

程式的流程控制-條件判斷

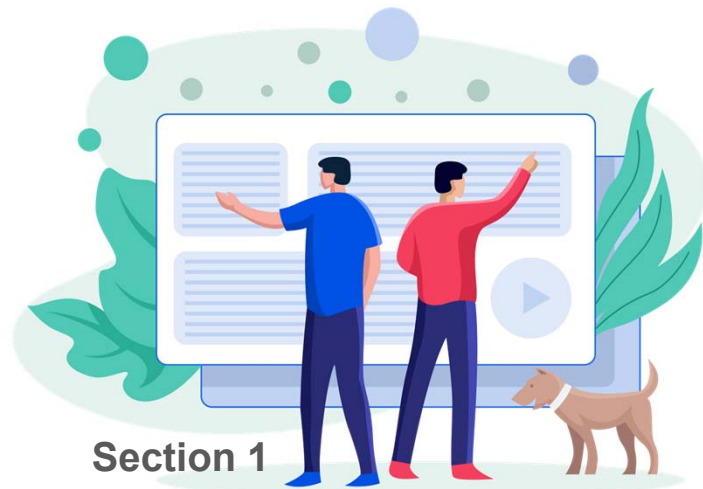
學習目標

- 詳細說明 **Python** 程式設計風格，縮排的重要性
- 條件判斷式 **if** 的用法
- 條件運算式
- 本章節將會學到Python的Coding style
- 從本章節開始，將開始進入到Python的程式設計領域(.py腳本)。
- Python是一個互動式的腳本語言。他可以透過Jupyter來操作，等同於操作套裝軟體(Software)；或是也可以寫成腳本(.py)，等同於開發程式(Program)



Python程式撰寫

- 介紹 Python 程式的風格與特性
- pass 敘述與程式碼區塊



Python的敘述

■ Python 是腳本語言(Scripting) , 一系列敘述

- `x=3`
`y=7`
`print('Result:',x+y)`

■ 一系列多個敘述

- 分號(;) 允許在一系列有多個敘述
- 以分號隔開 , 視為一個敘述 , 不需要縮排
- `x=3; y=7; print('Result:',x+y)`

■ pass 敘述 , 代表空敘述 , 不做任何運算(do nothing)

多列的敘述

■ Multi-Line Statements

■ Python 一行是一個敘述(Statement)

- 可使用 “\” 來接續下一列。使用反斜線Backslash(\)就可以讓Python程式將多行視為一個程式敘述(Statement)。下一行要不要往內縮或對其都是無所謂的，即接續的下一行跟縮排毫無關係。

- `total = score1 + \`
`score2 + \`
`score3`

- 在 [], {}, or () 內的敘述不需要 “\” 就可以接續下列

- `days = ['Sun', 'Mon', 'Tue', 'Wed',`
`'Thu', 'Fri', 'Sat']`



註解(Comments)

- # 代表註解 (必須不在字串內)，所有 # 後的整列視為註解，Python 直譯器會忽略它們，習慣上 # 後面空一格，再寫註解

- # First comment

- ```
print("Hello, Python!") # second comment
```

- Python 並沒有多列註解的功能，必須每一列分開註解

- # This is a comment

- ```
# This is a comment, too
```

- ```
This is a comment, too
```

- 空白列

- 只包含空格的列，Python 會完全忽略，可以當作程式碼的分隔，易於閱讀

# 縮排 -1

- Python 沒有提供大括號來表達一組敘述的功能
- Python 使用縮排的方式來表達一組敘述，Python 不建議使用 [TAB] 鍵縮排 (Indentation)
- PEP8規定：Use 4 spaces per indentation level.
- PEP8規定：Spaces are the preferred indentation method.
- 縮排的空格數可以不同，但是同一組敘述的空格數必須一致
  - 相同空格數的所有敘述組合成一個區塊
  - Python 建議使用 4 個空格來表達一個區塊
- 某一行以關鍵字開始，以 “:” 結尾，下一行就要縮排以表示程式碼區塊
- 不可以沒有理由縮排，必須是冒號(:)的下一行，才可以使用縮排

## 縮排 -2

### ■ 正確語法

```
□ if True:
 print('True')
else:
 print('False')
```

### ■ 以下區塊會產生錯誤

```
□ if True:
 print('Answer')
 print('True')
else:
 print('Answer')
 print('False')
```

■ Python 不是自由格式(Free format) 的語言，不能隨便空格

■ 縮排的空格數可以不同，但是同一組敘述的空格數必須一致

□ 相同空格數的所有敘述組合成一個區塊

□ Python 建議使用 4 個空格來表達一個區塊



# Python程式撰寫(Demo)

- 如何使用 IDLE 撰寫 Python 程式
- 如何執行 Python 程式



# 流程控制條件判斷

## ■ 使用條件判斷敘述

- if, if...else, if...elif...else

- Nested if (巢狀式的if，即if結構裡面含有另一個if結構)



