



JAVA 開發環境與程式語言 程式設計就業專班

修平科技大學 資訊網路技術系
陳文敬老師 jameschen@hust.edu.tw

Agenda

1. 開發環境介紹與安裝
2. 介紹 Java 程式基本架構與語法
3. 體驗開發工具 – Eclipse
4. 體驗除錯機制、設定中斷點、觀察記憶體
5. 變數與記憶體、觀察記憶體的變化
6. 運算子與運算式
7. 輸出與輸入

主要開發工具

- Java SE SDK

- <http://www.oracle.com/technetwork/java/javase/overview/index.html>

1. SDK – 8u92

2. DOCs

3. Demos/Samples and Tutorials ...

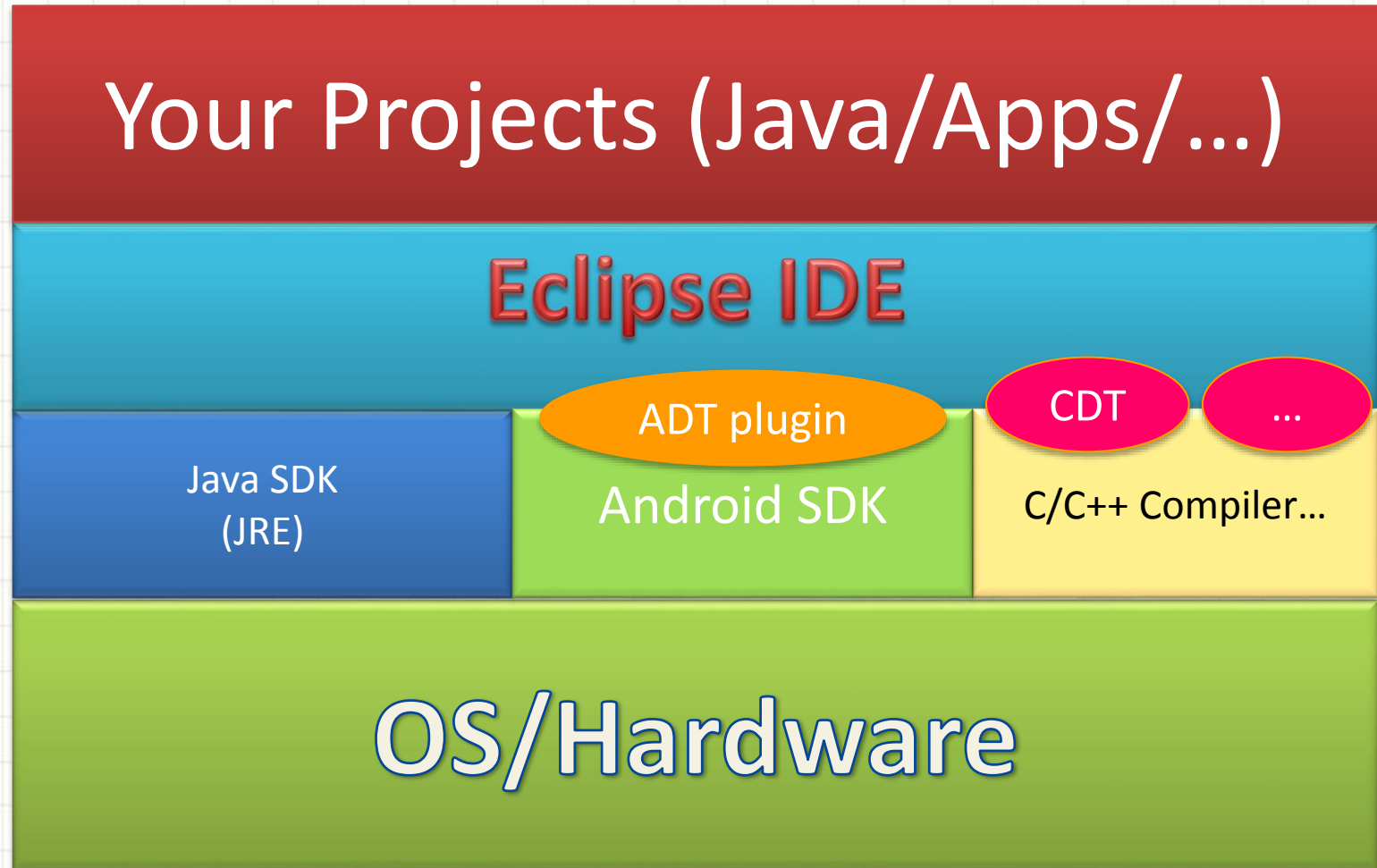
- 參考文件:

- http://docs.oracle.com/javase/8/docs/technotes/guides/install/install_overview.html

- 整合開發工具(IDE) : Eclipse

- <https://www.eclipse.org/downloads/eclipse-packages/>

Java SDK + **Eclipse** + ...



注意：目前專班的開發環境是 VM

- 專班課程需要安裝許多軟體，因此就利用 VM (虛擬機器)的方式先安裝好開發環境的VM，讓大家可以帶著走，回家可以方便練習；
 - JDK+Eclipse; Microsoft Visual Studio, SQL Server ...
- 主要步驟：
 - 安裝 VMware Player 14.1.1
 - 下載 VM Image (20GB → 解壓縮後 > 50GB)
 - 啟動 VMWare Player, 掛載開發環境的VM
 - 執行

開發環境安與檢查 - 1

(A) 安裝 Java SDK:

1. Windows X - 64 bits: jdk-XuYY-indows-**x64**.exe
2. Windows X - 32 bits : jdk-XuYY-windows-**i586**.exe
3. MAC, Linux : 參考 Oracle 官網說明

(B) 安裝 Eclipse (for Java) :

OS (Windows/Mac)?

32/64 bits?

開發環境安裝與檢查- 2

(C) 設定環境變數 (for Windows) : PATH, CLASSPATH

1. 開啟[控制台] -->[系統]-->[進階]-->[環境變數]，
增加 [系統]環境變數 宣告：

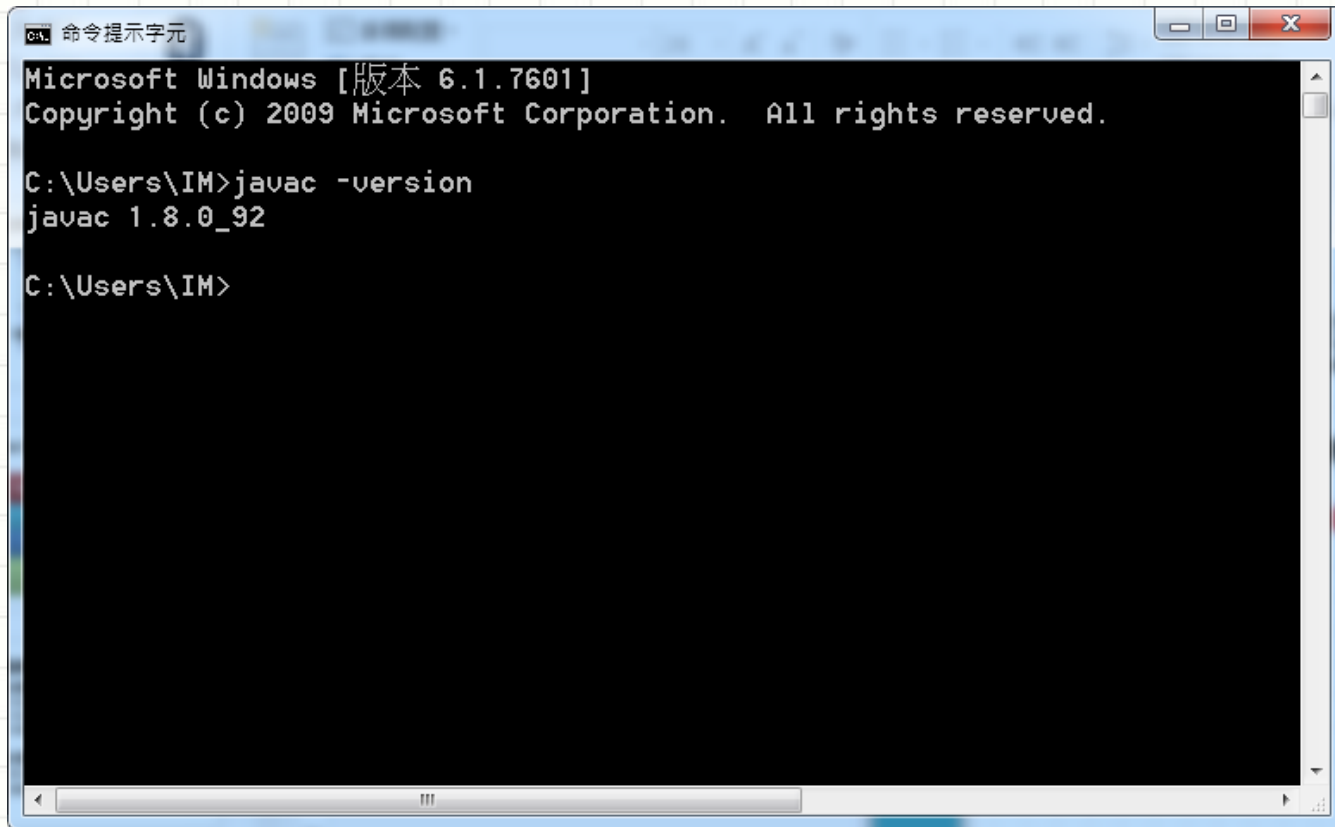
JAVA_HOME = C:\Program Files\Java\jdk1.7.0_51

