Assignment 7- Pandas

In [156…
```
import pandas as pd
```

Q1.

Create a DataFrame containing student information like Name, Age, Department, %Marks

In [157…
```
df=pd.DataFrame({'Name':['Rohit','Sunil','Sujita'],
                 'Age':[21,25,31],
                 'Department':['CSE','Civil','Mech'],
                 'Marks':[91,89,92]})
print(df)
```

```
    Name  Age Department  Marks
0   Rohit   21        CSE     91
1   Sunil   25      Civil     89
2  Sujita   31       Mech     92
```

a.Display column names of the DF

In [158…
```
print(df.loc[:-1,:])
print(df.columns)
```

```
Empty DataFrame
Columns: [Name, Age, Department, Marks]
Index: []
Index(['Name', 'Age', 'Department', 'Marks'], dtype='object')
```

b.Show min/max and average marks

In [159…
```
print("Min Marks: ",df.Marks.min())
print("Max Marks: ",df.Marks.max())
print("Avg Marks: ",df.Marks.mean())
print('\nagg()')
print(df.Marks.agg(['min','max','mean']))
```

```
Min Marks:  89
Max Marks:  92
Avg Marks:  90.66666666666667

agg()
min     89.000000
max     92.000000
mean    90.666667
Name: Marks, dtype: float64
```

c.Get unique values and student count for each Department

In [160…
```
print('Unique values:',df['Department'].unique())
```

```
print('\n',df['Department'].value_counts())
```

```
Unique values: ['CSE' 'Civil' 'Mech']

 Department
CSE      1
Civil    1
Mech     1
Name: count, dtype: int64
```

Q2. Read csv file solar.csv

In [161…
```
solar=pd.read_csv('solar.csv')
solar
```

Out[161…

	State	Number of Solar Plants	Installed Capacity (MW)	Average MW Per Plant	Generation (GWh)	Region
0	California	289	4395	15.3	10826	West
1	Arizona	48	1078	22.5	2550	West
2	Nevada	11	238	21.6	557	West
3	Chicago	33	261	7.9	590	North
4	Connecticut	20	118	5.9	235	East
5	Texas	12	187	15.6	354	South
6	New York	13	53	4.1	84	East
7	Florida	15	535	12.3	453	South

a. Display all the plants with capacity > 500 MW

In [162…
```
filtered=solar[solar['Installed Capacity (MW)']>500].dropna()
filtered
```

Out[162…

	State	Number of Solar Plants	Installed Capacity (MW)	Average MW Per Plant	Generation (GWh)	Region
0	California	289	4395	15.3	10826	West
1	Arizona	48	1078	22.5	2550	West
7	Florida	15	535	12.3	453	South

b. Display plant details for New York plant

In [163…
```
# Idiomatic approach
solar[solar['State']=='New York']
```

Out[163...

	State	Number of Solar Plants	Installed Capacity (MW)	Average MW Per Plant	Generation (GWh)	Region
6	New York	13	53	4.1	84	East

In [164...

```
# .query()
filtered=solar.query("State=='New York'")
filtered
```

Out[164...

	State	Number of Solar Plants	Installed Capacity (MW)	Average MW Per Plant	Generation (GWh)	Region
6	New York	13	53	4.1	84	East

In [165...

```
#  .loc
filtered=solar.loc[solar['State']=='New York']
filtered
```

Out[165...

	State	Number of Solar Plants	Installed Capacity (MW)	Average MW Per Plant	Generation (GWh)	Region
6	New York	13	53	4.1	84	East

c. For all the plants display Average MW Per Plant and Generation (GWh)

In [166...

```
filtered=solar[['State','Average MW Per Plant','Generation (GWh)']]
filtered
```

Out[166...

	State	Average MW Per Plant	Generation (GWh)
0	California	15.3	10826
1	Arizona	22.5	2550
2	Nevada	21.6	557
3	Chicago	7.9	590
4	Connecticut	5.9	235
5	Texas	15.6	354
6	New York	4.1	84
7	Florida	12.3	453

d. Sort the plants by Generation (GWh) (ascending) and Installed Capacity (MW)(descending)

In [167...

```
sorted=solar.sort_values(by=['Generation (GWh)','Installed Capacity (MW)'],ascendin
sorted
```

Out[167...

	State	Number of Solar Plants	Installed Capacity (MW)	Average MW Per Plant	Generation (GWh)	Region
6	New York	13	53	4.1	84	East
4	Connecticut	20	118	5.9	235	East
5	Texas	12	187	15.6	354	South
7	Florida	15	535	12.3	453	South
2	Nevada	11	238	21.6	557	West
3	Chicago	33	261	7.9	590	North
1	Arizona	48	1078	22.5	2550	West
0	California	289	4395	15.3	10826	West

e. List top 5 plants for their Generation

In [168...

```
top5=solar.sort_values('Generation (GWh)',ascending=False)[:5]
top5
```

Out[168...

	State	Number of Solar Plants	Installed Capacity (MW)	Average MW Per Plant	Generation (GWh)	Region
0	California	289	4395	15.3	10826	West
1	Arizona	48	1078	22.5	2550	West
3	Chicago	33	261	7.9	590	North
2	Nevada	11	238	21.6	557	West
7	Florida	15	535	12.3	453	South

f. Display all plant details for states – California, Nevada, Arizona and Texas

In [169...

```
targer_states=['California','Nevada','Arizona','Texas']
mask=solar['State'].isin(targer_states)
filtered=solar[mask]
filtered
```

Out[169...

	State	Number of Solar Plants	Installed Capacity (MW)	Average MW Per Plant	Generation (GWh)	Region
0	California	289	4395	15.3	10826	West
1	Arizona	48	1078	22.5	2550	West
2	Nevada	11	238	21.6	557	West
5	Texas	12	187	15.6	354	South

g. Group the plants by region, find – min/max capacity plant for each group

In [170…
```
filtered=solar.groupby('Region').agg({'Installed Capacity (MW)':['min','max']})
filtered
```

Out[170…

	Installed Capacity (MW)	
	min	max
Region		
East	53	118
North	261	261
South	187	535
West	238	4395

h. Add following information to the existing details for the plants

North Carolina	148	669	4.5	1162	North

In [171…
```
filtered=solar['Region']=='North Carolina'
filtered
```

Out[171…
```
0    False
1    False
2    False
3    False
4    False
5    False
6    False
7    False
Name: Region, dtype: bool
```

Advanced ----------

Q.3

Read csv gapminder2007.csv

In [172…
```
df_gap=pd.read_csv('gapminder2007.csv')
df_gap.head()
```

Out[172...

	country	pop	continent	lifeExp	gdpPercap
0	Afghanistan	31889923.0	Asia	43.828	974.580338
1	Albania	3600523.0	Europe	76.423	5937.029526
2	Algeria	33333216.0	Africa	72.301	6223.367465
3	Angola	12420476.0	Africa	42.731	4797.231267
4	Argentina	40301927.0	Americas	75.320	12779.379640

a.Display data for all Asian countries

In [173...

```
df_gap[df_gap['continent']=='Asia']
```

Out[173...

	country	pop	continent	lifeExp	gdpPercap
0	Afghanistan	3.188992e+07	Asia	43.828	974.580338
7	Bahrain	7.085730e+05	Asia	75.635	29796.048340
8	Bangladesh	1.504483e+08	Asia	64.062	1391.253792
18	Cambodia	1.413186e+07	Asia	59.723	1713.778686
24	China	1.318683e+09	Asia	72.961	4959.114854
55	Hong Kong, China	6.980412e+06	Asia	82.208	39724.978670
58	India	1.110396e+09	Asia	64.698	2452.210407
59	Indonesia	2.235470e+08	Asia	70.650	3540.651564
60	Iran	6.945357e+07	Asia	70.964	11605.714490
61	Iraq	2.749964e+07	Asia	59.545	4471.061906
63	Israel	6.426679e+06	Asia	80.745	25523.277100
66	Japan	1.274680e+08	Asia	82.603	31656.068060
67	Jordan	6.053193e+06	Asia	72.535	4519.461171
69	Korea, Dem. Rep.	2.330172e+07	Asia	67.297	1593.065480
70	Korea, Rep.	4.904479e+07	Asia	78.623	23348.139730
71	Kuwait	2.505559e+06	Asia	77.588	47306.989780
72	Lebanon	3.921278e+06	Asia	71.993	10461.058680
78	Malaysia	2.482129e+07	Asia	74.241	12451.655800
83	Mongolia	2.874127e+06	Asia	66.803	3095.772271
87	Myanmar	4.776198e+07	Asia	62.069	944.000000
89	Nepal	2.890179e+07	Asia	63.785	1091.359778
96	Oman	3.204897e+06	Asia	75.640	22316.192870
97	Pakistan	1.692706e+08	Asia	65.483	2605.947580
101	Philippines	9.107729e+07	Asia	71.688	3190.481016
109	Saudi Arabia	2.760104e+07	Asia	72.777	21654.831940
113	Singapore	4.553009e+06	Asia	79.972	47143.179640
119	Sri Lanka	2.037824e+07	Asia	72.396	3970.095407
124	Syria	1.931475e+07	Asia	74.143	4184.548089
125	Taiwan	2.317429e+07	Asia	78.400	28718.276840
127	Thailand	6.506815e+07	Asia	70.616	7458.396327

	country	pop	continent	lifeExp	gdpPercap
137	Vietnam	8.526236e+07	Asia	74.249	2441.576404
138	West Bank and Gaza	4.018332e+06	Asia	73.422	3025.349798
139	Yemen, Rep.	2.221174e+07	Asia	62.698	2280.769906

b.Display top and bottom 15 rows

In [174…

```
pd.concat([df_gap.head(15),df_gap.tail(15)])
```

Out[174...

	country	pop	continent	lifeExp	gdpPercap
0	Afghanistan	31889923.0	Asia	43.828	974.580338
1	Albania	3600523.0	Europe	76.423	5937.029526
2	Algeria	33333216.0	Africa	72.301	6223.367465
3	Angola	12420476.0	Africa	42.731	4797.231267
4	Argentina	40301927.0	Americas	75.320	12779.379640
5	Australia	20434176.0	Oceania	81.235	34435.367440
6	Austria	8199783.0	Europe	79.829	36126.492700
7	Bahrain	708573.0	Asia	75.635	29796.048340
8	Bangladesh	150448339.0	Asia	64.062	1391.253792
9	Belgium	10392226.0	Europe	79.441	33692.605080
10	Benin	8078314.0	Africa	56.728	1441.284873
11	Bolivia	9119152.0	Americas	65.554	3822.137084
12	Bosnia and Herzegovina	4552198.0	Europe	74.852	7446.298803
13	Botswana	1639131.0	Africa	50.728	12569.851770
14	Brazil	190010647.0	Americas	72.390	9065.800825
127	Thailand	65068149.0	Asia	70.616	7458.396327
128	Togo	5701579.0	Africa	58.420	882.969944
129	Trinidad and Tobago	1056608.0	Americas	69.819	18008.509240
130	Tunisia	10276158.0	Africa	73.923	7092.923025
131	Turkey	71158647.0	Europe	71.777	8458.276384
132	Uganda	29170398.0	Africa	51.542	1056.380121
133	United Kingdom	60776238.0	Europe	79.425	33203.261280
134	United States	301139947.0	Americas	78.242	42951.653090
135	Uruguay	3447496.0	Americas	76.384	10611.462990
136	Venezuela	26084662.0	Americas	73.747	11415.805690
137	Vietnam	85262356.0	Asia	74.249	2441.576404
138	West Bank and Gaza	4018332.0	Asia	73.422	3025.349798
139	Yemen, Rep.	22211743.0	Asia	62.698	2280.769906
140	Zambia	11746035.0	Africa	42.384	1271.211593
141	Zimbabwe	12311143.0	Africa	43.487	469.709298

c. Show all the rows where life expectancy is more than 50 years and less than 80 years

In [175…
```
df_gap[(df_gap['lifeExp']>50) & (df_gap['lifeExp']<80)]
```

Out[175…

	country	pop	continent	lifeExp	gdpPercap
1	Albania	3600523.0	Europe	76.423	5937.029526
2	Algeria	33333216.0	Africa	72.301	6223.367465
4	Argentina	40301927.0	Americas	75.320	12779.379640
6	Austria	8199783.0	Europe	79.829	36126.492700
7	Bahrain	708573.0	Asia	75.635	29796.048340
...	...	...	...	...	...
135	Uruguay	3447496.0	Americas	76.384	10611.462990
136	Venezuela	26084662.0	Americas	73.747	11415.805690
137	Vietnam	85262356.0	Asia	74.249	2441.576404
138	West Bank and Gaza	4018332.0	Asia	73.422	3025.349798
139	Yemen, Rep.	22211743.0	Asia	62.698	2280.769906

110 rows × 5 columns

d. Show data for India, America and France

In [176…
```
df_gap[df_gap['country'].isin(['India','United States','France'])]
```

