

PYTHON3

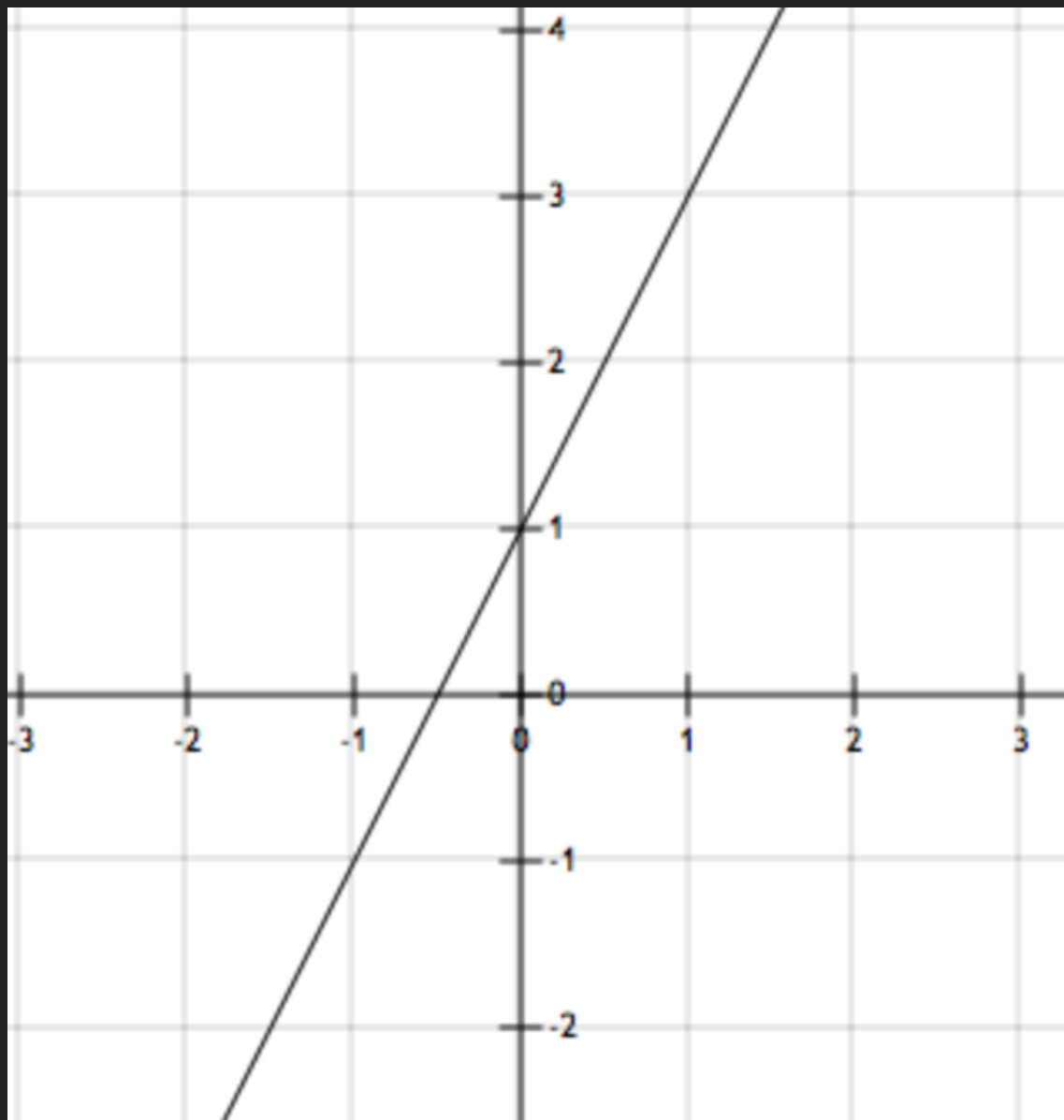
FUNCTION



二元一次方程式

▶ $y = 2x + 1$

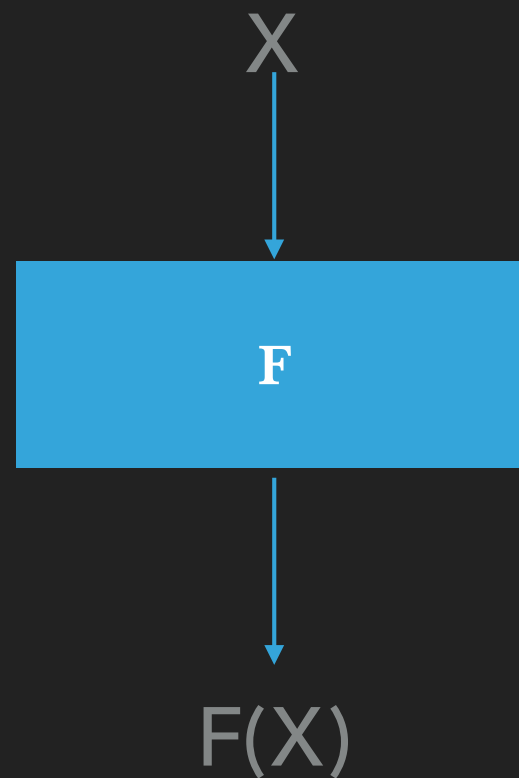
▶ $f(x) = 2x + 1$





函數是什麼？

- ▶ 塞一些東西進去，吐一些東西出來
- ▶ 像 $f(x) = 2x+1$ 就是塞一個實數進去，吐一個實數出來





定義語法

```
def [函數名稱](參數名稱1, 參數名稱2 ...):
```

```
[縮排]...
```

```
[縮排]return [回傳值]
```



EXAMPLE — 一開始的例子

```
def f(x):
```

```
    return 2*x + 1
```

```
def f(x, y):
```

```
    return x*y + x*x + y*y
```



EXAMPLE — 呼叫函數

```
def f(x):
```

```
    return 2*x + 1
```

$$\begin{array}{l} F(1) = 3 \\ F(2) = 5 \end{array}$$

```
print("F(1) = " + str(f(1)))
```

```
print("F(2) = " + str(f(2)))
```



EXAMPLE — 參數可以是字串或者不同型態

```
def appendMulti0(str1, n):
```

```
    return str1 + "0"*n
```

```
def combineString(str1, str2):
```

```
    return str1 + str2
```



EXAMPLE — 陣列 ?

```
def calAvg(arr):  
    s = 0  
  
    for e in arr:  
        s = s+e  
  
    return s/len(arr)
```




EXAMPLE — 陣列 ?

```
def filterOdd(arr):  
    ret_arr = []  
  
    for e in arr:  
        if e%2 == 0 :  
            ret_arr.append(e)  
  
    return ret_arr
```



EXAMPLE — 可以不需要參數或者不回傳東西

```
def 西瓜x10():
```

```
    for i in range(10):
```

```
        print("西瓜")
```

```
def 西瓜xN(n):
```

```
    for i in range(n):
```

```
        print("西瓜")
```



EXAMPLE — 又是⑨⑨乘法表

```
def drawNx1to9(n):  
    for i in range(1, 10):  
        print(str(n) + "x" + str(i) + "=" + str(i*n))  
  
def draw99():  
    for i in range(1, 10):  
        drawNx1to9(i)  
  
draw99()
```



MORE EXAMPLE — RETURN觸發的瞬間就要回家

```
def isPrime(n):  
    for i in range(2, n):  
        if n%i == 0:  
            return False  
    return True
```

問題一：當 $n = 20$ ，第三行會跑幾遍

問題二：當 $n = 107$ ，...(同上)



MORE EXAMPLE — 呼叫函數

```
def isPrime(n):
```

```
    ...
```

```
for i in range(2, 10):
```

```
    if isPrime(i) :
```

```
        print(i)
```



BUILT-IN FUNCTION

```
>>> len
<built-in function len>
>>> max
<built-in function max>
>>> min
<built-in function min>
>>> sum
<built-in function sum>
```



定義語法

```
def [函數名稱](參數名稱1, 參數名稱2 [= 預設值] ...):
```

```
[縮排]...
```

```
[縮排]return [回傳值]
```



MORE — 預設值

```
def filterMod(arr, a, m = 0):  
    ret_arr = []  
    for e in arr:  
        if e%a == m :  
            ret_arr.append(e)  
    return ret_arr
```




MORE — 預設值

`filterMod([1, 4, 7, 10, 12], 3, 1)` # 回傳[1, 4, 7, 10]

`filterMod([1, 4, 7, 10, 12], 3)` # 回傳[12]

`filterMod([1, 2, 3, 4], 2)` # 回傳 [2, 4]