2. 修改系統環境變數 **PATH**, 加入以下路徑(需包含分號)：
;%JAVA_HOME%\bin;
3. 增加系統環境變數 **CLASSPATH**, 設為以下值(或增加新值)：
.;%JAVA_HOME%\lib;

開發環境安裝與檢查-3

(D) 開啟文字命令列視窗(**cmd.exe**), 輸入以下指令

javac -version



```
命令提示字元
Microsoft Windows [版本 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\IM>javac -version
javac 1.8.0_92

C:\Users\IM>
```


練習時間：手動編譯+執行 Java 程式

- 從FB下載測試的 Java 原始程式(**Main.java**)
→ 存放於 **Z:\Disk_D\workspace** 資料夾內
- 開啟文字命令列視窗 (cmd.exe)，利用以下指令來切換工作目錄

D: ↩




cd \Disk_D\workspace ↩

- 編譯 Java 程式 (Main.java)
javac.exe Main.java ↩
- 執行 Java 程式 (Main.class)
java Main ↩


練習時間：Eclipse 開發環境

- 開啟桌面的 Eclipse
- 熟悉 Eclipse 開發環境
- 進行實驗
 - 新增專案
 - 新增 Java 類別
 - 編寫程式碼
 - 執行程式
 - 觀察結果

Package Explorer

專案瀏覽







程式編輯區



訊息、輸出輸入、
主控台

Task List









Find All Acti...

Connect Mylyn

[Connect](#) to your task and ALM tools or [create](#) a local task.

Outline

An outline is not available.

