

Java

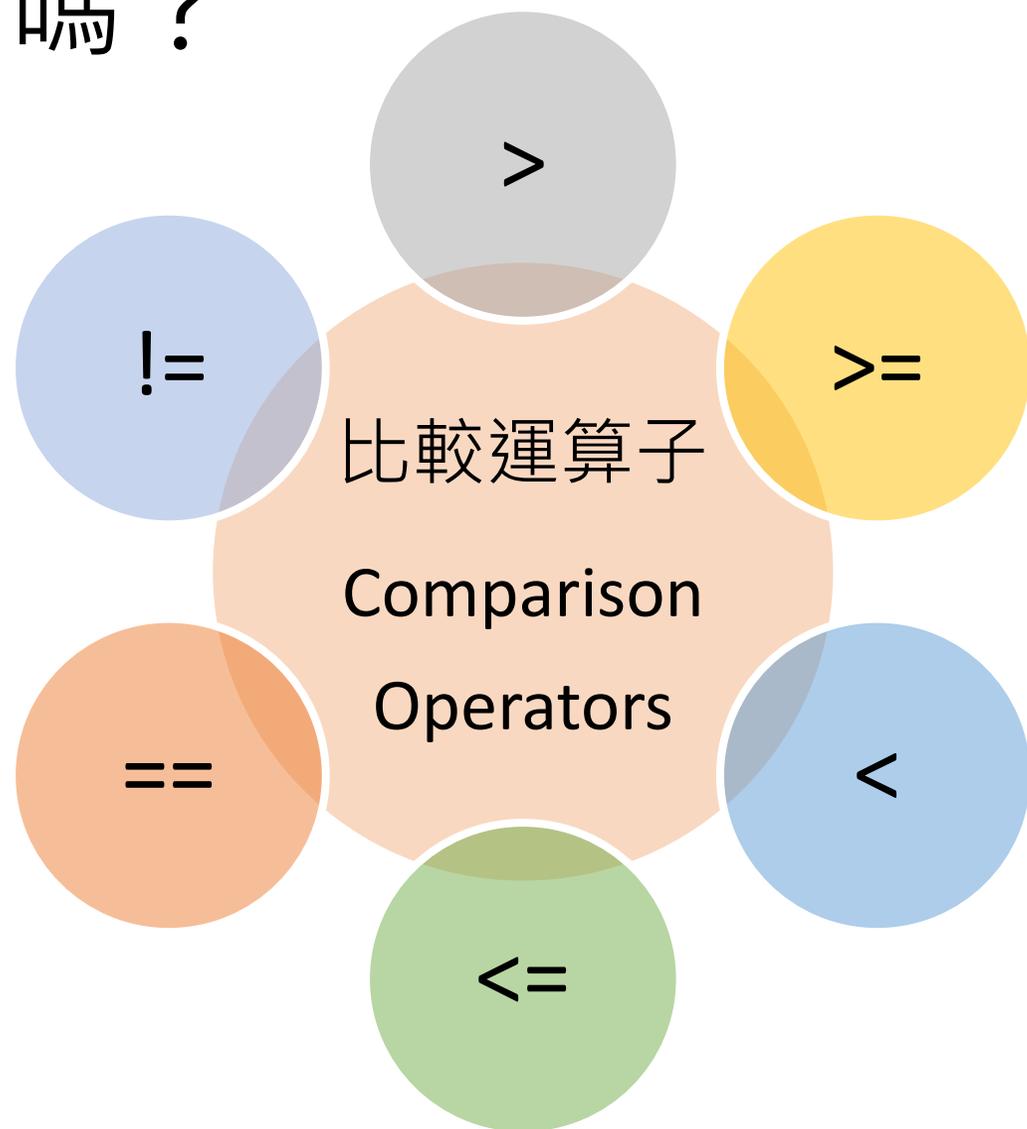
判斷式 if



較量

才剛開始呢

還記得這個嗎？

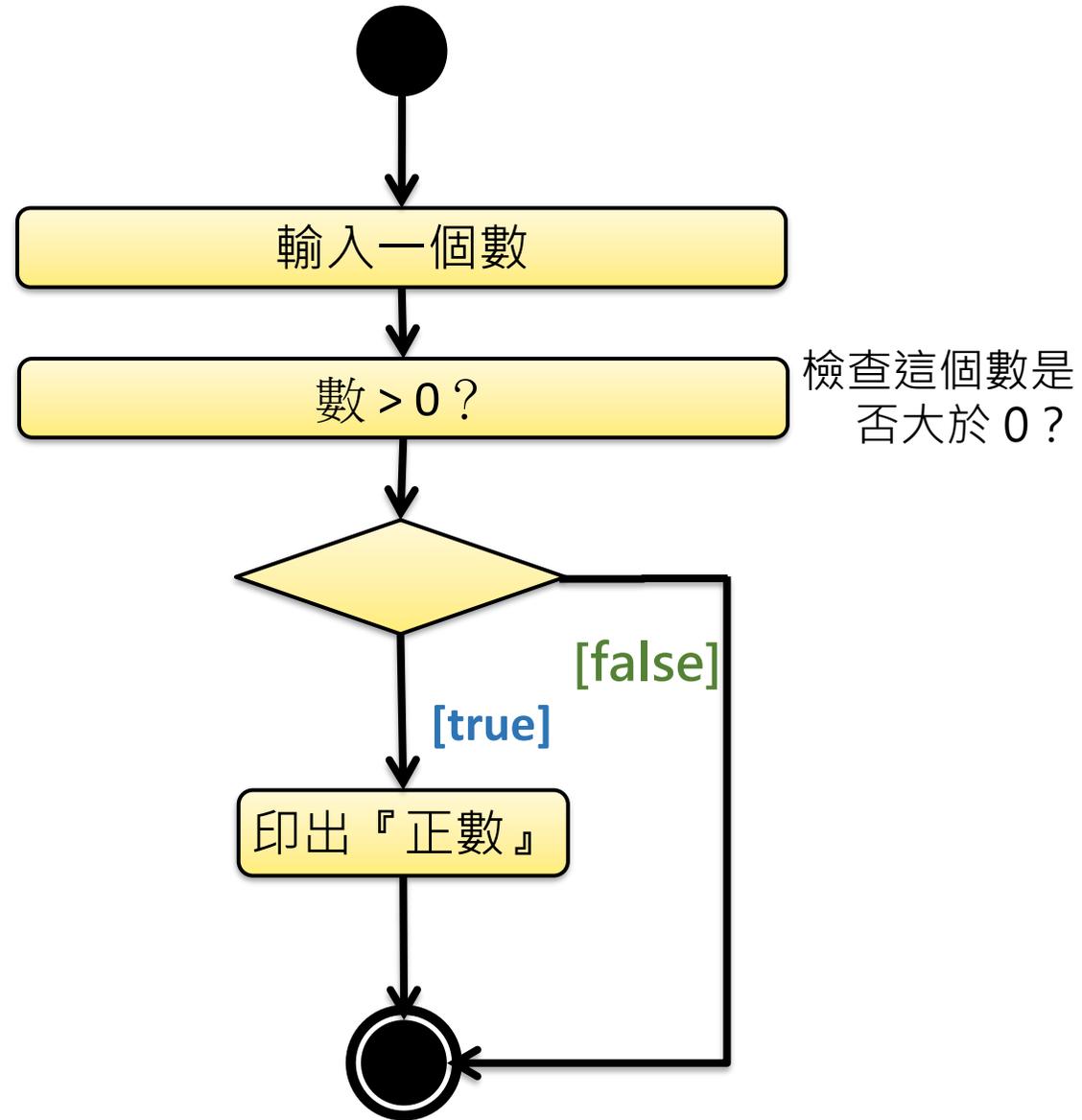


寫一個程式，
判斷一個數是正數嗎？

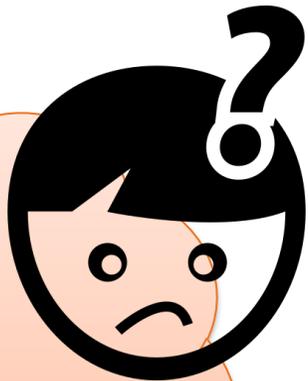
問題



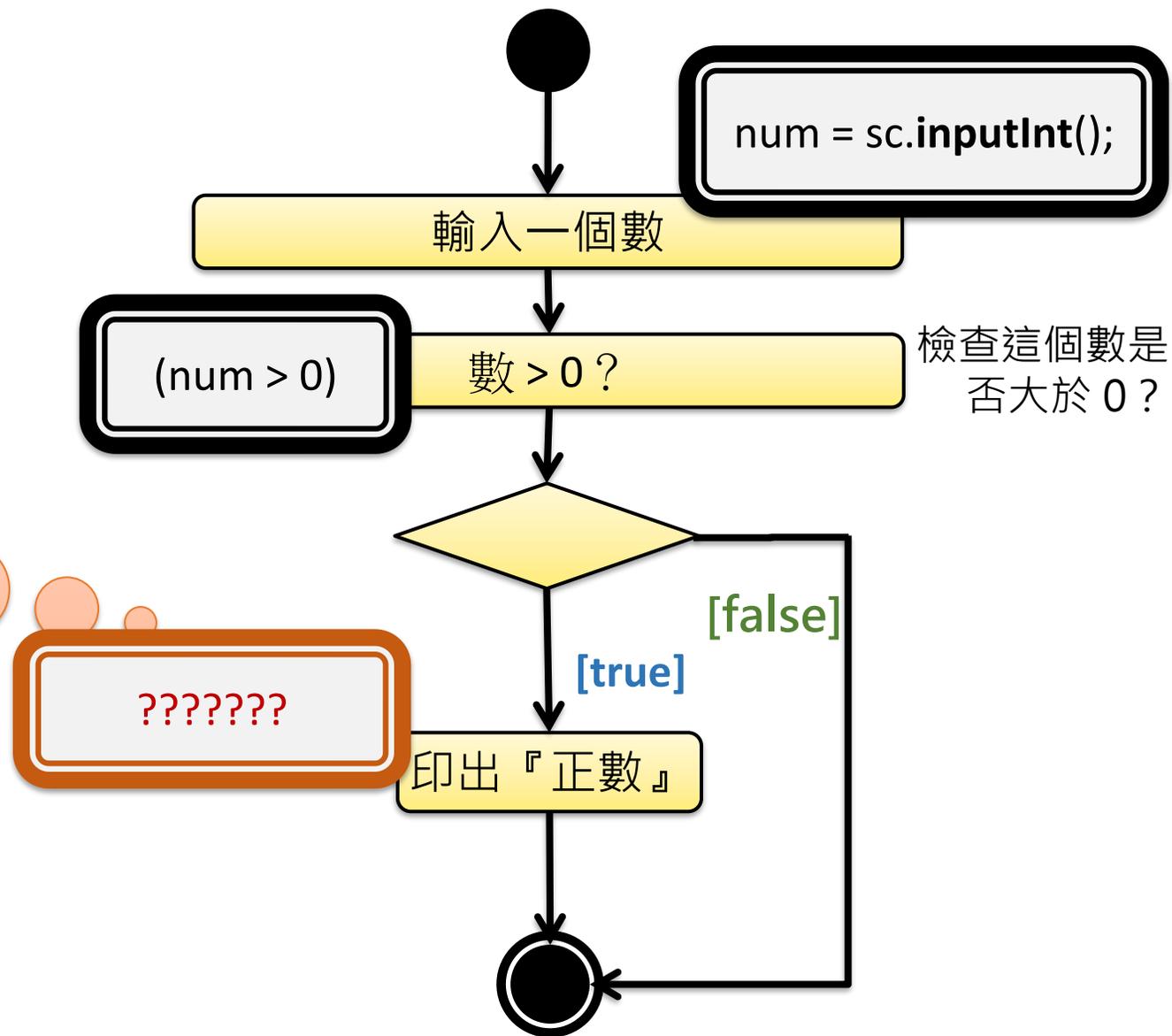
流程



流程 → 程式碼



(num > 0) 可以得到 true 或 false 的布林值結果，但如何依據這個結果執行對應的動作（印出正數）呢？



程式碼

```
import java.util.Scanner;
class web{
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int num;
        num = sc.nextInt();

        if (num > 0) {
            System.out.println("正數");
        }
    }
}
```

這是一個句子，
如果num>0印出 正數。

執行結果

20
正數



if 的十萬個為什麼？



JAVA

if 的十萬個為什麼？

- if 究竟是什麼？
- if 怎麼用？
- 條件判斷式是什麼？
- 條件判斷式與後續程式碼執行有什麼影響？
- if 條件式後面為什麼沒有分號反而有大括號？
- 大括號裡面的程式碼為什麼前面要多加空格？
- if 條件判斷如何執行？

