In [1]:

```
import pandas as pd
```

In [ ]:

```

```

Reading the data from csv files

In [3]:

```
data= pd.read_csv('addresses.csv')
```

In [4]:

```
data
```

4	5	5	24 Second Avenue	NaN	San Mateo	CA	94401	US
5	6	6	800 Middle Avenue	NaN	Menlo Park	CA	94025-9881	US
6	7	7	500 Arbor Road	NaN	Menlo Park	CA	94025	US
7	8	8	800 Middle Avenue	NaN	Menlo Park	CA	94025-9881	US
8	9	9	2510 Middlefield Road	NaN	Redwood City	CA	94063	US
9	10	10	1044 Middlefield Road	NaN	Redwood City	CA	94063	US
10	11	11	2140 Euclid Avenue.	NaN	Redwood City	CA	94061	US
			1044		Redwood			

In [5]:

```
type(data)
```

Out[5]:

```
pandas.core.frame.DataFrame
```

```
In pandas there are only two types of Data 1)Dataframe, 2)series

To read the data with pandas it will read through Dataframe data type only.
```

```
data.head()---> By default it will give the first five records, and we can also give
the value to get the required number of records.
```

```
data.tail()---> By default it will give the last five records, and we can also give the
value to get the requried number of records.
```

In [6]:

```
data.head()
```

Out[6]:

	id	location_id	address_1	address_2	city	state_province	postal_code	country
0	1	1	2600 Middlefield Road	NaN	Redwood City	CA	94063	US
1	2	2	24 Second Avenue	NaN	San Mateo	CA	94401	US
2	3	3	24 Second Avenue	NaN	San Mateo	CA	94403	US
3	4	4	24 Second Avenue	NaN	San Mateo	CA	94401	US
4	5	5	24 Second Avenue	NaN	San Mateo	CA	94401	US

In [7]:

```
data.head(9)
```

Out[7]:

	id	location_id	address_1	address_2	city	state_province	postal_code	country
0	1	1	2600 Middlefield Road	NaN	Redwood City	CA	94063	US
1	2	2	24 Second Avenue	NaN	San Mateo	CA	94401	US
2	3	3	24 Second Avenue	NaN	San Mateo	CA	94403	US
3	4	4	24 Second Avenue	NaN	San Mateo	CA	94401	US
4	5	5	24 Second Avenue	NaN	San Mateo	CA	94401	US
5	6	6	800 Middle Avenue	NaN	Menlo Park	CA	94025-9881	US
6	7	7	500 Arbor Road	NaN	Menlo Park	CA	94025	US
7	8	8	800 Middle Avenue	NaN	Menlo Park	CA	94025-9881	US
8	9	9	2510 Middlefield Road	NaN	Redwood City	CA	94063	US

In [8]:

```
data.tail()
```

Out[8]:

	id	location_id	address_1	address_2	city	state_province	postal_code	country
16	17	17	409 South Spruce Avenue	NaN	South San Francisco	CA	94080	US
17	18	18	114 Fifth Avenue	NaN	Redwood City	CA	94063	US
18	19	19	19 West 39th Avenue	NaN	San Mateo	CA	94403	US
19	20	21	123 El Camino Real	NaN	Belmont	CA	94002	US
20	21	22	2013 Avenue of the fellows	Suite 100	San Francisco	CA	94103	US

In [9]:

```
data.tail(8)
```

Out[9]:

	id	location_id	address_1	address_2	city	state_province	postal_code	country
13	14	14	660 Veterans Blvd.	NaN	Redwood City	CA	94063	US
14	15	15	1500 Valencia Street	NaN	San Francisco	CA	94110	US
15	16	16	1161 South Bernardo	NaN	Sunnyvale	CA	94087	US
16	17	17	409 South Spruce Avenue	NaN	South San Francisco	CA	94080	US
17	18	18	114 Fifth Avenue	NaN	Redwood City	CA	94063	US
18	19	19	19 West 39th Avenue	NaN	San Mateo	CA	94403	US
19	20	21	123 El Camino Real	NaN	Belmont	CA	94002	US
20	21	22	2013 Avenue of the fellows	Suite 100	San Francisco	CA	94103	US

In [11]:

```
# To get the column names from dataset

data.columns
```

Out[11]:

```
Index(['id', 'location_id', 'address_1', 'address_2', 'city', 'state_provi
nce',
       'postal_code', 'country'],
      dtype='object')
```

In [13]:

```
# To change the data type into list...

list(data.columns)
```

Out[13]:

```
['id',
 'location_id',
 'address_1',
 'address_2',
 'city',
 'state_province',
 'postal_code',
 'country']
```

```
Dataframe:- A Data frame is a two-dimensional data structure, i.e., data is aligned in
a tabular fashion in rows and columns.               Pandas DataFrame consists of three
principal components, the data, rows, and columns. DataFrame is two-dimensional
      size-mutable, potentially heterogeneous tabular data structure with labeled axes
(rows and columns).
```

In [14]:

```
# if we want to select any particular columns then

data[['id','address_2']]
```

Out[14]:

	id	address_2
0	1	NaN
1	2	NaN
2	3	NaN
3	4	NaN
4	5	NaN
5	6	NaN
6	7	NaN
7	8	NaN
8	9	NaN
9	10	NaN
10	11	NaN
11	12	2nd Floor
12	13	NaN
13	14	NaN
14	15	NaN
15	16	NaN
16	17	NaN
17	18	NaN
18	19	NaN
19	20	NaN
20	21	Suite 100

In [16]:

```
data[['city','address_2']]
```

Out[16]:

	city	address_2
0	Redwood City	NaN
1	San Mateo	NaN
2	San Mateo	NaN
3	San Mateo	NaN
4	San Mateo	NaN
5	Menlo Park	NaN
6	Menlo Park	NaN
7	Menlo Park	NaN
8	Redwood City	NaN
9	Redwood City	NaN
10	Redwood City	NaN
11	Redwood City	2nd Floor
12	Redwood City	NaN
13	Redwood City	NaN
14	San Francisco	NaN
15	Sunnyvale	NaN
16	South San Francisco	NaN
17	Redwood City	NaN
18	San Mateo	NaN
19	Belmont	NaN
20	San Francisco	Suite 100

# Reading data from the excel sheets

In [18]:

```
pd.read_excel('airline.xls')
```

Out[18]:

	YEAR	Y	W	R	L	K
0	1948	1.214	0.243	0.1454	1.415	0.612
1	1949	1.354	0.260	0.2181	1.384	0.559
2	1950	1.569	0.278	0.3157	1.388	0.573
3	1951	1.948	0.297	0.3940	1.550	0.564
4	1952	2.265	0.310	0.3559	1.802	0.574
5	1953	2.731	0.322	0.3593	1.926	0.711
6	1954	3.025	0.335	0.4025	1.964	0.776
7	1955	3.562	0.350	0.3961	2.116	0.827
8	1956	3.979	0.361	0.3822	2.435	0.800
9	1957	4.420	0.379	0.3045	2.707	0.921
10	1958	4.563	0.391	0.3284	2.706	1.067
11	1959	5.385	0.426	0.3856	2.846	1.083
12	1960	5.554	0.441	0.3193	3.089	1.481
13	1961	5.465	0.460	0.3079	3.122	1.736
14	1962	5.825	0.485	0.3783	3.184	1.926
15	1963	6.876	0.506	0.4180	3.263	2.041
16	1964	7.823	0.538	0.5163	3.412	1.997
17	1965	9.120	0.564	0.5879	3.623	2.257
18	1966	10.512	0.586	0.5369	4.074	2.742
19	1967	13.020	0.622	0.4443	4.710	3.564
20	1968	15.261	0.666	0.3052	5.217	4.767
21	1969	16.313	0.731	0.2332	5.569	6.511
22	1970	16.002	0.831	0.1883	5.495	7.627
23	1971	15.876	0.906	0.2023	5.334	8.673
24	1972	16.662	1.000	0.2506	5.345	8.331
25	1973	17.014	1.056	0.2668	5.662	8.557
26	1974	19.305	1.131	0.2664	5.729	9.508
27	1975	18.721	1.247	0.2301	5.722	9.062
28	1976	19.250	1.375	0.3452	5.762	8.262
29	1977	20.647	1.544	0.4508	5.877	7.474
30	1978	22.726	1.703	0.5877	6.108	7.104
31	1979	23.619	1.779	0.5346	6.852	6.874

when we are working with excel files, there may be a more than one sheet in the excel files.

In [23]:

```
pd.read_excel('airline1.xls', sheet_name = 'Sheet2')
```

Out[23]:

```
   YEAR      Y      W       R      L      K
0  1948  1.214  0.243  0.1454  1.415  0.612
1  1949  1.354  0.260  0.2181  1.384  0.559
2  1950  1.569  0.278  0.3157  1.388  0.573
3  1951  1.948  0.297  0.3940  1.550  0.564
4  1952  2.265  0.310  0.3559  1.802  0.574
5  1953  2.731  0.322  0.3593  1.926  0.711
6  1954  3.025  0.335  0.4025  1.964  0.776
7  1955  3.562  0.350  0.3961  2.116  0.827
8  1956  3.979  0.361  0.3822  2.435  0.800
```

In [28]:

```
data1 = pd.read_excel('airline1.xls', sheet_name = 'Sheet3',header=None, names=['a','b',
```

In [30]:

```
data1.to_csv('test1.csv', index=False) # Here i am going to convert the fileformat and a
```

In [33]:

```
data1.to_excel('test2.xlsx', index = False)
```

In [34]:

```
data1.head()
```

Out[34]:

```
      a      b      c       d      e      f
0  1960  5.554  0.441  0.3193  3.089  1.481
1  1961  5.465  0.460  0.3079  3.122  1.736
2  1962  5.825  0.485  0.3783  3.184  1.926
3  1963  6.876  0.506  0.4180  3.263  2.041
4  1964  7.823  0.538  0.5163  3.412  1.997
```

In [35]:

```
data1.columns
```

Out[35]:

```
Index(['a', 'b', 'c', 'd', 'e', 'f'], dtype='object')
```

In [37]:

```
list(data1.columns)
```

Out[37]:

```
['a', 'b', 'c', 'd', 'e', 'f']
```

Reading the data directly from the github

In [38]:

```
pd.read_csv('https://raw.githubusercontent.com/datasciencedojo/datasets/master/titanic.c
```

Out[38]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fa
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.25
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.28
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.92
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.10
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.05
...	...	...	...	...	...	...	...	...	...	
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.00
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.00
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.45
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.00
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.75

891 rows × 12 columns

Reading the data directly from the website

In [41]:

```
data2 = pd.read_html('https://www.basketball-reference.com/leagues/NBA_2015_totals.html'
```

In [42]:

```
len(data2)
```

Out[42]:

```
1
```

In [43]:

```
data2[0]
```

Out[43]:

	Rk	Player	Pos	Age	Tm	G	GS	MP	FG	FGA	...	FT%	ORB	DRB	TRB
0	1	Quincy Acy	PF	24	NYK	68	22	1287	152	331	...	.784	79	222	301
1	2	Jordan Adams	SG	20	MEM	30	0	248	35	86	...	.609	9	19	28
2	3	Steven Adams	C	21	OKC	70	67	1771	217	399	...	.502	199	324	523
3	4	Jeff Adrien	PF	28	MIN	17	0	215	19	44	...	.579	23	54	77
4	5	Arron Afflalo	SG	29	TOT	78	72	2502	375	884	...	.843	27	220	247
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
670	490	Thaddeus Young	PF	26	TOT	76	68	2434	451	968	...	.655	127	284	411
671	490	Thaddeus Young	PF	26	MIN	48	48	1605	289	641	...	.682	75	170	245
672	490	Thaddeus Young	PF	26	BRK	28	20	829	162	327	...	.606	52	114	166
673	491	Cody Zeller	C	22	CHO	62	45	1487	172	373	...	.774	97	265	362
674	492	Tyler Zeller	C	25	BOS	82	59	1731	340	619	...	.823	146	319	465

675 rows × 30 columns

In [44]:

```
data2[0].to_csv('players.csv')
```

In [ ]:

```

```