

# Python

匿名函数

# 有名字的體積計算函式

```
# area3 函式計算體積
def area3(r, h):
    return r * r * 3.14 * h

r = 3
volume = area3(r, 10)
print( '半徑 {0} 時的圓柱體體積: {1}'.format(r, volume))
```

# 沒名字的體積計算函式

```
# area3 函式計算體積
def area3(r, h):
    return r * r * 3.14 * h

r = 3
volume = area3(r, 10)
print( '半徑 {0} 時的圓柱體體積: {1}'.format(r, volume))
```

```
# area3 函式計算體積
volume = lambda r, h: r * r * 3.14 * h

r = 3
volume = volume(r, 10)
print( '半徑 {0} 時的圓柱體體積: {1}'.format(r, volume))
```

# 匿名函式 lambda

lambda 參數 : 運算式

# 匿名函式 lambda

匿名函式只會有一行！  
**Only One**

lambda 參數：**運算式**

匿名函式只能接受一行的運算式運算，不能執行敘述式喔！

如果運算式是  $x * 3$  這類的，lambda 會直接將運算結果回傳，不需要使用 return。

PYTHON

延伸的概念

## 為什麼要用匿名函式?

1. 程式一次行使用，所以不需要定義函式名，節省記憶體中變數定義空間
2. 如果想讓程式更加簡潔時。

## 匿名函式幾個規則:

1. 一般也就一行表示式，必須有返回值
2. 不能有return
3. 可以沒有引數，可以有一個或多個引數