

PANDAS - PART 1

Subset of a Data Frame

Subset represents a group of rows and cols from existing Data Frame.

Let us first create a Data Frame using a dictionary as:

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

```
[4]: mydict = {'col1': [1,2,3,4], 'col2': [5,6,7,8], 'col3': ['one', 'two', 'three', 'four']}
```

```
[5]: df = pd.DataFrame(mydict)
df
```

```
[5]:
```

	col1	col2	col3
0	1	5	one
1	2	6	two
2	3	7	three
3	4	8	four

```
[6]: # to know column names
df.columns
```

```
[6]: Index(['col1', 'col2', 'col3'], dtype='object')
```

```
[7]: # to know the row index
df.index
```

```
[7]: RangeIndex(start=0, stop=4, step=1)
```

```
[8]: # to create subset of df with a single col
df[['col1']]      # This is same as df.col1
```

```
[8]:
```

	col1
0	1
1	2
2	3
3	4

```
[9]: # to create subset of df with two cols
df[['col1', 'col3']]
```

```
[9]:
```

	col1	col3
0	1	one
1	2	two
2	3	three
3	4	four

```
[26]: # to create subset of df with one col based on condition
df[df['col1']>1]
```

[26]:

	col1	col2	col3
1	2	6	two
2	3	7	three
3	4	8	four

```
[28]: # to create subset of df with one col based on condition
df[df['col1']>1][df['col2']!=7]
```

```
<ipython-input-28-ce8a08a10467>:2: UserWarning: Boolean Series key will be reindexed to match DataFrame
df[df['col1']>1][df['col2']!=7]
```

[28]:

	col1	col2	col3
1	2	6	two
3	4	8	four

Slicing the data in a Data Frame

We can use `loc[]` or `iloc[]` on the Data Frame to do slicing.

```
[29]: df
```

[29]:

	col1	col2	col3
0	1	5	one
1	2	6	two
2	3	7	three
3	4	8	four

```
[32]: # retrieve rows from 1st to 3rd
df.loc[1:3]
```

[32]:

	col1	col2	col3
1	2	6	two
2	3	7	three
3	4	8	four

```
[34]: # retrieve rows from 1st to 3rd
df.iloc[1:3]
```

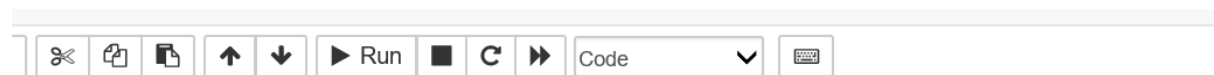
```
[34]:
```

	col1	col2	col3
1	2	6	two
2	3	7	three

```
[39]: # retrieve rows from 1st to 3rd from col1 and col3 only
df.loc[1:3, ['col1', 'col3']]
```

```
[39]:
```

	col1	col3
1	2	two
2	3	three
3	4	four



```
[40]: df
```

```
[40]:
```

	col1	col2	col3
0	1	5	one
1	2	6	two
2	3	7	three
3	4	8	four

```
[44]: # retrieve rows of col1 based on condition
df.loc[df['col1']<3] #df.loc[df.col1<3]
```

```
[44]:
```

	col1	col2	col3
0	1	5	one
1	2	6	two

Retrieving rows based on query

We can use `query()` method on the Data Frame object.

```
[46]: df
```

```
[46]:
```

	col1	col2	col3
0	1	5	one
1	2	6	two
2	3	7	three
3	4	8	four

```
[48]: df.query('col1>2')
```

```
[48]:
```

	col1	col2	col3
2	3	7	three
3	4	8	four

```
[49]: df.query('2*col1>=col2')
```

```
[49]:
```

	col1	col2	col3
3	4	8	four

```
[53]: df.query('col3 in ["three", "four"]')
```

```
[53]:
```

	col1	col2	col3
2	3	7	three
3	4	8	four

```
[54]: df.query('col2 != [5, 7]')
```

```
[54]:
```

	col1	col2	col3
1	2	6	two
3	4	8	four