Out[176…

	country	pop	continent	lifeExp	gdpPercap
44	France	6.108392e+07	Europe	80.657	30470.016700
58	India	1.110396e+09	Asia	64.698	2452.210407
134	United States	3.011399e+08	Americas	78.242	42951.653090

e. Show data for all countries where name starts with 'A'

In [177…
```
df_gap[df_gap['country'].str.startswith('A')]
```

Out[177…

	country	pop	continent	lifeExp	gdpPercap
0	Afghanistan	31889923.0	Asia	43.828	974.580338
1	Albania	3600523.0	Europe	76.423	5937.029526
2	Algeria	33333216.0	Africa	72.301	6223.367465
3	Angola	12420476.0	Africa	42.731	4797.231267
4	Argentina	40301927.0	Americas	75.320	12779.379640
5	Australia	20434176.0	Oceania	81.235	34435.367440
6	Austria	8199783.0	Europe	79.829	36126.492700

f. Show min/max and mean GDP and Population for each continent

In [178…

```
summ=df_gap.groupby(by='continent').agg({'pop':['min','max','mean'],'gdpPercap':['m
summ
```

Out[178…

	pop			gdpPercap		
	min	**max**	**mean**	**min**	**max**	**mean**
continent						
Africa	199579.0	1.350312e+08	1.787576e+07	277.551859	13206.48452	3089.032605
Americas	1056608.0	3.011399e+08	3.595485e+07	1201.637154	42951.65309	11003.031625
Asia	708573.0	1.318683e+09	1.155138e+08	944.000000	47306.98978	12473.026870
Europe	301931.0	8.240100e+07	1.953662e+07	5937.029526	49357.19017	25054.481636
Oceania	4115771.0	2.043418e+07	1.227497e+07	25185.009110	34435.36744	29810.188275

g. Show population and life expectancy for all countries

In [179…

```
df_gap[['country','pop','lifeExp']]
```

Out[179…

	country	pop	lifeExp
0	Afghanistan	31889923.0	43.828
1	Albania	3600523.0	76.423
2	Algeria	33333216.0	72.301
3	Angola	12420476.0	42.731
4	Argentina	40301927.0	75.320
...	...	...	...
137	Vietnam	85262356.0	74.249
138	West Bank and Gaza	4018332.0	73.422
139	Yemen, Rep.	22211743.0	62.698
140	Zambia	11746035.0	42.384
141	Zimbabwe	12311143.0	43.487

142 rows × 3 columns

h.Sort the data by country names (ascending) and GDP (descending)

In [180…
```
df_gap.sort_values(by=['country','gdpPercap'],ascending=[True,False])
```

Out[180…

	country	pop	continent	lifeExp	gdpPercap
0	Afghanistan	31889923.0	Asia	43.828	974.580338
1	Albania	3600523.0	Europe	76.423	5937.029526
2	Algeria	33333216.0	Africa	72.301	6223.367465
3	Angola	12420476.0	Africa	42.731	4797.231267
4	Argentina	40301927.0	Americas	75.320	12779.379640
...	...	...	...	...	...
137	Vietnam	85262356.0	Asia	74.249	2441.576404
138	West Bank and Gaza	4018332.0	Asia	73.422	3025.349798
139	Yemen, Rep.	22211743.0	Asia	62.698	2280.769906
140	Zambia	11746035.0	Africa	42.384	1271.211593
141	Zimbabwe	12311143.0	Africa	43.487	469.709298

142 rows × 5 columns

i.Find top 20 most populous countries

In [181…
```
df_gap.sort_values(by='pop',ascending=False)[:20]
```

Out[181…

	country	pop	continent	lifeExp	gdpPercap
24	China	1.318683e+09	Asia	72.961	4959.114854
58	India	1.110396e+09	Asia	64.698	2452.210407
134	United States	3.011399e+08	Americas	78.242	42951.653090
59	Indonesia	2.235470e+08	Asia	70.650	3540.651564
14	Brazil	1.900106e+08	Americas	72.390	9065.800825
97	Pakistan	1.692706e+08	Asia	65.483	2605.947580
8	Bangladesh	1.504483e+08	Asia	64.062	1391.253792
94	Nigeria	1.350312e+08	Africa	46.859	2013.977305
66	Japan	1.274680e+08	Asia	82.603	31656.068060
82	Mexico	1.087009e+08	Americas	76.195	11977.574960
101	Philippines	9.107729e+07	Asia	71.688	3190.481016
137	Vietnam	8.526236e+07	Asia	74.249	2441.576404
47	Germany	8.240100e+07	Europe	79.406	32170.374420
38	Egypt	8.026454e+07	Africa	71.338	5581.180998
42	Ethiopia	7.651189e+07	Africa	52.947	690.805576
131	Turkey	7.115865e+07	Europe	71.777	8458.276384
60	Iran	6.945357e+07	Asia	70.964	11605.714490
127	Thailand	6.506815e+07	Asia	70.616	7458.396327
27	Congo, Dem. Rep.	6.460676e+07	Africa	46.462	277.551859
44	France	6.108392e+07	Europe	80.657	30470.016700

