Removing and Fixing Columns with pandas

Chris Achard

@nanohopdev   www.nanohop.com

Overview

Fixing Columns

- Dropping columns
- Change case
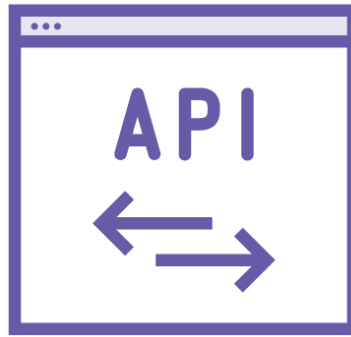- Rename columns

Why Fix Columns?

Collaboration
Naming clarity helps when datasets are shared

Interaction
Configure data for legacy or third party systems

Big Data
Don't lose track of data in a large systems with bad names

Demo

Drop irrelevant columns

Inplace=True

Batch changes (change case)

Renaming columns

Dropping Columns

Changing Column Casing

Renaming Columns

Review

```
df.drop('id', axis=1)
```

◄ **Drop a single column**

```
df.drop(columns = ['height', 'width'])
```

◄ **Drop several columns**

```
df.drop('id', axis=1, inplace=True)
```

◄ **Drop a column inplace**

```
df= pd.read_csv('file.csv',
usecols=['artist', 'title'])
```

◄ **Only import certain columns**

```
df.columns.str.lower()
```

◄ **Lowercase all column names**

```
[x.lower() for x in data.columns]
```

◄ **Use for...in**

```
map(lambda x: x.lower(), data.columns)
```

◄ **Use map**

```python
df.rename(columns={"start": "finish"})
```

◄ **Rename a dictionary of columns**

```python
df.rename(columns=lambda x: x.lower(),
inplace=True)
```

◄ **Rename inplace with a lambda**

```python
df = pd.read_csv('file.csv', names=['id',
'one', 'two'], header=0)
```

◄ **Rename on import**