

## 8.6 - MultiIndex

July 12, 2017

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

```
In [2]: df = pd.DataFrame(np.random.random((9, 5)),
                           index=pd.MultiIndex.from_product([(1, 2, 3), (4, 5, 6)]))
```

### 0.1 Ambiguous indexing

```
In [3]: df
```

```
Out [3]:
```

		0	1	2	3	4
1	4	0.040562	0.956871	0.277910	0.414775	0.991399
	5	0.241831	0.870444	0.735955	0.363697	0.546716
	6	0.841413	0.694261	0.315920	0.199795	0.204977
2	4	0.206391	0.709051	0.513527	0.185654	0.421419
	5	0.147146	0.566263	0.009528	0.824750	0.918497
	6	0.335040	0.161659	0.148115	0.039292	0.480922
3	4	0.529339	0.242498	0.431645	0.321784	0.342553
	5	0.801569	0.240662	0.802937	0.278161	0.415221
	6	0.673873	0.475225	0.944538	0.576891	0.239283

```
In [4]: df.loc[1, 4]
```

```
Out [4]:
```

0	0.040562
1	0.956871
2	0.277910
3	0.414775
4	0.991399

Name: (1, 4), dtype: float64

```
In [5]: df.loc[(1, 4), :]
```

```
Out [5]:
```

0	0.040562
1	0.956871
2	0.277910
3	0.414775
4	0.991399

Name: (1, 4), dtype: float64

```
In [6]: df.loc[(1, ), 4]
```

```
Out[6]: 4    0.991399  
        5    0.546716  
        6    0.204977  
        Name: 4, dtype: float64
```

## 0.2 Lists are not tuples

```
In [7]: df.loc[(1, 6), 4]
```

```
Out[7]: 0.20497667428651845
```

```
In [9]: df.loc[[1, 6], 4]
```

```
Out[9]: 1  4    0.991399  
        5    0.546716  
        6    0.204977  
        Name: 4, dtype: float64
```