

```
In [1]: import pandas as pd # USE FOR DATAFRAMES
```

```
In [2]: pd.__version__
```

```
Out[2]: '2.2.2'
```

```
In [3]: df=pd.read_csv(r'C:\DS & GEN AI\Pandas\data.csv')
df
```

Out[3]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
<b>0</b>	Aruba	ABW	10.244	78.9	High income
<b>1</b>	Afghanistan	AFG	35.253	5.9	Low income
<b>2</b>	Angola	AGO	45.985	19.1	Upper middle income
<b>3</b>	Albania	ALB	12.877	57.2	Upper middle income
<b>4</b>	United Arab Emirates	ARE	11.044	88.0	High income
...	...	...	...	...	...
<b>190</b>	Yemen, Rep.	YEM	32.947	20.0	Lower middle income
<b>191</b>	South Africa	ZAF	20.850	46.5	Upper middle income
<b>192</b>	Congo, Dem. Rep.	COD	42.394	2.2	Low income
<b>193</b>	Zambia	ZMB	40.471	15.4	Lower middle income
<b>194</b>	Zimbabwe	ZWE	35.715	18.5	Low income

195 rows × 5 columns

```
In [4]: len(df)
```

```
Out[4]: 195
```

```
In [5]: df.shape
```

```
Out[5]: (195, 5)
```

```
In [6]: df.columns
```

```
Out[6]: Index(['CountryName', 'CountryCode', 'BirthRate', 'InternetUsers',
       'IncomeGroup'],
       dtype='object')
```

```
In [7]: type(df)
```

```
Out[7]: pandas.core.frame.DataFrame
```

```
In [8]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 195 entries, 0 to 194
Data columns (total 5 columns):
 #   Column      Non-Null Count  Dtype  
--- 
 0   CountryName    195 non-null   object  
 1   CountryCode     195 non-null   object  
 2   BirthRate       195 non-null   float64 
 3   InternetUsers  195 non-null   float64 
 4   IncomeGroup     195 non-null   object  
dtypes: float64(2), object(3)
memory usage: 7.7+ KB
```

In [9]: `df.columns`

```
Out[9]: Index(['CountryName', 'CountryCode', 'BirthRate', 'InternetUsers',
               'IncomeGroup'],
              dtype='object')
```

In [10]: `len(df.columns)`

```
Out[10]: 5
```

In [11]: `df.head()`

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
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 [12]: `df.tail()`

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
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 [13]: `df.tail(2)`

Out[13]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
193	Zambia	ZMB	40.471	15.4	Lower middle income
194	Zimbabwe	ZWE	35.715	18.5	Low income

In [14]: `df.head(2)`

Out[14]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
0	Aruba	ABW	10.244	78.9	High income
1	Afghanistan	AFG	35.253	5.9	Low income

In [15]: `df[:::-1]`

Out[15]:

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

195 rows × 5 columns

In [16]: `df[:5]`

Out[16]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
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 [17]: df[6:]

Out[17]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
6	Armenia	ARM	13.308	41.9000	Lower middle income
7	Antigua and Barbuda	ATG	16.447	63.4000	High income
8	Australia	AUS	13.200	83.0000	High income
9	Austria	AUT	9.400	80.6188	High income
10	Azerbaijan	AZE	18.300	58.7000	Upper middle income
...	...	...	...	...	...
190	Yemen, Rep.	YEM	32.947	20.0000	Lower middle income
191	South Africa	ZAF	20.850	46.5000	Upper middle income
192	Congo, Dem. Rep.	COD	42.394	2.2000	Low income
193	Zambia	ZMB	40.471	15.4000	Lower middle income
194	Zimbabwe	ZWE	35.715	18.5000	Low income

189 rows × 5 columns

In [18]: df[0:200:10]

Out[18]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
0	Aruba	ABW	10.244	78.900000	High income
10	Azerbaijan	AZE	18.300	58.700000	Upper middle income
20	Belarus	BLR	12.500	54.170000	Upper middle income
30	Canada	CAN	10.900	85.800000	High income
40	Costa Rica	CRI	15.022	45.960000	Upper middle income
50	Ecuador	ECU	21.070	40.353684	Upper middle income
60	Gabon	GAB	30.555	9.200000	Upper middle income
70	Greenland	GRL	14.500	65.800000	High income
80	India	IND	20.291	15.100000	Lower middle income
90	Kazakhstan	KAZ	22.730	54.000000	Upper middle income
100	Libya	LBY	21.425	16.500000	Upper middle income
110	Moldova	MDA	12.141	45.000000	Lower middle income
120	Mozambique	MOZ	39.705	5.400000	Low income
130	Netherlands	NLD	10.200	93.956400	High income
140	Poland	POL	9.600	62.849200	High income
150	Sudan	SDN	33.477	22.700000	Lower middle income
160	Suriname	SUR	18.455	37.400000	Upper middle income
170	Tajikistan	TJK	30.792	16.000000	Lower middle income
180	Uruguay	URY	14.374	57.690000	High income
190	Yemen, Rep.	YEM	32.947	20.000000	Lower middle income

In [19]:

df

Out[19]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
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

In [20]: `df.describe() # DESCRIBE WILL GIVE NUMERICAL DATA.`

Out[20]:

	BirthRate	InternetUsers
<b>count</b>	195.000000	195.000000
<b>mean</b>	21.469928	42.076471
<b>std</b>	10.605467	29.030788
<b>min</b>	7.900000	0.900000
<b>25%</b>	12.120500	14.520000
<b>50%</b>	19.680000	41.000000
<b>75%</b>	29.759500	66.225000
<b>max</b>	49.661000	96.546800

In [21]: `df.describe().transpose() # TRANSPOSE CONVERT COLUMN INTO ROW`

Out[21]:

	count	mean	std	min	25%	50%	75%	max
<b>BirthRate</b>	195.0	21.469928	10.605467	7.9	12.1205	19.68	29.7595	49.6610
<b>InternetUsers</b>	195.0	42.076471	29.030788	0.9	14.5200	41.00	66.2250	96.5468

In [22]: `df.describe().T # TRANSPOSE CONVERT COLUMN INTO ROW`

	count	mean	std	min	25%	50%	75%	max
<b>BirthRate</b>	195.0	21.469928	10.605467	7.9	12.1205	19.68	29.7595	49.6610
<b>InternetUsers</b>	195.0	42.076471	29.030788	0.9	14.5200	41.00	66.2250	96.5468

In [23]: `df.columns`

Out[23]: `Index(['CountryName', 'CountryCode', 'BirthRate', 'InternetUsers', 'IncomeGroup'], dtype='object')`

In [24]: `df.columns=['a', 'b', 'c', 'd', 'e', ]`

In [25]: `df.head(1)`

Out[25]:

	a	b	c	d	e
<b>0</b>	Aruba	ABW	10.244	78.9	High income

In [26]: `df.columns`

Out[26]: `Index(['a', 'b', 'c', 'd', 'e'], dtype='object')`

In [27]: `df.columns=['CountryName', 'CountryCode', 'BirthRate', 'InternetUsers', 'IncomeGroup']`

In [28]: `df.head(1)`

Out[28]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
<b>0</b>	Aruba	ABW	10.244	78.9	High income

# Subsetting a DataFrame in Pandas #1.Rows #2.Columns #3.Combine the two

In [29]: `# Rows:  
df[21:26]`

Out[29]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
<b>21</b>	Belize	BLZ	23.092	33.60	Upper middle income
<b>22</b>	Bermuda	BMU	10.400	95.30	High income
<b>23</b>	Bolivia	BOL	24.236	36.94	Lower middle income
<b>24</b>	Brazil	BRA	14.931	51.04	Upper middle income
<b>25</b>	Barbados	BRB	12.188	73.00	High income

In [30]: `df[:]`

Out[30]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
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

In [31]: df[:10]

Out[31]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
0	Aruba	ABW	10.244	78.9000	High income
1	Afghanistan	AFG	35.253	5.9000	Low income
2	Angola	AGO	45.985	19.1000	Upper middle income
3	Albania	ALB	12.877	57.2000	Upper middle income
4	United Arab Emirates	ARE	11.044	88.0000	High income
5	Argentina	ARG	17.716	59.9000	High income
6	Armenia	ARM	13.308	41.9000	Lower middle income
7	Antigua and Barbuda	ATG	16.447	63.4000	High income
8	Australia	AUS	13.200	83.0000	High income
9	Austria	AUT	9.400	80.6188	High income

In [32]: df.head(10)

Out[32]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
0	Aruba	ABW	10.244	78.9000	High income
1	Afghanistan	AFG	35.253	5.9000	Low income
2	Angola	AGO	45.985	19.1000	Upper middle income
3	Albania	ALB	12.877	57.2000	Upper middle income
4	United Arab Emirates	ARE	11.044	88.0000	High income
5	Argentina	ARG	17.716	59.9000	High income
6	Armenia	ARM	13.308	41.9000	Lower middle income
7	Antigua and Barbuda	ATG	16.447	63.4000	High income
8	Australia	AUS	13.200	83.0000	High income
9	Austria	AUT	9.400	80.6188	High income

In [33]:

```
# How to reverse the dataframe
df[::-1]
```

Out[33]:

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

195 rows × 5 columns

In [34]:

```
df
```

Out[34]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
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

In [35]: df[::20]

Out[35]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
0	Aruba	ABW	10.244	78.9000	High income
20	Belarus	BLR	12.500	54.1700	Upper middle income
40	Costa Rica	CRI	15.022	45.9600	Upper middle income
60	Gabon	GAB	30.555	9.2000	Upper middle income
80	India	IND	20.291	15.1000	Lower middle income
100	Libya	LBY	21.425	16.5000	Upper middle income
120	Mozambique	MOZ	39.705	5.4000	Low income
140	Poland	POL	9.600	62.8492	High income
160	Suriname	SUR	18.455	37.4000	Upper middle income
180	Uruguay	URY	14.374	57.6900	High income

In [36]: df.dtypes

```

Out[36]: CountryName      object
CountryCode       object
BirthRate        float64
InternetUsers    float64
IncomeGroup      object
dtype: object

```

```
In [37]: # Columns
df.columns
```

```
Out[37]: Index(['CountryName', 'CountryCode', 'BirthRate', 'InternetUsers',
       'IncomeGroup'],
       dtype='object')
```

```
In [38]: df.head()
```

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
<b>0</b>	Aruba	ABW	10.244	78.9	High income
<b>1</b>	Afghanistan	AFG	35.253	5.9	Low income
<b>2</b>	Angola	AGO	45.985	19.1	Upper middle income
<b>3</b>	Albania	ALB	12.877	57.2	Upper middle income
<b>4</b>	United Arab Emirates	ARE	11.044	88.0	High income

```
In [39]: df['CountryName'].head()
```

```
Out[39]: 0          Aruba
1          Afghanistan
2          Angola
3          Albania
4  United Arab Emirates
Name: CountryName, dtype: object
```

```
In [40]: df[['CountryName', 'BirthRate']].head()
```

	CountryName	BirthRate
<b>0</b>	Aruba	10.244
<b>1</b>	Afghanistan	35.253
<b>2</b>	Angola	45.985
<b>3</b>	Albania	12.877
<b>4</b>	United Arab Emirates	11.044

```
In [41]: df.head()
```

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
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 [42]: `df['BirthRate']`

Out[42]:

0	10.244
1	35.253
2	45.985
3	12.877
4	11.044
	...
190	32.947
191	20.850
192	42.394
193	40.471
194	35.715

Name: BirthRate, Length: 195, dtype: float64

In [43]: `# Combine the two  
df[4:8][['CountryName', 'BirthRate']]`

	CountryName	BirthRate
4	United Arab Emirates	11.044
5	Argentina	17.716
6	Armenia	13.308
7	Antigua and Barbuda	16.447

In [44]: `df[['CountryName', 'BirthRate']][4:8]`

	CountryName	BirthRate
4	United Arab Emirates	11.044
5	Argentina	17.716
6	Armenia	13.308
7	Antigua and Barbuda	16.447

In [45]: `df_cat=df[['CountryName', 'CountryCode', 'IncomeGroup']]  
df_cat.head()`

Out[45]:

