



SiNGULAR 奇點創意

程式創客教室

機器人 / AI人工智慧 / 程式語言

*Join Singular!
Be a super inventor!*

主題：Python MTA 5



Maker + Coder =



Singular Super Inventor





讀寫檔案

Maker + Coder = Singular Super Inventor



SiNGULAR
奇點創意

程式創客教室

機器人 / AI人工智慧 / 程式語言

檔案讀寫參數

參數	敘述
r	讀檔案，檔案不存在，回報錯誤
r+	讀寫檔案，檔案不存在，回報錯誤
w	寫檔案，檔案不存在，創建檔案
w+	讀寫檔案，檔案不存在，創建檔案
a	續寫檔案，檔案不存在，創建檔案
a+	續讀寫檔案，檔案不存在，創建檔案

寫入檔案-1

#1. 要開啟的檔名

```
fileName = "test.txt"
```

#2. 指定w/ r /a mode

```
Mode = "w"
```

#3. 開啟檔案

```
myFile = open(fileName, Mode)
```

寫入檔案-2

#4. 寫入檔案 `\n` 換行符號

```
myFile.write("Hi\n")
```

```
myFile.write("How old are you?")
```

#5. 關閉檔案

```
myFile.close()
```

續寫入檔案-1

#1. 要開啟的檔名

```
fileName = "test.txt"
```

#2. 指定w/ r /a mode

```
Mode = "a"
```

#3. 開啟檔案

```
myFile = open(fileName, Mode)
```

續寫入檔案-2

#4. 寫入檔案 `\n` 換行符號

```
myFile.write("Hi\n")
```

```
myFile.write("My age is 18")
```

#5. 關閉檔案

```
myFile.close()
```

讀取檔案-1

#1. 要開啟的檔名

```
path = 'test.txt'
```

#2. 指定w/ r /a mode

```
f = open(path, 'r')
```



讀取檔案-2

#3. 讀取檔案並顯示

```
lines = f.read()
```

```
print(lines)
```

#4. 關閉檔案

```
f.close()
```



Math函數

Maker + Coder = Singular Super Inventor

Math函數

```
import math
```

#取餘數含小數點

- `math.fmod(x, y)`

#取絕對值

- `math.fabs(x)`



Math函數

#向下取整數

- `math.floor(x)`

#向上取整數

- `math.ceil(x)`

$x == m * 2^{**} e$

- `math.frexp(x)`



建立函式

Maker + Coder = Singular Super Inventor



SINGULAR
奇點創意

程式創客教室

機器人 / AI人工智慧 / 程式語言

建立函式

輸入變數

```
def name_add(name1, name2):  
    return name1 + name2
```

運算結果



使用函式

運算結果

輸入變數

```
val = name_add("abc", "def")  
print(val)
```

執行結果:

abcdef



區域變數 VS 全域變數

```
def area(a, b, c):
```

```
    test1=0
```

```
    test2=1
```

```
#Main Function
```

```
test3 = 2
```



區域變數 VS 全域變數

包在def中的變數test1, test2叫做區域變數，test3稱為全域變數

Main function中看不到 test1 and test2,

area function中可以看到 test3

area function中不要直接使用test3, 請把test3當作a, b, c參數傳入

e.g. area(test3, b, c)

區域變數 VS 全域變數

如在Main function中定義abc這個變數，和test function中的abc視為不同變數，但如果test function沒有定義abc變數，就會拿Main function來用

<pre>1 def test(): 2 abc=1 3 print(abc) 4 5 #Main Function 6 abc = 5 7 test() 8 print(abc)</pre> <p>Running: 變數test.py</p> <p>1 5</p>	<pre>1 def test(): 2 #abc=1 3 print(abc) 4 5 #Main Function 6 abc = 5 7 test() 8 print(abc)</pre> <p>Running: 變數test.py</p> <p>5 5</p>
---	--