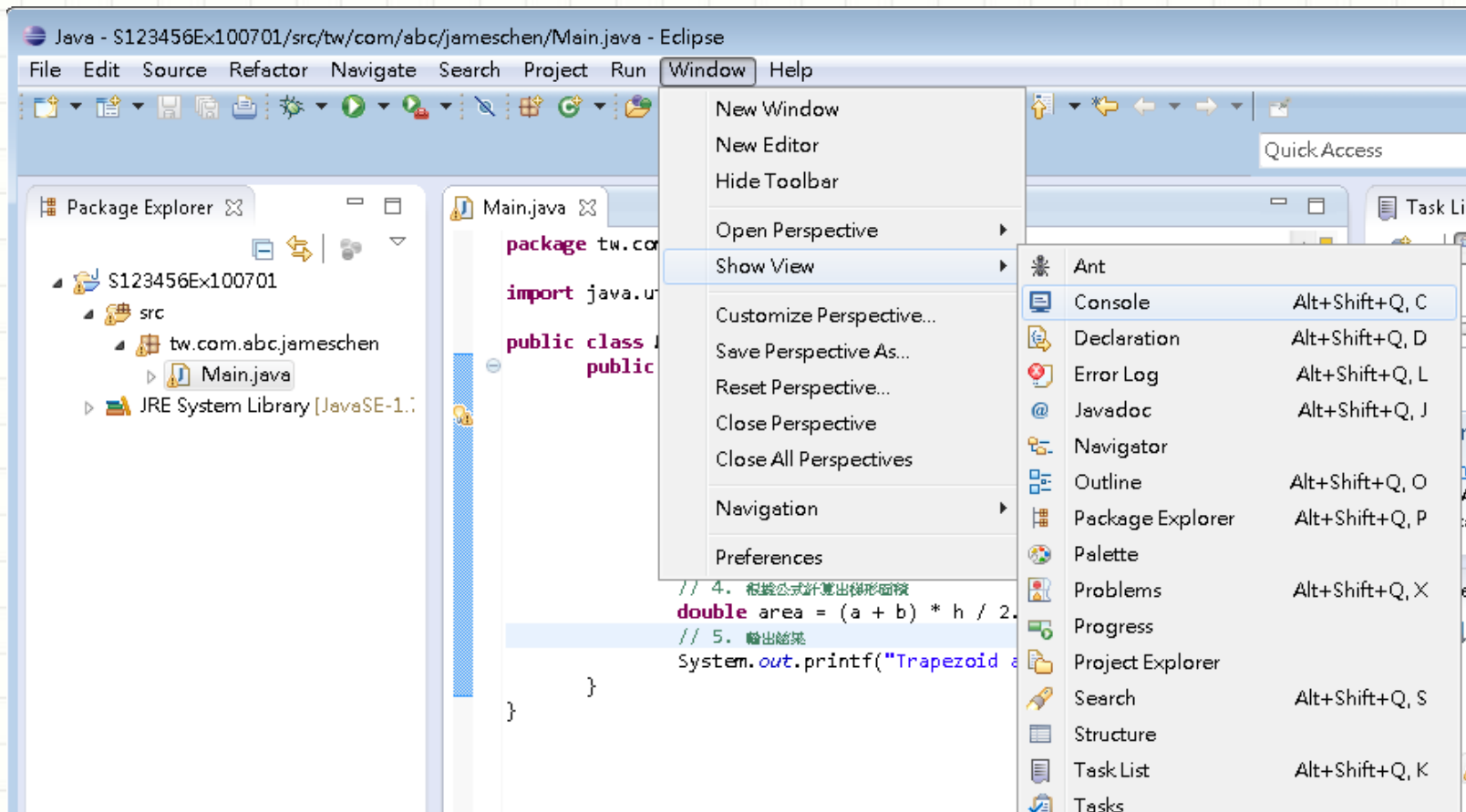
Problems @ Javadoc Declaration

0 items

Description	Resource	Path	Location	Type

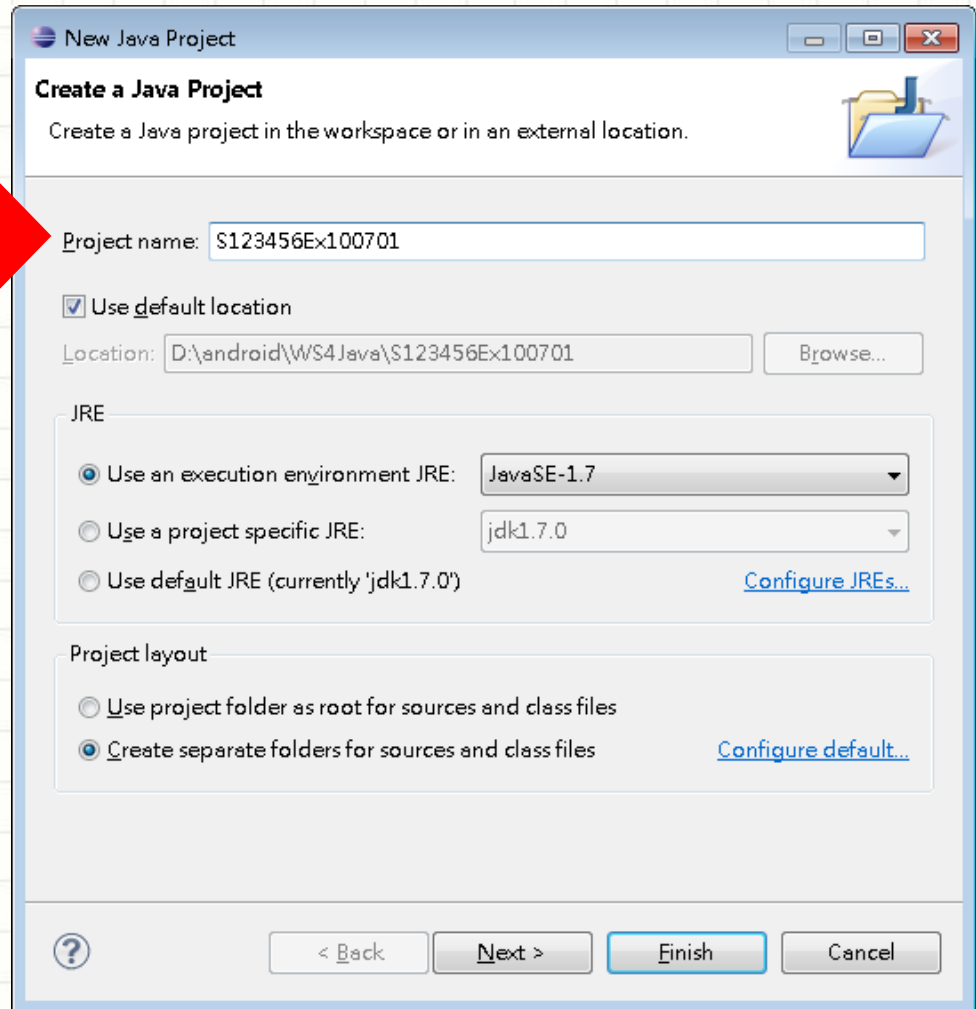
開啟 Console 主控台面板

- [Window] → [Show View] → [Console]



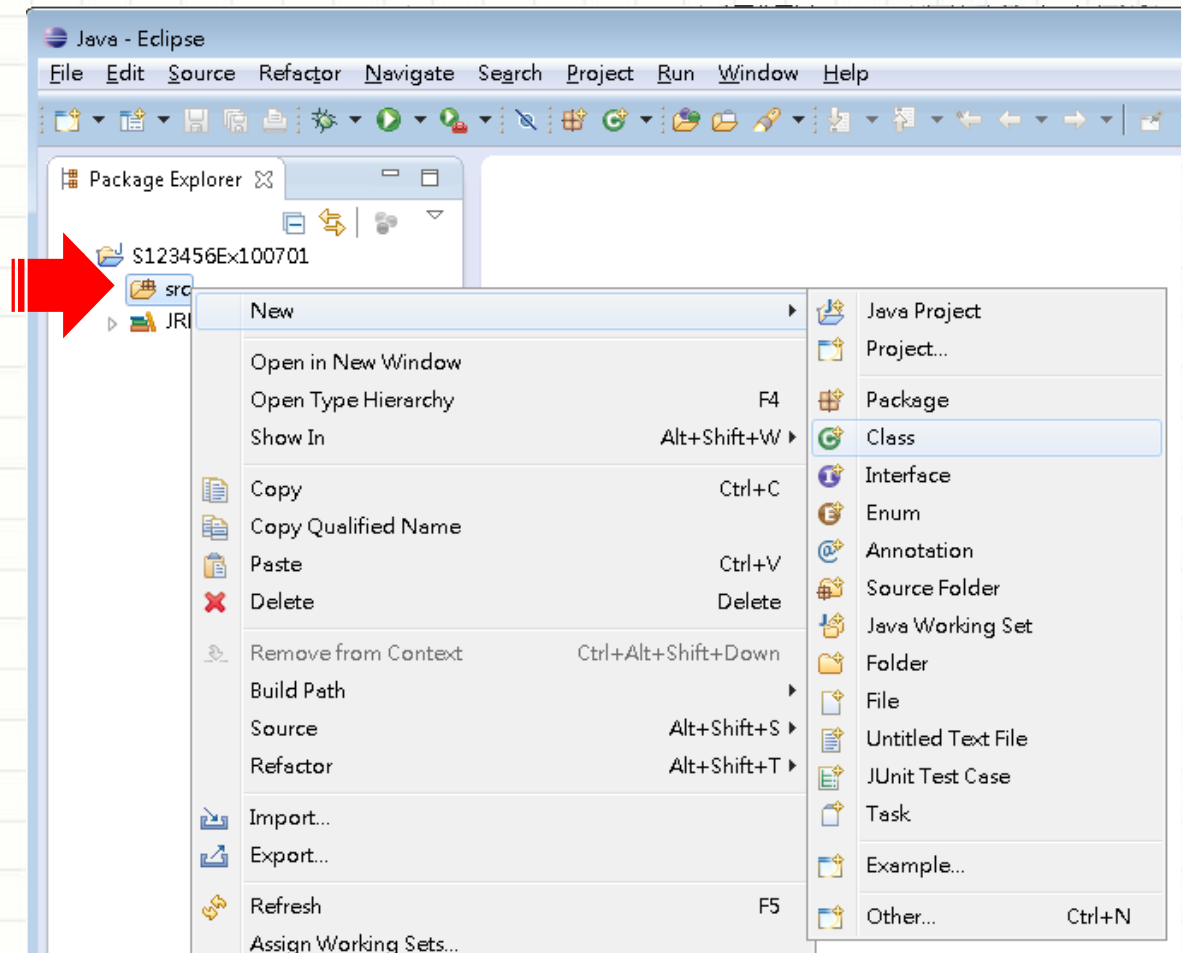
新增 Eclipse/Java 專案

- [File] → [New] → [Java Project]
- 輸入專案名稱 (Project name:) → [Finish]



新增一個類別檔案

- 以滑鼠左鍵
點擊專案
- 以右鍵點取
專案內原始
程式資料夾
src → [New]
→ [Class] →
輸入類別必
要參數



輸入參數

1. Package: 輸入套件名稱，
2. Name: 輸入類別名稱
3. 勾選 public static void main(String[] args)

New Java Class

Java Class

Create a new Java class.

Source folder: S123456Ex100701/src Browse...

Package: tw.com.abc.jameschen Browse...

Enclosing type: Browse...

Name: Main

Modifiers: public default private protected
 abstract final static

Superclass: java.lang.Object Browse...

Interfaces: Add...

Which method stubs would you like to create?

public static void main(String[] args)

Constructors from superclass

Inherited abstract methods

Finish Cancel

輸入以下程式碼並執行()

```
package tw.com.abc.jameschen;
```

```
public class Main {
```

```
public static void main(String[] args) {
```

```
System.out.println("Hello Java");
```

```
}
```

```
}
```


Java 基本程式架構

```
import java.lang.System; // 匯入所引用的外部元件
```

```
public class MyDemo {
```

```
public static void main(String[] args) {
```

```
    int x = 3, n=2;           // 宣告變數  
    long p = power(x, n);   // 計算 x 的n次方  
    System.out.println( p ); // 列印出計算結果;
```

```
}
```

```
public long power( int a, int b) {
```

```
    // 計算 a 的 b 次方;
```

```
    // 利用 return 回傳結果 ( long 的資料 );
```

```
}
```

```
}
```

Java 可能開發的過程

```
import java.lang.System; // 匯入所引用的外部元件
```

```
public class MyDemo {
```

```
    public static void main(String[] args) {
```

```
        int x = 3, n=2;           // 宣告變數  
        long p = power(x, n);     // 計算 x 的n次方  
        System.out.println( p ); // 列印出計算結果;
```

```
    }
```

```
    public long power( int a, int b) {
```

```
        // 計算 a 的 b 次方;
```

```
        // 利用 return 回傳結果 ( long 的資料 );
```

```
    }
```

```
}
```

學好程式設計的關鍵

- 掌握程式語言的基本架構

指令、副程式、模組化、套件

- 儲存資料: 了解各種資料型態的特性

數字、文字、布林；運算子

大量資料處理(陣列)、文字處理(字串)

- 控制流程: 熟悉基本語法 (+ 關鍵字)

```
if (條件) { /* 動作1 */ } else { /*動作2 */ }
```

```
while ( 條件) { /* 動作 */ }
```

- 善用開發工具進行除錯(找問題): 中斷點、單步執行、...

- 善用工具: 熟用內建程式庫 (類別庫、副程式、模組)

進階資料型別: Stack、List、Queue、... ..