	CountryName	CountryCode	IncomeGroup
<b>0</b>	Aruba	ABW	High income
<b>1</b>	Afghanistan	AFG	Low income
<b>2</b>	Angola	AGO	Upper middle income
<b>3</b>	Albania	ALB	Upper middle income
<b>4</b>	United Arab Emirates	ARE	High income

In [46]: `df.describe()`

Out[46]:

	BirthRate	InternetUsers
<b>count</b>	195.000000	195.000000
<b>mean</b>	21.469928	42.076471
<b>std</b>	10.605467	29.030788
<b>min</b>	7.900000	0.900000
<b>25%</b>	12.120500	14.520000
<b>50%</b>	19.680000	41.000000
<b>75%</b>	29.759500	66.225000
<b>max</b>	49.661000	96.546800

In [47]: `df_cat.describe()`

Out[47]:

	CountryName	CountryCode	IncomeGroup
<b>count</b>	195	195	195
<b>unique</b>	195	195	4
<b>top</b>	Aruba	ABW	High income
<b>freq</b>	1	1	67

In [48]: `#Mathematical operation  
df.BirthRate*df.InternetUsers`

```
Out[48]: 0    808.2516
         1    207.9927
         2    878.3135
         3    736.5644
         4    971.8720
         ...
        190   658.9400
        191   969.5250
        192   93.2668
        193   623.2534
        194   660.7275
Length: 195, dtype: float64
```

```
In [49]: df['mycalc']=df.BirthRate*df.InternetUsers # Adding a column
df
```

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup	mycalc
0	Aruba	ABW	10.244	78.9	High income	808.2516
1	Afghanistan	AFG	35.253	5.9	Low income	207.9927
2	Angola	AGO	45.985	19.1	Upper middle income	878.3135
3	Albania	ALB	12.877	57.2	Upper middle income	736.5644
4	United Arab Emirates	ARE	11.044	88.0	High income	971.8720
...	...	...	...	...	...	...
190	Yemen, Rep.	YEM	32.947	20.0	Lower middle income	658.9400
191	South Africa	ZAF	20.850	46.5	Upper middle income	969.5250
192	Congo, Dem. Rep.	COD	42.394	2.2	Low income	93.2668
193	Zambia	ZMB	40.471	15.4	Lower middle income	623.2534
194	Zimbabwe	ZWE	35.715	18.5	Low income	660.7275

195 rows × 6 columns

```
In [50]: len(df.columns)
```

```
Out[50]: 6
```

```
In [51]: # Remove a Column
df.drop('mycalc',axis = 1)
```

Out[51]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
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

In [52]: df.head()

Out[52]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup	mycalc
0	Aruba	ABW	10.244	78.9	High income	808.2516
1	Afghanistan	AFG	35.253	5.9	Low income	207.9927
2	Angola	AGO	45.985	19.1	Upper middle income	878.3135
3	Albania	ALB	12.877	57.2	Upper middle income	736.5644
4	United Arab Emirates	ARE	11.044	88.0	High income	971.8720

In [53]: df.columns[2]

Out[53]: 'BirthRate'

In [54]: df.InternetUsers&lt;2 #we are checking given condition if its correct true or false

```
Out[54]: 0    False
         1    False
         2    False
         3    False
         4    False
         ...
        190   False
        191   False
        192   False
        193   False
        194   False
Name: InternetUsers, Length: 195, dtype: bool
```

```
In [55]: Filter = df.InternetUsers < 2
```

```
In [56]: Filter
```

```
Out[56]: 0    False
         1    False
         2    False
         3    False
         4    False
         ...
        190   False
        191   False
        192   False
        193   False
        194   False
Name: InternetUsers, Length: 195, dtype: bool
```

```
In [57]: df[3:7]
```

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup	mycalc
3	Albania	ALB	12.877	57.2	Upper middle income	736.5644
4	United Arab Emirates	ARE	11.044	88.0	High income	971.8720
5	Argentina	ARG	17.716	59.9	High income	1061.1884
6	Armenia	ARM	13.308	41.9	Lower middle income	557.6052

```
In [58]: df[30:40]
```

Out[58]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup	mycalc
30	Canada	CAN	10.900	85.80	High income	935.2200
31	Switzerland	CHE	10.200	86.34	High income	880.6680
32	Chile	CHL	13.385	66.50	High income	890.1025
33	China	CHN	12.100	45.80	Upper middle income	554.1800
34	Cote d'Ivoire	CIV	37.320	8.40	Lower middle income	313.4880
35	Cameroon	CMR	37.236	6.40	Lower middle income	238.3104
36	Congo, Rep.	COG	37.011	6.60	Lower middle income	244.2726
37	Colombia	COL	16.076	51.70	Upper middle income	831.1292
38	Comoros	COM	34.326	6.50	Low income	223.1190
39	Cabo Verde	CPV	21.625	37.50	Lower middle income	810.9375

In [59]: df[Filter] #we will get only which row are false

Out[59]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup	mycalc
11	Burundi	BDI	44.151	1.3	Low income	57.3963
52	Eritrea	ERI	34.800	0.9	Low income	31.3200
55	Ethiopia	ETH	32.925	1.9	Low income	62.5575
64	Guinea	GIN	37.337	1.6	Low income	59.7392
117	Myanmar	MMR	18.119	1.6	Lower middle income	28.9904
127	Niger	NER	49.661	1.7	Low income	84.4237
154	Sierra Leone	SLE	36.729	1.7	Low income	62.4393
156	Somalia	SOM	43.891	1.5	Low income	65.8365
172	Timor-Leste	TLS	35.755	1.1	Lower middle income	39.3305

In [60]: df.BirthRate&gt;40

```
Out[60]: 0    False
         1    False
         2     True
         3    False
         4    False
        ...
       190   False
       191   False
       192     True
       193     True
       194   False
Name: BirthRate, Length: 195, dtype: bool
```

```
In [61]: Filter2 = df.BirthRate>40
Filter2
```

```
Out[61]: 0    False
         1    False
         2     True
         3    False
         4    False
        ...
       190   False
       191   False
       192     True
       193     True
       194   False
Name: BirthRate, Length: 195, dtype: bool
```

```
In [62]: Filter & Filter2
```

```
Out[62]: 0    False
         1    False
         2    False
         3    False
         4    False
        ...
       190   False
       191   False
       192   False
       193   False
       194   False
Length: 195, dtype: bool
```

```
In [63]: df[Filter & Filter2]
```

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup	mycalc
<b>11</b>	Burundi	BDI	44.151	1.3	Low income	57.3963
<b>127</b>	Niger	NER	49.661	1.7	Low income	84.4237
<b>156</b>	Somalia	SOM	43.891	1.5	Low income	65.8365

```
In [64]: df[(df.BirthRate > 40) & (df.InternetUsers < 2)]
```

Out[64]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup	mycalc
11	Burundi	BDI	44.151	1.3	Low income	57.3963
127	Niger	NER	49.661	1.7	Low income	84.4237
156	Somalia	SOM	43.891	1.5	Low income	65.8365

In [65]:

`df.head()`

Out[65]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup	mycalc
0	Aruba	ABW	10.244	78.9	High income	808.2516
1	Afghanistan	AFG	35.253	5.9	Low income	207.9927
2	Angola	AGO	45.985	19.1	Upper middle income	878.3135
3	Albania	ALB	12.877	57.2	Upper middle income	736.5644
4	United Arab Emirates	ARE	11.044	88.0	High income	971.8720

In [66]:

`df[df.IncomeGroup == 'High income']`

Out[66]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup	mycalc
0	Aruba	ABW	10.244	78.90	High income	808.25160
4	United Arab Emirates	ARE	11.044	88.00	High income	971.87200
5	Argentina	ARG	17.716	59.90	High income	1061.18840
7	Antigua and Barbuda	ATG	16.447	63.40	High income	1042.73980
8	Australia	AUS	13.200	83.00	High income	1095.60000
...	...	...	...	...	...	...
174	Trinidad and Tobago	TTO	14.590	63.80	High income	930.84200
180	Uruguay	URY	14.374	57.69	High income	829.23606
181	United States	USA	12.500	84.20	High income	1052.50000
184	Venezuela, RB	VEN	19.842	54.90	High income	1089.32580
185	Virgin Islands (U.S.)	VIR	10.700	45.30	High income	484.71000

67 rows × 6 columns

```
In [67]: df[df.IncomeGroup == 'Low income']
```

Out[67]:	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup	mycalc
1	Afghanistan	AFG	35.253	5.90	Low income	207.99270
11	Burundi	BDI	44.151	1.30	Low income	57.39630
13	Benin	BEN	36.440	4.90	Low income	178.55600
14	Burkina Faso	BFA	40.551	9.10	Low income	369.01410
29	Central African Republic	CAF	34.076	3.50	Low income	119.26600
38	Comoros	COM	34.326	6.50	Low income	223.11900
52	Eritrea	ERI	34.800	0.90	Low income	31.32000
55	Ethiopia	ETH	32.925	1.90	Low income	62.55750
64	Guinea	GIN	37.337	1.60	Low income	59.73920
65	Gambia, The	GMB	42.525	14.00	Low income	595.35000
66	Guinea-Bissau	GNB	37.503	3.10	Low income	116.25930
77	Haiti	HTI	25.345	10.60	Low income	268.65700
93	Cambodia	KHM	24.462	6.80	Low income	166.34160
99	Liberia	LBR	35.521	3.20	Low income	113.66720
111	Madagascar	MDG	34.686	3.00	Low income	104.05800
115	Mali	MLI	44.138	3.50	Low income	154.48300
120	Mozambique	MOZ	39.705	5.40	Low income	214.40700
123	Malawi	MWI	39.459	5.05	Low income	199.26795
127	Niger	NER	49.661	1.70	Low income	84.42370
132	Nepal	NPL	20.923	13.30	Low income	278.27590
148	Rwanda	RWA	32.689	9.00	Low income	294.20100
154	Sierra Leone	SLE	36.729	1.70	Low income	62.43930
156	Somalia	SOM	43.891	1.50	Low income	65.83650
158	South Sudan	SSD	37.126	14.10	Low income	523.47660
167	Chad	TCD	45.745	2.30	Low income	105.21350
168	Togo	TGO	36.080	4.50	Low income	162.36000
177	Tanzania	TZA	39.518	4.40	Low income	173.87920
178	Uganda	UGA	43.474	16.20	Low income	704.27880
192	Congo, Dem. Rep.	COD	42.394	2.20	Low income	93.26680

CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup	mycalc
194	Zimbabwe	ZWE	35.715	18.50	Low income 660.72750

```
In [68]: # How to get the unique categories
df.IncomeGroup.unique()
```

```
Out[68]: array(['High income', 'Low income', 'Upper middle income',
       'Lower middle income'], dtype=object)
```

```
In [69]: df.IncomeGroup.nunique() #It gives the count for the above one.
```

```
Out[69]: 4
```

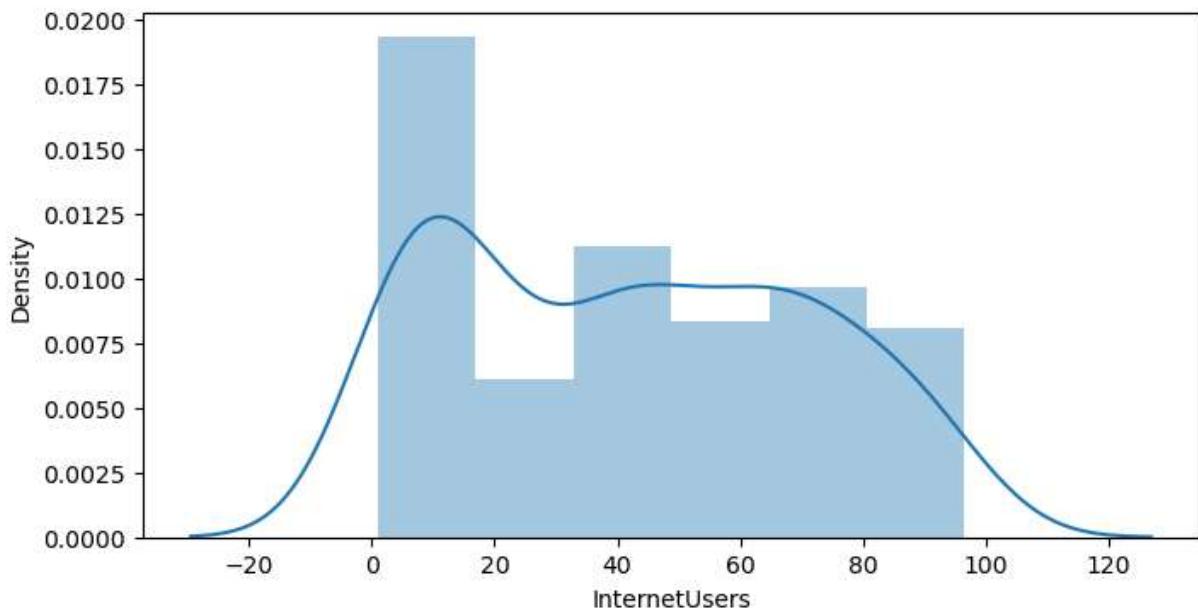
```
In [155... import matplotlib.pyplot as plt # Visualization
```

```
In [157... import seaborn as sns # Distribution and Advanced Visualization
```

