**Data Input and Output Classwork https://pandas.pydata.org/pandas-docs/stable/reference/io.html (https://pandas.pydata.org/pandas-docs/stable/reference/io.html)

**Data input and output exercises

In [39]:
```
#import the required libraries to read various files using pandas
import pandas as pd
import numpy as np
print("I am Preston")
```

```
I am Preston
```

In [ ]:
```

```

In [40]:
```
#let us display the file titanic.csv using the system commands
import sys
!type data\titanic.csv
print("I am Preston")
```

```
PassengerId,Survived,Pclass,Sex,Age,SibSp,Parch,Ticket,Fare,Cabin,Embarked
1,0,3,male,22.0,1,0,A/5 21171,7.25,,S
2,1,1,female,38.0,1,0,PC 17599,71.2833,C85,C
3,1,3,female,26.0,0,0,STON/O2. 3101282,7.925,,S
4,1,1,female,35.0,1,0,113803,53.1,C123,S
5,0,3,male,35.0,0,0,373450,8.05,,S
6,0,3,male,,0,0,330877,8.4583,,Q
7,0,1,male,54.0,0,0,17463,51.8625,E46,S
8,0,3,male,2.0,3,1,349909,21.075,,S
9,1,3,female,27.0,0,2,347742,11.1333,,S
10,1,2,female,14.0,1,0,237736,30.0708,,C
11,1,3,female,4.0,1,1,PP 9549,16.7,G6,S
12,1,1,female,58.0,0,0,113783,26.55,C103,S
13,0,3,male,20.0,0,0,A/5. 2151,8.05,,S
14,0,3,male,39.0,1,5,347082,31.275,,S
15,0,3,female,14.0,0,0,350406,7.8542,,S
16,1,2,female,55.0,0,0,248706,16.0,,S
17,0,3,male,2.0,4,1,382652,29.125,,Q
18,1,2,male,,0,0,244373,13.0,,S
```

In [42]:
```
# read the titanic csv file into a dataframe. Name the data frame df
df=pd.read_csv('data/titanic.csv')
df.head()



#show only the head of the dataframe df
```

Out[42]:

	PassengerId	Survived	Pclass	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarke
0	1	0	3	male	22.0	1	0	A/5 21171	7.2500	NaN	
1	2	1	1	female	38.0	1	0	PC 17599	71.2833	C85	
2	3	1	3	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	
3	4	1	1	female	35.0	1	0	113803	53.1000	C123	
4	5	0	3	male	35.0	0	0	373450	8.0500	NaN	

In [43]:
```
print("I am Preston")
```

```
I am Preston
```

In [7]:
```
#specifying 10 of rows to be read starting from the top
pd.read_csv('data/titanic.csv',nrows=10)
```

Out[7]:

	PassengerId	Survived	Pclass	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarke
0	1	0	3	male	22.0	1	0	A/5 21171	7.2500	NaN	
1	2	1	1	female	38.0	1	0	PC 17599	71.2833	C85	
2	3	1	3	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	
3	4	1	1	female	35.0	1	0	113803	53.1000	C123	
4	5	0	3	male	35.0	0	0	373450	8.0500	NaN	
5	6	0	3	male	NaN	0	0	330877	8.4583	NaN	
6	7	0	1	male	54.0	0	0	17463	51.8625	E46	
7	8	0	3	male	2.0	3	1	349909	21.0750	NaN	
8	9	1	3	female	27.0	0	2	347742	11.1333	NaN	
9	10	1	2	female	14.0	1	0	237736	30.0708	NaN	

In [44]:
```
print("I am Preston")
```

```
I am Preston
```

In [74]:
```
#read the titantic.csv file but skip 10 rows at the top while reading it and assi
df3=pd.read_csv('data/titanic.csv',skiprows=[I for I in range(1,10)])
df3.head()


#display onl the head of the data frame df3
```

Out[74]:

	PassengerId	Survived	Pclass	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	10	1	2	female	14.0	1	0	237736	30.0708	NaN	C
1	11	1	3	female	4.0	1	1	PP 9549	16.7000	G6	S
2	12	1	1	female	58.0	0	0	113783	26.5500	C103	S
3	13	0	3	male	20.0	0	0	A/5. 2151	8.0500	NaN	S
4	14	0	3	male	39.0	1	5	347082	31.2750	NaN	S

In [45]:
```
print("I am Preston")
```

```
I am Preston
```

In [9]:
```
#show only PassengerId, Survived, Sex , Age, Ticket columns. Save it to a datafra

df=pd.read_csv('data/titanic.csv',header=0,usecols=['PassengerId','Survived','Sex
df.head()
#Show only the head of the dataframe df
```

Out[9]:

	PassengerId	Survived	Sex	Age	Ticket
0	1	0	male	22.0	A/5 21171
1	2	1	female	38.0	PC 17599
2	3	1	female	26.0	STON/O2. 3101282
3	4	1	female	35.0	113803
4	5	0	male	35.0	373450

In [46]:
```
print("I am Preston")
```

```
I am Preston
```

In [38]:
```
# Handling missing values is an important and frequently nuanced part of the file
# Missing data is usually either not present (empty string) or marked by some ser
# By default, pandas uses a set of commonly occurring sentinels, such as NA and N
pd.isnull(df)
df.head()
```

Out[38]:

	Unnamed: 0	Currency	Code	Bank Selling	Bank Buying
0	NaN	EURO	EUR	NaN	NaN
1	NaN	BRITISH POUND	GBP	NaN	NaN
2	NaN	UNITED STATES DOLLAR	USD	NaN	NaN

In [47]:
```
print("I am Preston")
```

```
I am Preston
```

In [11]:
```
# The na_values option can take either a list or set of strings to consider missi

df3=pd.read_csv('data/titanic.csv', na_values=['NULL'])
df3.head()

#show only the head of the dataframe
```

Out[11]:

	PassengerId	Survived	Pclass	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarke
0	1	0	3	male	22.0	1	0	A/5 21171	7.2500	NaN	
1	2	1	1	female	38.0	1	0	PC 17599	71.2833	C85	
2	3	1	3	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	
3	4	1	1	female	35.0	1	0	113803	53.1000	C123	
4	5	0	3	male	35.0	0	0	373450	8.0500	NaN	

In [48]:
```
print("I am Preston")
```

```
I am Preston
```

**If I want to read any file. I type pd.read_ and click the tab. It will show me drop down menu of the variety of formats that pandas can read from. Pandas can read from clipboard, csv, excel, hdf, html, json, pickle, sas, sql, sql_query, sql_table,stata, and more. Let us see how can we use read_table to read a csv file

In [12]:
```
#use read_table to read the titanic.csv file. Save it to a dataframe call the dat

df1=pd.read_table('data/titanic.csv',sep=',')
df1.head()

#Show only the head of the dataframe df1
```

Out[12]:

	PassengerId	Survived	Pclass	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarke
0	1	0	3	male	22.0	1	0	A/5 21171	7.2500	NaN	
1	2	1	1	female	38.0	1	0	PC 17599	71.2833	C85	
2	3	1	3	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	
3	4	1	1	female	35.0	1	0	113803	53.1000	C123	
4	5	0	3	male	35.0	0	0	373450	8.0500	NaN	

In [49]:
```
print("I am Preston")
```

```
I am Preston
```

In [13]:
```
#write dataframe df to a csv file name it myTitanic.csv. Save it at data subfolde
df.to_csv('data/myTitanic.csv',index=False)
```

In [50]:
```
print("I am Preston")
```

```
I am Preston
```

In [14]:
```
#Let us see the myTitanic.csv using the type command
!type data\myTitanic.csv
import sys
```

```
5,0,male,35.0,373450
6,0,male,,330877
7,0,male,54.0,17463
8,0,male,2.0,349909
9,1,female,27.0,347742
10,1,female,14.0,237736
11,1,female,4.0,PP 9549
12,1,female,58.0,113783
13,0,male,20.0,A/5. 2151
14,0,male,39.0,347082
15,0,female,14.0,350406
16,1,female,55.0,248706
17,0,male,2.0,382652
18,1,male,,244373
19,0,female,31.0,345763
20,1,female,,2649
21,0,male,35.0,239865
22,1,male,34.0,248698
23,1,female,15.0,330923
24,1,male,28.0,113788
```

In [51]:
```
print("I am Preston")
```

```
I am Preston
```

Reading JSON

In [15]:
```
# Here is an example of what a JSON (JavaScript Object Notation) looks like:
obj = """
{
   "article": [

      {
         "id":"01",
         "language": "JSON",
         "edition": "first",
         "author": "Derrick Mwiti"
      },

      {
         "id":"02",
         "language": "Python",
         "edition": "second",
         "author": "Derrick Mwiti"
      }
   ],

   "blog":[
   {
       "name": "Datacamp",
       "URL":"datacamp.com"
   }
   ]
}
"""
```

In [16]:
```
#add required import statements

import numpy as np
from pandas import Series, DataFrame
import pandas as pd

#Let import json module
import json
#Let import json module
```

In [52]:
```
print("I am Preston")
```

```
I am Preston
```

In [17]:
```
# - To convert a JSON string to Python form, use json.loads:
result=json.loads(obj)
result
```

Out[17]:
```
{'article': [{'id': '01',
   'language': 'JSON',
   'edition': 'first',
   'author': 'Derrick Mwiti'},
  {'id': '02',
   'language': 'Python',
   'edition': 'second',
   'author': 'Derrick Mwiti'}],
 'blog': [{'name': 'Datacamp', 'URL': 'datacamp.com'}]}
```

In [53]:
```
print("I am Preston")
```

```
I am Preston
```

In [18]:
```
# json.dumps, on the other hand, converts a Python object back to JSON: use dumps

asjon=json.dumps(result)
```

In [55]:
```
print("I am Preston")
```

```
I am Preston
```

In [54]:
```
# How you convert a JSON object or list of objects to a DataFrame or some other d
# Conveniently, you can pass a list of dicts (which were previously JSON objects)
# constructor and select a subset of the data fields: Let us say blog
blog=pd.DataFrame(result['blog'], columns=['name','URL'])
blog
```

Out[54]:

	name	URL
0	Datacamp	datacamp.com

In [56]:
```
print("I am Preston")
```

```
I am Preston
```

**Working with Excel Files

In [20]:
```
#load sheet 2002 of battledeath.xlsx to a dataframe call it df1

#import required libraries


#read the required sheet of the excel file to a dataframe df1
import numpy as np
import pandas as pd
df1=pd.read_excel('data/battledeath.xlsx',sheet_name='2002')
df1


#show the data frame df1
```

Out[20]:

	War, age-adjusted mortality due to	2002
0	Afghanistan	36.083990
1	Albania	0.128908
2	Algeria	18.314120
3	Andorra	0.000000
4	Angola	18.964560
...	...	...
187	Venezuela	0.000000
188	Vietnam	0.040222
189	Yemen, Rep.	0.074510
190	Zambia	0.044548
191	Zimbabwe	33.796200

192 rows × 2 columns

In [57]:
```
print("I am Preston")
```

```
I am Preston
```

In [21]:
```
# Open the excel file data/Lec_28_test.xlsx as an object

xlsfile=pd.ExcelFile('data/Lec_28_test.xlsx')
```

In [58]:
```
print("I am Preston")
```

```
I am Preston
```

In [22]:
```
# Parse the first sheet of the excel file and set as DataFrame
dframe=xlsfile.parse('Sheet1')
```

In [59]:
```
print("I am Preston")
```

```
I am Preston
```

In [23]:
```
#Show dataframe

dframe
```

Out[23]:

	This is a test	Unnamed: 1	Unnamed: 2
0	23	6678	456
1	234	678	456
2	234	7	345
3	34	56	234
4	5	456	4365

In [60]:
```
print("I am Preston")
```

```
I am Preston
```

In [24]:
```
# Print sheet names

print(xlsfile.sheet_names)
```

```
['Sheet1', 'Sheet2', 'Sheet3']
```

In [61]:
```
print("I am Preston")
```

```
I am Preston
```

**Excel Output:

In [25]:
```
#write the line of code to  output the dataframe df to a sheet named sheet1 in th

df.to_excel('data/Excel_Sample2.xlsx',sheet_name='sheet1')
```

In [62]:
```
print("I am Preston")
```

```
I am Preston
```

***Retrieve information from html

**HTML Input Pandas read_html function will read tables off of a webpage and return a list of DataFrame objects:

In [26]:
```
#retrieve infromation from the following html https://www.fnb.co.za/Controller?nc
#read the html file using pandas and save the result in data
data = pd.read_html('https://www.fnb.co.za/Controller?nav=rates.forex.list.ForexF
```

In [63]:
```
print("I am Preston")
```

```
I am Preston
```

**Note this does not directly create a dataframe. If we checked the data type of df that is produced, you will find that it is a list. This is essentially what pandas tried to do was to find every table element that was in the html file. Try the following on the html page mentioned above. Right click on the page and view page source you'll see table references. So pandas will make a list of those table and make a list of them and convert each item in the list into a data frame. So to get the first table we can use the index zero

In [27]:
```
#display the data type of data
type(data)
```

Out[27]:
```
list
```

In [64]:
```
print("I am Preston")
```

```
I am Preston
```

In [37]:
```
#convert the result to dataframe
df=data[0]
df=pd.DataFrame(data[0])
df.head()

#show the dataframe
```

Out[37]:

	Unnamed: 0	Currency	Code	Bank Selling	Bank Buying
0	NaN	EURO	EUR	NaN	NaN
1	NaN	BRITISH POUND	GBP	NaN	NaN
2	NaN	UNITED STATES DOLLAR	USD	NaN	NaN

In [65]:
```
print("I am Preston")
```

```
I am Preston
```

In [29]:
```
# let us say you want to know the columns of this dataframe
df.columns.values
```

Out[29]:
```
array(['Unnamed: 0', 'Currency', 'Code', 'Bank Selling', 'Bank Buying'],
      dtype=object)
```

In [66]:
```
print("I am Preston")
```

```
I am Preston
```

In [ ]:
```

```