

執行期輸入與引數輸入

蔡尚融

2018-04-02

由鍵盤輸入

- 由 `input()` 函數讀入使用者鍵盤輸入的資料，回傳的型態為字串型態。

```
txtdata = input()
```

- 可以給「輸入提示字串」，提醒使用者輸入資料。

```
txtdata = input('Input some text: ')
```

```
prompt = 'What is the date today? '
```

```
todaydate = input(prompt)
```

範例：

等待使用者輸入「quit」結束執行。

```
#!/usr/bin/env python

testword = '' # Initial for loop condition be true

while testword != 'quit':
    testword = input('Input some word: ')
    print(testword)
```

隨堂練習：

由鍵盤輸入兩數字 x 與 y ，計算 $x + y$ 的值，並重複此步驟直到任一輸入為「exit」或「quit」結束程式執行。

腳本讀入引數

- 程式使用上更具有彈性，使程式能重複利用。
- 使用須匯入 sys 模組，由 sys.argv[索引值] 取得引數；
由 len(sys.argv) 取得引數個數。
- 編寫腳本 demoargv.py

```
#!/usr/bin/env python
import sys
print("Script name:", sys.argv[0])
print("Number of arguments:", len(sys.argv))
```

執行：

1. \$ python3 demoargv.py
2. \$ python3 demoargv.py hello

- 前述方法讀入固定個數的引數，如輸入個數少於程式碼中使用數量時，將導致執行期錯誤。修改腳本 `demoargv.py`

```
#!/usr/bin/env python
```

```
import sys
print("Script name:", sys.argv[0])
print("Number of arguments:", len(sys.argv))
print("Second:", sys.argv[1])
```

執行：

1. `$ python3 demoargv.py hello`
2. `$ python3 demoargv.py`

範例：求最大公因數

```
#!/usr/bin/env python
import sys

numa = int(sys.argv[1])
numb = int(sys.argv[2])

while numb != 0:
    tmp = numb
    numb = numa % numb
    numa = tmp

print(tmp)
```

選擇引數讀入

- 使用須要匯入 `sys` 與 `getopt` 模組。
- 由引數提示指定或輸入，如 `ls` 指令加引數 `-l`。
- 透過 `getopt.getopt()` 函式解析處理，指定字串解析引數，其中有須要讀入值的引數，須附加冒號在其後。

```
opts, args = getopt.getopt(sys.argv[1:], "a:bc:")
```

有輸入值引數為 `-a` 與 `-c`，無輸入者為 `-b`。

範例：

編寫腳本 demogetopt.py

```
#!/usr/bin/env python
import sys, getopt

opts, args = getopt.getopt(sys.argv[1:], "a:bc:")

for op, value in opts:
    if op == "-a":
        print("Option -a:", value)
    elif op == "-b":
        print("Option -b")
    elif op == "-c":
        print("Option -c:", value)
```

隨堂練習：最大公因數或最小公倍數

修改求最大公因數或最小公倍數的程式，並可引數控制選擇輸出最大公因數或最小公倍數。

- 預設輸出最大公因數。
- 選擇性引：
 - ▶ 引數 `-h` 只輸出說明，並直接結束。
 - ▶ 引數 `-g` 輸出最大公因數。
 - ▶ 引數 `-l` 改為輸出最小公倍數。

提示：

1. 需要自訂變數儲存輸出控制旗號（flag）。
2. 引數型態為字串。
3. 需要檢查數入引數數量與型態；本題目分為選擇性引數與必要引數，其中必要引數須為整數型態。
4. 使用字串（str）類別成員函式 `isdigit()` 檢查。
5. 用 `int()` 將數字字串轉成整數。

```
#!/usr/bin/env python
import sys, getopt

def usage():
    print("Compute GCD or LCM")
    print("    demogcdlcm.py [options] int1 int2")
    print("Option:")
    print("    -h  print help")
    print("    -g  compute GCD")
    print("    -l  compute LCM")
```

設定控制旗號：

```
flag = 0  # 0: GCD (default), 1: LCM
```

```
def gcd(numa, numb):  
    while numb != 0:  
        buf = numb  
        numb = numa % numb  
        numa = buf  
    return buf  
  
def lcm(numa, numb):  
    gcd_ab = gcd(numa, numb)  
    qa = numa // gcd_ab  
    qb = numb // gcd_ab  
    return gcd_ab * qa * qb
```

輸入引數測試：

```
opts, args = getopt.getopt(sys.argv[1:], "ghl")
for op, value in opts:
    if op == "-h":
        usage()
        exit()
    elif op == "-g":
        flag = 0
    elif op == "-l":
        flag = 1
    else:
        print("Unknow options.")
        exit()
```

測試輸入整數

```
if len(args) == 2:
    if args[0].isdigit():
        numa = int(args[0])
    else:
        print("Input 1 is not number")
    if args[1].isdigit():
        numb = int(args[1])
    else:
        print("Input 2 is not number")
else:
    print("Need input two integers")
```

```
if flag == 0:  
    print(gcd(numa, numb))  
elif flag == 1:  
    print(lcm(numa, numb))
```