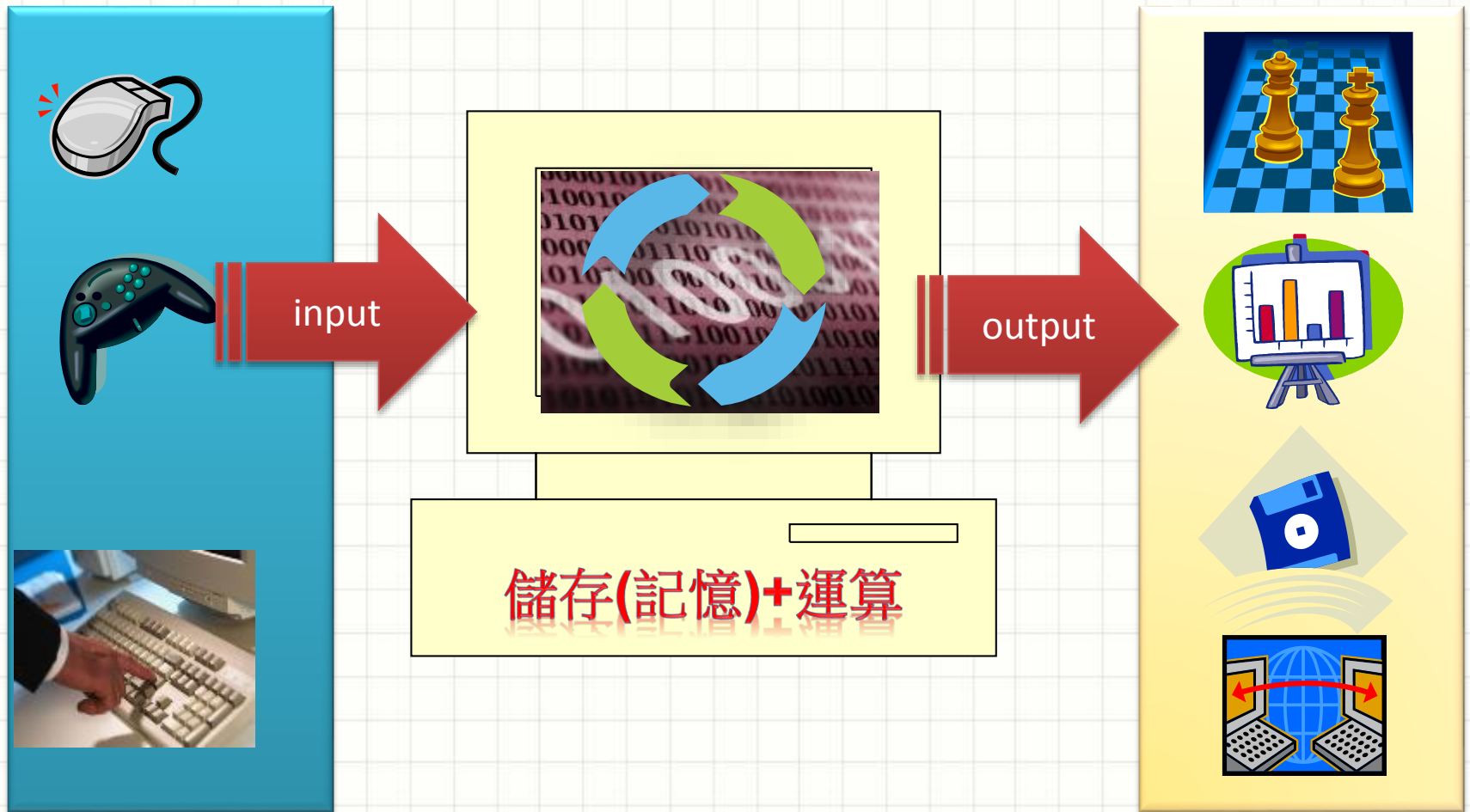


3*P

Java 程式語言的學習重點...

- 基本語法、程式基本框架、流程控制指令
- 資料儲存與運算(子)處理：數值、文字(字串)
- 物件導向的重要觀念 – 封裝、繼承與多型
 - 類別? 物件?
 - 抽象類別、抽象方法、介面
 - 繼承(類別) v.s 實作(介面)
- 基本元件(類別)的使用 – 如：資料轉換
- 檔案存取、例外處理
- 多執行緒(Thread/Runnable)與同步處理
- 容器(Collection)與泛型(Generics)的使用
- 網路連線(Socket/HTTP/...)、資料庫(JDBC) ...

Review : 電腦的主要工作 ... 計算



Lab01: 計算梯型面積

- 問題描述：

- 請以 JAVA 運算式計算下面梯形面積，並輸出面積結果。

- ※ 梯形面積公式為： $(\text{上底} + \text{下底}) \times \text{高} \div 2$ 。

- 輸入說明：

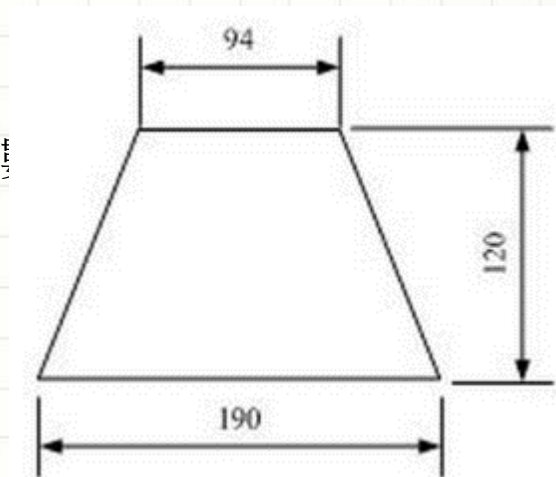
- 每一組依序分別輸入梯形的上底、下底及高的整數。

- 輸出說明：

- 輸出梯形面積。

- 範例：

輸入	輸出
94 190 120	梯形面積為:17040.0
99 54 47	梯形面積為:3595.5



輸入以下程式碼並執行()

```
import java.util.Scanner;
```

```
public class Main {
```

```
    public static void main(String[] args) {
```

```
        // 1.取得掃描物件，藉以讀取鍵盤所輸入的資料
```

```
        Scanner input = new Scanner(System.in);
```

```
        // 2. 宣告變數( a, b, h)：儲存梯形的上底、下底與高
```

```
        int a, b, h;
```

```
        // 3. 讀取資料
```

```
        a = input.nextInt();
```

```
        b = input.nextInt();
```

```
        h = input.nextInt();
```

```
        // 4. 根據公式計算出梯形面積
```

```
        double area = (a + b) * h / 2.0;
```

```
        // 5. 輸出結果
```

```
        System.out.printf("梯形面積為:%.1f\n", area);
```

```
    }
```

```
}
```

在下方主控台面板內輸入
測試資料(上底、下底、高)

儲存(記憶體)+運算(運算子)

- 宣告變數: 資料型別(=空間、內容)+ 變數名稱

```
int age;
```

```
long a=1, b=2, c;
```

```
float distance=18.5f;
```

```
double[ ] prices;
```

```
prices = new double[100];
```

- 使用變數: 利用運算子(Operator)

```
c = 3*a * a + 2*a + 4;
```

```
String msg = "我是" + 12 + "歲";
```