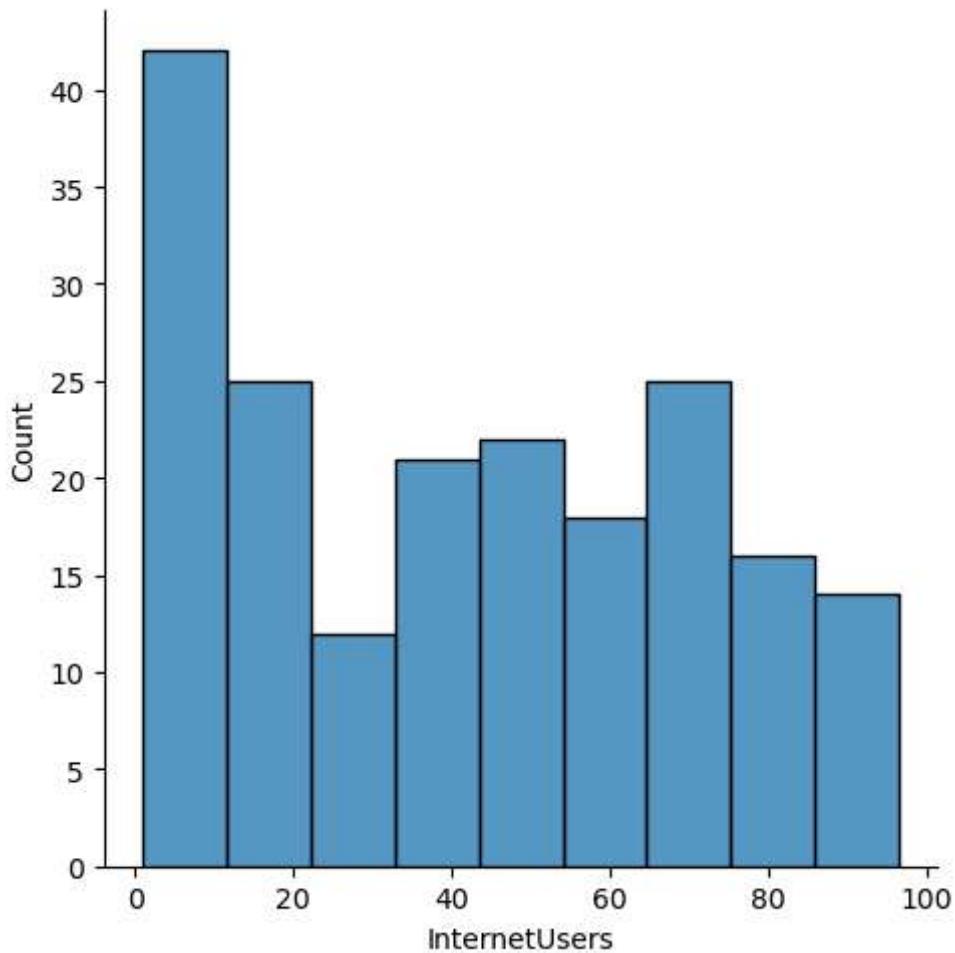
```
In [161... %matplotlib inline
plt.rcParams['figure.figsize'] = 8,4
```

```
In [163... import warnings
warnings.filterwarnings('ignore')
```

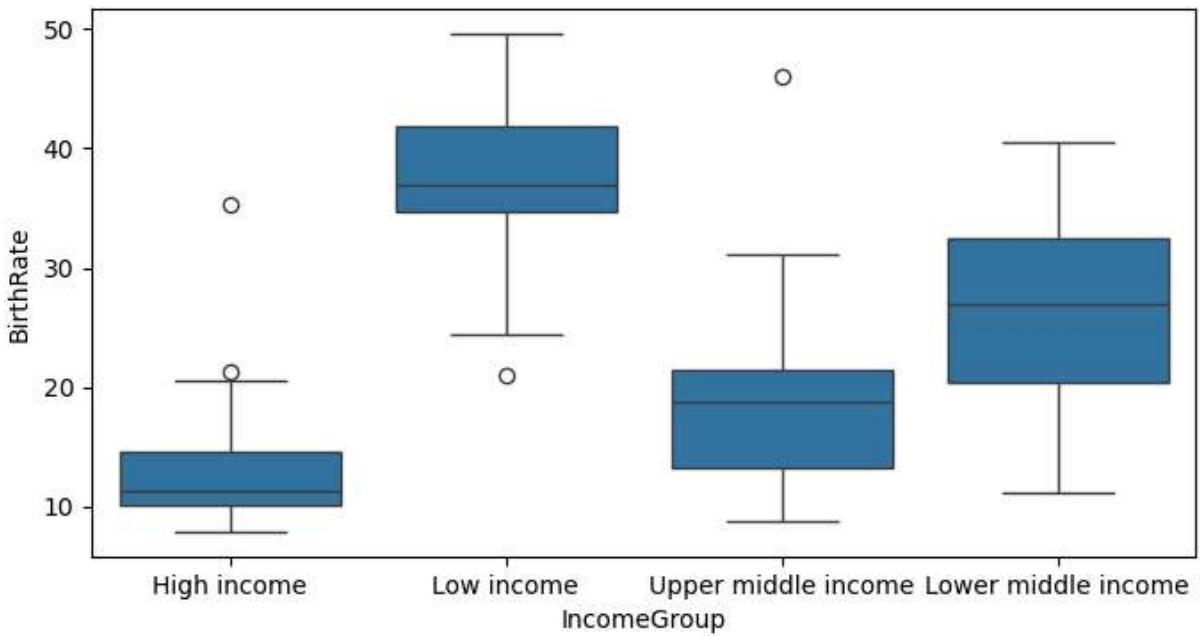
```
In [165... vis1 = sns.distplot(df["InternetUsers"]) # univariate analysis
plt.show(vis1) # it is advanced graph
```



```
In [167... vis1 = sns.distplot(df["InternetUsers"]) # t is removed. so distribution Line is removed
plt.show()
```

