

Java

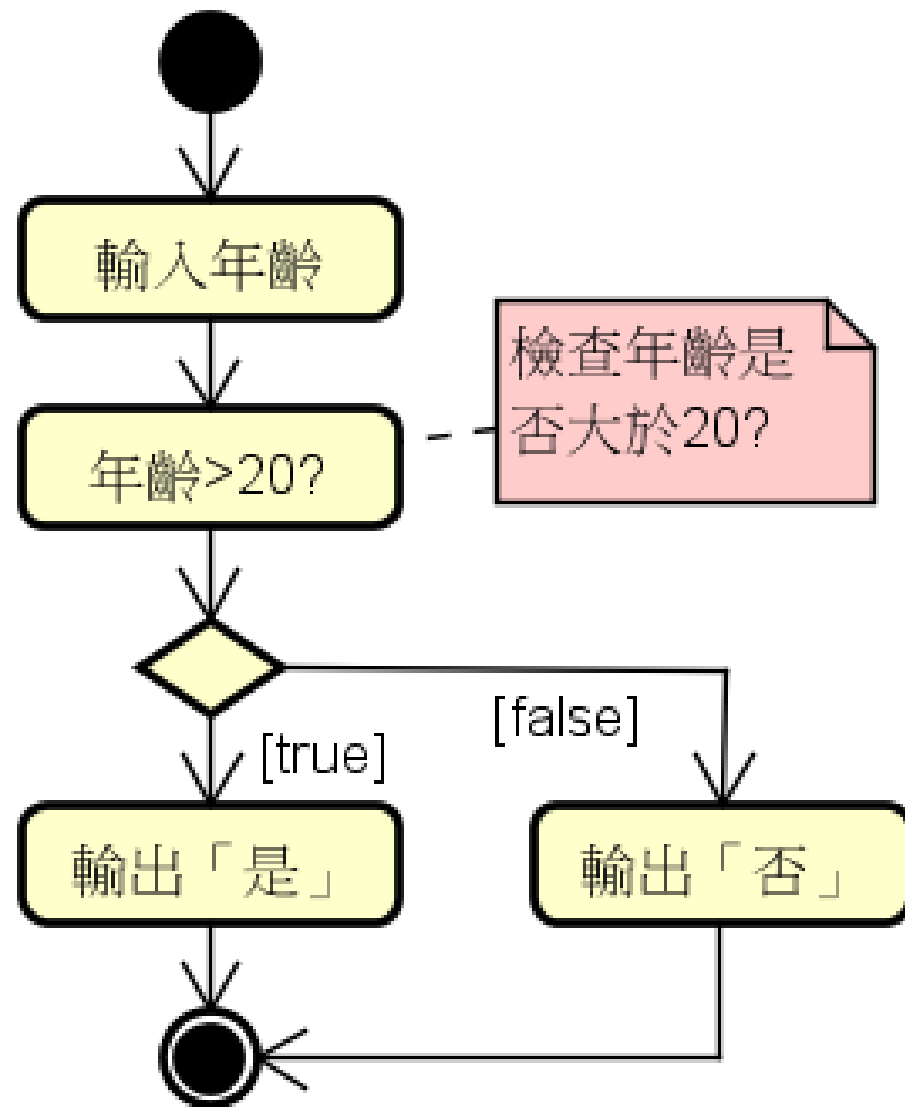
比較 - 真或假

請說出你的年齡，
並回答是否有大於20歲

問題



判斷流程



Java 來比較

```
import java.util.Scanner;

class web {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int age;

        age = sc.nextInt();
        System.out.println(age > 20);
    }
}
```



與數學運算比較一樣，用 > 符號比較左邊資料是否有大於右邊資料

執行結果

15
false

似曾相似的 false ?

比較結果輸出很熟悉？

```
import java.util.Scanner;

class web {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int age;

        age = sc.nextInt();
        System.out.println(age > 20);
        System.out.printf("%b\n", (age > 20));
        System.out.printf("%B\n", (age > 20));
    }
}
```



false 就是在【資料型態】裡面介紹過的『布林 boolean』型態，因此搭配輸出格式 **%b** 或 **%B** 也可正確的輸出喔！

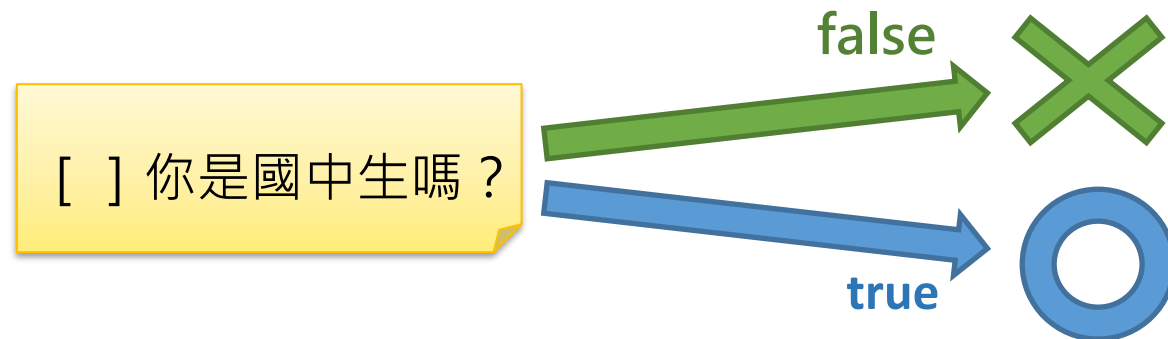
執行結果

```
15
false
false
FALSE
```

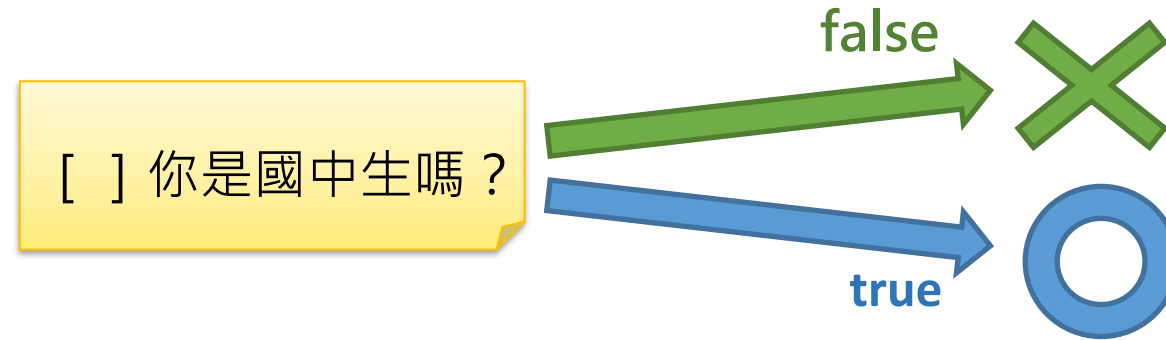
複習一下：布林 boolean 資料型態

- 布林資料型態專門儲存只有兩種答案的資料！
- 布林資料只有兩種可能內容：**true** 或 **false**
 - 可以將是非題的○視為 true，X 視為 false。

資料型態	名稱	可儲存的範圍	儲存空間 (單位:bits)	範例
boolean	布林	true、false	1	boolean a = true



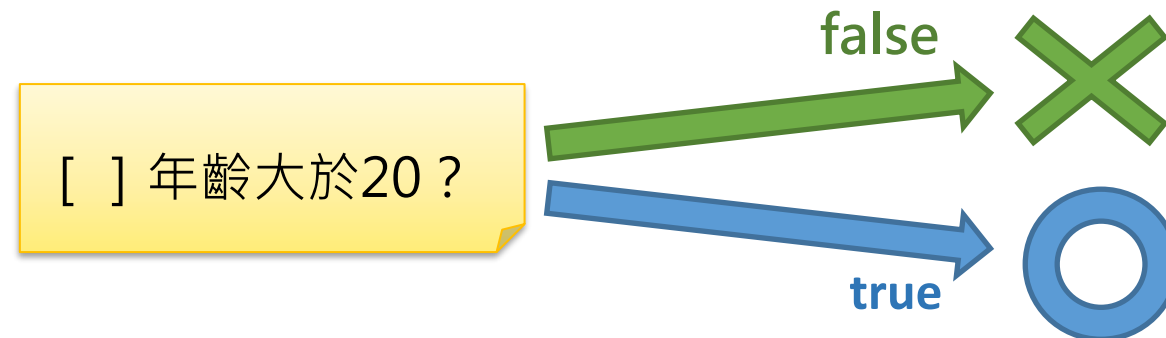
比較也是一種是非題



進行比較後同樣會得到兩種特別的值：

- True
- False

因此比較結果會是一種布林資料型態！



布林資料型態：True & False

True 可以當做是：

- 真的
- 對的
- 正確
- Yes
- 整數的 1

False 可以當做是：

- 假的
- 錯的
- 錯誤
- No
- 整數的 0



布爾	整數	代表的意義
True	1	真(對)
False	0	假(錯)

由於 True 跟 False 可以視為整數的 1 與 0，因此也可將布林型態視為整數型態的子集合

Java 嘛ㄟ通的數學比較符號

符號	數學意義	Java 比較運算子 (Comparison Operators)
=	等於	<code>==</code>
≠	不等於	<code>!=</code>
<	小於	<code><</code>
≤	小於等於	<code><=</code>
>	大於	<code>></code>
≥	大於等於	<code>>=</code>

前面有提過等號『=』是將右邊的內容指派給左邊變數的**指派運算子**，因此為了避免搞混，使用了兩個等號『==』代表等於的比較運算

限於打字輸入的方便性，比較符號的寫法會有部分差異，但代表的意義都是相同的喔！



Java 程式的比較運算

```
class web{  
    public static void main(String[] args) {  
        int age = 20;  
  
        System.out.println(age == 20);  
        System.out.println(age != 20);  
        System.out.println(age < 20);  
        System.out.println(age <= 20);  
        System.out.println(age > 20);  
        System.out.println(age >= 20);  
    }  
}
```



執行結果

```
true  
false  
false  
true  
false  
true
```

比較運算進階版



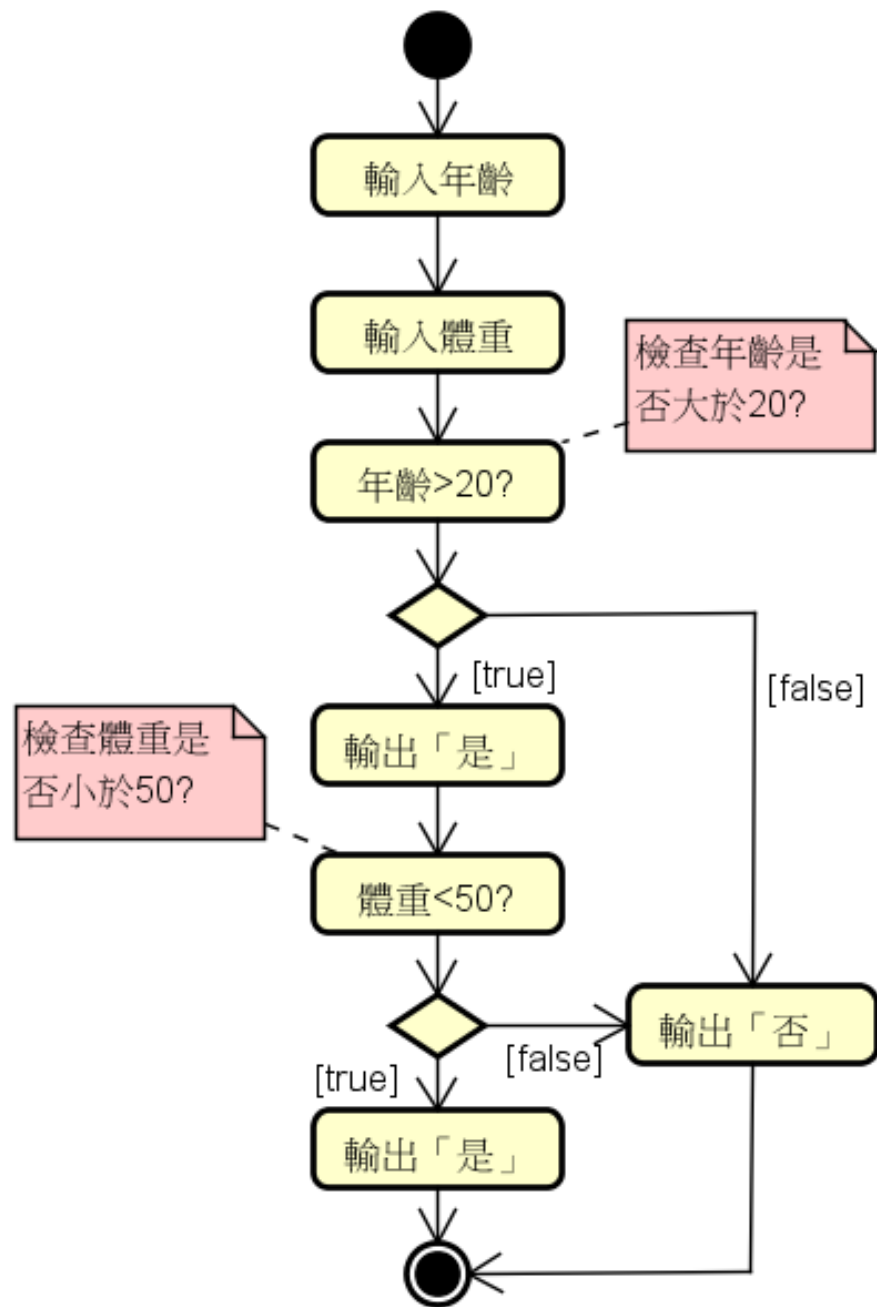
JAVA

請說出你的年齡與體重，
並回答是否有大於20歲，
且體重小於50公斤

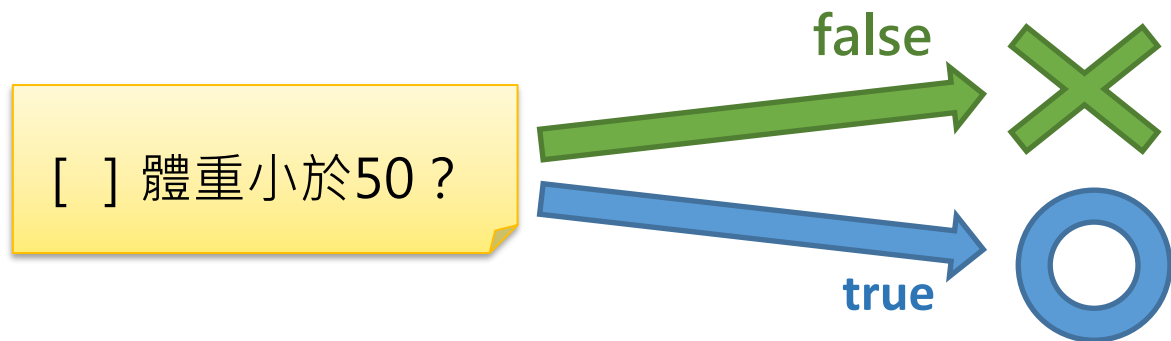
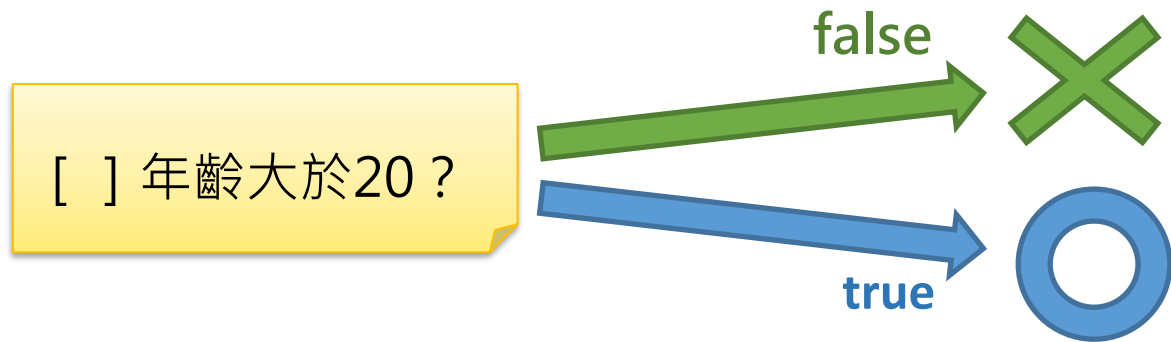
問題



判斷流程



一次比較無法決定結果



需要有可以將敘述組合處理的運算子！



&&: 所有敘述都為 true 才是 true

```
class web{  
    public static void main(String[] args) {  
        int age, weight;  
  
        age = 30;  
        weight = 40;  
        System.out.println(age > 20 && weight < 50);  
        age = 10;  
        weight = 40;  
        System.out.println(age > 20 && weight < 50);  
        age = 30;  
        weight = 60;  
        System.out.println(age > 20 && weight < 50);  
        age = 10;  
        weight = 60;  
        System.out.println(age > 20 && weight < 50);  
    }  
}
```



只有兩個敘述都符合時才會是 true，其他都會是 false

執行結果

true
false
false
false

||: 只要一個敘述是 true 就是 true

```
class web{  
    public static void main(String[] args) {  
        int age, weight;  
  
        age = 30;  
        weight = 40;  
        System.out.println(age > 20 || weight < 50);  
        age = 10;  
        weight = 40;  
        System.out.println(age > 20 || weight < 50);  
        age = 30;  
        weight = 60;  
        System.out.println(age > 20 || weight < 50);  
        age = 10;  
        weight = 60;  
        System.out.println(age > 20 || weight < 50);  
    }  
}
```



只要有一個敘述符合就會是 true，只有全部敘述都不符何時才會是 false

執行結果

true
true
true
false


```
class web{
    public static void main(String[] args) {
        int age, weight;
        age = 30;
        weight = 40;
        System.out.println("- age: 30 && weight: 40 -");
        System.out.println(age > 20 && weight < 50);
        System.out.println(!(age > 20 && weight < 50));
        System.out.println("- age: 30 || weight: 40 -");
        System.out.println(age > 20 || weight < 50);
        System.out.println(!(age > 20 || weight < 50));
        age = 10;
        weight = 40;
        System.out.println("- age: 10 && weight: 40 -");
        System.out.println(age > 20 && weight < 50);
        System.out.println(!(age > 20 && weight < 50));
        System.out.println("- age: 10 || weight: 40 -");
        System.out.println(age > 20 || weight < 50);
        System.out.println(!(age > 20 || weight < 50));
        age = 10;
        weight = 60;
        System.out.println("- age: 10 && weight: 60 -");
        System.out.println(age > 20 && weight < 50);
        System.out.println(!(age > 20 && weight < 50));
        System.out.println("- age: 10 || weight: 60 -");
        System.out.println(age > 20 || weight < 50);
        System.out.println(!(age > 20 || weight < 50));
    }
}
```



!: 不是 true 的才是 true

執行結果

```
- age: 30 && weight: 40
true
false
- age: 30 || weight: 40
true
false
- age: 10 && weight: 40 -
false
true
- age: 10 || weight: 40 -
true
false
- age: 10 && weight: 60 -
false
true
- age: 10 || weight: 60 -
false
true
```

! 就是將原本是 true 的改成 false，因此任何的敘述，不論是單一或是組合的敘述只要加上! 就會出現相反的結果。



一起比較吧

- 當需要多個一起比較時，就用 `&&` , `||` , `!` 把所有比較連起來
- `&&` , `||` , `!` 就稱為**邏輯運算子(Logical Operators)**

運算符號	意義	例子
<code>&&</code>	前後兩者都要符合才是 True	<code>3 > 5 && 5 < 10</code> False
		<code>3 < 5 && 5 < 10</code> True
		<code>3 > 5 && 5 > 10</code> False
<code> </code>	前後兩者只要有一個符合就是 True	<code>3 > 5 5 < 10</code> True
		<code>3 < 5 5 < 10</code> True
		<code>3 > 5 5 > 10</code> False
<code>!</code>	包住的條件不符合時就是 True	<code>!(3 > 5 && 5 < 10)</code> True
		<code>!(3 < 5 && 5 < 10)</code> False
		<code>!(3 > 5 && 5 > 10)</code> True

JAVA

延伸的概念

概念1：已知的運算子

類型	包含的符號
算數運算子 (Arithmetic Operators)	+ , - , * , / , %
指派運算子 (Assignment Operators)	+= , -= , *= , /= , %=
比較運算子 (Comparison Operators)	== , != , < , <= , > , >=
邏輯運算子 (Logical Operators)	&& , , !