Java 有八大
基本資料型別

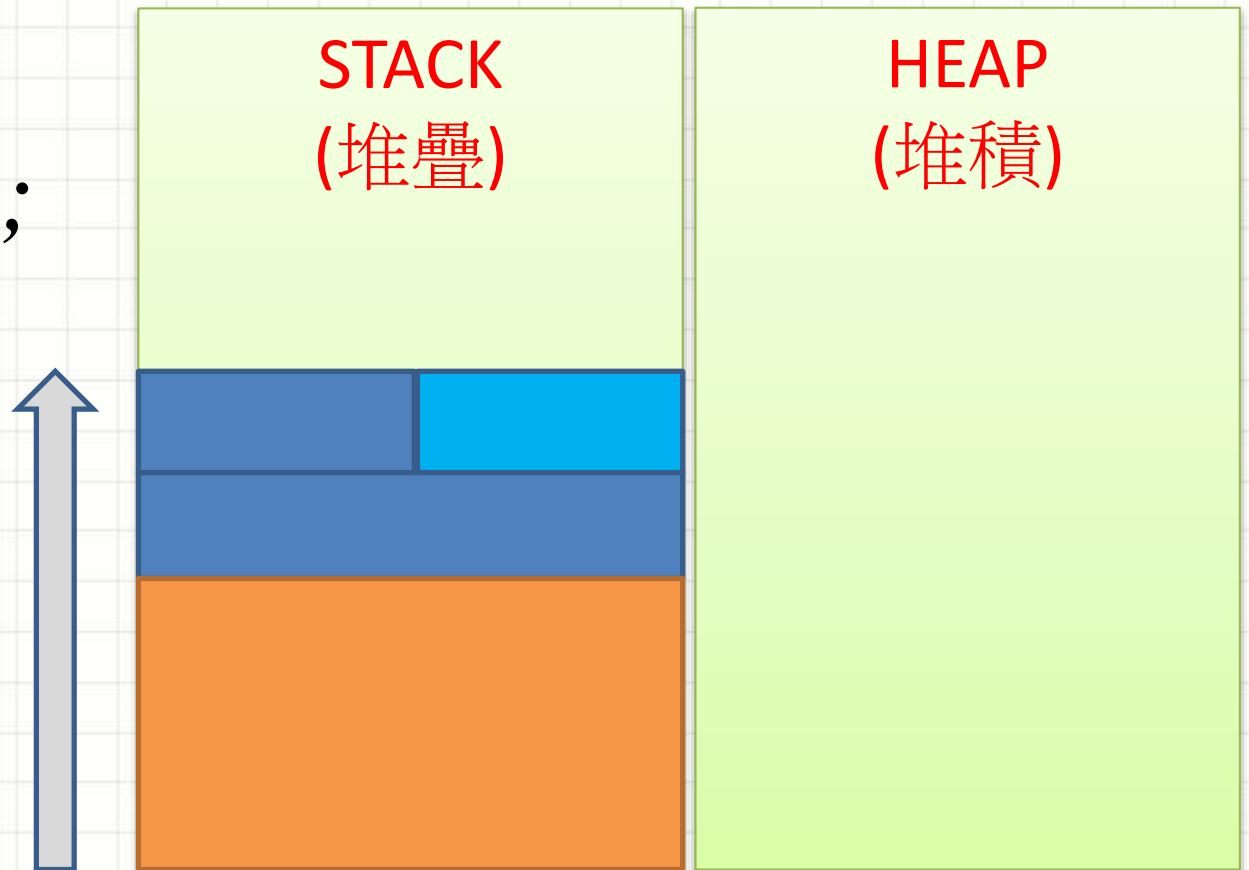
+ 這個符號有
不同功用!?

變數 (Variables) : 配置記憶體

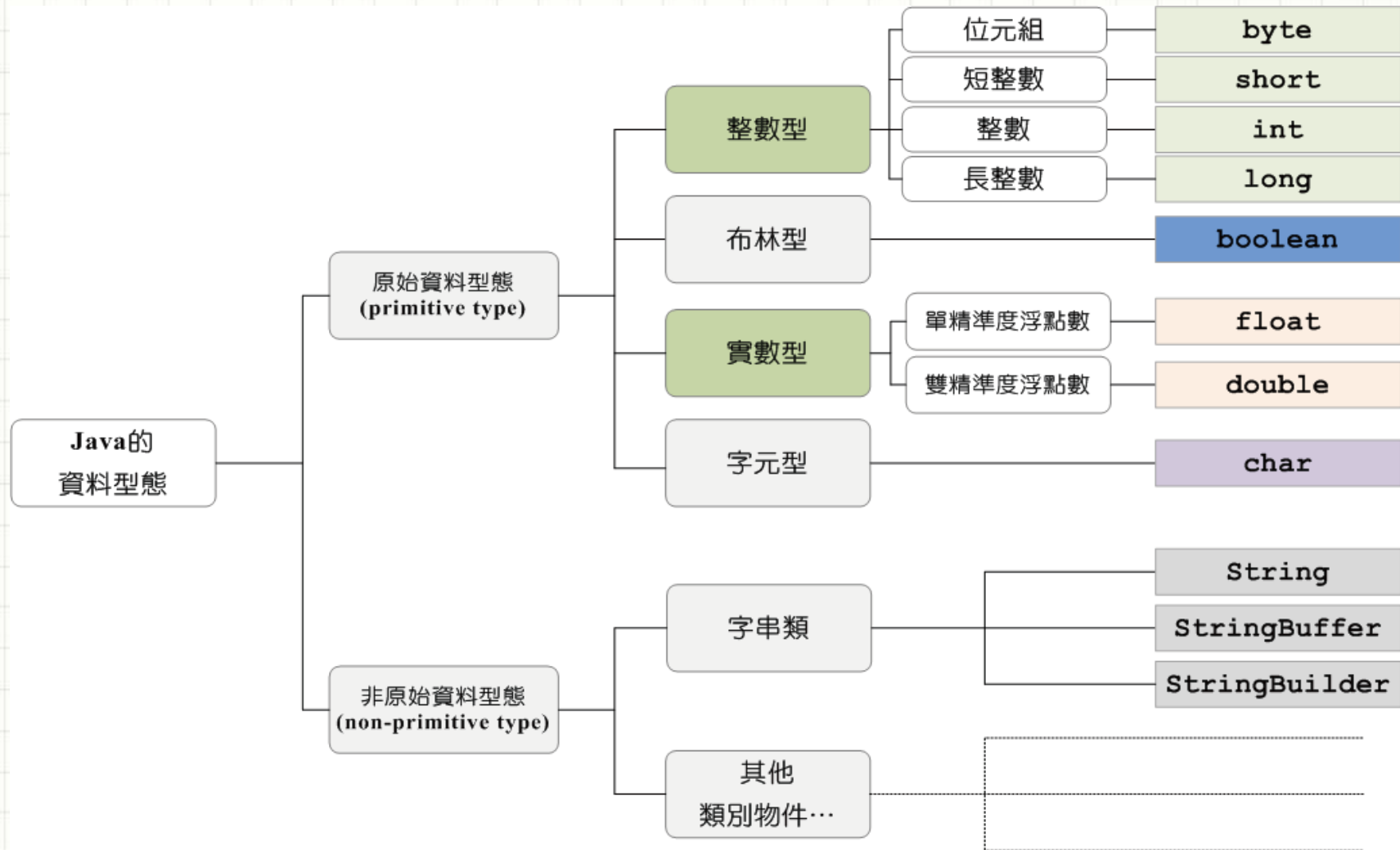
- 宣告變數: 資料型別(空間、內容) + 變數名稱

int age;

char ch;



資料型別(Data-type)

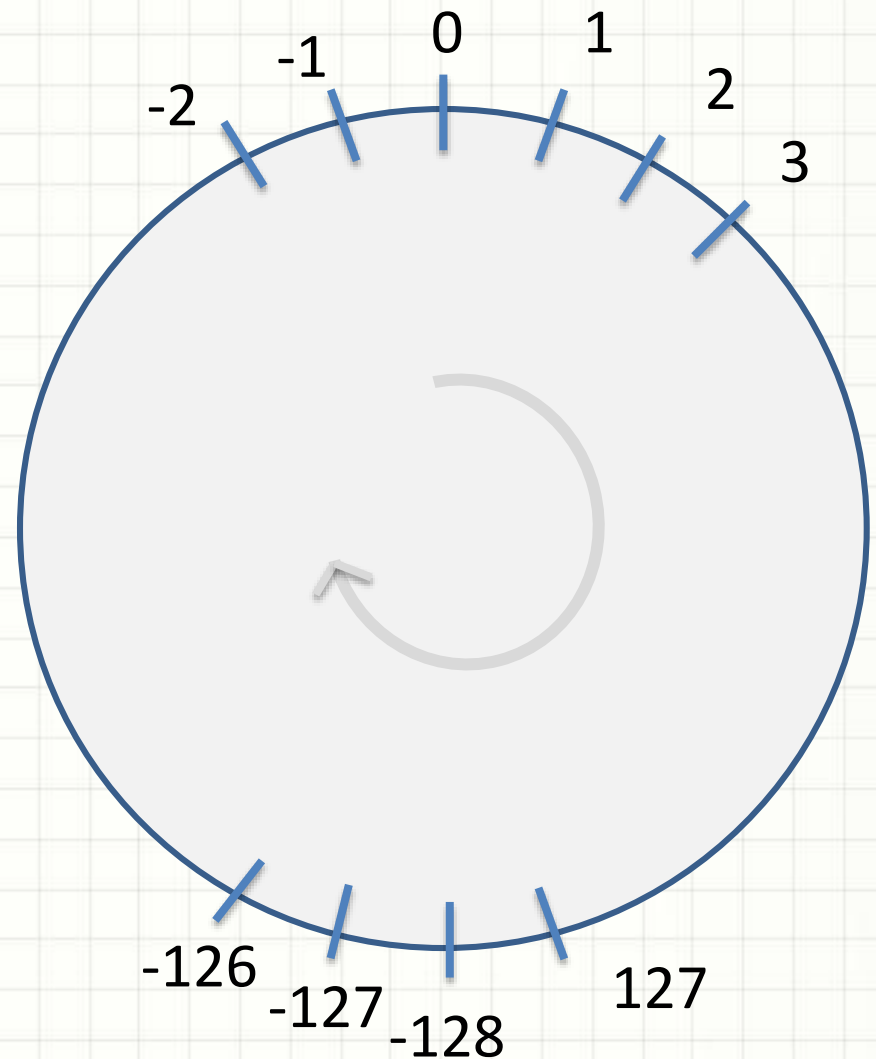


基本資料的相關資訊

資料型態名稱	關鍵字	位元組	說明
位元組	byte	1	-128 ~ 127之整數
短整數	short	2	-32,768 ~ 32,767之整數
整數	int	4	-2,147,483,648 ~ 2,147,483,647之整數
長整數	long	8	-9,223,372,036,854,775,808 ~ +9,223,372,036,854,775,807
字元	char	2	16-bit Unicode字元 最小值為'\u0000' (或0) ~ 最大值為'\uffff' (或65,535) 單引號 'A', '1', '#', ...
布林	boolean	1	true 或 false
單倍精準度浮點數	float	4	採用32-bit IEEE 754浮點數格式 精確位數為7位，在正數方面可表達3.4E-38~3.4E+38 (E後面的數字代表10的次方數)
雙倍精準度浮點數	double	8	採用64-bit IEEE 754浮點數格式 精確位數為15位，在正數方面可表達1.7E-308~1.7E+308 (E後面的數字代表10的次方數)

整數的表示法: 2的補數 (byte)

0 = 0000 0000
1 = 0000 0001
2 = 0000 0010
3 = 0000 0011
...
127 = 0111 1111
-128 = 1000 0000
-127 = 1000 0001
-126 = 1000 0010
...
-2 = 1111 1110
-1 = 1111 1111



最基本的運算子：指定(=)

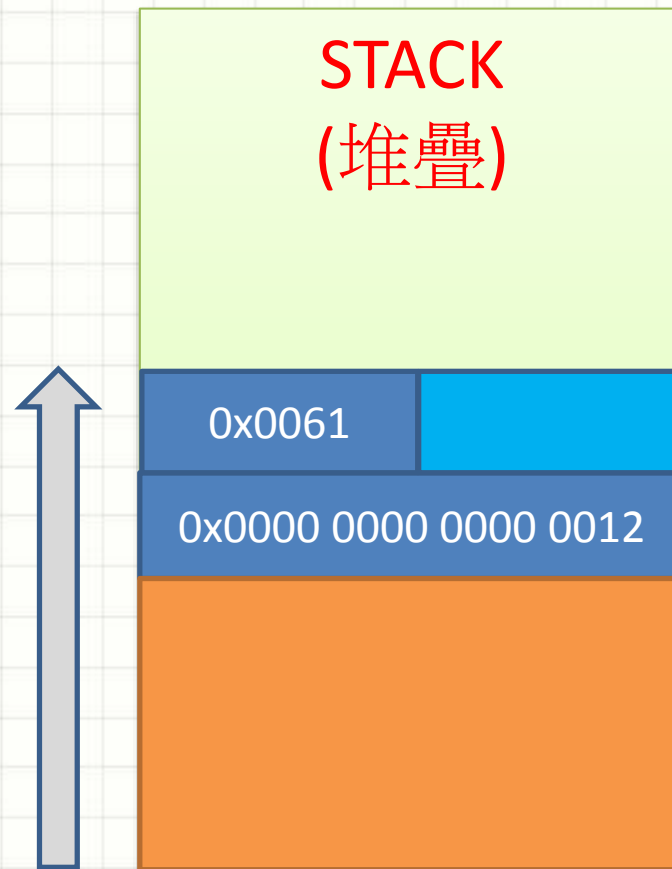
- 宣告變數：資料型別(空間、內容) + 變數名稱

```
int age;
```

```
char ch;
```

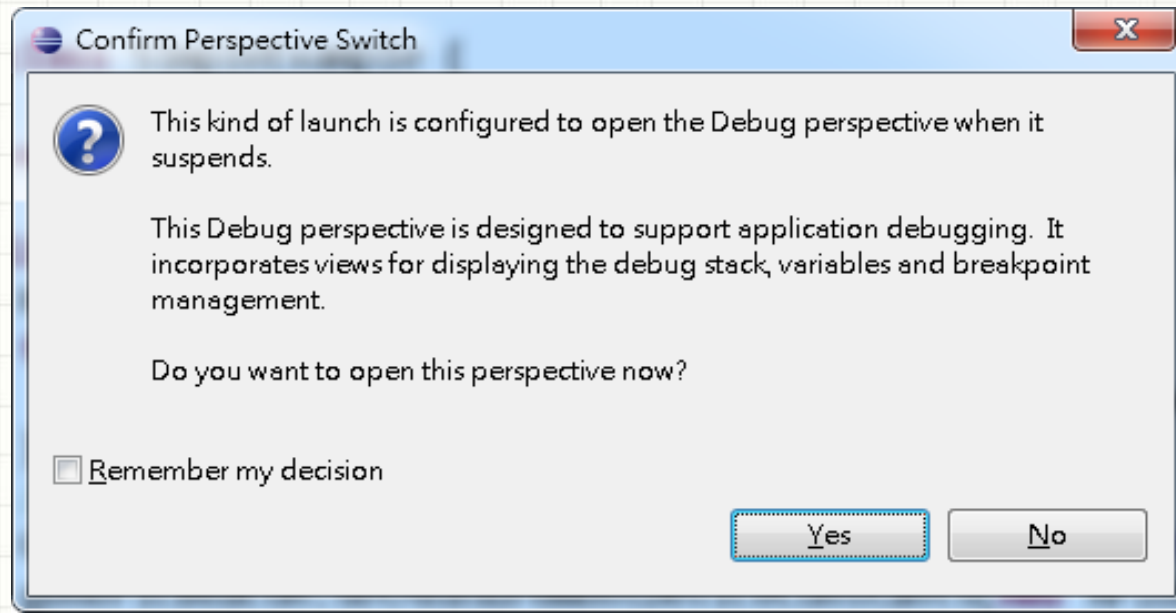
```
age = 18;
```

```
ch = 'A';
```



練習時間：利用Eclipse觀察變數

- 開啟測試程式
- 設定行號顯示
- 設定必要的中斷點, 進行除錯程序
- 過程中的確認視窗 ...



Eclipse 的除錯模式 (DEBUG)

The screenshot displays the Eclipse IDE in Debug mode. The top toolbar is highlighted with a red dashed box, and a red arrow points to the Debug button. The Debug console shows the execution of SimpleExample.java, with a breakpoint at line 18. The Variables view shows the state of the program, including the args array and the kalmanFilter object. The source code editor shows the main method, with line 18 highlighted. The Outline view shows the project structure and the current method being debugged.

```
public static void main(String[] args){
    KalmanFilter kalmanFilter = new KalmanFilter();
    for(int d : DATA){
        System.out.println((int)d * -1 + "," + (int)kalmanFilter.applyF
    }
    //
    GenericBeacon genericBeacon = new GenericBeacon("macHere",-59);
    genericBeacon.setMeasurementDeviceConstants(new RPiEmbeddedAntennaC
    genericBeacon.setRssiFilter(new KalmanFilter());
```

Name	Value
args	String[0] (id=16)
kalmanFilter	KalmanFilter (id=17)
errorCovarianceRSSI	0.0
estimatedRSSI	0.0
isInitialized	false
measurementNoise	0.8
processNoise	0.125

運算子：處理資料(運算元)

■ 運算式 = 運算子 + 運算元

```
int a = 2, c=0;
```

```
c = 3*a * a + 2*a + 4;
```

```
String msg;
```

```
msg = "我是" + 12 + "歲";
```

運算子：
一元、二元、三元

運算式的結果
是...?

運算子的分類

- 算術運算子：數值運算
- 關係運算子：比較大小
- 邏輯運算子：組合多重條件
- 位元運算子：逐位元處理
- 指定運算子：改變變數內容(寫入記憶體)
-

算術運算子

單元運算子	用法	說明
+	+3	正 3
-	-3	負 3

二元運算子	用法	說明	a	b	評估結果
+	$a + b$	a 加 b	5	3	8
-	$a - b$	a 減 b	5	3	2
*	$a * b$	a 乘以 b	5	3	15
/	a / b	a 除以 b	5	3	1
%	$a \% b$	a 除以 b 的餘數	5	3	2
++	$a++$	後置：先給再加	5		5 (a:6)
	$++a$	前置：先加再給	5		6 (a:6)
--	$a--$	後置：先給再減	5		5 (a:4)
	$--a$	前置：先減再給	5		4 (a:4)

關係運算子 – true/false

- 若運算子是由兩個符號組成，則符號之間**不能有空格**

運算子	用法	說明	a	b	評估結果
<	$a < b$	a是否小於 b	5	3	false
<=	$a <= b$	a是否小於或等於 b	5	3	false
==	$a == b$	a是否等於 b	5	3	false
!=	$a != b$	a是否不等於 b	5	3	true
>=	$a >= b$	a是否大於或等於 b	5	3	true
>	$a > b$	a是否大於 b	5	3	true

邏輯運算子：true/false

- 盡可能使用短迴路運算子

運算子	用法	說明
&&	$x \ \&\& \ y$	AND (且)運算: $x \ y$ 同時為 true, 就回傳 true, 否則回傳 false 短迴路 : 若 x 為 false, 則不再評估 y (效率佳)
&	$x \ \& \ y$	AND (且)運算: $x \ y$ 同時為 true, 就回傳 true, 否則回傳 false
	$x \ \ \ y$	OR(或)運算: $x \ y$ 有一個是 true, 就回傳 true, 否則回傳 false ($x \ y$ 同時為 false) 短迴路 : 若 x 為 true, 則不再評估 y (效率佳)
	$x \ \ y$	OR(或)運算: $x \ y$ 有一個是 true, 就回傳 true, 否則回傳 false ($x \ y$ 同時為 false)
!	$! \ x$	NOT(反相): 當 x 是 true, 則回傳 false, 否則回傳 true
^	$x \ \wedge \ y$	XOR(互斥)運算: 當 $x \ y$ 同時為 true 或 false, 則回傳 false, 否則回傳 true; 也就是 $x \ y$ 不相同時, 就回傳 true。

真值表

x	y	$x \& y$
true	false	false
true	true	true
false	false	false
false	true	false

x	y	x / y
true	false	true
true	true	true
false	false	false
false	true	true

x	y	$x \wedge y$
true	false	true
true	true	false
false	false	false
false	true	true

x	$!x$
true	false
false	true

位元(bitwise) 運算子

運算子	意義	用法	說明
~	反相	~x	將x的所有位元做NOT運算 (0變1, 1變0)
&	且	x & y	將x與y相對應的位元做AND運算 (1 & 1 = 1)
	或	x y	將x與y相對應的位元做OR運算 (0 0 = 0)
^	互斥	x ^ y	將x與y相對應的位元做XOR (互斥) 運算
>>	右移	x >> p	將x的內容往右移動p個bit (前：正補0, 負補1)
>>>	右移	x >>> p	將x的內容往右移動p個bit (左邊補0)
<<	左移	x << p	將x的內容往左移動p個bit (右邊補0)

x	y	x&y
0	0	0
0	1	0
1	0	0
1	1	1

x	y	x / y
0	0	0
0	1	1
1	0	1
1	1	1

x	y	x^y
0	0	0
0	1	1
1	0	1
1	1	0

範例：逐位元(bitwise) 運算

• **5 & 3 → 1**

先轉為2 進制

5 : 0000 0000 0000 0000 0000 0000 0000 0101

& 3 : 0000 0000 0000 0000 0000 0000 0000 0011

1 : 0000 0000 0000 0000 0000 0000 0000 0001

• **5 ^ 3 → 1**

5 : 0000 0000 0000 0000 0000 0000 0000 0101

^ 3 : 0000 0000 0000 0000 0000 0000 0000 0011

6 : 0000 0000 0000 0000 0000 0000 0000 0110

指定運算子

- 指定運算子 (=) 可以和其他運算元合併為精簡表示法 $x = ; (x : +, -, *, /, \%, \ll, \gg, \ggg, \dots)$

運算子	用法	說明	評估前		評估後	
			a	b	a	b
=	$a = b$	將 b 的值寫入 a 變數(記憶體)	5	3	3	3
+=	$a += b$	等同 $a = a + b$	5	3	8	3
-=	$a -= b$	等同 $a = a - b$	5	3	2	3
*=	$a *= b$	等同 $a = a * b$	5	3	15	3
/=	$a /= b$	等同 $a = a / b$	5	3	1	3
%=	$a \% = b$	等同 $a = a \% b$	5	3	2	3

還有 ...

- 特殊功能運算子

運算子	用法	說明
<code>?:</code>	<code>x ? a : b</code>	精簡式 if ... else 如果 x 為 true，傳回 a 如果 x 為 false，傳回 b
(資料型別)	<code>(int) 3.5</code> <code>(char) 65</code>	強制轉型轉換 Ex: 將浮點數當作整數 <code>(int)3.5 → 3</code> Ex: 將整數當作字元 <code>(char)65 → 'A'</code>
<code>instanceof</code>	<code>k instanceof C</code>	k 是否為 C 的實體物件 (Instance, object) Ex: <code>arFu instanceof Dog</code>

運算子: 優先順序與結合性

Precedence	Operator	Operand type	Description
1	++,	Arithmetic	Increment and decrement
1	+, -	Arithmetic	Unary plus and minus
1	~	Integral	Bitwise complement
1	!	Boolean	Logical complement
1	(type)	Any	Cast
2	*, /, %	Arithmetic	Multiplication, division, remainder
3	+, -	Arithmetic	Addition and subtraction
3	+	String	String concatenation
4	<<	Integral	Left shift
4	>>	Integral	Right shift with sign extension
4	>>>	Integral	Right shift with no extension
5	<, <=, >, >=	Arithmetic	Numeric comparison
5	instanceof	Object	Type comparison
6	==, !=	Primitive	Equality and inequality of value
6	==, !=	Object	Equality and inequality of reference
7	&	Integral	Bitwise AND
7	&	Boolean	Boolean AND
8	^	Integral	Bitwise XOR
8	^	Boolean	Boolean XOR
9		Integral	Bitwise OR
9		Boolean	Boolean OR
10	&&	Boolean	Conditional AND
11		Boolean	Conditional OR
12	?:	N/A	Conditional ternary operator
13	=	Any	Assignment

R -> L

L -> R

R -> L

運算式的評估

int a = 2, c=0;

*c = 3 * a * a + 2 * a + 4*

→ *c = 20*

String result1 = 1 + 2 + “元”;

→ “3元”

String result2 = “共” + 1 + 2 + “元”;

→ “共12元”?

練習時間：觀察變數的變化

■ 預測以下執行結果

```
int a=2, b=3, c=0;
```

```
c = a++ + ++b;
```

```
c = a + b * a + b;
```

```
c = a ^ b;
```

a	b	c
2	3	0
3	4	6
3	4	19
3	4	7

■ 請利用 **Eclipse** 驗證結果

輸出資料 System.out

- **System.out : 標準輸出= 主控台**
 - 內建功能模組: 可以列印各類型的資訊
 - 使用輸出功能時須利用 小括號(...)傳入列印的資料
- **直接呼叫相關的副程式 printxxx(...)**
 - System.out.print(123);*
 - System.out.println("123");*
 - System.out.printf("%d*%d=%d\n",2,3,2*3);*
- **使用 printf 需搭配格式化標籤 (佔位符號) %X**
 - %d : 整數
 - %f : 浮點數


常用的格式化標籤(佔位符號)

標籤	功能	Notes
%d	整數(十進位)	%2d % 0 2d
%f	浮點數	%.3f %10.3f
%x	整數(16進位)	%X
%s	字串	%10s % - 10s
%c	字元	

買受人註記欄		
區分	進貨及費用	固定資產
得扣抵		
不得扣抵		

檢查號碼：A85

金額	備註
200	
13,000	
6,000	
20	
800	
500	
30	
25	營業人蓋用統一發票專用章
15	
20,590	
1,030	
21,620	
零元	


