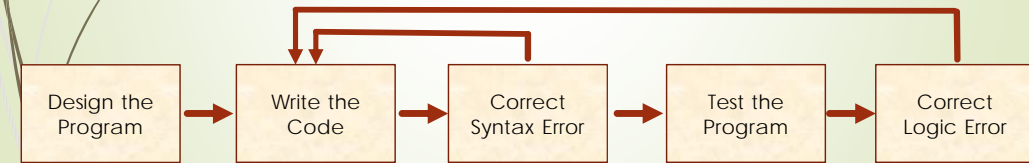


Review

The Program Development Cycle

1. **Design** the Program (程式設計)
2. **Write/Change** the **Code** (程式撰寫)
3. **Correct Syntax** Error (更正語法錯誤)
4. **Test** the Program (程式測試)
5. **Correct Logic/Semantic** Error (更正邏輯/語意錯誤)



Review Topics

1. Install Python 3.5.1 (IDLE, Enlarge Font)
2. Arithmetic Expression (Comment)
3. Boolean Expression (comparison operator)
4. String Expression (expression, print)
5. Program Design (input, type conversion)
6. Assignment
7. If

Syntax Error

所謂syntax error說的是在給電腦指令的表達上不符合程式在語法上的要求。

例一: `print 1`則給予error message。

```
>>> print(1)
1
>>> print 1
SyntaxError: Missing parentheses in call to 'print'
>>> |
```

例二: `'3'*3`會產生無法evaluate的錯誤

```
>>> 3*3
'333'
>>> '3'*3
'333'
>>> '3'*'3'
Traceback (most recent call last):
  File "<pyshell#9>", line 1, in <module>
    '3'*'3'
TypeError: can't multiply sequence by non-int of type 'str'
>>> |
```

Logic Error

- 所謂logic error是說雖然你的syntax都是正確的，但是邏輯上是不正確的。例如說請你計算6是否是偶數，結果計算結果說6不是偶數，就表示你的程式執行的結果是錯誤的。

Flow Control (via Statement)

Python使用Statement來讓程式運轉，包括：

1. input statement
2. print statement (print function)
3. assignment statement
4. if statement
5. for loop statement
6. while loop statement
7. return
8. break
9. continue
10. raise
11. try
12. except
13.

學習步驟

	語言(English)	程式語言(Python)
Lexical 語彙	單字、片語 Ex. Cat、clothes、phone	語言中的token Ex. int、bool、str、if、.(Reserve words) 1、2、3.14..... (numbers)
Syntax 語法	S+V+O 主詞+動詞+受詞 Ex. I like you. (o) I favorite you. (x)	If <condition>: True Blocks Ex. if a>0: if a>0 a = a - 1 a = a - 1 (o) (x)
Semantic 語意	邏輯上的正確性 Ex. Cat eats mouse. (o) Dog eats mouse.(x)	邏輯上的正確性 Ex. 程式判斷6是否為 偶數，答案說不是
Pragmatics 意境	作文程度 Ex. 修辭、押韻	程式的好壞程度，可讀性，執行效率等 Ex. O(n)、O(log n) 之 program

Data Type

- Numbers: integers, float, complex, ...
- Boolean
- String
- List
- Tuple
- Dictionary
-
- User Defined Data

Expression

- Expression的兩個元素:value(數值)和operator(運算)。
- 目前所學的Expression有
 - Arithmetic Expression (算術式)
 - Boolean Expression (布林式)
 - String Expression (字串式)

Precedence(整數運算的順序)

- Python定義的運算優先順序為：
 - **最高
 - *, //, %第二
 - +, -最後
- 如果兩個優先順序相同應如何處理?在PYTHON中採取由左到右的方式,當優先順序相同時,左邊的運算先算。
- 口訣:先次方,再乘除,後加減,同順序由左到右。

Arithmetic Expression

- 整數間的運算(結果仍數整數),包括:
 - + : 加法。例如 $5+3$,結果為8。
 - - : 減法。例如 $5-3$,結果為2。
 - * : 乘法。例如 $5*3$,結果為15。
 - **: 次方。例如 $5**3$ 表示5的3次方,也就是5自己乘以自己三次($5*5*5$),結果為125。
 - // : 商。例如 $5//3$ 表示5除以3的商,結果為1。
 - % : 餘。例如 $5\%3$ 表示5除以3的餘數,結果為2。

Boolean Expression布林式

- Boolean有兩個值,個別是True(是、真、對)和False(否、偽、假)

and	True	False
True	True	False
False	False	False

- Boolean的運算有三,依優先順序排列為:

not			
and			
or			

or	True	False
True	True	True
False	True	False

String(字串)

有三種方式來表達string

- 用兩個'將字串包在一起。如'123'。
- 用兩個"將字串包在一起。如"I'm fine."。
- 用兩個'''(打三次的')將字串包在一起。通常用在跳行的長字串中。且字串的內部可以有'和"。

例如：

```
''' This is an example.  
"How do you do?", I asked.  
He said "Can't complain."  
'''
```

Variable Name (Rules):

- Key words are not allowed.
- 第一個字母只能是a-z, A-Z, _.
- 第二個以後的字母只能是a-z, A-Z, 0-9, _

String操作

字串的操作包括：

- +(concatenation): 將兩個字串連(加)在一起。例如說'abc' + 'cba'的結果是字串'abccba'。
- *(multiple): 將字串重複幾次連在一起。例如3*'ab'='ababab'。
- len(string): 計算字串的長度。例如len('abc')=3, len("")=0。

if statement

```
if bool :  
    True Block  
Next Statement
```

```
if bool :  
    True Block  
else :  
    False Block  
Next Statement
```

```
if bool1:  
    statement block 1  
elif bool2:  
    statement block 2  
elif bool3:  
    statement block 3  
.....  
else:  
    statement block n
```