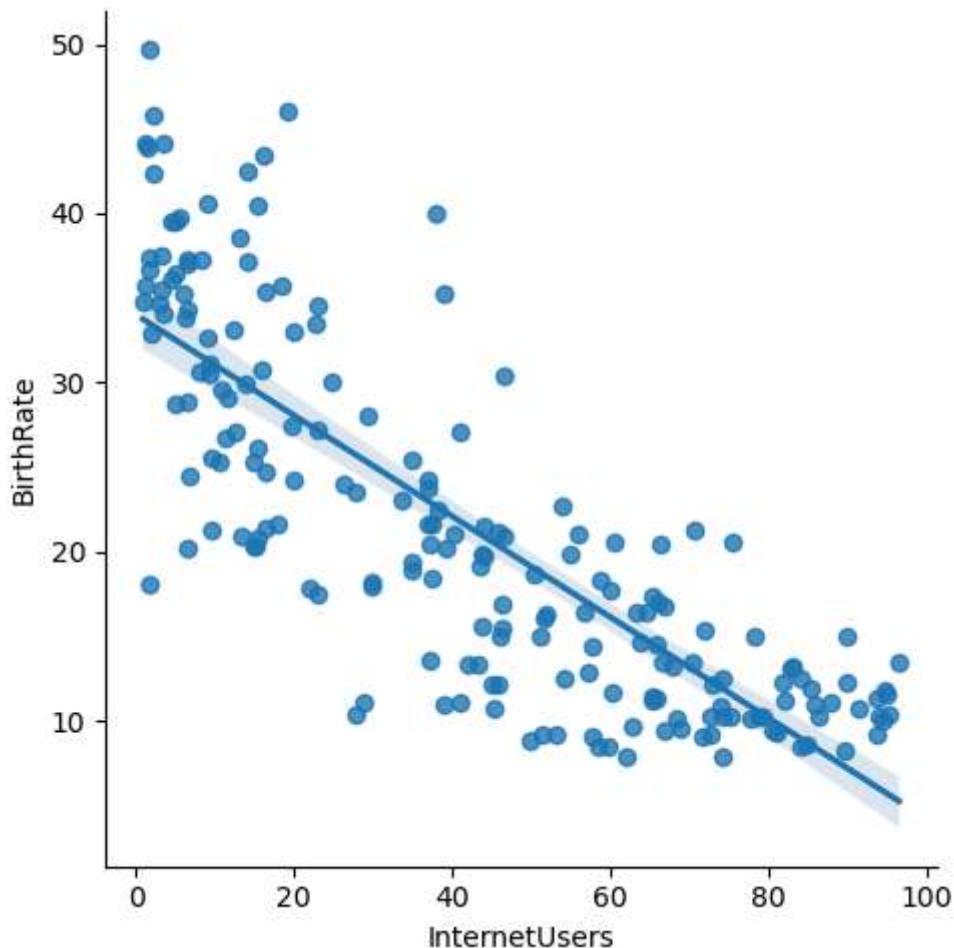


```
In [169]: vis2 = sns.boxplot(data = df, x= 'IncomeGroup', y = 'BirthRate')
plt.show()
```

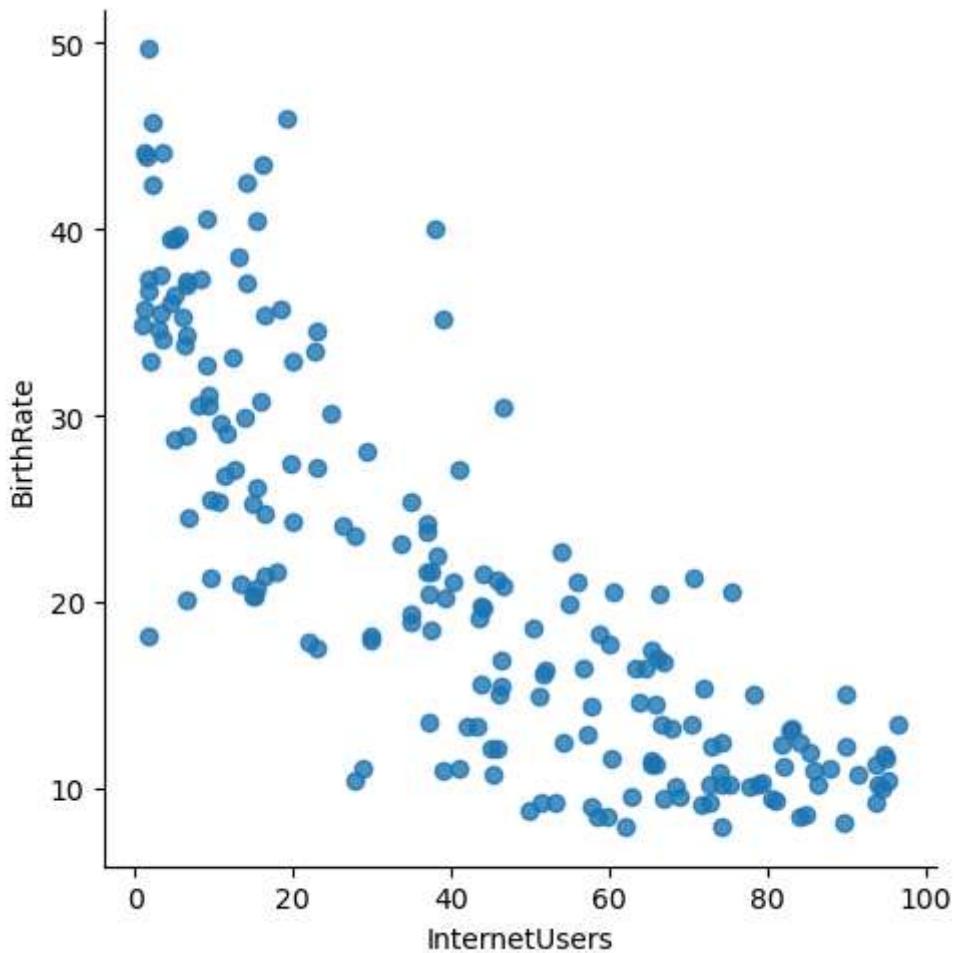


```
# outlier = anomaly detection # Its very very important in machine learning # outliers are nothing but they are far from other observations. # they are not the neihgbours, they stay far from others # the diamond or circled shaped in boxplot graph is an outlier # Blue box = interquartile range [IQR] # midline is called average lines # lower line = minimum # Upper line = maximum
```

