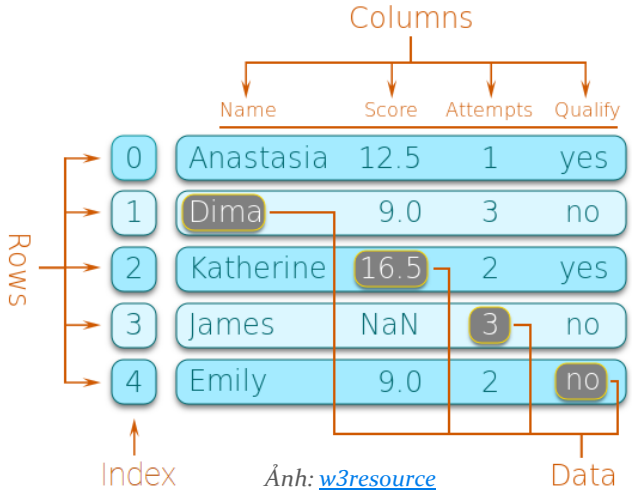
mindX Technology School

Computer Science Advanced

BÀI 10. TRUY XUẤT DỮ LIỆU VỚI PANDAS

1. Truy Xuất Dòng và Cột từ DataFrame

Ta sử dụng ***DataFrame*** **`df`** chứa dữ liệu như ảnh bên cho các ví dụ bên dưới.



Ảnh: w3resource

CÚ PHÁP THÔNG THƯỜNG

```
# truy xuất 1 cột
df['Name']
df.Name
```

```
# truy xuất nhiều cột
df[['Name', 'Attemps']]
```

```
# truy xuất 1 dòng
df[1:2]
```

```
# truy xuất nhiều dòng
df[1:4]
df[1:]
```

Ghi chú: Nếu truy vấn là *một dòng hoặc một cột* thì kết quả trả về là một *Series*. Nếu truy vấn bao gồm *nhiều hơn một dòng hoặc một cột* thì kết quả là một *DataFrame*.

CÚ PHÁP LOC VÀ ILOC

DataFrame hỗ trợ thuộc tính **`loc`** và **`iloc`** để truy xuất dòng và cột.

Cú pháp chung: `df.loc[<danh sách dòng>, <danh sách cột>]`
và `df.iloc[<danh sách dòng>, <danh sách cột>]`

Nếu lược bỏ danh sách cột, ***pandas*** *mặc định lấy tất cả các cột*.

Trong đó, *danh sách dòng* và *danh sách cột* có thể là:

- một dòng/cột
- một list các dòng/cột
- cú pháp list slicing
  `[<bắt đầu>:<kết thúc>]`

Thuộc tính **`iloc`** Truy xuất theo số thứ tự	***Thuộc tính*** **`loc`** Truy xuất theo tên
`# truy xuất 1 dòng` `df.iloc[1]`	`# truy xuất 1 dòng` `df.loc[1]`
`# truy xuất 1 cột` `df.iloc[:, 1]`	`# truy xuất 1 cột` `df.loc[:, 'Score']`
`# truy xuất 1 ô dữ liệu` `df.iloc[1, 2]  # dòng 1, cột 2`	`# truy xuất 1 ô dữ liệu` `df.loc[1, 'Attemps']`

`# truy xuất nhiều dòng`
`df.iloc[1:3]`
`df.iloc[[0, 2, 3]]`
	`# truy xuất nhiều dòng`
	`df.loc[1:3]`
	`df.loc[[0, 2, 3]]`

`# truy xuất nhiều cột`
`df.iloc[:, 1:3]`
`df.iloc[:, [0, 2, 3]]`
	`# truy xuất nhiều cột`
	`df.loc[:, 'Score':'Qualify']`
	`df.loc[:, ['Name', 'Attemps', 'Qualify']]`

`# truy xuất nhiều dòng và cột` `df.iloc[1:, [0, 2, 3]]`	`# truy xuất nhiều dòng và cột` `df.loc[1:, ['Name', 'Attemps', 'Qualify']]`

**Các dòng loc và iloc ngang hàng nhau ở bảng trên trả về cùng giá trị.*
***Trong ví dụ này, index được tự động đánh số nên tên dòng và số thứ tự các dòng giống hệt nhau.*

TRUY XUẤT DÒNG THEO ĐIỀU KIỆN

Pandas hỗ trợ *truy xuất các dòng thỏa một điều kiện nào đó* theo cú pháp bên dưới.