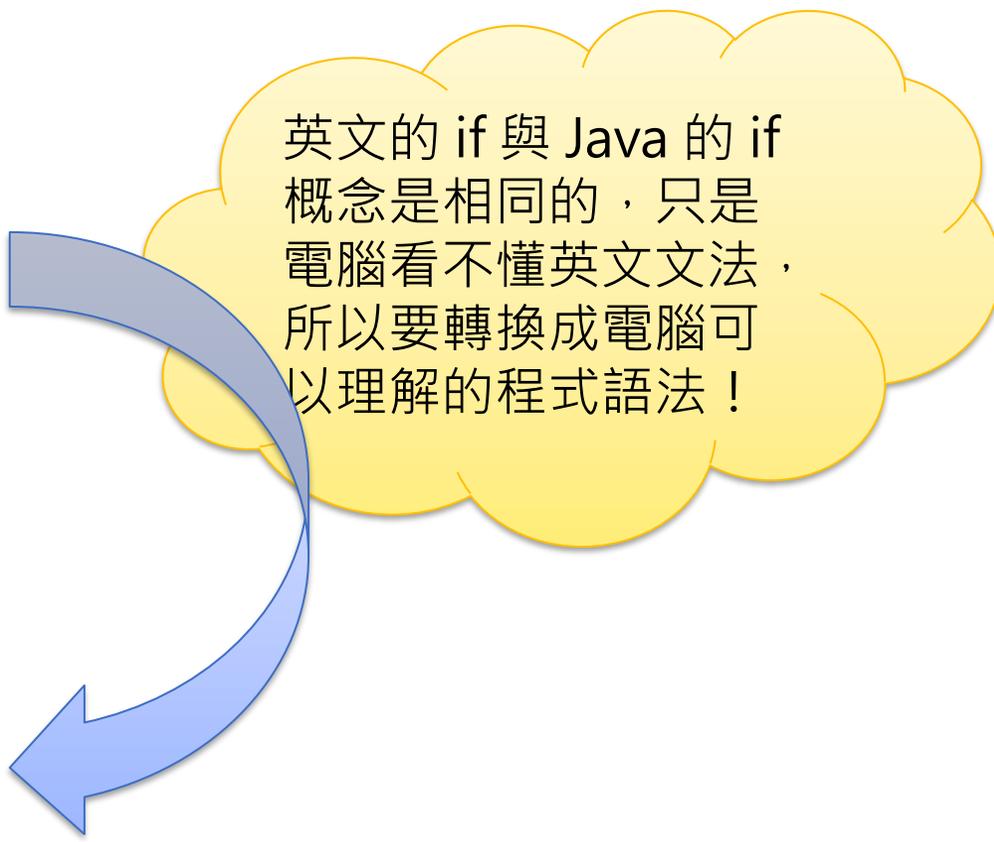


[if 的十萬個為什麼？]

if 究竟是什麼？

- 英文的 if 是用來表達假設語法
 - If the sun shines all day, it gets hot.
- Java 的 if 是用來處理條件判斷

```
if (the sun shines all day) {  
    System.out.println("Hot!");  
}
```



英文的 if 與 Java 的 if 概念是相同的，只是電腦看不懂英文文法，所以要轉換成電腦可以理解的程式語法！

「if 的十萬個為什麼？」

if 怎麼用？

- Java 程式有基本 **框架**

```
class web {  
    public static void main(String[] args) {  
        → 程式碼寫在這 ←  
    }  
}
```



{if 的十萬個為什麼?}

if 怎麼用?

- **if** 條件判斷式也有一個 **框架**

```
if (條件判斷式) {  
    → 判斷符合時要執行的程式碼 ←  
}
```



條件判斷式是什麼？

- 可以根據 **true / false** 執行對應的指令，稱為條件判斷。
- 一個執行(運算)後可以得到 true/false 結果的式子就稱為條件判斷式。

