```
In [1]:  import numpy as np
         from pandas import Series, DataFrame
         import pandas as pd
```

```
In [2]:  #Now we'll learn DataFrames

         #Let's get some data to play with. How about the NFL?
         import webbrowser
         website = 'http://en.wikipedia.org/wiki/NFL_win-loss_records'
         webbrowser.open(website)
```

```
Out[2]:  True
```

```
In [3]:  #Copy and read to get data
         nfl_frame = pd.read_clipboard()
```

```
In [5]:  #Show
         nfl_frame
```

Out[5]:

	Rank	Team	Won	Lost	Tied*	Pct.	First Season	Total Games	Conference
0	1	Dallas Cowboys	510	378	6	0.574	1960	894	NFC East
1	2	Chicago Bears	752	563	42	0.570	1920	1357	NFC North
2	3	Green Bay Packers	741	561	37	0.567	1921	1339	NFC North
3	4	Miami Dolphins	443	345	4	0.562	1966	792	AFC East
4	5	Baltimore Ravens	182	143	1	0.560	1996	326	AFC North

```
In [6]:  # We can grab the oclumn names with .columns
         nfl_frame.columns
```

```
Out[6]:  Index([u'Rank', u'Team', u'Won', u'Lost', u'Tied*', u'Pct.', u'First Season',
                u'Total Games', u'Conference'], dtype='object')
```

```
In [22]:    #Lets see some specific data columns
            DataFrame(nfl_frame,columns=['Team','First Season','Total Games'])
```

Out[22]:

	Team	First Season	Total Games
0	Dallas Cowboys	1960	894
1	Chicago Bears	1920	1357
2	Green Bay Packers	1921	1339
3	Miami Dolphins	1966	792
4	Baltimore Ravens	1996	326
5	San Francisco 49ers	1950	1003

```
In [11]:    #What happens if we ask for a column that doesn't exist?
            DataFrame(nfl_frame,columns=['Team','First Season','Total Games','Stadium'])
```

Out[11]:

	Team	First Season	Total Games	Stadium
0	Dallas Cowboys	1960	894	0
1	Chicago Bears	1920	1357	1
2	Green Bay Packers	1921	1339	2
3	Miami Dolphins	1966	792	3
4	Baltimore Ravens	1996	326	4

```
In [13]:    # Call columns
            nfl_frame.columns
```

Out[13]:   Index([u'Rank', u'Team', u'Won', u'Lost', u'Tied*', u'Pct.', u'First Season',
             u'Total Games', u'Conference', u'Stadium'], dtype='object')

```
In [18]:    #We can retrieve individual columns
            nfl_frame.Team
```

Out[18]:   0   Dallas Cowboys
            1   Chicago Bears
            2   Green Bay Packers
            3   Miami Dolphins
            4   Baltimore Ravens
            Name: Team, dtype: object

```
In [19]:  # Or try this method for multiple word columns
          nfl_frame['Total Games']
```

```
Out[19]:  0     894
          1    1357
          2    1339
          3     792
          4     326
          Name: Total Games, dtype: int64
```

```
In [25]:  #We can retrieve rows through indexing
          nfl_frame.ix[3]
```

```
Out[25]:  Rank                        4
          Team            Miami Dolphins
          Won                       443
          Lost                      345
          Tied*                       4
          Pct.                    0.562
          First Season             1966
          Total Games               792
          Conference            AFC East
          Name: 3, dtype: object
```

```
In [26]:  #We can also assign value sto entire columns
          nfl_frame['Stadium']="Levi's Stadium" #Careful with the ' here
```

```
In [28]:  nfl_frame
```

Out[28]:

	Rank	Team	Won	Lost	Tied*	Pct.	First Season	Total Games	Conference	Stadium
0	1	Dallas Cowboys	510	378	6	0.574	1960	894	NFC East	Levi's Stadium
1	2	Chicago Bears	752	563	42	0.570	1920	1357	NFC North	Levi's Stadium
2	3	Green Bay Packers	741	561	37	0.567	1921	1339	NFC North	Levi's Stadium
3	4	Miami Dolphins	443	345	4	0.562	1966	792	AFC East	Levi's Stadium
4	5	Baltimore Ravens	182	143	1	0.560	1996	326	AFC North	Levi's Stadium
5	6	San Francisco 49ers	545	444	14	0.550	1950	1003	NFC West	Levi's Stadium

```
In [9]:  #Putting numbers for stadiums
         nfl_frame["Stadium"] = np.arange(5)

         #Show
         nfl_frame
```

Out[9]:

	Rank	Team	Won	Lost	Tied*	Pct.	First Season	Total Games	Conference	Stadium
0	1	Dallas Cowboys	510	378	6	0.574	1960	894	NFC East	0
1	2	Chicago Bears	752	563	42	0.570	1920	1357	NFC North	1
2	3	Green Bay Packers	741	561	37	0.567	1921	1339	NFC North	2
3	4	Miami Dolphins	443	345	4	0.562	1966	792	AFC East	3
4	5	Baltimore Ravens	182	143	1	0.560	1996	326	AFC North	4

```
In [10]:  # Call columns
          nfl_frame.columns
```

```
Out[10]:  Index([u'Rank', u'Team', u'Won', u'Lost', u'Tied*', u'Pct.', u'First Season',
           u'Total Games', u'Conference', u'Stadium'], dtype='object')
```

```
In [14]:  #Adding a Series to a DataFrame
          stadiums = Series(["Levi's Stadium","AT&T Stadium"],index=[4,0])
```

```
In [15]:  #Now input into the nfl DataFrame
          nfl_frame['Stadium']=stadiums

          #Show
          nfl_frame
```

Out[15]:

	Rank	Team	Won	Lost	Tied*	Pct.	First Season	Total Games	Conference	Stadium
0	1	Dallas Cowboys	510	378	6	0.574	1960	894	NFC East	AT&T Stadium
1	2	Chicago Bears	752	563	42	0.570	1920	1357	NFC North	NaN
2	3	Green Bay Packers	741	561	37	0.567	1921	1339	NFC North	NaN
3	4	Miami Dolphins	443	345	4	0.562	1966	792	AFC East	NaN
4	5	Baltimore Ravens	182	143	1	0.560	1996	326	AFC North	Levi's Stadium

```
In [16]:  #We can also delete columns
          del nfl_frame['Stadium']

          nfl_frame
```

Out[16]:

	Rank	Team	Won	Lost	Tied*	Pct.	First Season	Total Games	Conference
0	1	Dallas Cowboys	510	378	6	0.574	1960	894	NFC East
1	2	Chicago Bears	752	563	42	0.570	1920	1357	NFC North
2	3	Green Bay Packers	741	561	37	0.567	1921	1339	NFC North
3	4	Miami Dolphins	443	345	4	0.562	1966	792	AFC East
4	5	Baltimore Ravens	182	143	1	0.560	1996	326	AFC North

```
In [17]:  #DataFrames can be constructed many ways. Another way is from a dictionary of equ
          data = {'City':['SF','LA','NYC'],
                   'Population':[837000,3880000,8400000]}

          city_frame = DataFrame(data)

          #Show
          city_frame
```

Out[17]:

	City	Population
0	SF	837000
1	LA	3880000
2	NYC	8400000

```
In [40]:  #For full list of ways to create DataFrames from various sources go to teh docume
          website = 'http://pandas.pydata.org/pandas-docs/dev/generated/pandas.DataFrame.ht
          webbrowser.open(website)
```

Out[40]: True

```
In [ ]:
```