# 條件判斷式

## ■ if statement

- if 敘述包含一個結果為 Boolean 的運算，伴隨一個程式碼區塊(要有:以及下一行內縮)

## ■ if...else statement

- 如果 if 敘述的判斷式為真，則執行 if 後面的敘述，否則執行 else 後面的敘述

## ■ Nested if statement

- if 敘述內可以再包含 if 敘述，形成巢狀的 if 敘述

## ■ Python 的 False 如下：

- False、None、0、0.0
- 空字串 ""、空串列 []、空元組 ()
- 空字典 {}、空集合 set()

## ■ 除此以外，都視為 True

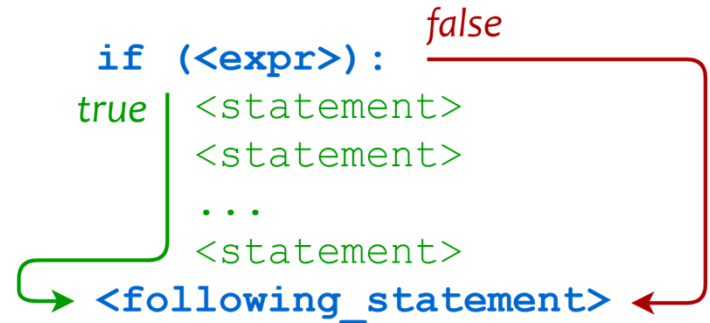
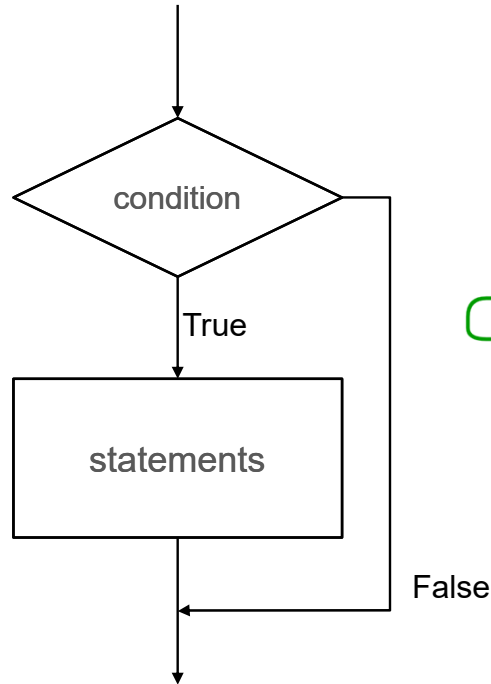
# if 敘述

## ■ 語法

- if expression:  
statement(s)

## ■ 範例

- var1 = 100  
if var1:  
    print('True expr 1')  
    print(var1)  
var2 = 0  
if var2:  
    print('True expr 2')  
    print(var2)  
print('Finished!')



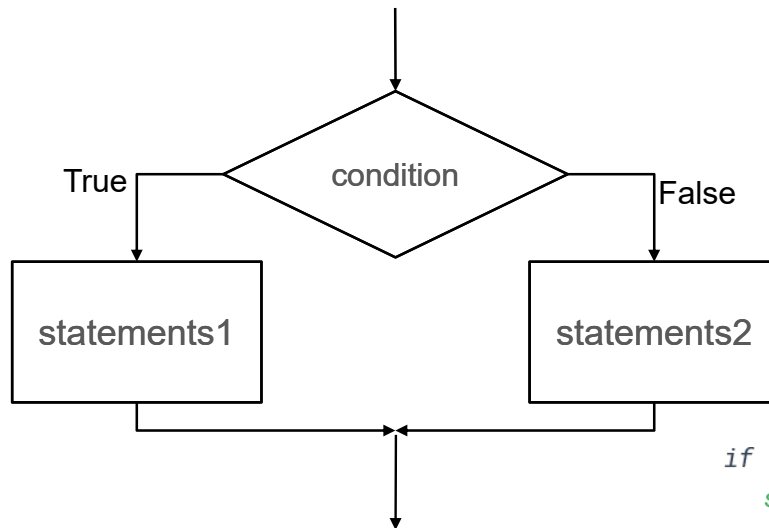
# if...else 敘述

## ■ 語法

```
if expression:
 statement(s)
else:
 statement(s)
```

## ■ 範例

```
if amt < 1000:
 discount = amt * 0.05
 print('Discount',discount)
else:
 discount = amt * 0.10
 print('Discount',discount)
```



```
if condition:
 statement
 statement
 ...
else:
 statement
 statement
 ...
following_statement
```

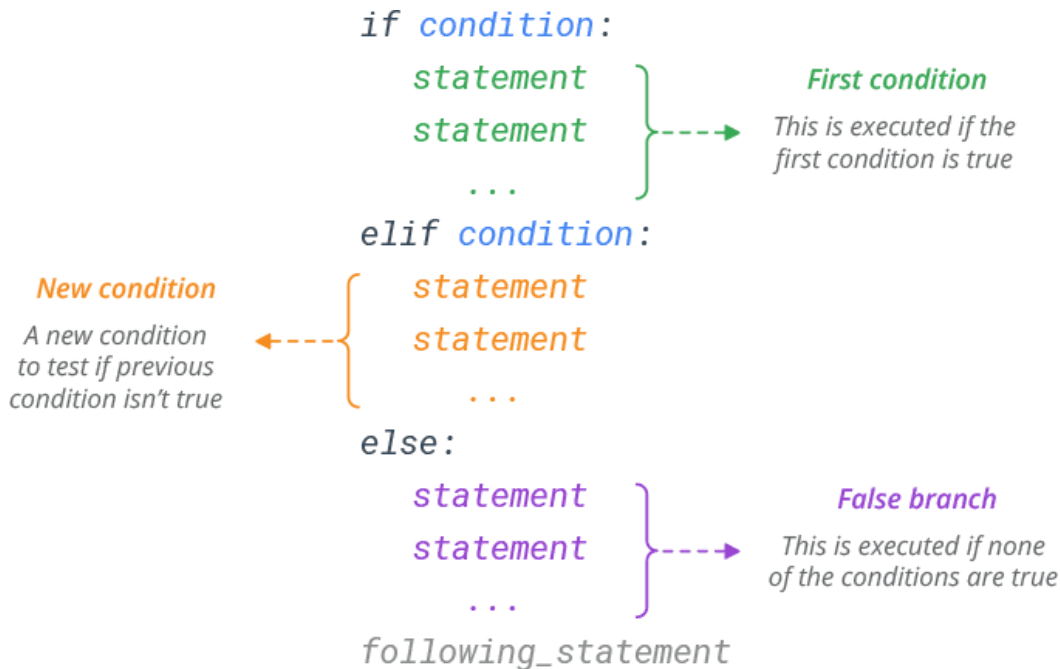
*True branch*  
This is executed if the condition is true

*False branch*  
This is executed if the condition is false

# if...elif...else 敘述 -1

## ■ 語法

```
if expression1:
 statement(s)
elif expression2:
 statement(s)
elif expression3:
 statement(s)
else:
 statement(s)
```



# if...elif...else 敘述 -2

## ■ 範例

```
□ amt = int(input('Enter amount: '))
```

```
if amt < 1000:
```

```
 discount = amt * 0.05
```

```
 print('Discount',discount)
```

```
elif amt < 5000:
```

```
 discount = amt * 0.10
```

```
 print('Discount',discount)
```

```
else:
```

```
 discount = amt * 0.15
```

```
 print('Discount',discount)
```

```
print('Net price:',amt - discount)
```

# Nested if 敘述 -1

## ■ 語法

```
□ if expression1:
 statement(s)
 if expression2:
 statement(s)
 elif expression3:
 statement(s)
 else:
 statement(s)
elif expression4:
 statement(s)
else:
 statement(s)
```



# Nested if 敘述 -2

## ■ 範例

```
num = int(input('enter number'))
if num%2 == 0:
 if num%3 == 0:
 print('Divisible by 3 and 2')
 else:
 print('Divisible by 2, not divisible by 3')
else:
 if num%3 == 0:
 print('Divisible by 3, not divisible by 2')
 else:
 print('Not Divisible by 2, not divisible by 3')
```

# 單一敘述套件

- Single Statement Suites

- 假如 if 子句只包含一個敘述，則可以直接寫在一行

- 範例

- `var = 50`  
# Only one statement  
`if var==50: print('Value is 50')`  
`print('Finished!')`

# 條件運算式

## ■使用條件運算式



# 條件運算式

- Conditional expressions
- 條件運算又稱為三元運算子(ternary operator)
- 是所有Python運算子中優先順序最低的
- 語法
  - `x if C else y`
  - 先評估C的條件，如果 C 的結果為真則得到 x，否則結果為 y

# 流程控制條件判斷(Demo)

- 如何使用 if - else 敘述
- 如何使用條件運算式



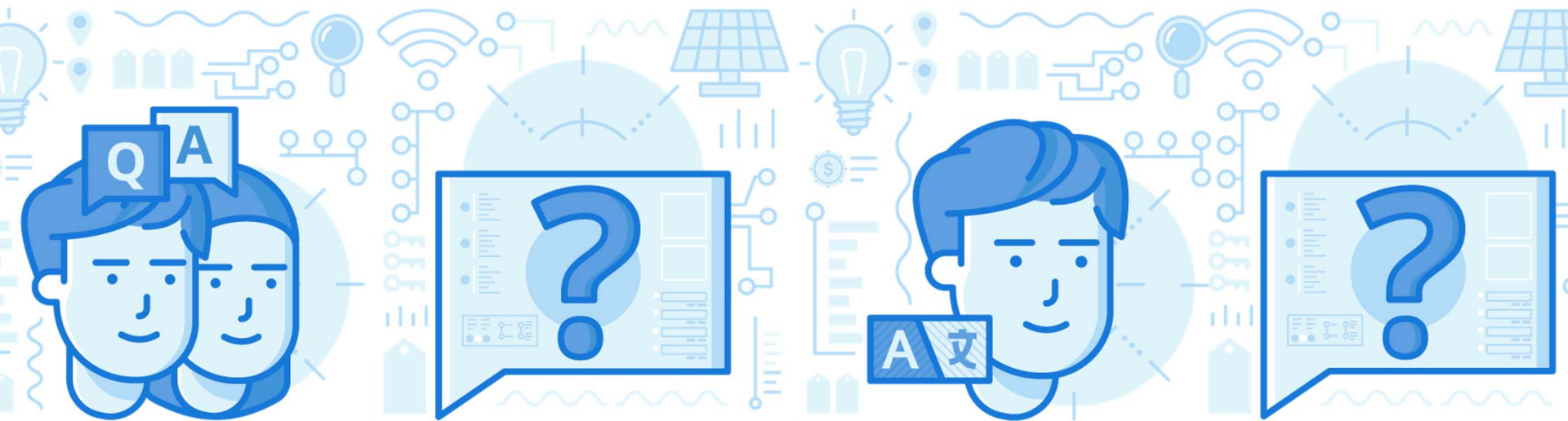
# 本章重點精華回顧

- Python 腳本程式的特性
- 流程控制 if 敘述
- 條件運算式



# Lab: 程式的流程控制條件判斷

- Lab01: 撰寫 Python 程式
- Lab02: 使用 if, elif, else



# Lab01: 撰寫 Python 程式

1. 啟動Python IDLE環境，做以下練習

2. 使用 “File/New File” 開啟新視窗

3. 輸入以下程式：

```
ex7_1.py
name = input('Your name please: ')
print('Hello',name)
print('Nice to meet you!')
```

4. 使用 “File/Save...” 將程式存成 “ex7\_1.py”

5. 使用 “Run/Run module” 執行程式，程式會在 “Python Shell” 視窗執行

6. 輸入資料，然後按 “Enter”，觀察程式執行結果

7. 關閉ex7\_1.py程式視窗



# Lab02: 使用 if, elif, else

1. 啟動Python IDLE環境，做以下練習
2. 使用 “File/Open...” 開啟 “ifstmt1.py” 程式，了解if敘述的用法

```
if stmt

s = int(input('Input score: '))

if s < 60:
 print('Put more effort!')

print('Exam socre is',s)
```

3. 關閉ifstmt1.py程式視窗

# Lab02: 使用 if, elif, else

4. 使用 “File/Open...” 開啟 “ifstmt2.py” 程式，了解if-else敘述的用法

```
if-else stmt
s = int(input('Input score: '))
if s >= 60:
 print("Pass! ")
else:
 print("Not pass! ")
```

5. 關閉ifstmt2.py程式視窗

# Lab02: 使用 if, elif, else

6. 使用 “File/Open...” 開啟 “ifstmt3.py” 程式，了解if-elif-else敘述的用法

```
if-elif-else stmt
age = int(input('Input age: '))
if age < 13:
 print('Child')
elif age < 18:
 print('Youth')
else:
 print('Adult')
```

7. 關閉ifstmt3.py程式視窗

# Lab02: 使用 if, elif, else

8. 使用 “File/Open...” 開啟 “ifstmt4.py” 程式，了解conditional expression的用法

```
conditional expression
```

```
h = int(input('Enter hour(0-23): '))
```

```
h = h%24 if h>=24 else h
```

```
print('Current time = {}:00'.format(h))
```

9. 關閉ifstmt4.py程式視窗