Ví dụ: Tìm các thí sinh có *Score* > 10.

```
rows = df['Score'] > 10
result = df[rows]
```

rows là một *Series* chứa các giá trị *boolean* với **True** và **False** lần lượt tương ứng với các dòng thỏa và không thỏa điều kiện.

```
>> rows :
0     True
1    False
2     True
3    False
4    False
Name: Score, dtype: bool
>> result :
        Name  Score  Attemps Qualify
0  Anastasia   12.5        1     yes
2  Katherine   16.5        2     yes
```

Viết trên cùng một dòng:

```
result = df[df['Score'] > 10]
```

Sử dụng trong **loc** ***và*** **iloc**: *Tìm tên và điểm các thí sinh có Score > 10.*

```
result1 = df.loc[df['Score'] > 10, ['Name', 'Score']]
result2 = df.iloc[df['Score'] > 10, [0, 1]]
```

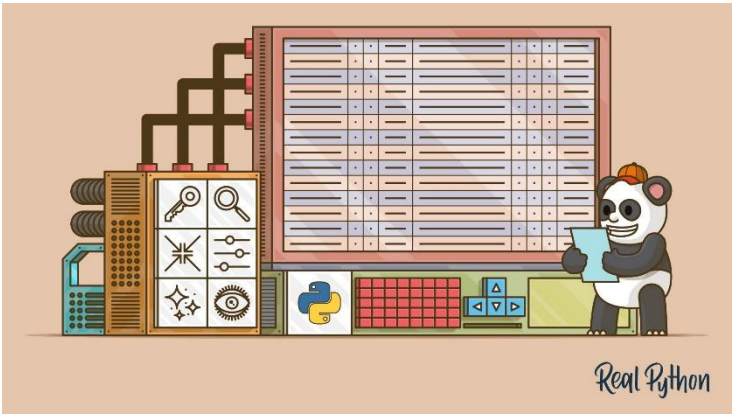
2. Tổng Kết Dữ Liệu trong DataFrame

CÁC HOẠT ĐỘNG TỔNG KẾT CƠ BẢN

Đếm số dòng: `len(df)` `>> 5`

Đếm số dòng và cột: `df.shape` `>> (5, 4)`

Ảnh: Real Python



CÁC PHƯƠNG THỨC TỔNG KẾT CƠ BẢN

sum(), count(), min(), max(), mean()

Ví dụ: `df['Attemps'].sum()` `>> 11`

Khi áp dụng các phương thức trên cho cả ***DataFrame***, ta nhận được *kết quả tổng kết trên tất cả các cột*.

Ví dụ: `result = df.max()`

```
>> result :
Name        Katherine
Score            16.5
Attemps             3
Qualify           yes
dtype: object
```

MỘT SỐ PHƯƠNG THỨC KHÁC

Đếm số lần xuất hiện của từng giá trị:

`df['Qualify'].value_counts()`

```
>>

no     3
yes    2
Name: Qualify, dtype: int64
```

Tổng kết chung các cột chứa dữ liệu số:

`df.describe()`

```
>>
           Score  Attemps
count   4.000000  5.00000
mean   11.750000  2.20000
std     3.570714  0.83666
min     9.000000  1.00000
25%     9.000000  2.00000
50%    10.750000  2.00000
75%    13.500000  3.00000
max    16.500000  3.00000
```