條件判斷式

```
if ( num > 0 ) {  
    System.out.println("正數");  
}
```



if 的十萬個為什麼？

條件判斷式是什麼？

- Java 使用 **if** 實作**條件判斷**。
 - if 是一個關鍵字，看到 if 就知道要處理條件判斷。
 - 就像看到 System.out.println 就知道要輸出資料一樣。

if 關鍵字

```
if (num > 0) {  
    System.out.println("正數");  
}
```



if 的十萬個為什麼？

條件判斷式與後續程式碼執行有什麼影響？

```
import java.util.Scanner;
class web {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int num;
        num = sc.nextInt();

        if (num > 0) {
            System.out.printf("輸入正數: %d\n", num);
        }

        System.out.printf("num: %d\n", num);
        System.out.println("執行結束！");
    }
}
```



```
-2
num: -2
執行結束！
```

不論 num > 0 是 true 還是 false，判斷式後面 (紅框內) 的程式碼都一定會執行的！



```
20
輸入正數: 21
num: 21
執行結束！
```

if 的十萬個為什麼？

條件判斷式與後續程式碼執行有什麼影響？

```
import java.util.Scanner;
class web {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int num;
        num = sc.nextInt();

        if (num > 0) {
            num = num + 1;
            System.out.printf("輸入正數: %d\n", num);
        }

        System.out.printf("num: %d\n", num);
        System.out.println("執行結束！");
    }
}
```

影響 num
變數值



num > 0 是 false

```
-2
num: -2
執行結束！
```

雖然 num > 0 是 true 還是 false 不會影響判斷式後面的程式碼執行與否，但如果判斷式內有更改到資料變數(num)，就會影響執行結果！

num > 0 是 true

```
20
輸入正數: 21
num: 21
執行結束！
```

if 的十萬個為什麼？

if 條件式後面為什麼沒有分號反而有大括號？

```
if ( num > 0 ) {  
    System.out.println("正數");  
}
```

```
class web{  
    public static void main(String[] args) {  
        → 程式碼寫在這 ←  
    }  
}
```

if 後面接著的程式碼與
if 是一體的

當前後程式碼是一體的時候，
或是某程式碼前面一定要是某
一個指定的程式碼時，那就不
需要分號，反而會用大括號標
示出那些程式碼是一體的。

main 後面接著的程式碼與
main 是一體的

大括號裡面的程式碼為什麼前面要多加空格？

- 程式碼前面的空格稱為『**縮排**』，是為了保持程式碼的可讀性。
 - 有縮排的程式碼可以讓人在查看程式的時候立即明白那些程式碼是一體的，或是那些程式碼的執行是有相互影響關係的。
 - 『**縮排**』是給人看的，是否有縮排對電腦解讀程式碼是無關的！

if 的十萬個為什麼？

大括號裡面的程式碼為什麼前面要多加空格？

有縮排的程式碼

```
class web{
    public static void main(String[] args) {
        int num;
        num = 20;
        if (num > 0) {
            System.out.printf("%d 是正數\n", num);
        }
        num = -1;
        if (num > 0) {
            System.out.printf("%d 是正數\n", num);
        }
        System.out.println("執行結束！");
    }
}
```

