實作輔導

- 日期: 3/11 09:10~16:00
- 地點:臺北市立大學　臺北市中正區愛國西路一號 (中正紀念堂站7號出口)
- 公誠樓三樓 G315 電腦教室(資訊科學系(含碩士班))
- 可自行攜帶筆電
- 目標:協助習題、安裝java 環境、path設定

- 下次預定: 3/24（星期六）、4/14（星期六）09:10~16:00

搶答I

1. 負責編譯(compile)Java程式為?
2. 你所編寫Java的來源程式副檔名?

搶答I

1. 負責編譯(compile)Java程式為? Javac.exe
2. 你所編寫Java的來源程式副檔名? java

搶答II

1. 若a1.java編譯成功，會產生的檔案為何?
2. 執行Java編譯成功檔案的程式名稱?
3. 如何執行上述1編譯成功之檔案?
4. Java的主程式名稱?

搶答II

1. 若a1.java編譯成功，會產生的檔案為何? a1.class
2. 執行Java編譯成功檔案的程式名稱? Java.exe
3. 如何執行上述1編譯成功之檔案? Java a1
4. Java的主程式名稱? main

輸出、輸入、變數
循序敘述

臺北市立大學 資訊科學系(含碩士班)
賴阿福

複習第1支程式

學習程式之觀念

- 漸進式學習
- From unclear to clear
- 不用全部學完語法再開始程式設計
  - 不可能一步登天
- 從基本功能進化、優化
- 養成規劃、設計習慣
- 不要怕出現Bug
- 思考、動手寫程式

第1支程式回顧

類別名稱，Class name，一定與主檔案名稱相同

主程式名稱(固定)

```
import java.util.*; //匯入類別/函數庫
public class welcomea{
public static void main(String args[]){
  System.out.println("Java is an excellent computer language.");//輸出
  System.out.println("Let's start to learn Java!");
 }//main()
}//class
```

主程式之引數

敘述以分號結束

println之引數, 現為字串常數

System.out: 標準輸出設備，即螢幕
println:輸出/顯示之函數/方法，ln: Line，即顯示後游標跳到下行(換行)

註解

輸出：換行(二種方式) 或 不換行

- System.out.println(“Drink Java coffee!”);//換行
- System.out.println(“excellent .”);//不換行

相同

- System.out.print (“Drink Java coffee\n”); //不換行變換行
- System.out.println(“excellent .”);

\n :Escape 字元，new line，即換行

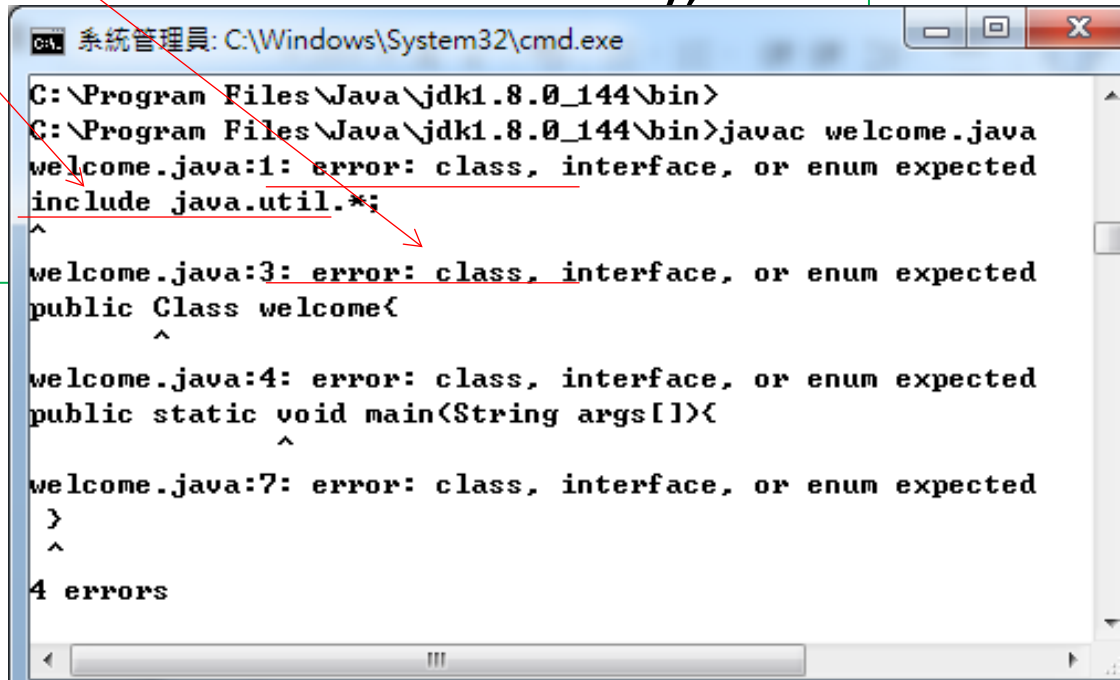
\n\n ：換行二次

程式出現錯誤，如何是好?
Error(錯誤)! Bug(臭蟲)!
Debug(抓臭蟲)

Error! Bug! debug

```
include java.util.*;


public Class welcome{
public static void main(String args[]){
  System.out.println("Java is an excellent computer language.")
  System.out.println("Let's start to learn Java!");
 }
}//class
```



- include: import
- Class: class
- ;

debug(偵錯)

在二戰結束後，霍普（Grace Murray Hopper，1906-1992）在哈佛大學與同事開始建造第二代的電腦（Mark II）。夏日的某一天，Mark II忽然不動了。霍普打開電腦檢查，看到一隻蛾卡在裡面；於是就拿了一把鑷子把蟲給夾出來。從此電腦的除錯就叫做debug了。

參考文獻：Darlene R. Stille, Extraordinary women scientists.

