For reading files there arte two methods: 1. reading the file directly 2. changing the working directory to the path the file exist and reading the file.

first method (Reading the file directly)

In [ ]:
```
import pandas as pd
stats = pd.read_csv('E:\Agile\Python Data Science\P4-Demographic-Data.csv')
```

In [ ]:
```
stats
```

Out[ ]:

	Country Name	Country Code	Birth rate	Internet users	Income Group
0	Aruba	ABW	10.244	78.9	High income
1	Afghanistan	AFG	35.253	5.9	Low income
2	Angola	AGO	45.985	19.1	Upper middle income
3	Albania	ALB	12.877	57.2	Upper middle income
4	United Arab Emirates	ARE	11.044	88.0	High income
...	...	...	...	...	...
190	Yemen, Rep.	YEM	32.947	20.0	Lower middle income
191	South Africa	ZAF	20.850	46.5	Upper middle income
192	Congo, Dem. Rep.	COD	42.394	2.2	Low income
193	Zambia	ZMB	40.471	15.4	Lower middle income
194	Zimbabwe	ZWE	35.715	18.5	Low income

195 rows × 5 columns

Second method is to change the directory

In [ ]:
```
import os
paththh = os.getcwd()
print(paththh)
```

```
e:\Agile\Python Data Science
```

In [ ]:
```
os.chdir('E:\Agile\Python Data Science')
stats2 = pd.read_csv('P4-Demographic-Data.csv')
```

In [ ]:
```
stats2
```

Out[ ]:

	Country Name	Country Code	Birth rate	Internet users	Income Group

	Country Name	Country Code	Birth rate	Internet users	Income Group
0	Aruba	ABW	10.244	78.9	High income
1	Afghanistan	AFG	35.253	5.9	Low income
2	Angola	AGO	45.985	19.1	Upper middle income
3	Albania	ALB	12.877	57.2	Upper middle income
4	United Arab Emirates	ARE	11.044	88.0	High income
...	...	...	...	...	...
190	Yemen, Rep.	YEM	32.947	20.0	Lower middle income
191	South Africa	ZAF	20.850	46.5	Upper middle income
192	Congo, Dem. Rep.	COD	42.394	2.2	Low income
193	Zambia	ZMB	40.471	15.4	Lower middle income
194	Zimbabwe	ZWE	35.715	18.5	Low income

195 rows × 5 columns

Working with Dataframes

In [ ]:
```
len(stats)
```

Out[ ]:
```
195
```

In [ ]:
```
stats.columns       #no need to put perentesis
```

Out[ ]:
```
Index(['Country Name', 'Country Code', 'Birth rate', 'Internet users',
       'Income Group'],
      dtype='object')
```

In [ ]:
```
len(stats.columns)      #checking number of columns
```

Out[ ]:
```
5
```

In [ ]:
```
stats.head()        #number of heads can be specified
```

Out[ ]:

	Country Name	Country Code	Birth rate	Internet users	Income Group
0	Aruba	ABW	10.244	78.9	High income
1	Afghanistan	AFG	35.253	5.9	Low income
2	Angola	AGO	45.985	19.1	Upper middle income
3	Albania	ALB	12.877	57.2	Upper middle income
4	United Arab Emirates	ARE	11.044	88.0	High income

In [ ]:
```
stats.tail()        #number of tails can be specified
```

Out[ ]:

	Country Name	Country Code	Birth rate	Internet users	Income Group
190	Yemen, Rep.	YEM	32.947	20.0	Lower middle income
191	South Africa	ZAF	20.850	46.5	Upper middle income
192	Congo, Dem. Rep.	COD	42.394	2.2	Low income
193	Zambia	ZMB	40.471	15.4	Lower middle income
194	Zimbabwe	ZWE	35.715	18.5	Low income

In [ ]:
```
stats.info()    # like str() function in R programming
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 195 entries, 0 to 194
Data columns (total 5 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   Country Name    195 non-null    object 
 1   Country Code    195 non-null    object 
 2   Birth rate      195 non-null    float64
 3   Internet users  195 non-null    float64
 4   Income Group    195 non-null    object 
dtypes: float64(2), object(3)
memory usage: 7.7+ KB
```

In [ ]:
```
stats.describe()        #like summary() in R
```

Out[ ]:

	Birth rate	Internet users
count	195.000000	195.000000
mean	21.469928	42.076471
std	10.605467	29.030788
min	7.900000	0.900000
25%	12.120500	14.520000
50%	19.680000	41.000000
75%	29.759500	66.225000
max	49.661000	96.546800

In [ ]:
```
stats.describe().transpose()
```

Out[ ]:

	count	mean	std	min	25%	50%	75%	max
Birth rate	195.0	21.469928	10.605467	7.9	12.1205	19.68	29.7595	49.6610
Internet users	195.0	42.076471	29.030788	0.9	14.5200	41.00	66.2250	96.5468

In [ ]: