



深度學習TensorFlow實務

Python 基礎教學

Pre-Lab 2

-TA-

李偉弘

廖宜健

林佑昌

蔡明諺

彭冠偉

什麼是 Python ?

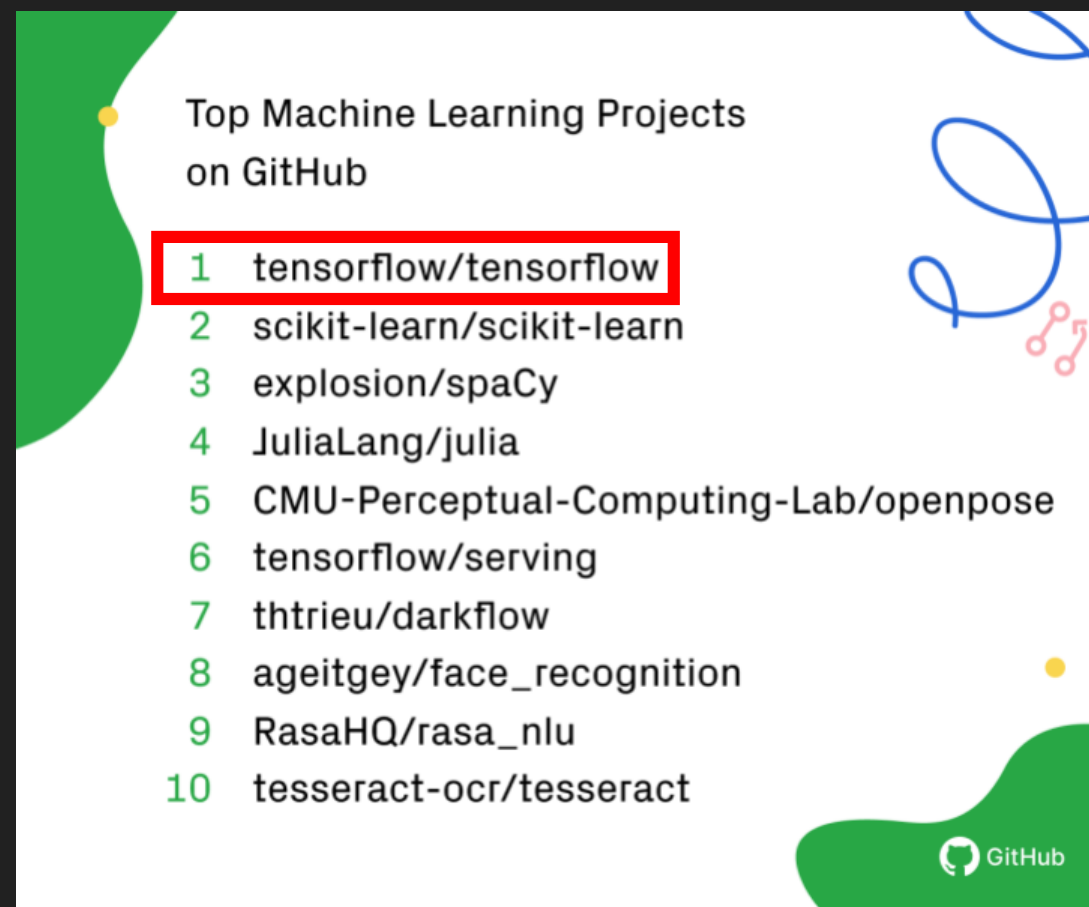
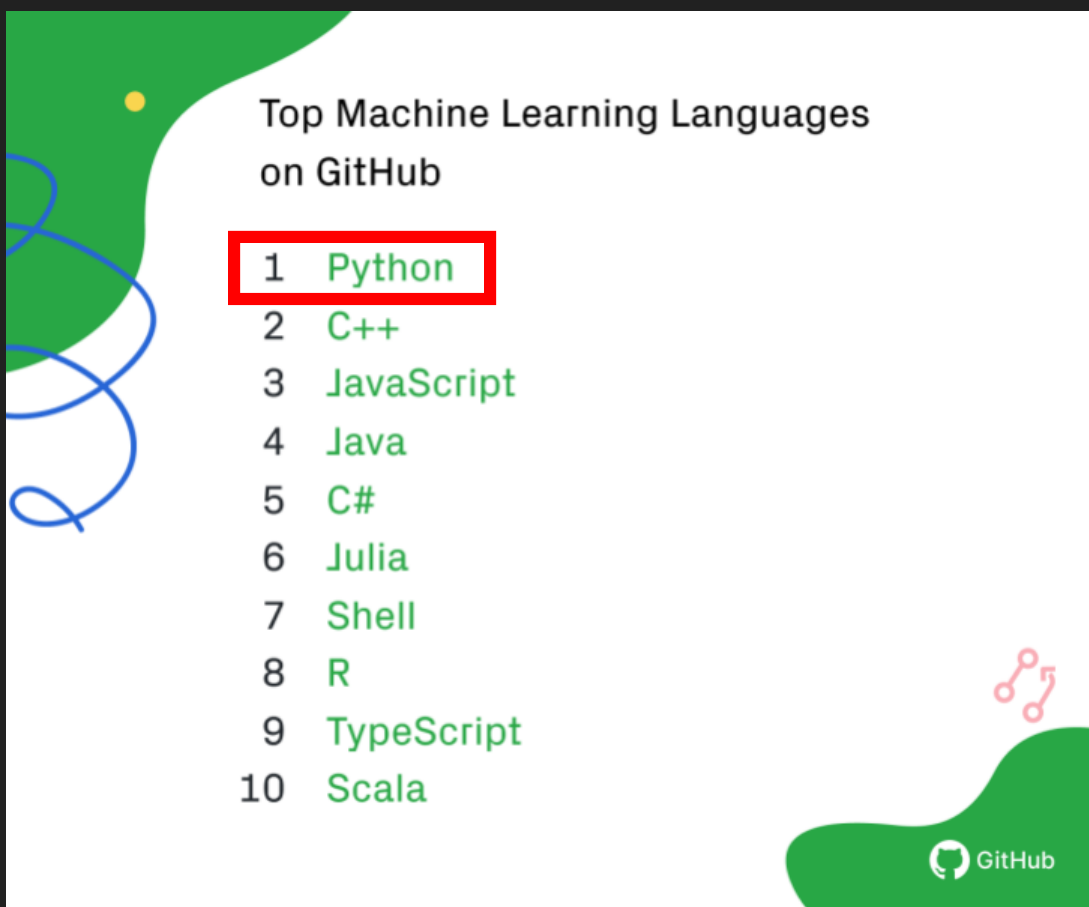
Python 大補帖

- 高階程式語言
- 語法簡潔，容易上手
- 跨平台
- 多種第三方函式庫
- 多用途
 - GUI 設計、機器學習、Web 開發等



Python 大補帖

- 2018 年在 Github 機器學習領域中 Python 為最受歡迎的程式語言



Python 資料型態介紹

Python 資料型態

- 變數(Variable)
- 字串(String)
- 清單(List)
- 字典(Dictionary)
- 布林(Bool)

變數(Variable)

- 使用英文字母，即可定義變數
- 常用變數有整數(Integer)、浮點數(Float)等

```
#賦予 x變數數值 10  
x = 10
```

```
#輸出 x  
print(x)
```

10

```
x = 10  
y = 3.14
```

```
#輸出 x + y  
print(x + y)
```

13.14

```
t = 50  
print(t / 3)  
print(t // 3)  
print(t % 3)
```

16.6666666666666668

16

2

字串(String)

- 使用英文字母，即可定義字串
- 字串必需要加上單引號 ' '，或雙引號 " "

```
a = "hello tensorflow"  
print(a)
```

```
hello tensorflow
```

```
a = "hello "  
b = "tensorflow"  
print(a + b)
```

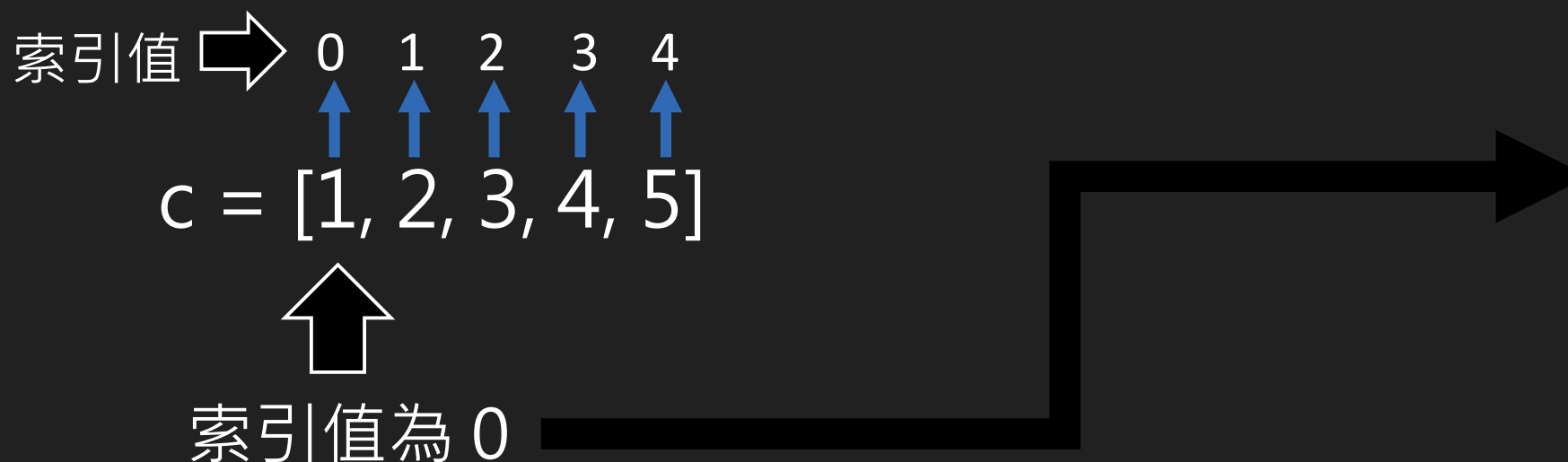
```
hello tensorflow
```


清單(List)

- 整合資料變成陣列
- 清單必須用中括號 `[]` 來表示
- 清單中的索引值(index，中括號中的值)

從 0 開始算

索引值 \Rightarrow 0 1 2 3 4
 \uparrow \uparrow \uparrow \uparrow \uparrow
`c = [1, 2, 3, 4, 5]`
 \uparrow
索引值為 0



```
#建立空清單  
non_list = []  
print(non_list)
```

```
[]
```

```
#新增物件到清單中  
non_list.append(11)  
print(non_list)
```

```
[11]
```

```
#建立清單  
c = [1, 2, 3, 4, 5]  
print(c)
```

```
[1, 2, 3, 4, 5]
```

```
#取得清單長度  
len(c)
```

```
5
```

```
#取得清單中的第一個數  
c[0]
```

```
1
```

```
#改變清單中的數值  
c[0] = 99  
print(c)
```

```
[99, 2, 3, 4, 5]
```

清單(List)

- 使用 list 中的切片(Slicing)存取子清單

a為一陣列，取a的子清單語法為：

a[m:n]

注意！！

從第m個原素開始存取，取到第n-1個原素為止

```
a = [1, 3, 5, 7, 9]
```

```
#取出list中第一到三個值  
print(a[0:3])
```

```
[1, 3, 5]
```

```
a = [1, 3, 5, 7, 9]
```

```
#取出list中第二到四個值  
print(a[1:4])
```

```
[3, 5, 7]
```

清單(List)

- 另一種切片(Slicing)方式

a為一陣列，切片語法：

`a[m:n]`

⇒ m 或 n 不輸入

```
a = [1, 3, 5, 7, 9]
```

```
#取出list中第二到最後一個值  
print(a[1:])
```

```
[3, 5, 7, 9]
```

```
a = [1, 3, 5, 7, 9]
```

```
#取出list中第一到第三個值  
print(a[:3])
```

```
[1, 3, 5]
```

字典(Dictionary)

- 類似 list，索引值可自由定義
- 字典必須用大括號 {} 來表示
- 存取值需用中括號 []

```
#建立字典  
me = {'height':180}
```

```
#用中括號取出值  
print(me['height'])
```

180

```
#使用中括號新增元素  
me['weight'] = 70
```

```
#顯示字典  
print(me)
```

{'height': 180, 'weight': 70}

布林(Bool)

- 僅有 True 與 False 兩種值
- 運算子有 and、not、or 等

```
#肚子餓嗎
```

```
hungry = True
```

```
#想睡覺嗎
```

```
sleepy = False
```

```
#肚子餓且想睡覺嗎
```

```
print(hungry and sleepy)
```

```
#肚子餓或想睡覺嗎
```

```
print(hungry or sleepy)
```

```
False
```

```
True
```

Python 資料型態查詢

- 碰到未知的資料時，我們可以使用 `type` 指令查詢其資料型態

```
a = 1
b = 'hello tensorflow'
list1 = [2, 4, 6, 8]
mydata = {'sex': 'man', 'height': 180}
hungry = False

print(type(a))
print(type(list1))
print(type(mydata))
print(type(hungry))

<class 'int'>
<class 'list'>
<class 'dict'>
<class 'bool'>
```

Python 資料型態轉換

- 轉換資料時，我們可以直接在要轉換型態的資料前加上目標型態

```
a = 1
b = 'class'

#輸出"class 1"
print(b + ' ' + str(a))

class 1
```

Python 常用指令

- 條件陳述式(if / else)
- 迴圈陳述式(for)
- 函數(Function)

條件陳述式(if / else)

- if 語法：

```
if condition_1:  
    statement_block_1  
elif condition_2:  
    statement_block_2  
else:  
    statement_block_3
```

- 注意冒號下一行的縮排(4個空格)

```
a = 1  
  
if a == 1:  
    print('a = 1')  
else:  
    print('a != 1')  
  
a = 1
```

迴圈陳述式(for)

- for 迴圈語法：

for <variable> in <sequence>:

 <statements>

else:

 <statements>

```
for i in [1, 2, 3]:  
    print(i)
```

```
1  
2  
3
```

```
for i in 'hello':  
    print(i)
```

```
h  
e  
l  
l  
o
```

迴圈陳述式(for)

■ range函數

語法： range(stop)

range(start, stop[, step])

```
range_list = range(10)
```

```
print(range_list)
```

```
range(0, 10)
```

```
print(list(range_list))
```

```
for i in range_list:  
    print(i)
```

```
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

```
0
```

```
1
```

```
2
```

```
3
```

```
4
```

```
5
```

```
6
```

```
7
```

```
8
```

```
9
```

函數(Function)

- 具有整合性的處理定義為函數(Function)

函數未取得參數

```
def hello():  
    print("hello tensorflow")
```

```
hello()
```

```
hello tensorflow
```

函數取得參數

```
def hello(object):  
    print("hello" + object)
```

```
hello(" ntut")
```

```
hello ntut
```

練習題

- 第一題：找出 1 到 100 中，能被 3 整除的整數，並建立一個列表(list)，由小到大排列這些整數，並輸出。

(提示：可使用創建空 list 指令)

輸出結果：

```
[3, 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48, 51, 54, 57, 60, 63, 66, 69, 72, 75, 78, 81, 84, 87, 90, 93, 96, 99]
```

- 第二題：承上題，將 list 中，數值超過 50 整數的加總，並輸出。

輸出結果：1275

-END-