何謂偵(抓)錯(debug)?語法(syntax)、語意(semantic)錯誤?

- **語法(syntax)錯誤?**
  - 指的是句子文法上的錯誤
    - 我是學生 → 我學生是
  - **Compiler**可抓到語法錯誤
- **語意(semantic)錯誤?**
  - 指的是句子意思上的錯誤
  - 求矩形面積
    - **Area=(Length+width)*2;**
  - **Compiler**抓不到語意錯誤

Debug抓錯蟲

```
import java.util.*; /匯入函數庫
public class 9welcomea{
public static void Main(String args[]){
  system.Out.println(“Java is an excellent computer language.);/輸出
  system.Out.println(‘Let's start to learn Java!’)
 }/main()
}/class
```

輸出、輸入敘述
(input & output)

輸出目的?可輸出何種資料?

輸出目的?

- 傳送訊息(如回饋、分析結果)給操作者
- 提示訊息: 輸入資料時

輸出裝置?輸出到電腦哪裡?

- 螢幕(CRT、LCD)、喇叭、

可輸出資料類型:

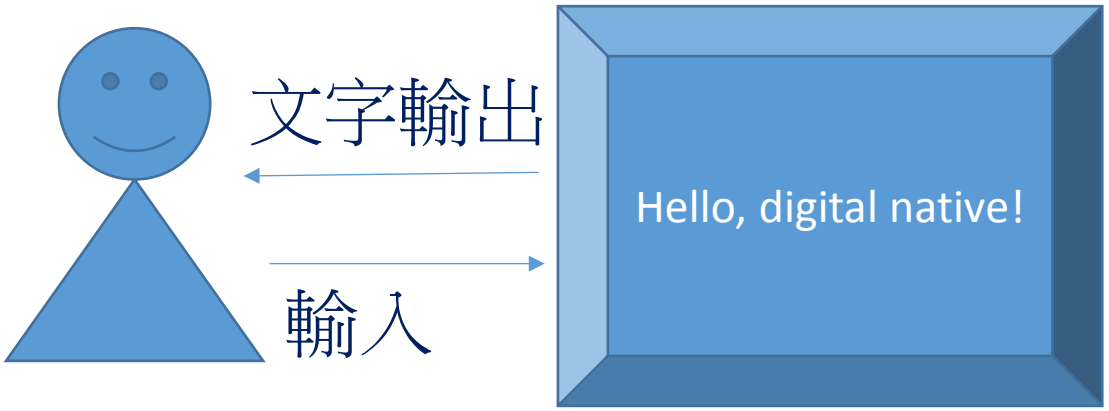
文字介面:

- 文字:字串(string)、數值(numerical data)
  - System.out.println("Let's start to learn Java!");

圖形介面:

- 文字:字串(string)、數值(numerical data)
  - label物件
- 聲音
- 圖形
- 動畫
- 震動

不能只有輸出，也要有輸入
:互動(interaction)
第2支程式



輸出裝置?

從何裝置輸入?keyboard、mouse、觸控平板、joystick、microphone

從keyboard裝置可輸入何種資料?

可輸入資料類型:

- 文字:字串(string)
- 數值(numerical data)
  - 整數: integer
  - 小數:浮動點 (floating point)
    - 單精度float
    - 倍精度Double

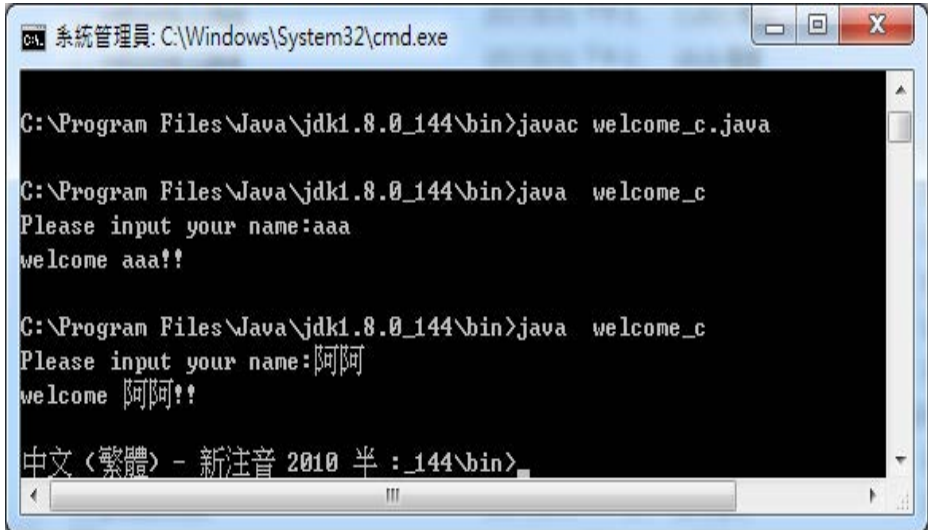
輸入後放電腦哪裡?

輸入後放電腦哪裡?

- 輸入目的?
  - 要處理、分析、計算、判斷
- 放電腦哪裡??
  - 主記憶體:RAM
    - 如同人的腦部
    - 容量??:可能有8G(=2^{30})byte(位元組)
    - 位元組:1位元組可存放一個英文字
    - 主記憶體以位址管理，非常複雜
    - COMPILER為便於程式設計師管理，因此以英文名稱取代主記憶體位址，即變數名稱
  - 程式設計師如何取用處理?
    - 利用輸入指令/敘述/物件
    - 輸入後放在主記憶體:放在指定變數 (程式設計師不用管真正位址)

Second program :鍵盤輸入

```
import java.util.*;
public class welcome_c{
public static void main(String args[]){
  System.out.print("Please input your name:");
  Scanner sca = new Scanner(System.in);//產生Scanner物件, 提供鍵盤輸入
  String name = sca.next(); //運用next方法/函數讀取輸入資料，再放入name變數
  System.out.println("welcome "+name+"!!"); //字串連結
 }//main()
}//class
```



Second program:程式概念

Scanner sca = new Scanner(System.in);

- Scanner :類別 (class)
  - Scanner原意為掃描器，代表輸入
- new Scanner(): 產生Scanner物件, 可提供鍵盤輸入
- System.in: 標準輸入設備，即鍵盤
- sca: new之後，sca是Scanner物件(object)， sca是物件名稱

String name = sca.next(); //sca物件的(.)next方法或nextLine

- 運用next方法/函數讀取輸入資料，再放入name變數
- =是指定運算符號，將右邊處理或運算結果指定給左邊變數 (next方法/函數讀取資料，放入name變數)
- String 是一種變數型態，即字串變數，用以存放文字資料

System.out.println("welcome "+name+"!!");

- +: 字串連結，字串常數與字串變數連結
- “12”+”34”字串連結變成”1234” ，不是”46”

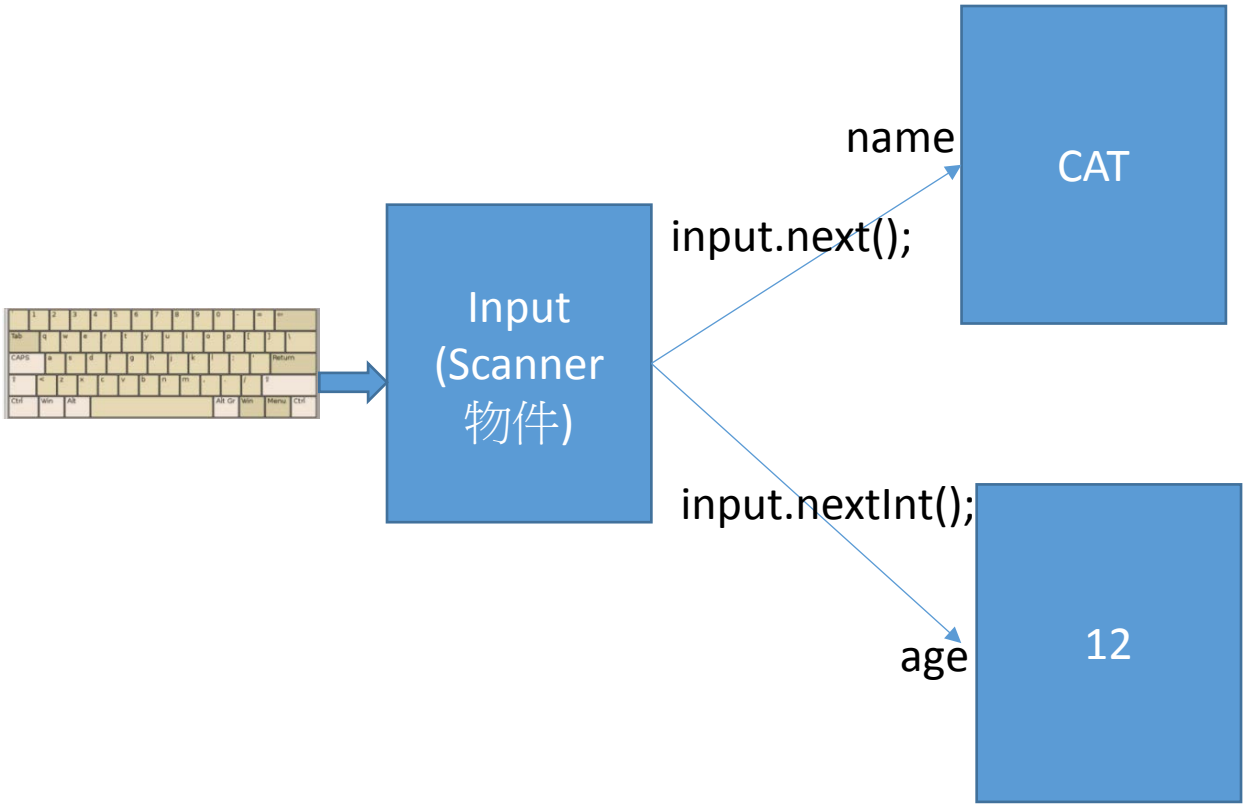
- int result=12+34; //+:算數運算，加法結果放入result數

比較: 字串連結、加法運算

- “12”+”34”
  - 字串連結(concatenation, combining)
  - 結果變成”1234” ，不是”46”
- 12+34
  - +:算數運算
  - 加法結果46
  - int result=12+34;
    - 加法結果放入result數

多輸入

```
import java.util.*;
public class welcome_D{
public static void main(String args[]){
  Scanner input = new Scanner(System.in);//產生Scanner物件
  System.out.print("Please input your name:");
  String name = input.next();
  System.out.print("Please input your age:");
  int age = input.nextInt();
  System.out.println("Nice to meet you, "+name+"!!");
  System.out.print("\n你的年齡: "+age+5+"歲.");
  age=age+5;//指定、指派、set敘述
  System.out.print("\n五年後你的年齡: "+age+"歲.");
 }//main()
}//class
```


name
CAT
input.next();
Input
(Scanner
物件)
input.nextInt();
age
12

變數 (variable)
記憶體代言人

變數(Variable)?

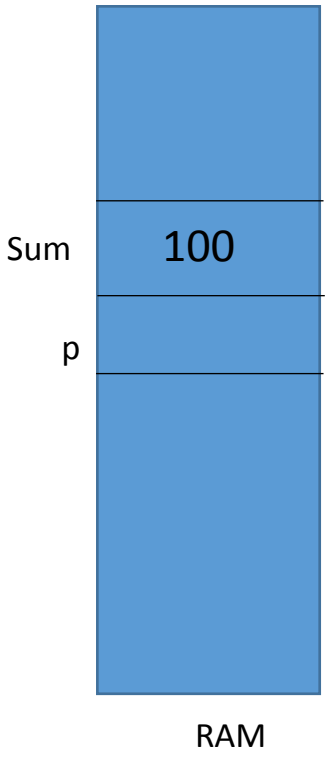
- 為了提供還不錯的、有用的功能，程式必須能記得住事情。
- 想想看，如果你不能記得任何事情你要如何過生活。
  - 如果你要把兩個數字相加，你必須記得這兩個數字才能相加，然後你必須記得結果才能在處理其他事情時使用它。
- 電腦用變數來記憶資料，他們之所以叫做變數是因為他們的值是可變的。
  - 你可以把變數想像成有不同名稱在上面的信箱：信箱裡面裝的東西叫做變數的內容。
- 變數的名稱必須遵守一些規則。
  - 首先他們的第一個字必須是英文字母A到Z，然後可以接著字母或數字的任意組合。
  - 可以結合大寫和小寫字母。
  - 要小心的是電腦對待大小寫字母的看法比你更仔細，如java及所有C-like語言把TIME、time、Time當作不同的變數。
- 變數的型態(data type)
  - String, int, double…
  - 為何需要不同型的態變數

為何變數要宣告型態的理由:不同型態的變數

- 人的腦袋非常靈活可變，可處理各種不同型式的資料，看到「在9:00見面」、「32」或「2.5」就知道該如何處理它。
- 電腦不像人那樣精明，它要正確地知道該處理什麼型式的資料。這就是為何變數要宣告型態的理由。

變數 (variable)

- 變數代表一塊記憶體
  - 每一塊記憶體都有位址，但不可能記得起來，因此用英文字來替代記憶體
- 如何將資料放入到記憶體?
  - 指定/派敘述(assignment statement)
    - sum=100;
    - sum=p+10;
    - sum=sum+70; //取出sum，加70，再放入sum
- 如何從記憶體取出資料
  - printf(“%d”, sum); //取出sum
  - System.out.println(sum);//取出sum
- 如何為記憶體取名?
  - 分大小寫
  - 英文開頭
  - 有意義名字



變數的名稱不合法invalid?合法valid?

