1. Import CSV files

It is important to note that **a single backslash does not work** when specifying the file path**.** You need to either **change it to forward slash or add one more backslash** like below

```
import pandas as pd
mydata= pd.read_csv("C:\\Users\\Deepanshu\\Documents\\file1.csv")
```

If no header (title) in raw data file

```
mydata1  = pd.read_csv("C:\\Users\\Deepanshu\\Documents\\file1.csv", header = None)
```

You need to include **header = None** option to tell Python there is no column name (header) in data.

Add Column Names

We can include column names by using names= option.

```
mydata2  = pd.read_csv("C:\\Users\\Deepanshu\\Documents\\file1.csv", header = None, names = ['ID', 'first_name', 'salary'])
```

The variable names can also be added separately by using the following command.

```
mydata1.columns = ['ID', 'first_name', 'salary']
```

Create sample data for import

```
dt = {'ID': [11, 12, 13, 14, 15],
            'first_name': ['David', 'Jamie', 'Steve', 'Stevart', 'John'],
            'company': ['Aon', 'TCS', 'Google', 'RBS', '.'],
            'salary': [74, 76, 96, 71, 78]}
mydt = pd.DataFrame(dt, columns = ['ID', 'first_name', 'company', 'salary'])
```

Save data as CSV in the working directory

The following command tells python to write data in CSV format.

```
mydt.to_csv('workingfile.csv', index=False)
```

Example 1 : Read CSV file with header row

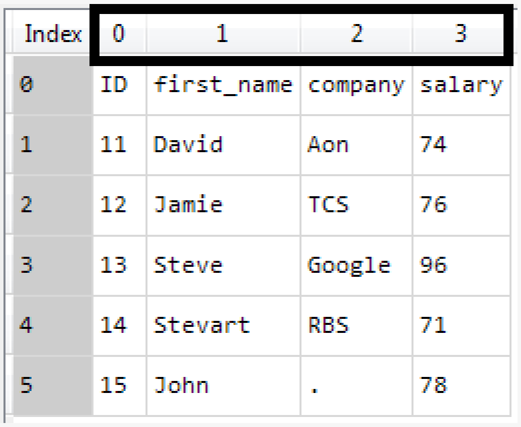
It's the basic syntax of read_csv() function. You just need to mention the filename.

```
mydata  = pd.read_csv("workingfile.csv")
```

Example 2 : Read CSV file without header row

```
mydata0  = pd.read_csv("workingfile.csv", header = None)
```

If you specify **"header = None",** python would assign a series of numbers starting from 0 to (number of columns - 1). See the output shown below -



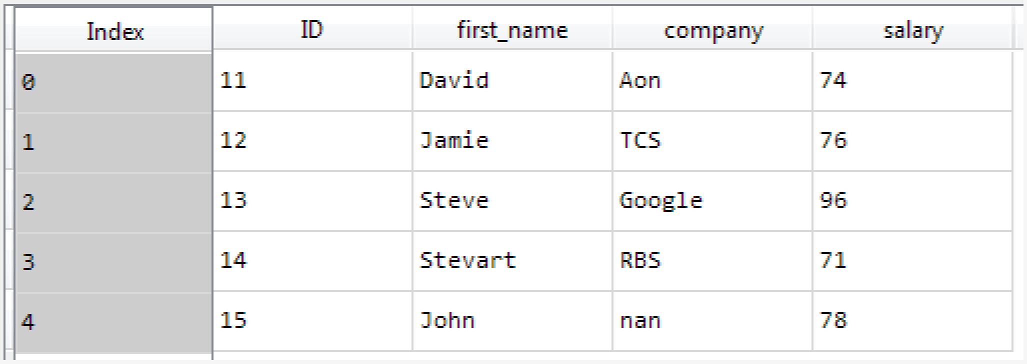
Index	0	1	2	3
0	ID	first_name	company	salary
1	11	David	Aon	74
2	12	Jamie	TCS	76
3	13	Steve	Google	96
4	14	Stevart	RBS	71
5	15	John	.	78

Output

Example 3 : Specify missing values

The na_values= options is used to set some values as blank / missing values.

```
mydata00  = pd.read_csv("workingfile.csv", na_values=['.'])
```

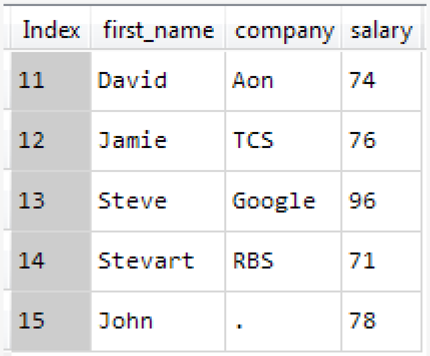


Index	ID	first_name	company	salary
0	11	David	Aon	74
1	12	Jamie	TCS	76
2	13	Steve	Google	96
3	14	Stevart	RBS	71
4	15	John	nan	78

Set Missing Values

Example 4 : Set Index Column to ID

```
mydata01  = pd.read_csv("workingfile.csv", index_col ='ID')
```



Index	first_name	company	salary
11	David	Aon	74
12	Jamie	TCS	76
13	Steve	Google	96
14	Stevart	RBS	71
15	John	.	78

Python : Setting Index Column

As you can see in the above image, the column ID has been set as index column.

Example 5 : Read CSV File from URL

You can directly read data from the CSV file that is stored on a web link.

```
mydata02  = pd.read_csv("http://winterolympicsmedals.com/medals.csv")
```

Example 6 : Skip First 5 Rows While Importing CSV

```
mydata03  = pd.read_csv("http://winterolympicsmedals.com/medals.csv",
skiprows=5)
```

It reads data from 6th row (6th row would be a header row)

Example 7 : Skip Last 5 Rows While Importing CSV

```
mydata04  = pd.read_csv("http://winterolympicsmedals.com/medals.csv",
skip_footer=5)
```

It excludes last5 rows.

Example 8 : Read only first 5 rows

```
mydata05  = pd.read_csv("http://winterolympicsmedals.com/medals.csv",
nrows=5)
```

Example 9 : Read only specific columns

```
mydata07 = pd.read_csv("http://winterolympicsmedals.com/medals.csv",
usecols=(1,5,7))
```

The above code reads only columns placed at first, fifth and seventh position.

Example 10 : Read some rows and columns

```
mydata08 = pd.read_csv("http://winterolympicsmedals.com/medals.csv",
usecols=(1,5,7),nrows=5)
```