113-1基礎程式設計(13)

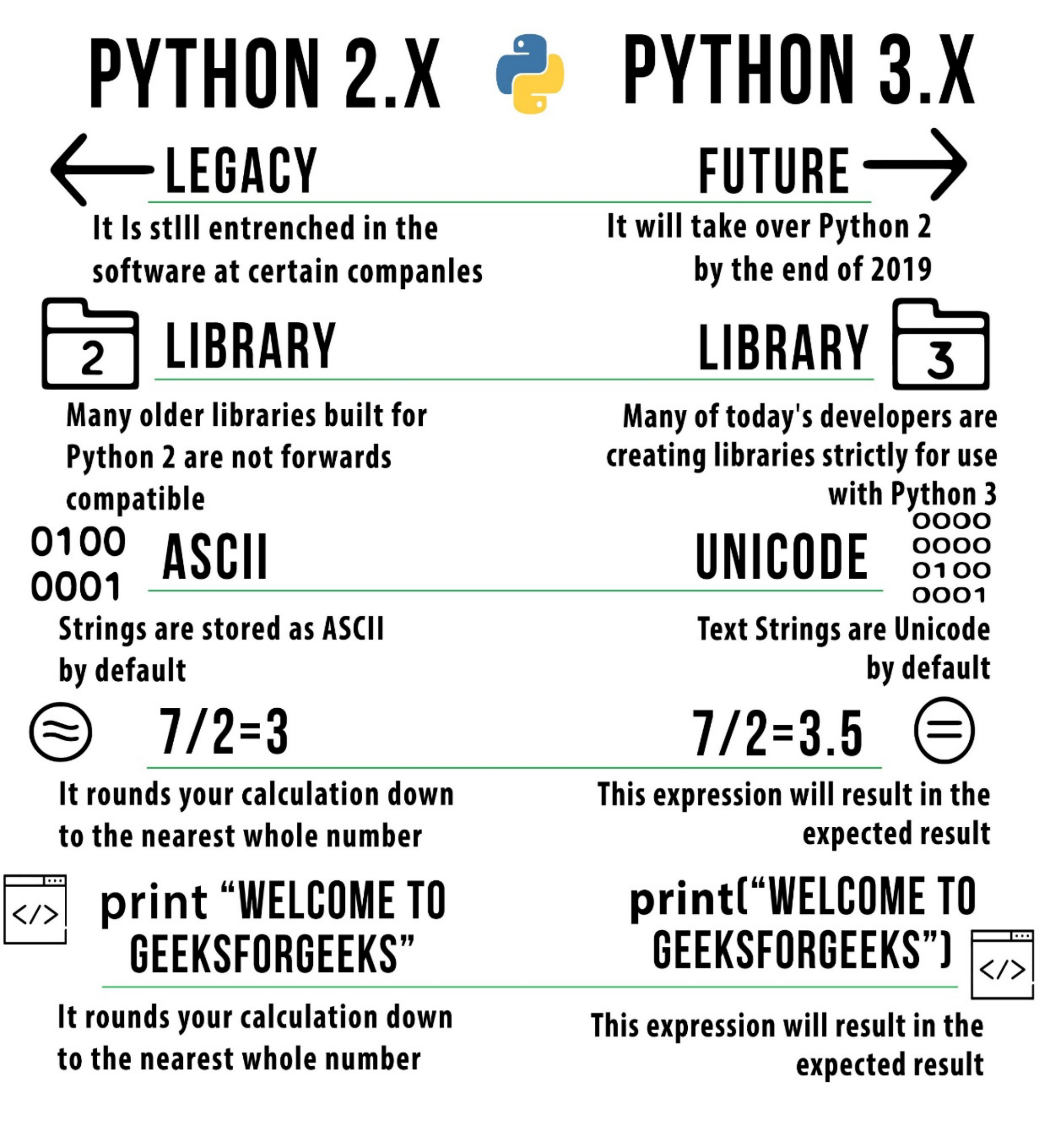
亞大資工系

課程大綱

- W1-Python簡介及程式工具
- W2
- W3-變數和運算
- W4-迴圈和格式化輸出
- W5-判斷式和容器
- W6-字串處理和輸出入
- W7-字典容器
- W8-檔案處理
- W9-期中測驗

- W10-函數
- W11-進階流程控制
- W12-進階運算和生成器
- W13-進階函數
- W14-類別
- W15-進階類別
- W16-模組和套件
- W17-進階設計
- W18-期末測驗

Python Versions



PYTHON 2.X	PYTHON 3.X
LEGACY It Is stIll entrenched in the software at certain companles	FUTURE It will take over Python 2 by the end of 2019
2 LIBRARY Many older libraries built for Python 2 are not forwards compatible	LIBRARY 3 Many of today's developers are creating libraries strictly for use with Python 3
0100 0001 ASCII Strings are stored as ASCII by default	UNICODE 0000 0000 0100 0001 Text Strings are Unicode by default
≈ 7/2=3 It rounds your calculation down to the nearest whole number	7/2=3.5 = This expression will result in the expected result
</> print “WELCOME TO GEEKSFORGEEKS” It rounds your calculation down to the nearest whole number	print(“WELCOME TO GEEKSFORGEEKS”) </> This expression will result in the expected result

Python new features:

- Python 3.10: Structural Pattern Matching
- Python 3.6 : f-Strings
- Python 3.3 : Virtual Environments
- Python 3.2: Argparse

Python powerful features:

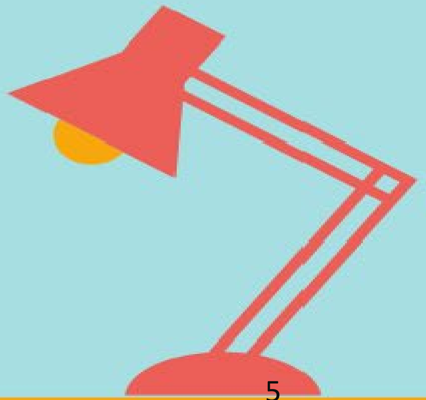
- Iterators
- Generators
- Decorators
- Context Managers

本週主題-進階函數

- Week13-進階函數
  - Topic 1(主題1)-函數定義和呼叫的複習
  - Topic 2(主題2)-參數和傳回值的資料型別
  - Topic 3(主題3)-遞迴函數 (Recursion)
  - Topic 4(主題4)-Lambda 函數(Anonymous Function)
  - Topic 5(主題5)-JSON格式處理
  - Topic 6(主題6)-亂數(random)
  - Topic 7(主題7)-用圖表呈現資料

Topic 1-函數定義和呼叫的複習

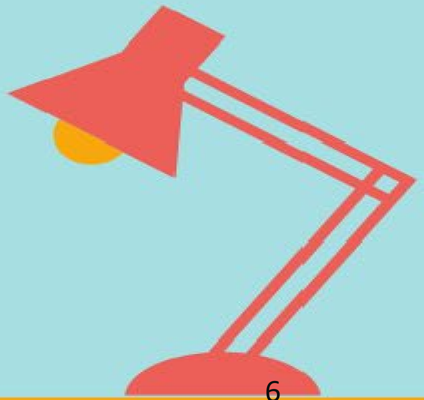
- 函數的定義(definition)
- 函數的呼叫(call)
- 可變參數*參數
- 可變參數**參數



Topic 2-參數和傳回值的資料型別(3.5 版新加入)

```
def gcd1(x, y):
  r=x%y
  while r!=0:
    x=y;y=r
    r=x%y
  return y

a=240; b=96
print(f"{a}, {b}, {gcd1(a,b)}")
```

```
def gcd2(x:int, y:int)->int:
  r=x%y
  while r!=0:
    x=y;y=r
    r=x%y
  return y

a=240; b=96
print(f"{a}, {b}, {gcd2(a,b)}")
```



Topic 3-遞迴函數 (Recursion)

```
def gcd1(x, y):
  r=x%y
  while r!=0:
    x=y;y=r
    r=x%y
  return y

a=240; b=96
print(f"{a}, {b}, {gcd1(a,b)}")
```

```
def gcd2(x, y):
    if y == 0:
        return x
    else:
        return gcd2(y, x%y)


a=240; b=96
print(f"{a}, {b}, {gcd2(a,b)}")
```

Topic 4-Lambda 函數

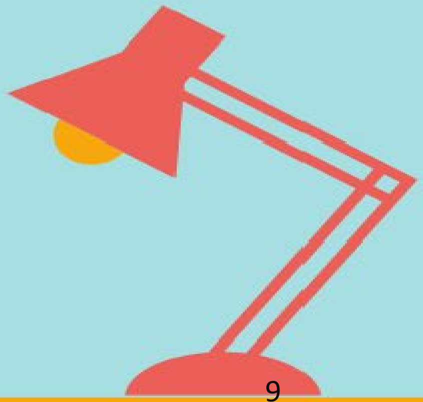
```
def gcd1(x, y):
  r=x%y
  while r!=0:
    x=y;y=r
    r=x%y
  return y

a=240; b=96
print(f"{a}, {b}, {gcd1(a,b)}")
```

```
gcd = lambda x,y : (gcd(y,x%y)) if x%y else y

a=240; b=96
print(f"{a}, {b}, {gcd(a,b)}")
```

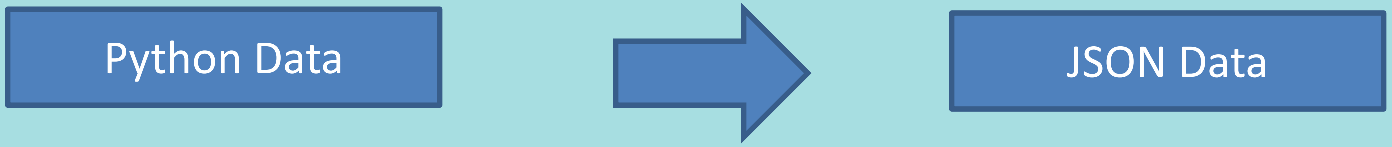
在map()和filter()中使用Lambda函數

```
base=[1, 2, 3, 4]

square=list(map(lambda x: x**2, base))
print(square)
```

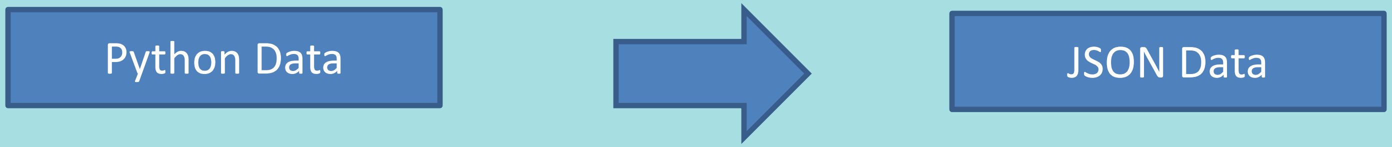
```
data=[0,1,2,3,4,7,8,14,19,31,34]

odds =list(filter(lambda x: x%2 > 0, data))
print(odds)
evens=list(filter(lambda x: x%2 ==0, data))
print(evens)
```



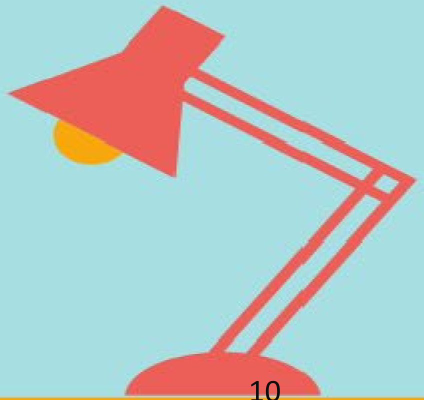
Topic 5-JSON



```
import json
data = [ { 'a' : 1, 'b' : 2, 'c' : 3, 'd' : 4, 'e' : 5 } ]
file = open("test.json", "w")
json.dump(data, file)
file.close()
```



```
import json
file = open ('test.json')
json_data = json.load(file)
file.close()
```

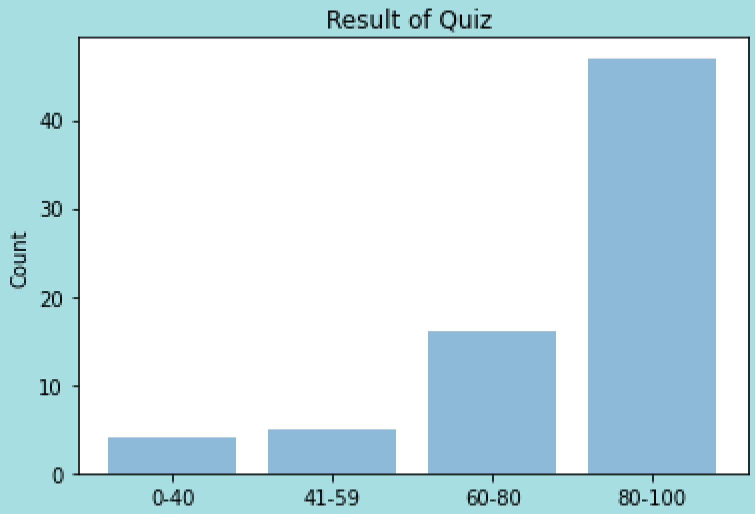


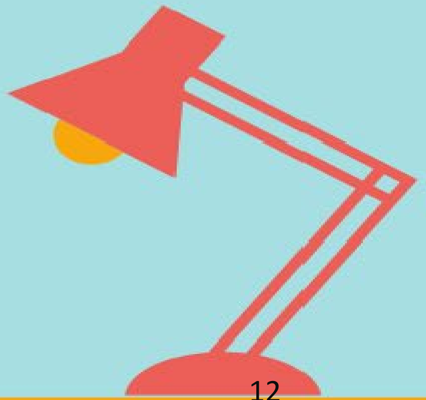
Topic 6-亂數(random)

```
rd.seed(12345)
a=rd.random()                                  # Random float:  0.0 <= x < 1.0
b=rd.uniform(2.5, 10.0)                        # Random float:  2.5 <= x < 10.0
c=rd.randrange(10)                             # Integer from 0 to 9 inclusive
d=rd.choice(['win', 'lose', 'draw'])           # Single random element from a seq
uence
print(a,b,c,d)
```

Topic 7-用圖表呈現資料-長條圖

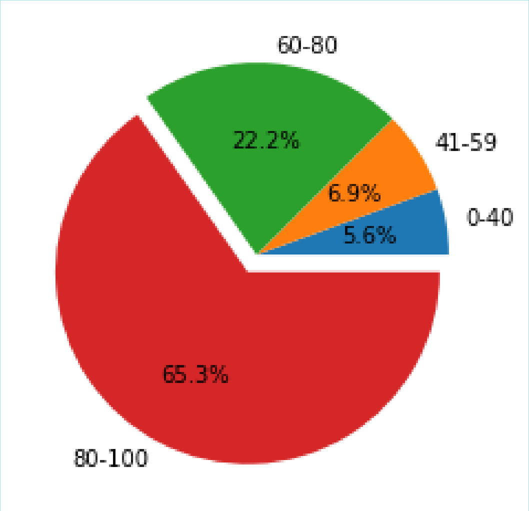
```
#畫出人數分佈的長條圖
import matplotlib.pyplot as plt
import numpy as np
stats = [4, 5 ,16 , 47]
labels = ('0-40', '41-59', '60-80', '80-100')
y_pos = np.arange(len(stats))
plt.bar(y_pos, stats, align='center', alpha=0.5)
plt.xticks(y_pos, labels)
plt.ylabel('Count')
plt.title('Result of Quiz ')
plt.show()
```





Topic 7-用圖表呈現資料-圓餅圖

```
import matplotlib.pyplot as plt

# Pie chart

stats = [4, 5 ,16 , 47]
labels = ('0-40', '41-59', '60-80', '80-100')
explode = (0, 0, 0,  0.1)  # 切開'80-100'的人數
fig1, ax1 = plt.subplots()
ax1.pie(stats, labels=labels, explode=explode, autopct='%.1f%%')
plt.show()
```



Thanks!

Q&A