 奇機資訊股份有限公司
 統一發票專用章
 統一編號
 70429505
 負責人：李清培
 TEL: 07-5585585
 高雄市中區博愛二路366號4樓之5

00000號函核准使用。

練習時間：格式化輸出

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

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

現在時間是 2018/01/31 15:01:35

輸入資料 Scanner+System.in

- **System.in** : 標準輸入 = 鍵盤
 - 不方便直接輸入、處理各種類型的資料
- 利用 Java 內建模組(類別) **Scanner** 處理

```
Scanner kb = new Scanner(System.in);
```

```
int a = kb.nextInt();
```

```
int b = kb.nextInt();
```

```
double f = kb.nextFloat();
```


建議練習: ITSA (溫度轉換)

- Step1. 登入 ITSA
- Step2. 點擊練習題目數學(C_MM10)
 - <http://e-tutor.itsa.org.tw/e-Tutor/mod/programming/view.php?id=6882>
- Step3. 分析題目
- Step 4. 嘗試利用 Eclipse 解題、自我測試
- Step 5. 上傳、查看結果

C_MM10 題目資訊

輸入範例	輸出範例
27	80.6
32.5	90.5

- 題目敘述:

請撰寫一個程式，依據代表攝氏溫度的變數 c 的值，顯示華氏溫度

公式: 攝氏溫度 = 華氏溫度減32度再乘上5/9

- 輸入:

- 輸入攝氏溫度(數值不一定是整數)。

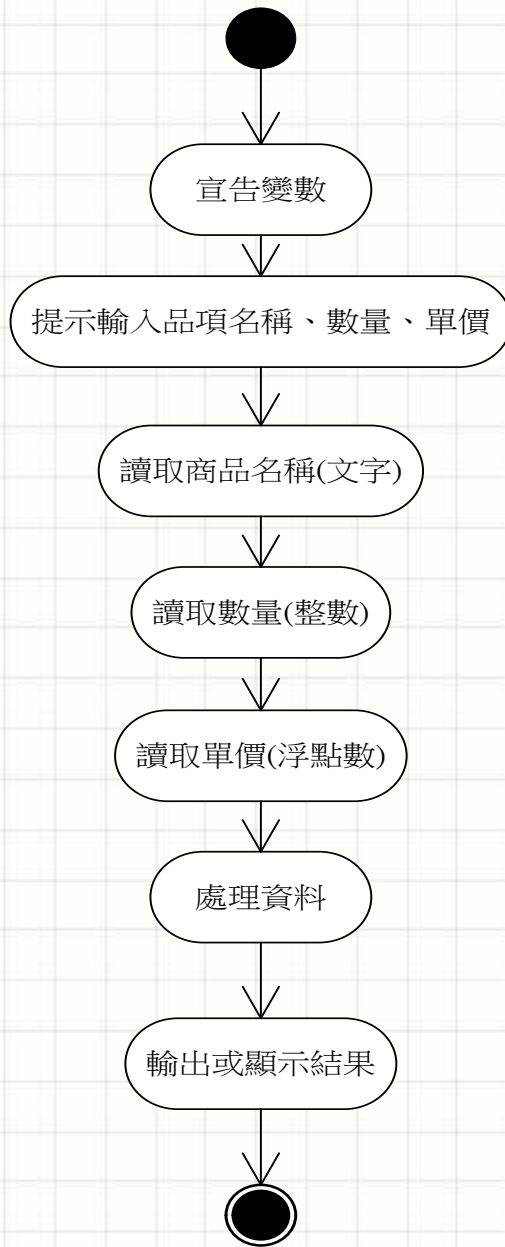
- 輸出:

- 華氏溫度(四捨五入取至小數點第一位，整數需補零)。

練習時間：請問你要買甚麼？

- 輸入要購買的物品名稱、數量、單價
 - 例如：**蘋果**, **10**, **39**
- 並顯示出以下結果：
 - 品名：**蘋果**, 單價:**10**元, 小計:**390**元。

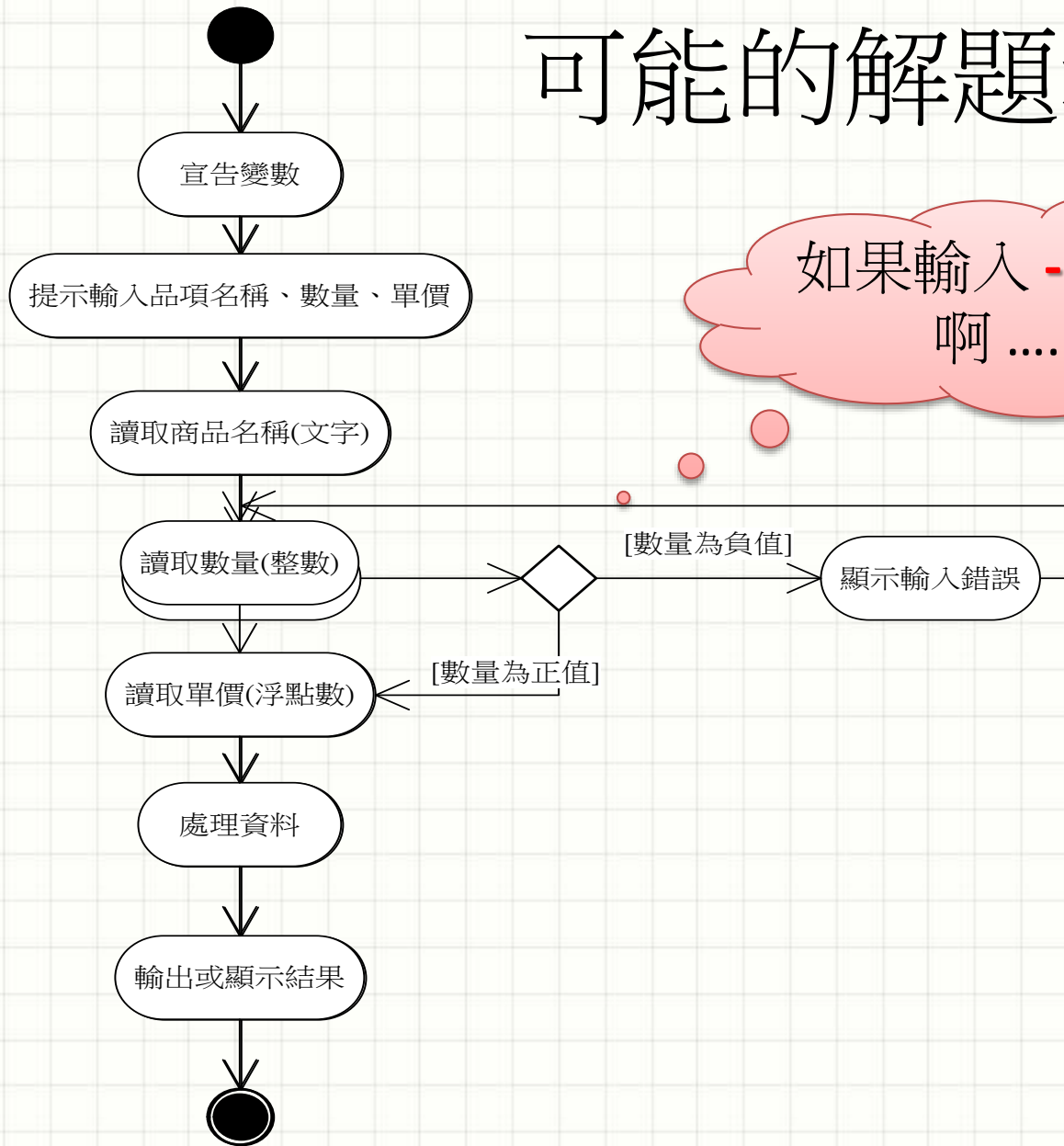
可能的解題步驟-1



題目要求：

- 輸入要購買的物品名稱、數量、單價
例如：**蘋果, 10, 39**
- 顯示出以下結果：
品名：**蘋果**, 單價:**10**元, 小計:**390**元。

可能的解題步驟-2



如果輸入 **-10** 呢?
啊

沒有一成不變的事 ... 要控制流程

- 循序結構 : <built-in>
- 選擇結構: if ... else, switch...case...default
- 重覆結構: for, while, do ...while;