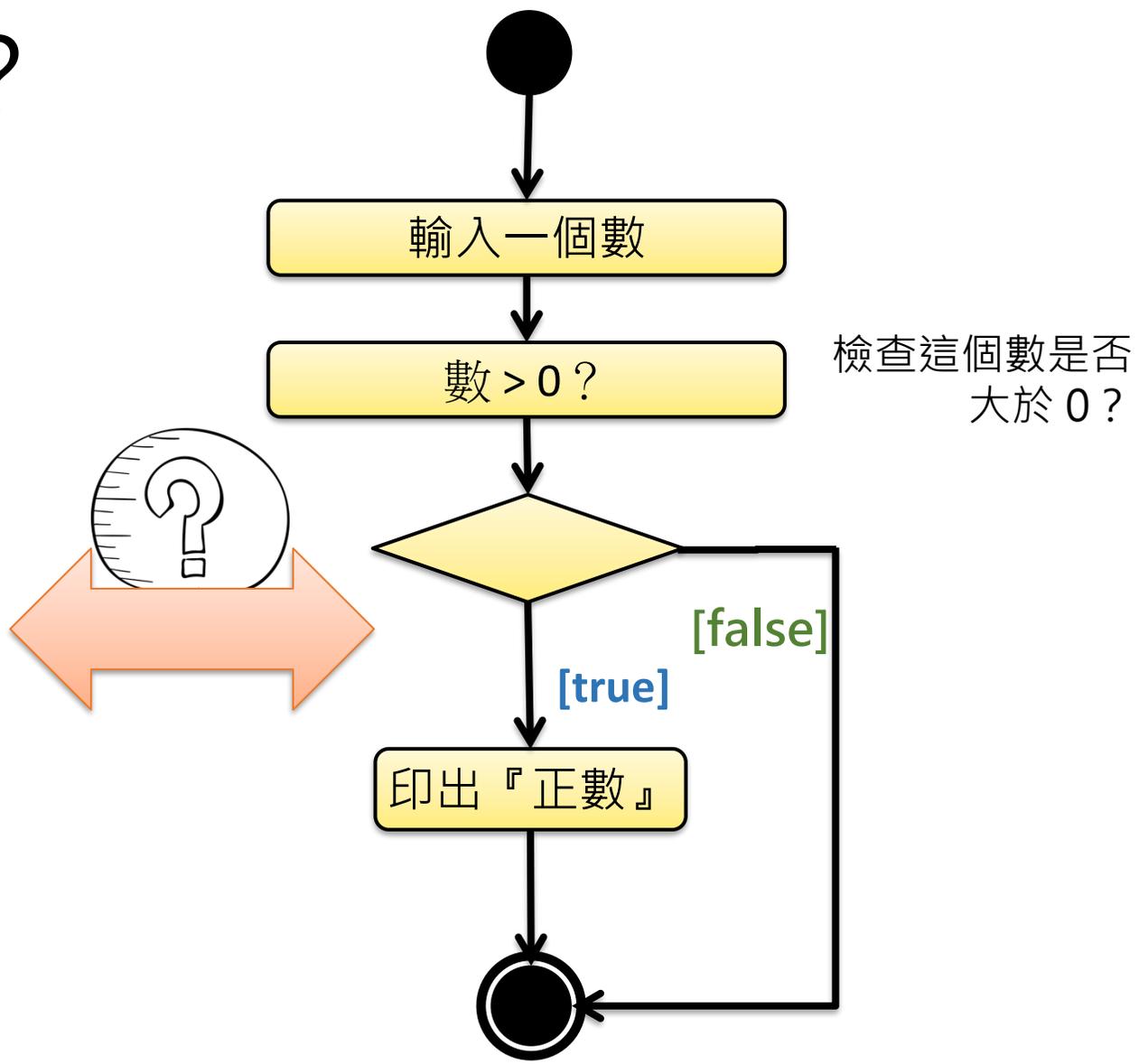
沒縮排的程式碼

```
class web{
public static void main(String[] args) {
int num;
num = 20;
if (num > 0) {
System.out.printf("%d 是正數\n", num);
}
num = -1;
if (num > 0) {
System.out.printf("%d 是正數\n", num);
}
System.out.println("執行結束！");
}
}
```

