

```
In [3]: import pandas as pd
```

```
In [ ]: TASK
```

we will test our jupyter notebook functionality to write data to file **and** also read data

```
In [ ]: # Open C drive
# create a folder and name it sandbox
# Create a sample dataframe
```

```
In [5]: # Create a sample dataframe
```

```
In [ ]: #lets create a simple dataframe(table)

Tunde has 5 students in his class taking 3 subjects ( chem, Bio, math).
He has their scores as a list and wants us to save them to a dataframe
```

```
In [9]: SNames = ['Chioma', 'Joshua', 'Moses', 'Prince', 'Yusuf']
mathScore = [87,77,90,67,80]
chemScore = [57,73,84,77,95]
bioScore = [30,58,88,69,55]

# Converting the above records into a dataframe

# create an empty dataframe

studResults = pd.DataFrame()

# populate the dataframe

studResults['Name'] = SNames
studResults['Math'] = mathScore
studResults['Chem'] = chemScore
studResults['Bio'] = bioScore

print(studResults)
```

	Name	Math	Chem	Bio
0	Chioma	87	57	30
1	Joshua	77	73	58
2	Moses	90	84	88
3	Prince	67	77	69
4	Yusuf	80	95	55

```
In [ ]: # save studResults to sandbox folder as a csv file
# to save the file we need to use the pandas function to_csv()
#SYNTAX
#dataframe.to_csv(full path to save the file including file name and extension)
```

```
In [24]: studResults.to_csv('c:/sandbox/studResults.csv')
```

```
In [25]: studResults.to_excel('c:/sandbox/studResults.xls')
```

<ipython-input-25-39b2ffaa1942>:1: FutureWarning: As the xlwt package is no longer maintained, the xlwt engine will be removed in a future version of pandas. This is the only engine in pandas that supports writing in the xls format. Install openpyxl and write to a n xlsx file instead. You can set the option io.excel.xls.writer to 'xlwt' to silence this warning. While this option is deprecated and will also raise a warning, it can be globally set and the warning suppressed.

```
studResults.to_excel('c:/sandbox/studResults.xls')
```

```
In [ ]: studResults.to_csv('c:/studResults.csv')
```

```
In [26]: # Read back the file
```

```
df = pd.read_csv('c:/sandbox/studResults.csv')
```

```
In [27]: df
```

```
Out[27]:
```

	Unnamed: 0	Name	Math	Chem	Bio
0	0	Chioma	87	57	30
1	1	Joshua	77	73	58
2	2	Moses	90	84	88
3	3	Prince	67	77	69
4	4	Yusuf	80	95	55

```
In [ ]: # subsetting a dataframe
```

```
In [16]: # get the names of the columns in your dataframe
```

```
df.columns
```

```
Out[16]: Index(['Unnamed: 0', 'Name', 'Math', 'Chem', 'Bio'], dtype='object')
```

```
In [17]: # converting to a list
```

```
headerList = list(df.columns)
headerList
```

```
Out[17]: ['Unnamed: 0', 'Name', 'Math', 'Chem', 'Bio']
```

```
In [18]: # extracting cols 'Name', 'Math', 'Chem', 'Bio'
```

```
subDF = df[['Name', 'Math', 'Chem', 'Bio']]
subDF
```

```
Out[18]:
```

	Name	Math	Chem	Bio
0	Chioma	87	57	30
1	Joshua	77	73	58
2	Moses	90	84	88
3	Prince	67	77	69
4	Yusuf	80	95	55

```
In [ ]: # get the names of the columns in your dataframe

df.columns

# converting to a List

headerList = list(df.columns)
headerList

# extracting cols 'Name', 'Math', 'Chem', 'Bio'

subDF = df[['Name', 'Math', 'Chem', 'Bio']]
subDF
```

```
In [ ]: diab = pd.read_csv('c:/sandbox/diabetes.csv')
```

```
In [19]: df
```

```
Out[19]:
```

	Unnamed: 0	Name	Math	Chem	Bio
0	0	Chioma	87	57	30
1	1	Joshua	77	73	58
2	2	Moses	90	84	88
3	3	Prince	67	77	69
4	4	Yusuf	80	95	55

```
In [21]: df.columns
```

```
Out[21]: Index(['Unnamed: 0', 'Name', 'Math', 'Chem', 'Bio'], dtype='object')
```

```
In [22]: diab = pd.read_csv('c:/sandbox/diabetes.csv')
```

```
In [29]: diab
```

Out[29]:

	ID	No_Pation	Gender	AGE	Urea	Cr	HbA1c	Chol	TG	HDL	LDL	VLDL	BMI	CLASS
0	502	17975	F	50	4.7	46	4.9	4.2	0.9	2.4	1.4	0.5	24.0	N
1	735	34221	M	26	4.5	62	4.9	3.7	1.4	1.1	2.1	0.6	23.0	N
2	420	47975	F	50	4.7	46	4.9	4.2	0.9	2.4	1.4	0.5	24.0	N
3	680	87656	F	50	4.7	46	4.9	4.2	0.9	2.4	1.4	0.5	24.0	N
4	504	34223	M	33	7.1	46	4.9	4.9	1.0	0.8	2.0	0.4	21.0	N
...
995	200	454317	M	71	11.0	97	7.0	7.5	1.7	1.2	1.8	0.6	30.0	Y
996	671	876534	M	31	3.0	60	12.3	4.1	2.2	0.7	2.4	15.4	37.2	Y
997	669	87654	M	30	7.1	81	6.7	4.1	1.1	1.2	2.4	8.1	27.4	Y
998	99	24004	M	38	5.8	59	6.7	5.3	2.0	1.6	2.9	14.0	40.5	Y
999	248	24054	M	54	5.0	67	6.9	3.8	1.7	1.1	3.0	0.7	33.0	Y

1000 rows × 14 columns

In []:

```
diab = pd.read_csv('c:/sandbox/diabetes.csv')
```