# Working with Tables III: Sorting and Grouping

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

Example:

In [2]:
```
t = pd.DataFrame()
t['Firm']      = ['A','B','C','D','E','F']
t['Sector']    = ['Healthcare','Industrials','Healthcare','Healthcare','Industri
t['Style']     = ['Growth','Growth','Value','Growth','Value','Value']
t['Revenue']   = [800,1600,500,4000,6000,400]
t['Earnings']  = [200,400,100,800,600,5000]
t
```

Out[2]:

| | Firm | Sector | Style | Revenue | Earnings |
|---|---|---|---|---|---|
| **0** | A | Healthcare | Growth | 800 | 200 |
| **1** | B | Industrials | Growth | 1600 | 400 |
| **2** | C | Healthcare | Value | 500 | 100 |
| **3** | D | Healthcare | Growth | 4000 | 800 |
| **4** | E | Industrials | Value | 6000 | 600 |
| **5** | F | Industrials | Value | 400 | 5000 |

Total earnings:

In [3]:
```
t.Earnings.sum()
```

Out[3]:
```
7100
```

Total earnings per sector:

In [4]:
```
t.groupby('Sector').sum()
```

Out[4]:

| Sector | Revenue | Earnings |
|---|---|---|
| **Healthcare** | 5300 | 1100 |
| **Industrials** | 8000 | 6000 |

If we don't want the group column to become the index:

In [5]:
```
t.groupby('Sector', as_index=False).sum()
```

Out[5]:

| | Sector | Revenue | Earnings |
|---|---|---|---|
| **0** | Healthcare | 5300 | 1100 |

| | Sector | Revenue | Earnings |
|---|---|---|---|
| **1** | Industrials | 8000 | 6000 |

Total earnings per sector and style:

In [9]:
```
t.groupby(['Sector','Style'])[['Earnings']].sum()
```

Out[9]:

| | | Earnings |
|---|---|---|
| **Sector** | **Style** | |
| **Healthcare** | **Growth** | 1000 |
| | **Value** | 100 |
| **Industrials** | **Growth** | 400 |
| | **Value** | 5600 |

Unstack this table:

In [10]:
```
t.groupby(['Sector','Style'])[['Earnings']].sum().unstack()
```

Out[10]:

| | Earnings | |
|---|---|---|
| **Style** | **Growth** | **Value** |
| **Sector** | | |
| **Healthcare** | 1000 | 100 |
| **Industrials** | 400 | 5600 |

Or unstack level 0:

In [11]:
```
t.groupby(['Sector','Style'])[['Earnings']].sum().unstack(level=0)
```

Out[11]:

| | Earnings | |
|---|---|---|
| **Sector** | **Healthcare** | **Industrials** |
| **Style** | | |
| **Growth** | 1000 | 400 |
| **Value** | 100 | 5600 |

Sort table by earnings:

In [12]:
```
t.sort_values('Earnings')
```

Out[12]:

| | Firm | Sector | Style | Revenue | Earnings |
|---|---|---|---|---|---|
| **2** | C | Healthcare | Value | 500 | 100 |
| **0** | A | Healthcare | Growth | 800 | 200 |

| | Firm | Sector | Style | Revenue | Earnings |
|---|---|---|---|---|---|
| **1** | B | Industrials | Growth | 1600 | 400 |
| **4** | E | Industrials | Value | 6000 | 600 |
| **3** | D | Healthcare | Growth | 4000 | 800 |
| **5** | F | Industrials | Value | 400 | 5000 |

Sort by sector and earnings:

In [13]:
```
t.sort_values(['Sector','Earnings'])
```

Out[13]:

| | Firm | Sector | Style | Revenue | Earnings |
|---|---|---|---|---|---|
| **2** | C | Healthcare | Value | 500 | 100 |
| **0** | A | Healthcare | Growth | 800 | 200 |
| **3** | D | Healthcare | Growth | 4000 | 800 |
| **1** | B | Industrials | Growth | 1600 | 400 |
| **4** | E | Industrials | Value | 6000 | 600 |
| **5** | F | Industrials | Value | 400 | 5000 |

Change the sorting order:

In [14]:
```
t.sort_values(['Sector','Earnings'], ascending=[True,False])
```

Out[14]:

| | Firm | Sector | Style | Revenue | Earnings |
|---|---|---|---|---|---|
| **3** | D | Healthcare | Growth | 4000 | 800 |
| **0** | A | Healthcare | Growth | 800 | 200 |
| **2** | C | Healthcare | Value | 500 | 100 |
| **5** | F | Industrials | Value | 400 | 5000 |
| **4** | E | Industrials | Value | 6000 | 600 |
| **1** | B | Industrials | Growth | 1600 | 400 |

Select firm with highest earnings per sector:

In [16]:
```
t.sort_values(['Sector','Earnings']).groupby(['Sector']).last()
```

Out[16]:

| Sector | Firm | Style | Revenue | Earnings |
|---|---|---|---|---|
| **Healthcare** | D | Growth | 4000 | 800 |
| **Industrials** | F | Value | 400 | 5000 |