Date
DATE
date
TimeOfDay
yr1991
AbCdEfGhiJ
i2
Stats_1981
Total_Cost
even_
1981stats
The%Given
The+Given
Total Cost
*
A+b-c

不合法

合法

不同變數的名稱：

下面是合法的：

Date

DATE

date

TimeOfDay

yr1991

AbCdEfGhiJ

i2

Stats_1981

Total_Cost

even_

下面是不合法的：

1981stats	(開始不可以是數字)
The%Given	(中間不可有%)
The+Given	(中間不可有+)
Total Cost	(中間不可有空隔)
*	(開始不可以是*符號)
A+b-c	(中間不可有+)

數值型態numeric data type

- 整數integer:
  - 正負
  - Byte, short, int, long
- 浮動點floating point : 小數點，採用IEEE 754格式
- 不同型態可表示數值範圍不同，占用RAM空間不同
- 型態名稱都是keyword /reserved word

	型態名稱	值域(Range)	貯存空間
整數	byte	-2^7 to 2^7 (–128 to 127)	8-bit signed
	short	-2^{15} to 2^{15} (–32768 to 32767)	16-bit signed
	int	-2^{31} to 2^{31} (–2147483648 to 2147483647)	32-bit signed
	long	-2^{63} to 2^{63} (–9223372036854775808 to 9223372036854775807)	64-bit signed
浮動點	float	Negative range:–3.4028235E+38 to –1.4E – 45 Positive range: 1.4E – 45 to 3.4028235E + 38	32-bit IEEE 754
	double	Negative range:–1.7976931348623157E+308 to –4.9E–324 Positive range: 4.9E–324 to 1.7976931348623157E+308	64-bit IEEE 754

計算BMI

BMI說明

- 身體質量指數 Body Mass Index，簡稱BMI
  世界衛生組織建議以BMI來衡量肥胖程度

成人肥胖定義	身體質量指數(BMI)(kg/m^2)
體重過輕	BMI<18.5
健康體位	18.5<=BMI<24
體位異常	過重：24<=BMI<27 輕度肥胖：27 <= BMI < 30 中度肥胖：30 <= BMI < 35 重度肥胖：BMI >= 35

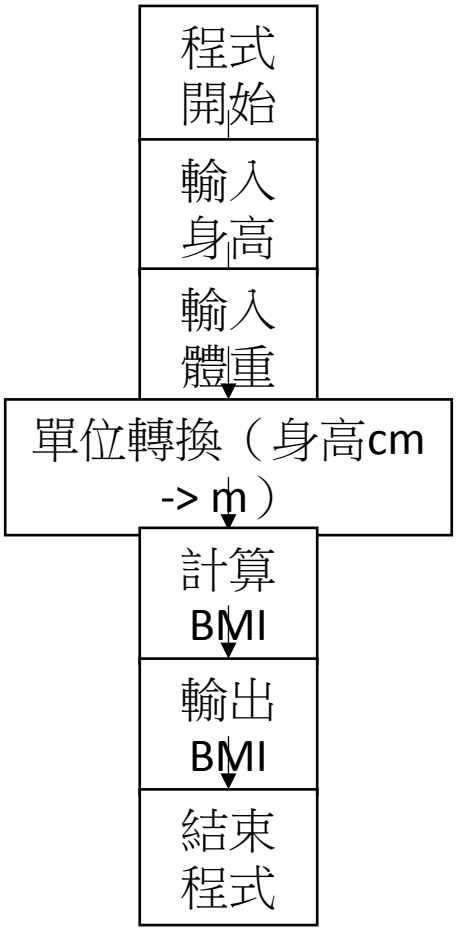
※ $\text{BMI}=\frac{\text{體重(公斤)}}{\text{身高}^2\text{(公尺}^2\text{)}}$

- 出處：衛生福利部國民健康署 健康九九網站
  http://health99.hpa.gov.tw/OnlinkHealth/Onlink_BMI.aspx

BMI程式一: 只計算BMI

- 解析: 了解問題、規劃、程式設計
  - (1) 讓使用者輸入身高、體重；先取二者變數名稱
  - (2) 計算BMI（算術運算式）:結果之變數名稱及型態、指定敘述(assignment statement)
  - (3) 輸出BMI
  - 更熟悉java程式結構

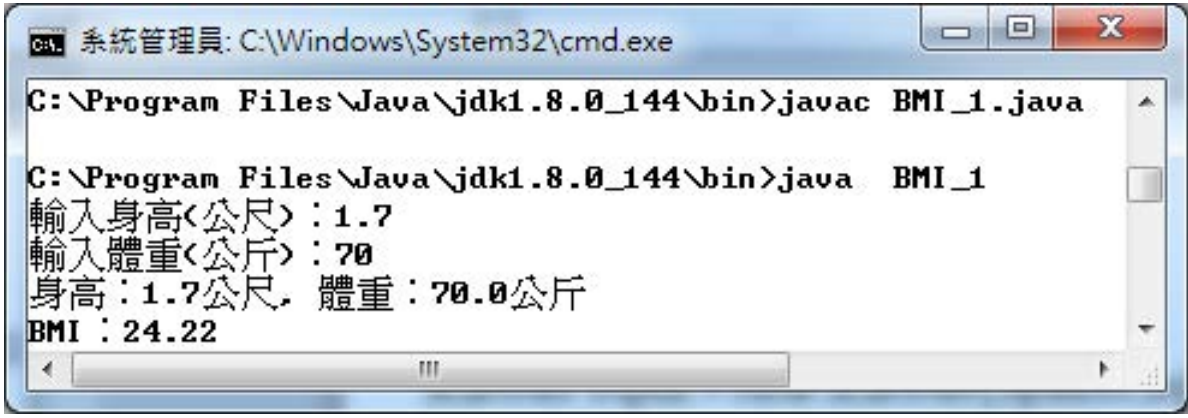
BMI程式—流程圖 (flow chart)



循序(sequence)

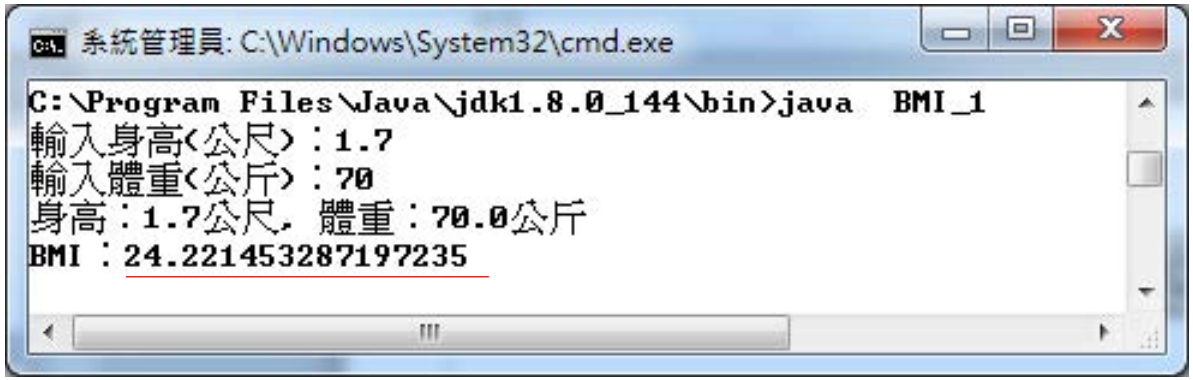
只計算BMI

```
import java.util.Scanner;
public class BMI_1 {
 public static void main(String[] args) {
   Scanner input = new Scanner(System.in);
   System.out.print(“輸入身高(公尺)：”);//提示訊息
   double height = input.nextDouble();//讀取浮動點資料(double)
   System.out.print("輸入體重(公斤)：");
   double weight = input.nextDouble();
   double bmi = Math.round((weight/ (height*height) )* 100) / 100.00;
   System.out.println("身高："+height+"公尺, 體重："+weight+"公斤");
   System.out.println("BMI："+bmi);
          }//end of main
}//end of class
```

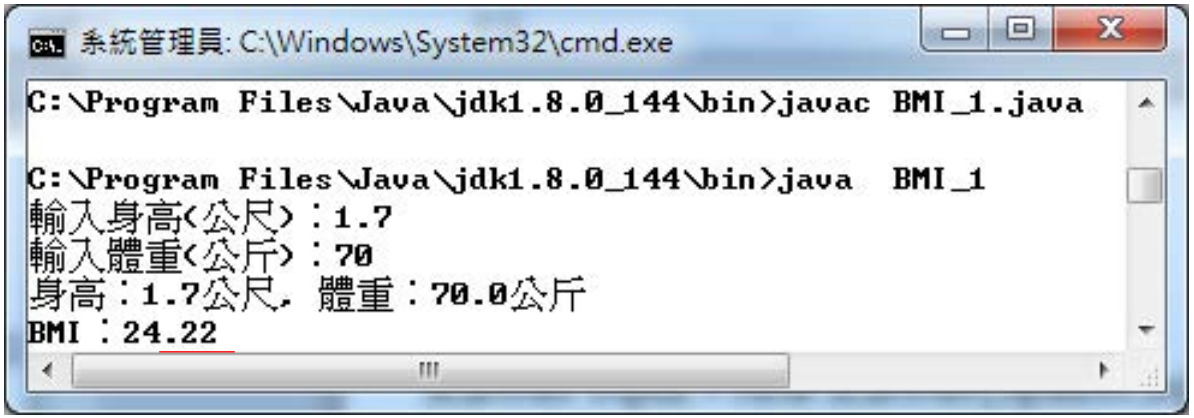


浮動點之小數位數

- double bmi = weight/ (height*height);



- double bmi = Math.round((weight/ (height*height))* 100) / 100.00;



輸入整數資料及整數變數

int nn = ipt.nextInt();

- 輸入整數資料
- 運用nextInt方法讀取使用者所輸入整數資料，再放入nn整數變數
- int :是一種變數型態，用以存放整數資料(integer)

int n1=nn%10;

- 求得除以10之餘數

int n2=nn/10;

- 求得除以10之商

習題

- 習題1:輸入三位整數資料，進行位數翻轉且顯示結果
  - 如輸入789，顯示翻轉結果為987
  - 先假設一定會輸入三位整數
  - 如何解決? 請規畫其過程 (解法)
  - 熟悉運算式及其應用
- 習題2:輸入矩形長、寬(整數)資料，求面積及邊長
  - 如何解決? 請規畫其過程 (解法)

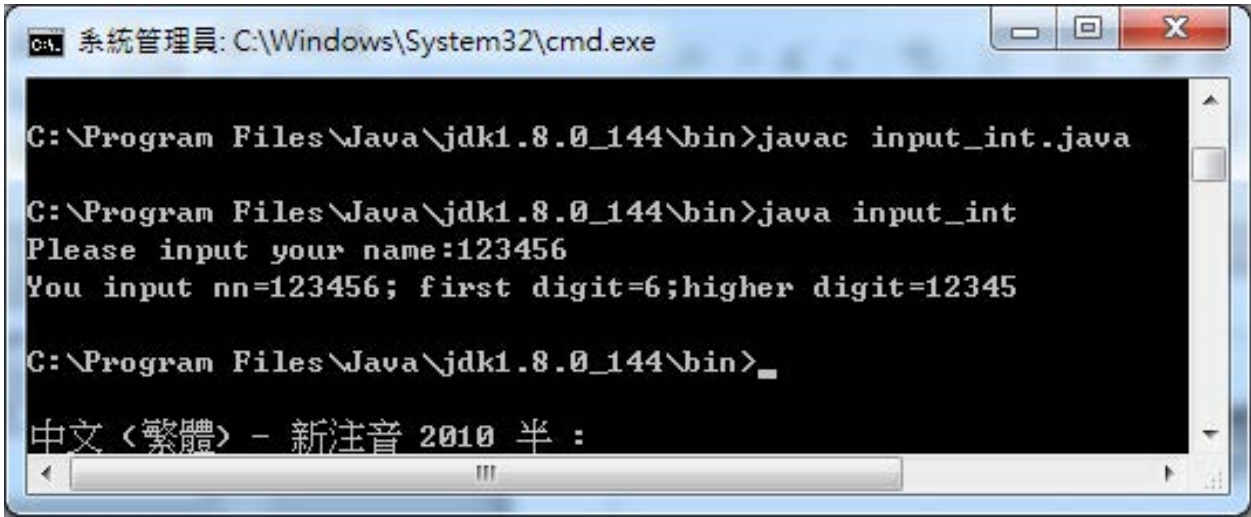
未完成之程式

```
import java.util.*;

public class input_int{
public static void main(String args[]){
  System.out.print("Please input a 3-digit decimal :");
  Scanner ipt = new Scanner(System.in);//產生Scanner物件
  int nn = ipt.nextInt(); //讀取整數(integer)資料
  int n1=nn%10;  //求餘數
  int n2=nn/10; //除法
  System.out.print("You input nn="+nn+"; first digit="+n1+";");
  System.out.println("higher digit="+n2);

 }//main()
}//class
```

```
System.out.print("Please input a 3-digit decimal:");
 Scanner ipt = new Scanner(System.in);
 int nn = ipt.nextInt();
 int n1=nn%10;
 int n2=nn/10;
 System.out.print("You input nn="+nn+"; first digit="+n1+";");
 System.out.println("higher digit="+n2);
```



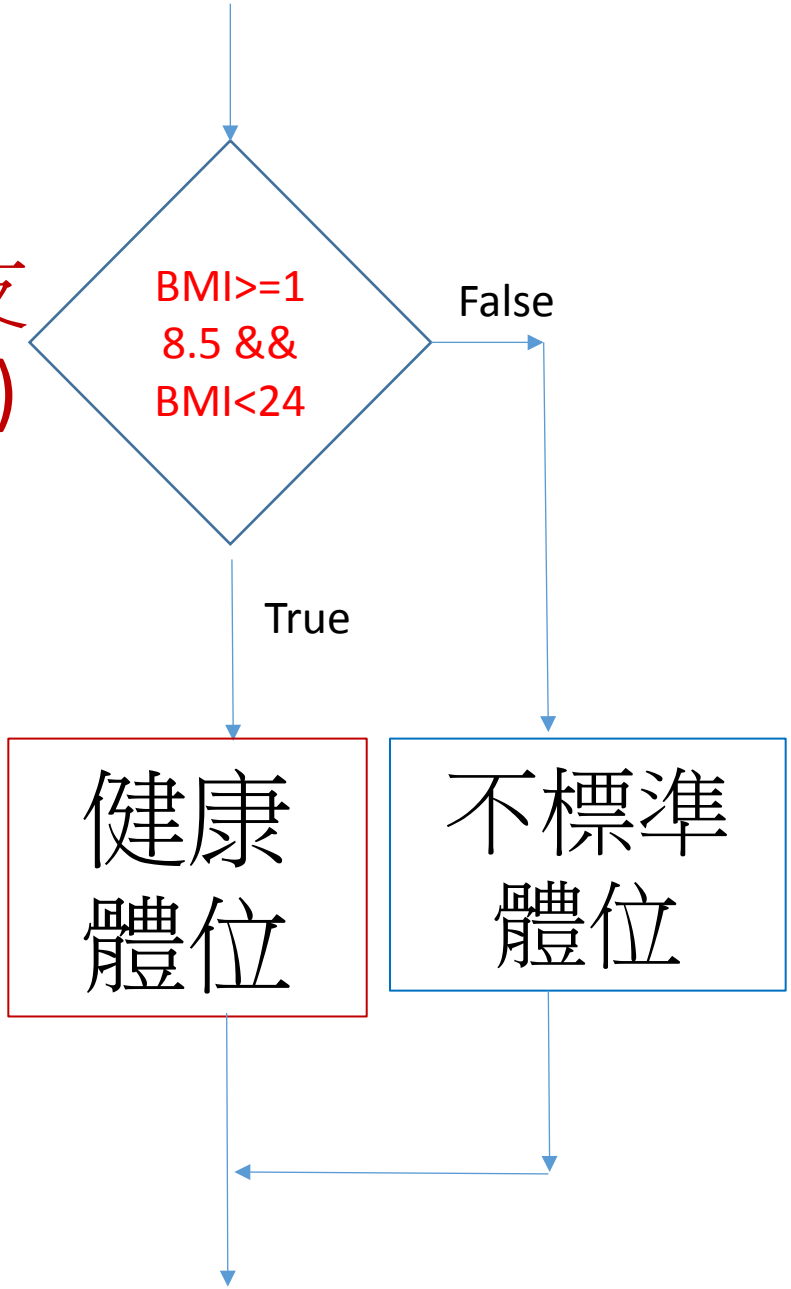
為何需要類別/ 函式庫(Library)/package?

- 站在巨人肩膀
- 有些程式細節非常複雜如**Input/Output**、圖形之細部工作,因此必須會提供系統函數，讓設計者直接引用
- import java.util.*;
  - *代表萬用(全部)Wildcard import
  - Scanner class就在java.util package
  - import java.util.Scanner;
    - Specific import
- **c語言用include**

從循序到分支
程式會轉彎

判斷體位是否標準?

選擇/分支 (selection)




```
if (BMI>=18.5 && BMI<24)
   System.out.print("健康體位 Normal\n\n");
else
   System.out.print("不標準體位\n\n");
```

//source file: BMI-2a

BMI診斷分成二層次

```
import java.util.Scanner;
public class BMI_2 {
public static void main(String[] args) {
    Scanner input = new Scanner(System.in);
    System.out.print("輸入身高：");
    double height = input.nextDouble();
    System.out.print("輸入體重：");
    double weight = input.nextDouble();
    double bmi = Math.round((weight/ (height*height) )* 100) / 100.0;
    String status;
     if  (bmi>=18.5 && bmi < 24)
          status = "正常Normal";
     else
          status = "不標準體位";
   System.out.println("BMI："+bmi+"，狀態: "+status);
  }//main
}//class
```

將BMI診斷分成三層次(簡化)

```
import java.util.Scanner;
public class BMI_2 {
public static void main(String[] args) {
    Scanner input = new Scanner(System.in);
    System.out.print("輸入身高：");
    double height = input.nextDouble();
    System.out.print("輸入體重：");
    double weight = input.nextDouble();
    double bmi = Math.round((weight/ (height*height) )* 100) / 100.0;
    String status;

    if (bmi < 18.5) {
        status = "體重過輕Underweight";
        }
    else if (bmi < 24) {//(bmi>=18.5 && bmi < 24)
        status = "正常Normal";
         }
    else
        status = "過重Overweight";

   System.out.println("BMI："+bmi+"，狀態: "+status);
  }//main
}//class
```

成人肥胖定義	身體質量
體重過輕	BMI<18.5
健康體位	18.5<=BMI<24
體位異常	過重：24<=BMI<27 輕度肥胖：27 <= BMI < 30 中度肥胖：30 <= BMI < 35 重度肥胖：BMI >= 35

未輸入三位整數時: 從循序到分支

```
import java.util.*;
public class input_int_a{
public static void main(String args[]){
  System.out.print("Please input 3-digit decimal:");
  Scanner ipt = new Scanner(System.in);//產生Scanner物件
  int nn = ipt.nextInt();
  if (nn>=100 && nn<=999)
    { //compound statements
     int n1=nn%10;
     int n2=nn/10;
     System.out.print("You input nn="+nn+"; first digit="+n1+";");
     System.out.println("higher digit="+n2);
    }
   else
     System.out.println(“你輸入資料不是三位數!!”); //單一敘述single
statement
 }//main()
}//class
```

(nn>=100 || nn<=999)

複合敘述compound statements

條件:邏輯運算式 (立即搶答)

if (nn>=100 && nn<=999)

- && : and (及)
- 數值範圍?

If (nn>=100 || nn<=999)

- || : or (或)
- 數值範圍?

If (nn<100 || nn>999)

- 數值範圍?

If (nn<100 && nn>999)

- 數值範圍?

if !(nn>=100 && nn<=999)

- ! : not (反)
- 數值範圍?

作業

- 將BMI診斷分成六層次

再談指定/派敘述 = (assignment statement)

- 何者正確?

  X+1=y-2;

  If (X+1=y-2) .....

  If (X+1==y-2) .....

- 何者正確?

  X=y;

  If (X=y) ....

  If (X==y) ...