{if 的十萬個為什麼?}

if 條件判斷如何執行?

```
if (條件判斷式) {  
    → 判斷符合時要執行的程式碼 ←  
}
```



「if 的十萬個為什麼？」

當 **if** 條件判斷式是 **true** 的時候

當 **if** 條件式符合時，
就會執行紅框內的程式



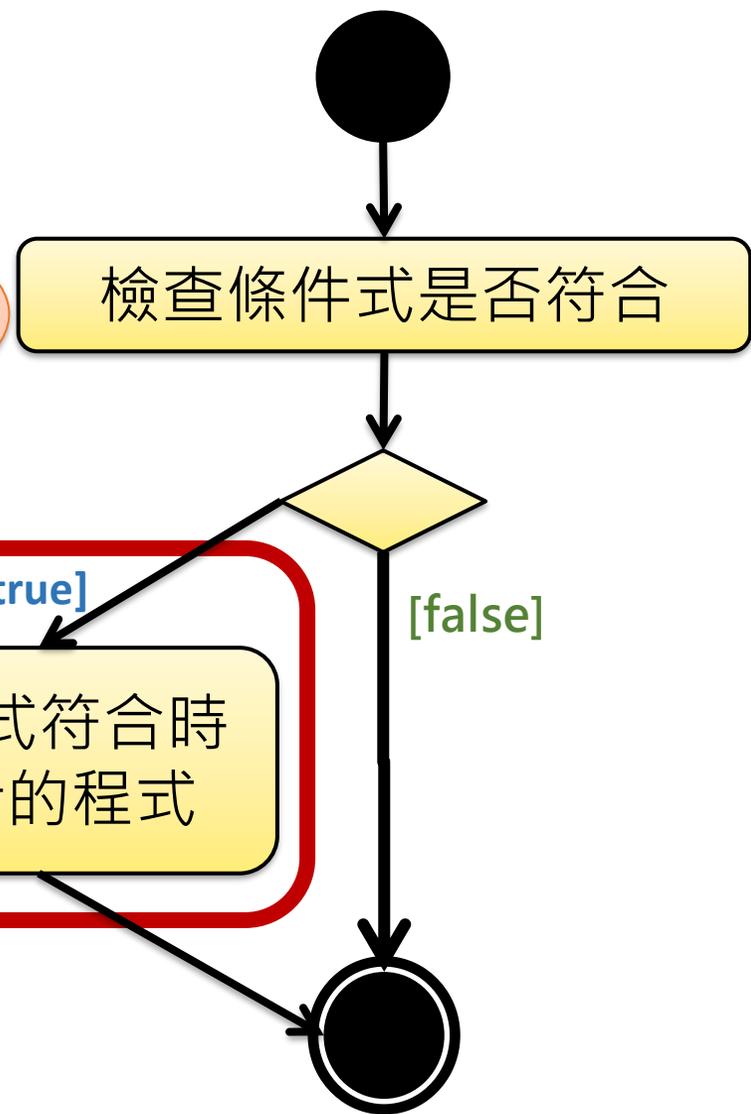
if 條件式

true

```
if (條件判斷式) {
```

→ 判斷符合時要執行的程式碼 ←

```
}
```



[true]

[false]

條件式符合時
執行的程式

[if 的十萬個為什麼?]

當 if 條件判斷式是 false 的時候

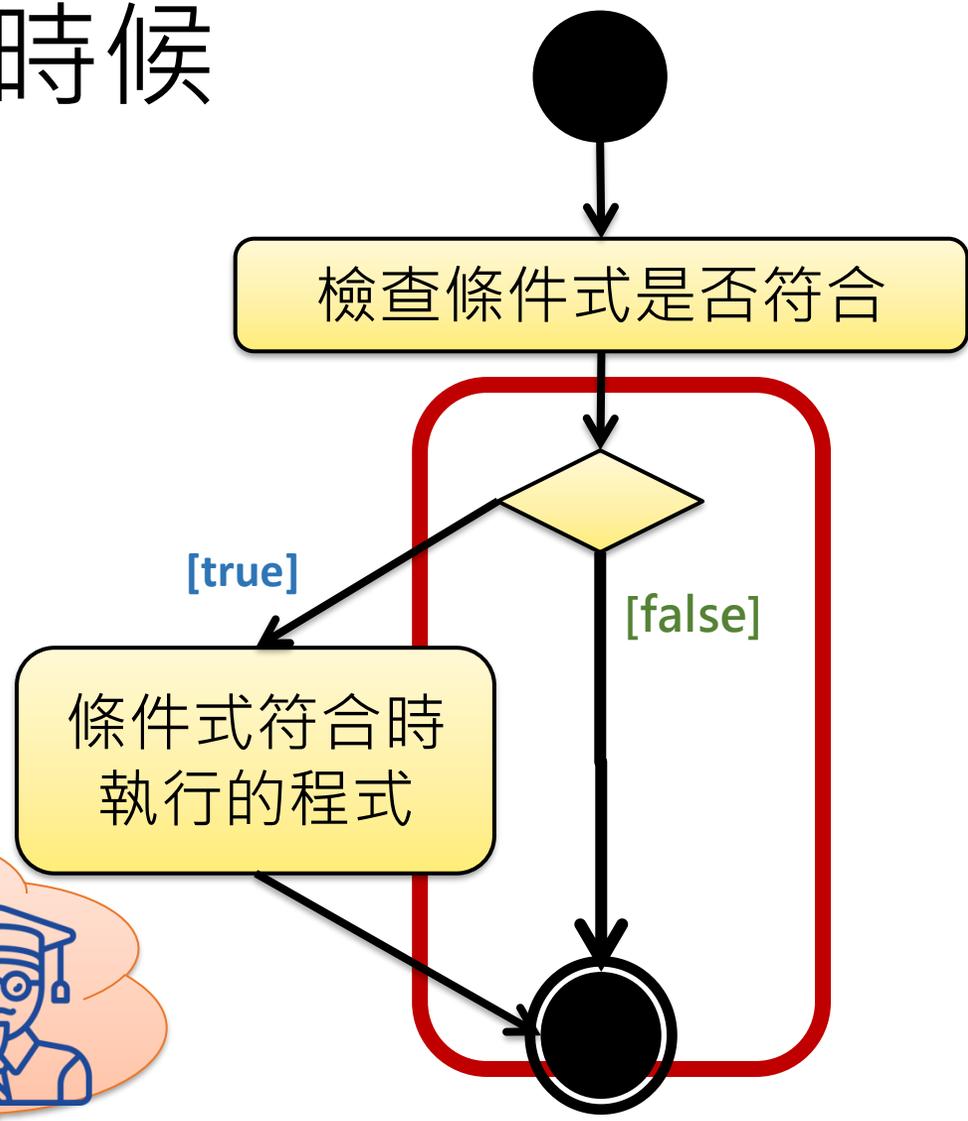
if 條件式

```
if (條件判斷式) {  
    → 判斷符合時要執行的程式碼 ←  
}  
與條件判斷無關的後續程式碼
```

