

RenameColumns -> UFO Dataset

November 16, 2016

```
In [1]: import pandas as pd
```

```
In [2]: ufo = pd.read_csv('http://bit.ly/uforeports')
```

```
In [3]: ufo.head()
```

```
Out[3]:
```

	City	Colors Reported	Shape Reported	State	Time
0	Ithaca	NaN	TRIANGLE	NY	6/1/1930 22:00
1	Willingboro	NaN	OTHER	NJ	6/30/1930 20:00
2	Holyoke	NaN	OVAL	CO	2/15/1931 14:00
3	Abilene	NaN	DISK	KS	6/1/1931 13:00
4	New York Worlds Fair	NaN	LIGHT	NY	4/18/1933 19:00

```
In [5]: ufo.columns # list of Columns listed
```

```
Out[5]: Index(['City', 'Colors Reported', 'Shape Reported', 'State', 'Time'], dtype=object)
```

```
In [9]: ufo.rename( columns = {'Colors Reported' : 'Colors_Reported', 'Shape Reported' : 'Shape_Reported'},
# rename old : new
# inplace => pandas.DataFrame.drop => if True, do operation inplace and return None)
```

```
In [10]: ufo.columns # returns columns as needed
```

```
Out[10]: Index(['City', 'Colors_Reported', 'Shape_Reported', 'State', 'Time'], dtype=object)
```

```
In [11]: # rename everything => create Python list of strings
ufo_cols = ['city', 'colors reported', 'shape reported', 'state', 'time']
```

```
In [12]: ufo.columns = ufo_cols # overwriting old column names with new ones (lowercase)
```

```
In [13]: ufo.head()
```

```
Out[13]:
```

	city	colors reported	shape reported	state	time
0	Ithaca	NaN	TRIANGLE	NY	6/1/1930 22:00
1	Willingboro	NaN	OTHER	NJ	6/30/1930 20:00
2	Holyoke	NaN	OVAL	CO	2/15/1931 14:00
3	Abilene	NaN	DISK	KS	6/1/1931 13:00
4	New York Worlds Fair	NaN	LIGHT	NY	4/18/1933 19:00

```
In [16]: ufo = pd.read_csv('http://bit.ly/uforeports', names=ufo_cols, header=0) #a
```

```
In [15]: ufo.head()
```

```
Out[15]:
```

	city	colors	reported	shape	reported	state		t
0	Ithaca		NaN	TRIANGLE	NY	6/1/1930	22	
1	Willingboro		NaN	OTHER	NJ	6/30/1930	20	
2	Holyoke		NaN	OVAL	CO	2/15/1931	14	
3	Abilene		NaN	DISK	KS	6/1/1931	13	
4	New York Worlds Fair		NaN	LIGHT	NY	4/18/1933	19	

```
In [20]: ufo.columns
```

```
Out[20]: Index(['city', 'colors_reported', 'shape_reported', 'state', 'time'], dtype=object)
```

```
In [18]: ufo.columns = ufo.columns.str.replace(' ', '_') #replace all spaces in head
```

```
In [19]: ufo.columns # check that all spaces in column names were replaced with '_'
```

```
Out[19]: Index(['city', 'colors_reported', 'shape_reported', 'state', 'time'], dtype=object)
```

```
In [ ]:
```