

資料拿取

常用資料型態的資料拿取：

List	
1D	2D
list[切片]	list[row 切片][column 切片]
#切片順序不可顛倒(Must be 小:大) #List切片只能針對同一維度的資料進行切片	

```
65  92  78  83  70
90  72  76  93  56
81  85  91  89  77
79  53  47  94  80
```

```
print(list1[1][1:4])
print(list1[2][1:4])
print(list1[1:3])

[72, 76, 93]
[85, 91, 89]
[[90, 72, 76, 93, 56],
 [81, 85, 91, 89, 77]]
```

```
ndarray1[1:3,1:4]

array([[72, 76, 93],
       [85, 91, 89]])
```

Numpy	
1D	2D
ndarray[切片]	ndarray[row 切片][column 切片]#同list用法 ndarray[row 切片, column 切片]
ndarray[ndarray + 布林運算]	ndarray[[row],[column]] #前衛索引Fancy Indexing Nparray[[1,1,0,0],[0,2,1,0]]→(1,0),(1,2),(0,1),(0,0)
ndarray[['數值索引1','數值索引2']] #可跳順序	ndarray[np.ix_([row],[column]]) #笛卡爾乘積 x[np.ix_([1,5,7,2],[0,3,1,2])]→ ndarray([[x[1,0], x[1,3], x[1,1], x[1,2]], [x[5,0], x[5,3], x[5,1], x[5,2]], [x[7,0], x[7,3], x[7,1], x[7,2]], [x[2,0], x[2,3], x[2,1], x[2,2]]])

Dict	
1D	2D
dic[key]	dic[key][subkey]
dict.get(key)	dict.get(key,'if not found value').get(key) #如未找到key會回傳None · 可指定未找到要回傳的值
dict.setdefault(key) #如果key值不存在會新增	dict.setdefault(key).setdefault(key) #如果key值不存在會新增
#2D dict.get第一層如未找到 · 會回傳'NoneType' object has no attribute 'get'錯誤 建議可直接寫成dict.get(key, {}).get(key)	

Pandas	
Series	DataFrame
Series[切片]	head(n)#前n筆資料
Series[索引名稱]	tail(n)#後n筆資料
Sereies[['索引名稱1','索引名稱2']] #可跳順序	df.loc[['row索引名稱'],[column索引名稱]] #可跳順序
Sereies[[數值索引1,數值索引2]] #可跳順序	df.iloc[[row數值索引],[column數值索引]] #可跳順序
Series.loc[索引名稱]	df.loc[['row索引名稱切片'],[column索引名稱切片]]
Series.iloc[數值索引]	df.iloc[[row數值索引切片],[column數值索引切片]]
Series[Series + 布林運算]	df[df + 布林運算]
#Pandas有自己的邏輯運算子(& ~.....) · 多個布林運算條件一定要加小括號分開() #只取一格資料可以用df[row][column]	