

一、單向判斷式

1. 是 if 指令中最簡單的型態，語法為：

```
if (條件句):           #「()」括號亦可省略不寫
    程式區塊           #當「條件句」為 True 時，執行該程式區塊
```

例如：

```
if (score>=60):
    grade="及格"      #條件句為 True 時，才會執行
    sum = sum + score #條件句為 True 時，才會執行
print(score)         #下一列程式，不在判斷式的範圍裡，所以無論如何都會執行
```

2. 如果程式區塊只有一列程式碼時，則可以合併成一行，語法為：

```
if (條件句):  程式區塊
```

例如：

```
if (score>=60):  grade="及格"  #條件句為 True 時，才會執行
sum = sum + score #下一列程式，不在判斷式的範圍裡，所以無論如何都會執行
```

二、雙向判斷式

1. 「if...else...」為雙向判斷式，語法為：

```
if (條件句):           #「()」括號亦可省略不寫
    程式區塊一         #當「條件句」為 True 時，執行該程式區塊
else:
    程式區塊二         #當「條件句」為 False 時，執行該程式區塊
```

例如：

```
if (score>=60):       #「()」括號亦可省略不寫
    grade="及格"      #當「條件句」為 True 時，執行該程式區塊
else:
    grade="不及格"    #當「條件句」為 False 時，執行該程式區塊
```

三、多向判斷式

1. 「if...elif...else...」為雙向判斷式，語法為：

```
if (條件句一):
    程式區塊一         #當「條件句一」為 True 時，執行該程式區塊
elif (條件句二):
    程式區塊二         #當「條件句二」為 True 時，執行該程式區塊
elif (條件句三):
    程式區塊三         #當「條件句三」為 True 時，執行該程式區塊
. . . . .
else:
```

程式區塊 else

#當以上皆非時，執行該程式區塊

例如：

```
if (score>=60) :  
    grade="及格"           #當「條件句一」為 True 時，執行該程式區塊  
elif (score>=40) :  
    grade="可補考"       #當「條件句二」為 True 時，執行該程式區塊  
else :  
    grade="死當"         #當以上皆非時，執行該程式區塊
```

四、巢狀判斷式

1. 判斷式之內可以再包含判斷式，例如：

```
if (customer=="會員") :  
    if (money>=10000) :  
        discount="打 5 折"  
    elif (money>=5000) :  
        discount="打 7 折"  
    else :  
        discount="打 9 折"  
else :  
    discount="未打折"
```