false



當 if 條件式不符合時，
就會不執行紅框內的程式碼，
直接執行大括號後面的程式



JAVA

延伸的概念

概念1：if 條件判斷式的本質: true/false

```
class web {  
    public static void main(String[] args) {  
        int num;  
        num = 20;  
        if (num > 0) {  
            System.out.printf("%d 是正數\n", num);  
        }  
        if (true) {  
            System.out.println("is true!");  
        }  
        if (false) {  
            System.out.println("is false!");  
        }  
        System.out.println("執行結束!");  
    }  
}
```



- if 是根據後面的布林(Boolean)值決定是否執行。
 - if 後面若是 True，就會執行接著的程式
 - if 後面若是 False，就不會執行
- 但是，True 跟 False 都是完全確定的值，完全不需要使用 if 的假設判斷邏輯。
- 所以，一般使用時 if 後面接的會是**條件式**，當條件符合(True)時就執行，條件不符合(False)就不執行。

概念1：if 條件判斷式的本質: true/false

```
class web {  
    public static void main(String[] args) {  
        int num;  
        num = 20;  
        if (num > 0) {  
            System.out.printf("%d 是正數\n", num);  
        }  
        if (true) {  
            System.out.println("is true!");  
        }  
        if (false) {  
            System.out.println("is false!");  
        }  
        System.out.println("執行結束!");  
    }  
}
```



```
class web {  
    public static void main(String[] args) {  
        int num;  
        num = 20;  
        if (num > 0) {  
            System.out.printf("%d 是正數\n", num);  
        }  
        System.out.println("is true!");  
        System.out.println("執行結束!");  
    }  
}
```

既然 True 跟 False 都已經是完全確定的值，那就沒有必要使用 if，就可以直接簡化程式阿！

概念2：條件運算式的組合

符號	數學意義	比較運算子
=	等於	==
≠	不等於	!=
<	小於	<
≤	小於等於	<=
>	大於	>
≥	大於等於	>=



運算符號	意義	例子	
&&	前後兩者都要符合才是 True	3 > 5 && 5 < 10	False
		3 < 5 && 5 < 1	True
		3 > 5 && 5 > 10	False
	前後兩者只要有一個符合就是 True	3 > 5 5 < 10	True
		3 < 5 5 < 10	True
		3 > 5 5 > 10	False
!	包住的條件不符合時就是 True	!(3 > 5 && 5 < 10)	True
		!(3 < 5 && 5 < 10)	False
		!(3 > 5 && 5 > 10)	True

只要可以得到 true / false 結果的運算式都可以作為 if 的條件判斷式，不會只侷限單一的比較運算式喔！

概念2：偷懶的寫法

```
class web {  
    public static void main(String[] args) {  
        int num;  
        num = 20;  
        if (num > 0) {  
            System.out.printf("%d 是正數\n", num);  
        }  
    }  
}
```

```
class web{  
    public static void main(String[] args) {  
        int num;  
        num = 20;  
        if (num > 0) System.out.printf("%d 是正數\n", num);  
    }  
}
```

當 if 符合時要執行的程式碼只有一行時，可直接放到 if 後面，並省略大括號，但如縮排的作用一樣，為了維持程式碼的可讀性，不建議使用此寫法喔！