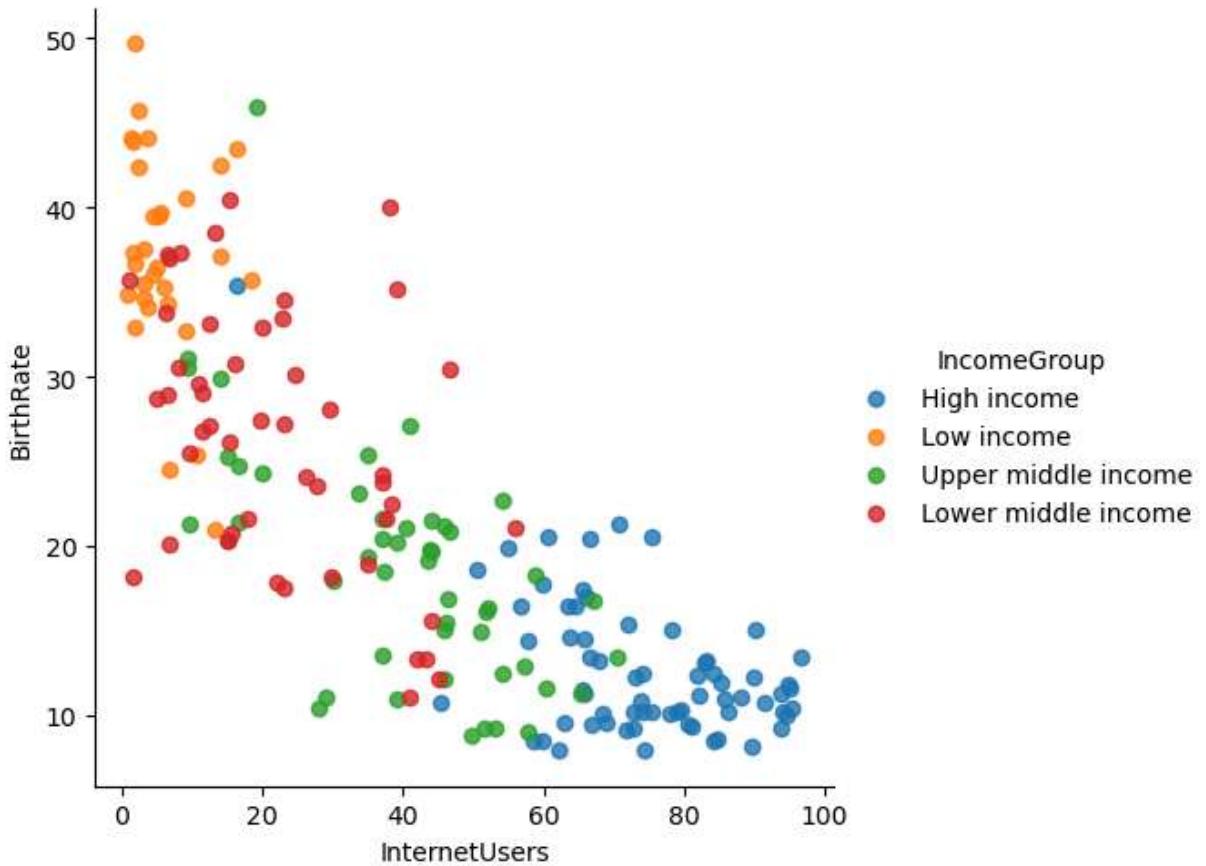
```
In [171... vis3 = sns.lmplot(data=df, x = "InternetUsers", y = "BirthRate")  
plt.show() # the blue dots are each data present in the excel
```



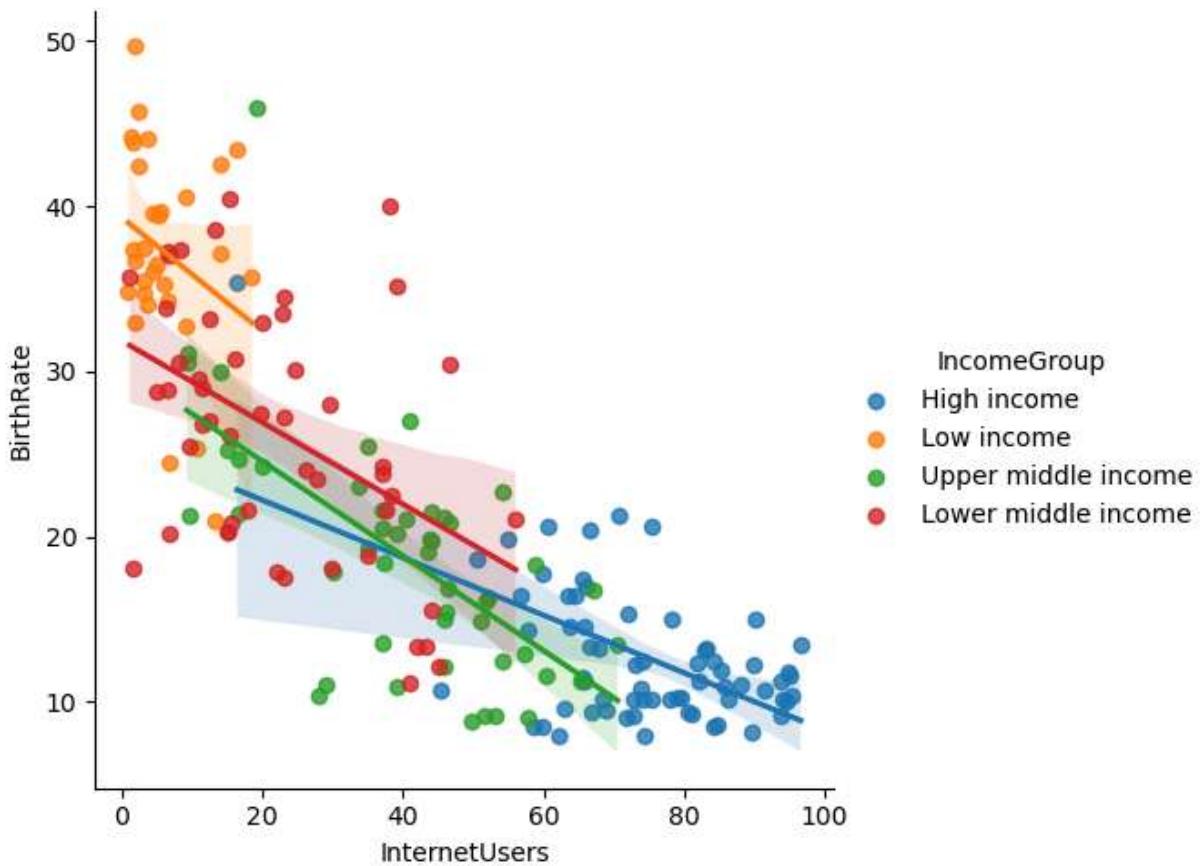
```
In [173... vis3 = sns.lmplot(data=df, x = "InternetUsers", y = "BirthRate", fit_reg = False)  
plt.show()
```



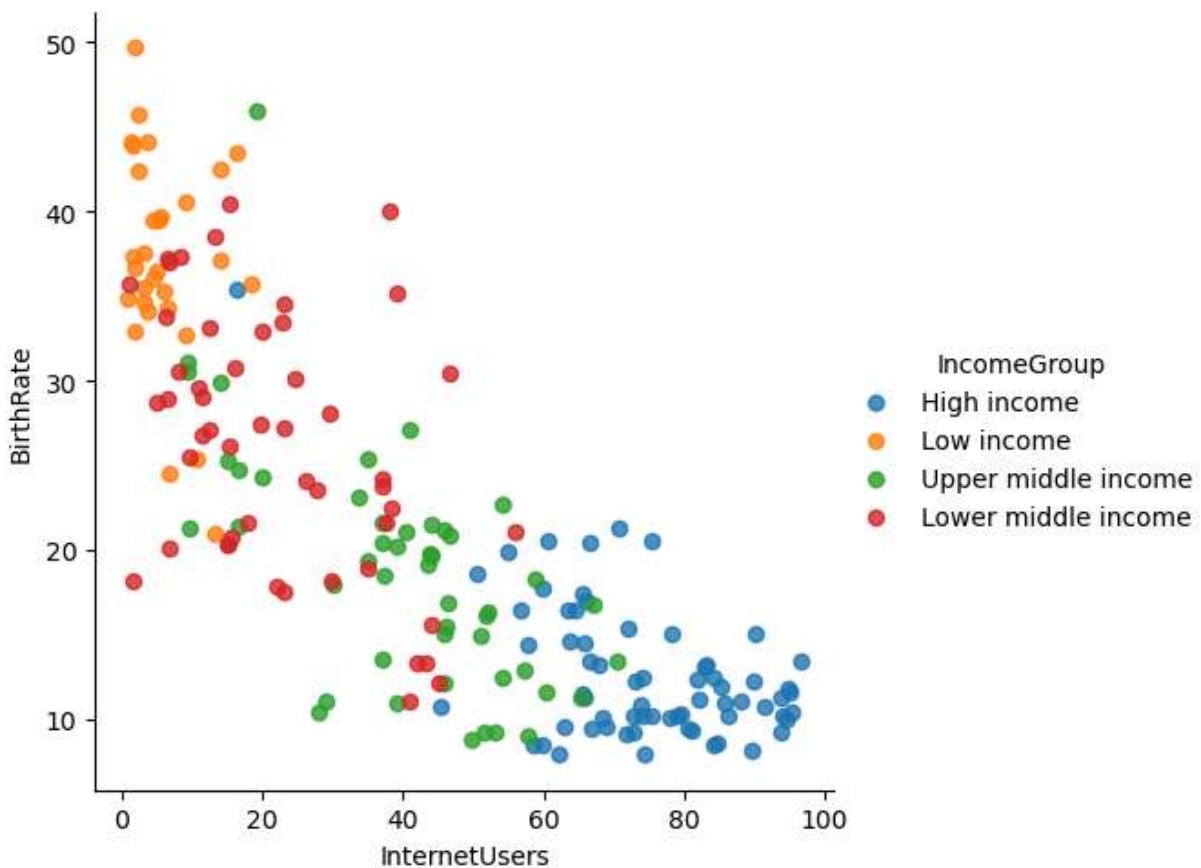
```
In [175]: vis3 = sns.lmplot(data=df, x = "InternetUsers", y = "BirthRate", fit_reg = False, h  
plt.show() # "hue" is same as "Legend"
```



```
In [177]: vis3 = sns.lmplot(data=df, x = "InternetUsers", y = "BirthRate", fit_reg = True, hue=IncomeGroup)
```



```
In [191]: vis5 = sns.lmplot(data = df,x = 'InternetUsers', y = 'BirthRate', fit_reg = False,h  
plt.show()
```



In [ ]:

In [ ]: