*/

```

29

課堂範例

```

public class app5_4_1 {
    public static void main(String[] argv) {
        int i;
        for(i=0;i<3;i++)
        {
            System.out.println("for內部i : " + i);
        }
        System.out.println("for跳出的i : " + i);
    }
}

```

```

for內部i : 0
for內部i : 1
for內部i : 2
for跳出的i : 3

```

30

5.3 課堂練習

- **ex5_3_1.java**
- 試利用for迴圈，計算 $1+3+5+\dots+n$ 的總和，其中 n 為101。
 - hint :
 - 初始值多少？
 - 判斷條件為何？
 - 增減量為何？

```
/* output-----
1+3+...+101=2601
-----*/
```

31

5.3 課堂練習

- **ex5_3_2.java**
- 試撰寫一程式，列出20的所有因數1、2、4、5、10、20，與共有幾個20的因數。
- 提示：
 - 初始值多少？
 - 判斷條件多少？
 - 增減量？
 - For迴圈裡面要做什麼？
 - 因數：可以被20整除的數字

```
/* output-----
20的因素分別為：1 2 4 5 10 20 ，共有6個
-----*/
```

32

5.3 迴圈

while 迴圈 (1/2)

先想想看For迴圈需要哪些條件？

while迴圈敘述格式

```

設定迴圈初值;
while(判斷條件)
{
    迴圈主體;
    設定增減量;
}
          
```

這兒不可以加分號

這兒不可以加分號

```

graph TD
    Start([ ]) --> Init[設定迴圈初值]
    Init --> Cond{判斷條件}
    Cond -- true --> Body[迴圈主體]
    Body --> Inc[設定增減量]
    Inc --> Cond
    Cond -- false --> Other[其它敘述]
    Other --> End([ ])
          
```

- 第一次進入while迴圈前，就必需設定迴圈控制變數的起使值。
- 根據判斷條件的內容，檢查是否要繼續執行迴圈，如果條件判斷值為true，則繼續執行迴圈主體；如果條件判斷為false，則跳出迴圈執行後續的敘述。
- 執行完迴圈主體內的敘述後重新設定(增加或減少)迴圈控制變數的值，由於while迴圈不會主動更改迴圈控制變數的內容，所以在while迴圈中，設定迴圈控制變數的工作要由我們自己來做，再回到步驟2重新判斷是否繼續執行迴圈。

33

5.3 迴圈

while 迴圈 (2/2)

- 利用while迴圈計算1+2+...+10：

```

01 // app5_5,while迴圈
02 public class app5_5
03 {
04     public static void main(String args[])
05     {
06         int i=1,sum=0;
07
08         while(i<=10)
09         {
10             sum+=i; // 累加計算
11             i++;
12         }
13         System.out.println("1+2+...+10="+sum); // 印出結果
14     }
15 }
          
```

在程式設計的慣例上，會在確定迴圈次數時選擇for迴圈，而在不確定迴圈次數時選擇while迴圈，這樣的做法能讓語意更清楚的表達

i=1	sum=0,	sum=sum+i=0+1=1
i=2	sum=1,	sum=sum+i=1+2=3
i=3	sum=3,	sum=sum+1=3+3=6
.....		
i=10	sum=?	?

/* app5_5 OUTPUT---
 1+2+...+10=55
 -----*/

34

5.3 迴圈

do while 迴圈 (1/2)

- do while 是用於迴圈執行的次數未知時
- do while 至少會執行1次迴圈主體

do while 迴圈敘述格式

```

設定迴圈初值;
do
{
    迴圈主體;
    設定增減量;
} while (判斷條件);
        
```

要加分號

35

補充

下面的例子是利用while迴圈計算1累加到10：

```

01 // app5_5,while迴圈
02 public class app5_5
03 {
04     public static void main(String args[])
05     {
06         int i=1,sum=0;
07
08         while(i<=10)
09         {
10             sum+=i; // 累加計算
11             i++;
12         }
13         System.out.println("1+2+...+10="+sum); // 印出結果
14     }
15 }
        
```

/* app5_5 OUTPUT---
 1+2+...+10=55
 -----*/

```

do{
    sum+=i;
    i++;
}while(i<=10);
        
```

36

5.3 迴圈

do while 迴圈 (2/2)

```

01 // app5_6, do while 迴圈
02 import java.util.Scanner;
03 public class app5_6
04 {
05     public static void main(String args[])
06     {
07         Scanner scn=new Scanner(System.in);
08         int n,i=1,sum=0;
09
10         do{
11             System.out.print("請輸入累加的最大值: ");
12             n=scn.nextInt();
13         }while (n<1); // 輸入 n, n 要大於等於 1, 否則會一直重複輸入
14
15         do
16             sum+=i++; // 計算 sum=sum+i, 然後 i 值再加 1
17         while (i<=n);
18
19         System.out.println("1+2+...+n="+sum); // 印出結果
20     }
21 }

```

/* app5_6 OUTPUT-----
 請輸入累加的最大值: -8
 請輸入累加的最大值: 10
 1+2+...+10=55
 -----*/

37

5.3 課堂練習

- ex5_3_3.java, ex5_3_4.java, ex5_3_5.java
- 試撰寫一程式，印出從1到200之間，所有可以被9整除，又可以被7整除的數值。
- 同時以for/while/do-while完成。

```

/* output-----
1~200中可同時被9與7整除的數有 63 126 189, 共有3個數字可被整除
-----*/

```

38

5.3 迴圈

巢狀迴圈 (nested loops)

- 迴圈敘述中又有其它迴圈敘述時，稱為巢狀迴圈
- 以列印部份的九九乘法表為例，練習巢狀迴圈：

```

01 // app5_7,巢狀for迴圈求9*9乘法表
02 public class app5_7
03 {
04     public static void main(String args[])
05     {
06         int i,j;
07
08         for (i=1;i<=3;i++)    // 外層迴圈
09         {
10             for (j=1;j<=3;j++) // 內層迴圈
11                 System.out.print(i+"*"+j+"="+i*j+"\t");
12             System.out.println();
13         }
14     }
15 }

```

```

/* app5_7 OUTPUT----
1*1=1  1*2=2  1*3=3
2*1=2  2*2=4  2*3=6
3*1=3  3*2=6  3*3=9
-----*/

```

39

5.3 課堂練習

- ex5_3_6.java
- 寫一個4*9的乘法表

```

1*1=1  1*2=2  1*3=3  1*4=4  1*5=5  1*6=6  1*7=7  1*8=8  1*9=9
2*1=2  2*2=4  2*3=6  2*4=8  2*5=10 2*6=12 2*7=14 2*8=16 2*9=18
3*1=3  3*2=6  3*3=9  3*4=12 3*5=15 3*6=18 3*7=21 3*8=24 3*9=27
4*1=4  4*2=8  4*3=12 4*4=16 4*5=20 4*6=24 4*7=28 4*8=32 4*9=36

```

40

講解

- 試利用for迴圈撰寫出一個能產生如下圖結果的程式。請先繪製出流程圖後，根據流程圖撰寫程式。

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

```
public class hw5_13
{
    public static void main(String args[])
    {
        int i,j;

        for(i=1;i<=5;i++)
        {
            for(j=1;j<=i;j++)
            {
                System.out.print('*');
                System.out.println();
            }
        }
    }
}
```

41

5.3 課堂練習

- ex5_3_7.java
- hint:以2個for loop來完成

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

42

```

public class ex5_3_8 {
    public static void main(String args[]) {
        int i=0;

        do {
            for (i=0;i<10;i++)
                while (i < 4)
                    System.out.print(i);

        } while (i < 10);
    }
}

```

結果: 0000.....無限個，那無限個0 如何終止？

43

```

public class ex5_3_9 {
    public static void main(String args[]) {
        int i=0;

        do {
            for (i=0;i<10;i++)
                while (i++ < 4)
                    System.out.print(i);

        } while (i < 10);
    }
}

```

結果: 1234

44

```
public class ex5_3_10 {  
    public static void main(String args[]) {  
        int i=0;  
  
        do {  
            for (i=0;i<10;i++)  
            {  
                while (++i < 4)  
                {  
                    System.out.print(i);  
                }  
            } while (i < 10);  
        }  
    }  
}
```

結果: 123

45

```
public class ex5_3_11 {  
    public static void main(String args[]) {  
        int i=0;  
  
        do {  
            for (i=0;i<10;i++)  
            {  
                while (++i < 4)  
                {  
                    System.out.print(i);  
                }  
                System.out.print ( " " );  
            } while (i < 10);  
        }  
    }  
}
```

結果: 12311

46

```

public class ex5_3_12 {
    public static void main(String args[]) {
        int i=0;

        do {
            for (i=0;i<10;i++)
                while (++i < 4)
                    System.out.print(i);
            System.out.print ( i );
        } while (i < 12);
    }
}

```

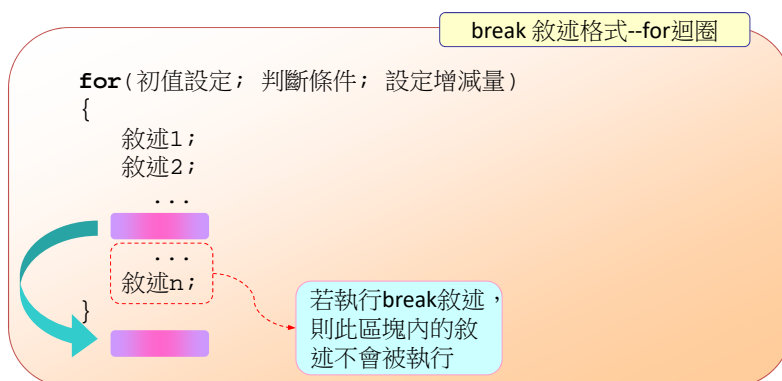
結果: 123111231112311.....無限個

47

5.4 迴圈的跳離

break敘述 (1/2)

- break敘述格式：



48

break敘述 (2/2)

- 在for迴圈中使用break敘述的範例：

```

01 // app5_8,break的使用
02 public class app5_8
03 {
04     public static void main(String args[])
05     {
06         int i;
07
08         for (i=1;i<=10;i++)
09         {
10             if(i%3==0) // 判斷i%3是否為0
11                 break;
12             System.out.println("i="+i); // 印出i的值
13         }
14         System.out.println("when loop interrupted,i="+i);
15     }
16 }

```

```

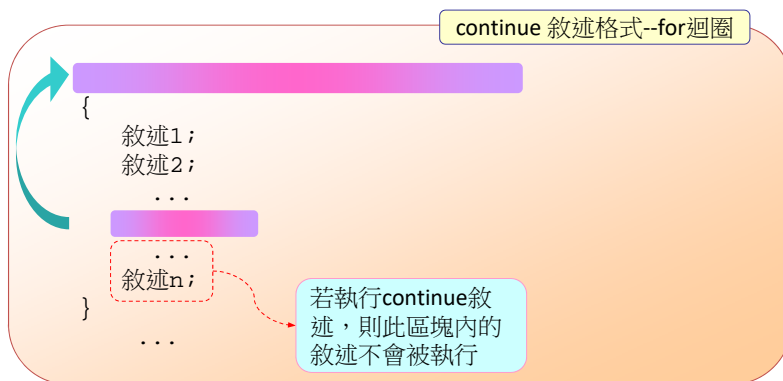
/* app5_8 OUTPUT-----
i=1
i=2
when loop interrupted,i=3
-----*/

```

49

continue敘述 (1/2)

- continue敘述會強迫程式跳到迴圈的起頭
- continue敘述的格式：



50

5.4 迴圈的跳離

continue敘述 (2/2)

- 使用continue敘述的範例：

```

01 // app5_9,continue的使用
02 public class app5_9
03 {
04     public static void main(String args[])
05     {
06         int i;
07
08         for (i=1;i<=10;i++)
09         {
10             if(i%3==0) // 判斷i%3是否為0
11                 continue;
12             System.out.println("i="+i); // 印出i的值
13         }
14         System.out.println("when loop interrupted,i="+i);
15     }
16 }

```

```

/* app5_9 OUTPUT-----
i=1
i=2
i=4
i=5
i=7
i=8
i=10
when loop interrupted,i=11
-----*/

```

51

5.4 課堂練習

- 課本習題

– ex5_4_1.java

- 請將下述程式改以do-while完成。

– ex5_4_2.java

- 請將下述程式改以break完成。

```

public class test5 {
    public static void main(String argv[]) {
        int day=1;
        float length=1500;
        for(length=1500; length>=6; day++)
        {
            length=length/2;
            System.out.println("第"+day+"天，長度："+length);
        }
        System.out.println("剪第"+(day-1)+"天短於6公尺");
    }
}

```

```

/* output-----
第1天，長度：750.0
第2天，長度：375.0
第3天，長度：187.5
第4天，長度：93.75
第5天，長度：46.875
第6天，長度：23.4375
第7天，長度：11.71875
第8天，長度：5.859375
剪第8天短於6公尺
-----*/

```

52

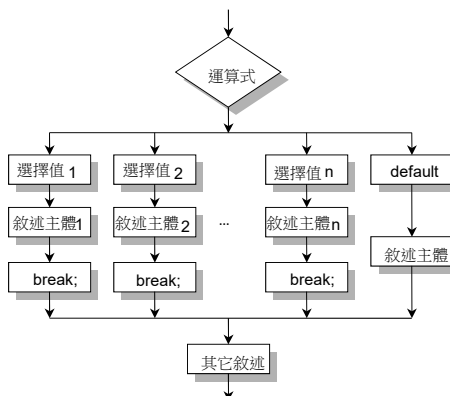
5.5 更好用的多重選擇--switch敘述

switch敘述 (1/2)

- switch敘述可將多選一的情況簡化，格式如下：

switch 敘述格式

```
switch(運算式)
{
    case 選擇值1:
        敘述主體1;
        break;
    case 選擇值2:
        敘述主體2;
        break;
    ...
    case 選擇值n:
        敘述主體n;
        break;
    default:
        敘述主體;
}
```



53

5.5 更好用的多重選擇--switch敘述

switch敘述 (2/2)

```
01 // app5_10,switch敘述
02 public class app5_10
03 {
04     public static void main(String args[])
05     {
06         int a=100,b=7;
07         char oper='/' ;
08
09         switch (oper)
10         {
11             case '+': // 印出a+b
12                 System.out.println(a+"+"+b+"="+ (a+b));
13                 break;
14             case '-': // 印出a-b
15                 System.out.println(a+"-"+b+"="+ (a-b));
16                 break;
17             case '*': // 印出a*b
18                 System.out.println(a+"*"+b+"="+ (a*b));
19                 break;
20             case '/': // 印出a/b
21                 System.out.println(a+"/"+b+"="+ ((float)a/b));
22                 break;
23             default: // 印出字串
24                 System.out.println("Unknown expression!!");
25         }
26     } }
```

```
/* app5_10 OUTPUT--
100/7=14.285714
-----*/
```

如果沒有在case敘述結尾處加上break，則會一直執行到switch敘述的尾端，才會離開switch敘述，如此將造成執行結果的錯誤

54

5.5 課堂練習

- ex5_5_1.java
- 試由程式中直接設定一個1~7之間的整數day，代表星期一到星期日。若day的值是1~5，則印出"今天要上班"，若day的值是6~7，則印出"今天休息"，若day的值不是1~7，則印出"輸入錯誤"。

```
請輸入1-7的整數:3  
今天要上班
```

```
請輸入1-7的整數:6  
今天休息
```

55

Ch5 總複習

- Sequence
- Selection
 - Switch case
 - If...else if...else
- Iteration
 - For
 - While
 - Do while

56

迴圈的樣式

```
int i,sum=0;
for(i=1; i<=5; i++)
{
    sum=sum+i;
}
```

```
int i=1,sum=0;
while(i<=5)
{
    sum=sum+i;
    i++;
}
```

```
int i=1,sum=0;
do
{
    sum=sum+i;
    i++;
} while(i<=5);
```

57

```
int a=15;
if (a%2==0)
    System.out.println(a+"even");
else
    System.out.println(a+"odd");
```

```
int a=8,b=3, max;
max=(a>b)?a:b;
System.out.println(max+"較大");
```

```
int i,sum=0;
for(i=1; i<=5; i++)
{
    sum=sum+i;
}
```

```
int i=1,sum=0;
while(i<=5)
{
    sum=sum+i;
    i++;
}
```

```
int i=1,sum=0;
do
{
    sum=sum+i;
    i++;
} while(i<=5);
```

58

回家作業

- hw5_1.java
- 試撰寫一程式，可由鍵盤輸入一個整數，然後判斷該數值是正數還是負數。
- 以條件運算式來撰寫

```
/* output-----
請輸入一個整數:-6
-6是負數
-----*/
```

59

回家作業

- hw5_2.java
- 試撰寫一程式，可由鍵盤輸入一個整數，然後判斷該數值是正數還是負數。
- 以if-else來撰寫

```
/* output-----
請輸入一個整數:-6
-6是負數
-----*/
```

60

回家作業

- hw5_3.java
- 撰寫一個程式，可以將十進制轉換成八進制。
- % /的運用

請輸入一個十進位的數字：**666**
表示成八進制為：**1232**

其他說明：若要一次印出多個整數數字，可用下述語法，較為簡便。
`System.out.format("表示成八進制為:%d %d %d %d",a,b,c,d)`

61

10進位→任何進位

//-----

輸入進位值：**8**

輸入轉換的數字：**666**

10進位：666 → 8進位

$666 \div 8 = 83 \text{ ---- } 2$

$83 \div 8 = 10 \text{ ---- } 3$

$10 \div 8 = 1 \text{ ---- } 2$

$1 \div 8 = 0 \text{ ---- } 1$

答案：1232

使用十進制的數據不斷除以8，直到商為0為止。從下往上取餘數就是對應的八進制。

<https://codertw.com/%E7%A8%BB%E5%BC%8F%E8%AA%9E%E8%A8%80/711310/>

10進位→任何進位

//-----

輸入進位值：**2**

輸入轉換的數字：**256**

10進位：256 → 2進位

$256 \div 2 = 128 \text{ ---- } 0$

$128 \div 2 = 64 \text{ ---- } 0$

$64 \div 2 = 32 \text{ ---- } 0$

$32 \div 2 = 16 \text{ ---- } 0$

$16 \div 2 = 8 \text{ ---- } 0$

$8 \div 2 = 4 \text{ ---- } 0$

$4 \div 2 = 2 \text{ ---- } 0$

$2 \div 2 = 1 \text{ ---- } 0$

$1 \div 2 = 0 \text{ ---- } 1$

答案：100000000

62

回家作業

- hw5_4.java
- 撰寫一個程式，可以將八進制轉換成十進制。
- 運算問題

請輸入一個00~77的八進位整數： 77
八進制 77 轉換成十進制= 63

說明：%s 印出字串 %d印出整數 %c印出字元

```
System.out.format("/八進制%s 轉換成十進制= %d",str,n1);
System.out.format("/八進制%c%c 轉換成十進制= %d",a[0],a[1],n1);
```

63

任何進位→10進位

//-----

輸入進位值：8

輸入轉換的數字：1234567

8進位：1234567 → 10進位

$$123456<7> \rightarrow 7 \times 8^0 = 7 \times 1 = 7$$

$$12345<6>7 \rightarrow 6 \times 8^1 = 6 \times 8 = 48$$

$$1234<5>67 \rightarrow 5 \times 8^2 = 5 \times 64 = 320$$

$$123<4>567 \rightarrow 4 \times 8^3 = 4 \times 512 = 2048$$

$$12<3>4567 \rightarrow 3 \times 8^4 = 3 \times 4096 = 12288$$

$$1<2>34567 \rightarrow 2 \times 8^5 = 2 \times 32768 = 65536$$

$$<1>234567 \rightarrow 1 \times 8^6 = 1 \times 262144 = 262144$$

$$7+48+320+2048+12288+65536+262144$$

$$\text{答案} = 342391$$

64

回家作業

- hw5_5.java
- 試寫一個程式, 輸入學生的成績, 成績在 90~100 分之間為 A ; 成績在 80~89 分為 B ; 範圍在 70~79 分為 C ; 而範圍落在 60~69 為 D ; 未滿 60 為 E (使用 if 條件運算式)。
- **If-else if.....-else**

請輸入成績：88
您的成績等第為：B

65

回家作業

- hw5_6.java
- 試寫一程式, 可計算出 起始數字 到 中止數字 間所有 3 的倍數之總和。
- **for**

請輸入兩個整數(起始數字與終止數字)：3 21
3-21間所有3的倍數的總和為：84

66

回家作業

- `hw5_7.java`
- 試寫一程式, 讓使用者輸入任意正整數 N , 並利用 `for` 迴圈在螢幕上輸出 $1 * 1$ 、 $2 * 2$ 、...、 $N * N$ 之結果。
- `for`

```
請輸入一整數：5
1*1=1
2*2=4
3*3=9
4*4=16
5*5=25
```

67

回家作業

- `hw5_8.java`
- 請撰寫一個程式, 接受使用者輸入一個字元, 如果字元為小寫英文字母就將之轉換為大寫英文字母後輸出; 如果字元為大寫英文字母, 就將之轉換為小寫英文字母後輸出; 如果輸入的並非英文字母, 則輸出 " 您所輸入的並不是英文字母"。
- `If-else ifelse`

```
請輸入一個英文字母:A
您輸入的第一個字元經大小寫轉換後為:a
```

```
請輸入一個英文字母:d
您輸入的第一個字元經大小寫轉換後為:D
```

```
請輸入一個英文字母:4
您所輸入的並不是英文字母
```

68

回家作業

- `hw5_9.java`
- 使用迴圈設計一個程式，找出1~40中，可以被3整除，但不能被5整除的所有正整數，輸出結果如下
- **for-if**

```
/* output-----
3 6 9 12 18 21 24 27 33 36 39
-----*/
```

69

回家作業

- `hw5_10.java`
- 試撰寫一程式，印出從1到100之間，所有可以被41 整除，或被50整除的數值。
- **for-if**

```
/* output-----
1~100 中可同時被41或50 整除的數有41 50 82 100
-----*/
```

70

回家作業

- **hw5_11.java**
- 輸入三個數字，並找出三個數字之間，兩兩相乘之後最小的兩個數字。例如：3,6,9，兩兩相乘之後，結果最小的數3,6，值為18。
- **請測試以下各種情況，以確保程式撰寫周全。
- **If-else if....else**

請輸入3個相異整數: 3 6 9
最小與次小的數為3 6, 兩數相乘結果為18

請輸入3個相異整數: 9 9 9
最小與次小的數為9 9, 兩數相乘結果為81

請輸入3個相異整數: 6 9 3
最小與次小的數為3 6, 兩數相乘結果為18

請輸入3個相異整數: 3 9 9
最小與次小的數為3 9, 兩數相乘結果為27

請輸入3個相異整數: 9 6 3
最小與次小的數為3 6, 兩數相乘結果為18

71

回家作業

- **hw5_12.java**
- 請以多個for迴圈，列印出以下星號。
- **for-for-for**

請輸入您想出現幾排的星號: 5

```

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

```

72

回家作業

- `hw5_13.java`
- 請輸入一個整數(0除外)，並計算這些整數相乘之後的結果。
- `for`

請輸入1個正整數: 2345
120

請輸入1個正整數: 232345
720

請輸入1個正整數: 3456789
181440

73

回家作業

- `hw5_14.java`
- 試寫一個程式, 用來計算三角型面積、矩形面積及梯形面積。選擇三角形時, 會要求輸入底及高, 選擇矩形時會要求使用者輸入長與寬, 選擇梯形時, 則要求使用者輸入上底、下底、高。
- `switch-case`

請選擇(1)三角形(2)矩形(3)梯形
→1
請輸入(下)底長
→5
請輸入高
→4
三角形面積為: 10.0

請選擇(1)三角形(2)矩形(3)梯形
→2
請輸入(下)底長
→4
請輸入高
→5
矩形面積為: 20

請選擇(1)三角形(2)矩形(3)梯形
→3
請輸入(下)底長
→7
請輸入高
→4
請輸入上底長
→5
梯形面積為: 24.0

請選擇(1)三角形(2)矩形(3)梯形
→5
選錯了

74

回家作業

- hw5_15.java
- 試寫一程式, 讓使用者輸入兩個整數, 並計算兩整數間所有整數的和。
- **for**

請輸入兩個整數
第一個整數：3
第二個整數：8
3到8間的整數和為33

請輸入兩個整數
第一個整數：9
第二個整數：2
2到9間的整數和為44

75

回家作業

- hw5_16.java
- 試寫一程式, 可以繪製出如下的圖形：
- **for-for-for**

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

76

回家作業

- hw5_17.java
- 輸入一個四位數密碼，如果輸入錯誤的話，請重新輸入，錯誤三次終止程式。
- While

```

請輸入一個四位數密碼：
123
密碼錯誤，請從新輸入。
請輸入一個四位數密碼：
4564
密碼錯誤，請從新輸入。
請輸入一個四位數密碼：
1235
登入失敗，終止程式。

```

```

請輸入一個四位數密碼：
1212
密碼錯誤，請從新輸入。
請輸入一個四位數密碼：
2323
密碼錯誤，請從新輸入。
請輸入一個四位數密碼：
1234
登入成功，終止程式。

```

77

回家作業

- hw5_18.java
- 使用for 迴圈設計一個(0~9 * 0~9) 的偶數乘法表。執行結果如下（請特別注意乘數與被乘數的排列方式）。
- for-for

0*0=0	2*0=0	4*0=0	6*0=0	8*0=0
0*2=0	2*2=4	4*2=8	6*2=12	8*2=16
0*4=0	2*4=8	4*4=16	6*4=24	8*4=32
0*6=0	2*6=12	4*6=24	6*6=36	8*6=48
0*8=0	2*8=16	4*8=32	6*8=48	8*8=64

78

回家作業

- hw5_19.java
- 使用迴圈計算 $1^1 + 2^2 + 3^3 + \dots + n^n$ 的值（ n 由使用者輸入， n 為個位數的正整數）（不得使用公式，也不得使用 `java.lang.Math` 數學類別）。執行結果如下。
- n 介於 1-10，超出範圍則重新輸入。
- **do-while for-for**

n=6
Sum=50069

n=15
n=0
n=5
Sum=3413

79

回家作業

- hw5_20.java
- 撰寫一程式，有三個 選項，選擇出要顯示出的圖案，有矩形，正三角形，倒三角形，菱形

請輸入選擇(1)矩形(2)正三角形(3)倒三角形(4)菱形(請輸入奇數)：1
輸入寬為=>2
輸入高為=>3

請輸入選擇(1)矩形(2)正三角形(3)倒三角形(4)菱形(請輸入奇數)：2
請輸入高度：3
*
**

請輸入選擇(1)矩形(2)正三角形(3)倒三角形(4)菱形(請輸入奇數)：3
請輸入高度:3

**
*

請輸入選擇(1)矩形(2)正三角形(3)倒三角形(4)菱形(請輸入奇數)：4
高為:3
*
**

*

80

$$1^1 + 2^2 + 3^3 + \dots + n^n$$

```
for(i=1;i<=3 ; i++)//這裡決定n,假設n=3
{
    powerS = 1;
    for(j=1;j<=i ; j++)//這裡決定n被乘幾次
    {
        powerS = powerS * i; //第1次(i=1) 1*1,第二次(i=2)1*2,2*2
                               第3次(i=3) 1*3, 3*3, 3*3*3
    }
    Sum = Sum + powerS;//把所有乘過的n加總起來
}
System.out.println("Sum=" + Sum);
}
```

81