j.Delete all rows for a specific country

In [182…
```
country_to_drop='Iran'
index_to_drop=df_gap[df_gap['country']==country_to_drop].index
new_df=df_gap.drop(index=index_to_drop,axis=0)
'Iran' in new_df['country'].values
```

Out[182… `False`

k.Change all column names to title case

In [183…
```
new_df.columns=df_gap.columns.str.title()
new_df.head()
```

Out[183...

	Country	Pop	Continent	Lifeexp	Gdppercap
0	Afghanistan	31889923.0	Asia	43.828	974.580338
1	Albania	3600523.0	Europe	76.423	5937.029526
2	Algeria	33333216.0	Africa	72.301	6223.367465
3	Angola	12420476.0	Africa	42.731	4797.231267
4	Argentina	40301927.0	Americas	75.320	12779.379640

l.Change column names: pop to Population, gdpPercap to GDP_Percap

In [184...

```
new_df=df_gap.rename(columns={'pop':'Population',
                              'gdpPercap':'GDP_Percap' })
new_df.head()
```

Out[184...

	country	Population	continent	lifeExp	GDP_Percap
0	Afghanistan	31889923.0	Asia	43.828	974.580338
1	Albania	3600523.0	Europe	76.423	5937.029526
2	Algeria	33333216.0	Africa	72.301	6223.367465
3	Angola	12420476.0	Africa	42.731	4797.231267
4	Argentina	40301927.0	Americas	75.320	12779.379640

m.Increase Life Expectancy by 2 years for all African countries

In [185...

```
new_df=df_gap.copy()
mask=new_df['continent']=='Africa'
new_df.loc[mask,'lifeExp']=new_df.loc[mask,'lifeExp']+2
print("Old \n",df_gap[2:4])
print("New \n",new_df[2:4])
```

```
Old
    country         pop continent  lifeExp    gdpPercap
2  Algeria  33333216.0    Africa   72.301  6223.367465
3   Angola  12420476.0    Africa   42.731  4797.231267
New
    country         pop continent  lifeExp    gdpPercap
2  Algeria  33333216.0    Africa   74.301  6223.367465
3   Angola  12420476.0    Africa   44.731  4797.231267
```

n.Find all the rows that contain NA values

In [186...

```
new_df=df_gap.copy()
new_df.loc[len(df_gap)]=['Atlantis',12.0,'Ocenia',85.0,float('nan')]
mask=new_df.isna().any(axis=1)
new_df[mask]
```

Out[186…

	country	pop	continent	lifeExp	gdpPercap
142	Atlantis	12.0	Ocenia	85.0	NaN

o.Fill NA values with 0 for numeric columns

In [187…
```
new_df=new_df.fillna(0)
# idiomatic approach
print(new_df.iloc[-1])
```

```
country      Atlantis
pop              12.0
continent      Ocenia
lifeExp          85.0
gdpPercap         0.0
Name: 142, dtype: object
```

In [188…
```
new_df=df_gap.copy()
new_df.loc[len(df_gap)]=['Atlantis',12.0,'Ocenia',85.0,float('nan')]
new_df.iloc[-1]
```

Out[188…
```
country      Atlantis
pop              12.0
continent      Ocenia
lifeExp          85.0
gdpPercap         NaN
Name: 142, dtype: object
```

In [189…
```
# smarter approach
numeric_cols=new_df.select_dtypes(include=['float64','int64']).columns
new_df[numeric_cols]=new_df[numeric_cols].fillna(0)
new_df.iloc[-1]
```

Out[189…
```
country      Atlantis
pop              12.0
continent      Ocenia
lifeExp          85.0
gdpPercap         0.0
Name: 142, dtype: object
```

In [ ]: