In [1]:
```
import pandas as pd
```

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

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

In [3]:
```
df = pd.read_csv(r'C:\Users\Admin\Downloads\Data Science\Daily work\20th, 21st
```

In [4]:
```
df
```

Out[4]:

	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 [5]:
```
# Explore data in python
#1. Full dataframe
#2. How many rows & columns.

len(df) #195 rows imported (this is for tracking later part )
```

Out[5]:
```
195
```

In [6]:
```
#3. see columns
df.columns
```

Out[6]:

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

In [7]:

```
#4. Number of columns

len(df.columns)
```

Out[7]:

```
5
```

In [8]:

```
#5. top rows

df.head() # it will print top 5 rows
```

Out[8]:

	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 [9]:

```
df.head(2)
```

Out[9]:

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

In [10]:

```
#6. Bottom rows
df.tail() #last 5 rows
```

Out[10]:

	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 [11]:
```
df.tail(3)
```

Out[11]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
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 [12]:
```
#7. information of the column

df.info()  #strings are called as object
```

```
<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 [13]:
```
#8. get stats on the columns

df.describe() #it will work like a statistic fun
```

Out[13]:

	BirthRate	InternetUsers
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 [14]:
```
df.describe().transpose() #transpose convert column into rows
```

Out[14]:

	count	mean	std	min	25%	50%	75%	ma
BirthRate	195.0	21.469928	10.605467	7.9	12.1205	19.68	29.7595	49.661
InternetUsers	195.0	42.076471	29.030788	0.9	14.5200	41.00	66.2250	96.546

In [15]:

```
# Renaming columns of a dataframe

df.head()
```

Out[15]:

	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 [16]:

```
df.columns
```

Out[16]:

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

In [17]:

```
df.columns = ['a','b','c','d','e']
df.head()
```

Out[17]:

	a	b	c	d	e
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 [18]:

```
df.columns = ['CountryName', 'CountryCode', 'BirthRate', 'InternetUsers','Inco
```

In [19]:

```
df.head()
```

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

In [20]:

```
# subsetting a dataframes in pandas

#1. Rows
#2. Columns
#3. combine the two
```

In [21]:

```
# Rows:

df[21:26] #how python know that only this is rows based on index
```

Out[21]:

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

In [22]:

```
df[:]
```

Out[22]:

	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 [23]:

```
df[:10]
```

Out[23]:

	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 [24]:

```
df.head(10)
```

Out[24]:

	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 [25]:

```
# How to reverse the dataframe

df[ : : -1]
```

Out[25]:

	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 [26]:

```
df
```

Out[26]:

	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 [27]:

```
# get only every 20th row

df[: : 20]
```

Out[27]:

	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 [28]:

```
# COLUMNS:

df.columns
```

Out[28]:

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

In [29]:

```
df.head()
```

Out[29]:

	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 [30]:

```
df['CountryName'].head()
```

```
Out[30]: 0                  Aruba
         1            Afghanistan
         2                 Angola
         3                Albania
         4   United Arab Emirates
         Name: CountryName, dtype: object
```

```
In [31]: ['CountryName','BirthRate']
```

```
Out[31]: ['CountryName', 'BirthRate']
```

```
In [32]: df[['CountryName','BirthRate']].head()
```

Out[32]:

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

```
In [33]: df.head()
```

Out[33]:

	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 [34]: df['BirthRate']
```

Out[34]:

```
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 [35]:

```
# combine the two

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

Out[35]:

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

In [36]:

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

Out[36]:

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

In [37]:

```
stats = df [['CountryName','BirthRate']]
```

In [38]:

```
stats
```

Out[38]:

	CountryName	BirthRate
0	Aruba	10.244
1	Afghanistan	35.253
2	Angola	45.985
3	Albania	12.877
4	United Arab Emirates	11.044
...	...	...
190	Yemen, Rep.	32.947
191	South Africa	20.850
192	Congo, Dem. Rep.	42.394
193	Zambia	40.471
194	Zimbabwe	35.715

195 rows × 2 columns

In [39]:

```
stats = df[4:8]
```

In [40]:

```
stats
```

Out[40]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
4	United Arab Emirates	ARE	11.044	88.0	High income
5	Argentina	ARG	17.716	59.9	High income
6	Armenia	ARM	13.308	41.9	Lower middle income
7	Antigua and Barbuda	ATG	16.447	63.4	High income

In [41]:

```
# Basic operatioin of dataframe
stats.head()
```

Out[41]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
4	United Arab Emirates	ARE	11.044	88.0	High income
5	Argentina	ARG	17.716	59.9	High income
6	Armenia	ARM	13.308	41.9	Lower middle income
7	Antigua and Barbuda	ATG	16.447	63.4	High income

In [42]:

```
df[['CountryCode','BirthRate','InternetUsers']][4:8] #subet dataframe
```

Out[42]:

	CountryCode	BirthRate	InternetUsers
4	ARE	11.044	88.0
5	ARG	17.716	59.9
6	ARM	13.308	41.9
7	ATG	16.447	63.4

In [43]:

```
df.head()
```

Out[43]:

	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 [44]:

```
#Mathmetical operation =
df.BirthRate * df.InternetUsers
```

Out[44]:

```
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 [45]:

```
# Add a column

df['myCalc'] = df.BirthRate * df.InternetUsers
```

In [46]:

```
df.head()
```

Out[46]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup	myCal
0	Aruba	ABW	10.244	78.9	High income	808.251
1	Afghanistan	AFG	35.253	5.9	Low income	207.992
2	Angola	AGO	45.985	19.1	Upper middle income	878.313
3	Albania	ALB	12.877	57.2	Upper middle income	736.564
4	United Arab Emirates	ARE	11.044	88.0	High income	971.872

In [47]:

```
#Remove a column

df.drop('myCalc',axis = 1)
```

Out[47]:

	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 [48]:

```
df = df.drop('myCalc',axis = 1)
```

In [49]:

```
df['InternetUsers']
```

Out[49]:

```
0      78.9
1       5.9
2      19.1
3      57.2
4      88.0
       ...
190    20.0
191    46.5
192     2.2
193    15.4
194    18.5
Name: InternetUsers, Length: 195, dtype: float64
```

In [50]:

```
df.head()
```

Out[50]:

	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 [51]:
```
df.columns[2]
```

Out[51]:
```
'BirthRate'
```

In [52]:
```
df.InternetUsers<2 #we are checking given condition if its correct true or fal
```

Out[52]:
```
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 [53]:
```
Filter = df.InternetUsers < 2
```

In [54]:
```
Filter
```

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]:
```
df[3:7]
```

Out[55]:

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

In [56]:

```
df[30:40]
```

Out[56]:

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

In [57]:

```
df[Filter] # IT WILL take that row which are false
```

Out[57]:

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

In [58]:

```
len(df[Filter])
```

Out[58]:

```
9
```

In [59]:

```
df.BirthRate>40
```

Out[59]:

```
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 [60]:

```
Filter2 = df.BirthRate>40
```

In [61]:

```
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]:

```
df[Filter2]
```

Out[62]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
2	Angola	AGO	45.985	19.1	Upper middle income
11	Burundi	BDI	44.151	1.3	Low income
14	Burkina Faso	BFA	40.551	9.1	Low income
65	Gambia, The	GMB	42.525	14.0	Low income
115	Mali	MLI	44.138	3.5	Low income
127	Niger	NER	49.661	1.7	Low income
128	Nigeria	NGA	40.045	38.0	Lower middle income
156	Somalia	SOM	43.891	1.5	Low income
167	Chad	TCD	45.745	2.3	Low income
178	Uganda	UGA	43.474	16.2	Low income
192	Congo, Dem. Rep.	COD	42.394	2.2	Low income
193	Zambia	ZMB	40.471	15.4	Lower middle income

In [63]:

```
len(df[Filter2])
```

Out[63]:

```
12
```

In [64]:

```
#Filter and Filter2
Filter & Filter2
```

Out[64]:

```
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 [65]:

```
df[Filter & Filter2]
```

Out[65]:

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

In [66]:

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

Out[66]:

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

In [67]:

```
df.head()
```

Out[67]:

	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 [68]:

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

Out[68]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
1	Afghanistan	AFG	35.253	5.90	Low income
11	Burundi	BDI	44.151	1.30	Low income
13	Benin	BEN	36.440	4.90	Low income
14	Burkina Faso	BFA	40.551	9.10	Low income
29	Central African Republic	CAF	34.076	3.50	Low income
38	Comoros	COM	34.326	6.50	Low income
52	Eritrea	ERI	34.800	0.90	Low income
55	Ethiopia	ETH	32.925	1.90	Low income
64	Guinea	GIN	37.337	1.60	Low income
65	Gambia, The	GMB	42.525	14.00	Low income
66	Guinea-Bissau	GNB	37.503	3.10	Low income
77	Haiti	HTI	25.345	10.60	Low income
93	Cambodia	KHM	24.462	6.80	Low income
99	Liberia	LBR	35.521	3.20	Low income
111	Madagascar	MDG	34.686	3.00	Low income
115	Mali	MLI	44.138	3.50	Low income
120	Mozambique	MOZ	39.705	5.40	Low income
123	Malawi	MWI	39.459	5.05	Low income
127	Niger	NER	49.661	1.70	Low income
132	Nepal	NPL	20.923	13.30	Low income
148	Rwanda	RWA	32.689	9.00	Low income
154	Sierra Leone	SLE	36.729	1.70	Low income
156	Somalia	SOM	43.891	1.50	Low income
158	South Sudan	SSD	37.126	14.10	Low income
167	Chad	TCD	45.745	2.30	Low income
168	Togo	TGO	36.080	4.50	Low income
177	Tanzania	TZA	39.518	4.40	Low income
178	Uganda	UGA	43.474	16.20	Low income
192	Congo, Dem. Rep.	COD	42.394	2.20	Low income
194	Zimbabwe	ZWE	35.715	18.50	Low income

In [69]:

```
# How to get the unique categories

df.IncomeGroup.unique()
```

Out[69]:

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

In [101…

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

Out[101…

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

67 rows × 5 columns

In [70]:

```
# How to get the unique categories

df.IncomeGroup.nunique()
```

Out[70]:

```
4
```

In [71]:

```
# Introduction to seaborn # seaborn is very powerfull visualizatio(STATISTIC V

import matplotlib.pyplot as plt # visulaiztion
import seaborn as sns # distribution visualtion

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

import warnings
warnings.filterwarnings('ignore')
```

```
In [72]: df.head()
```

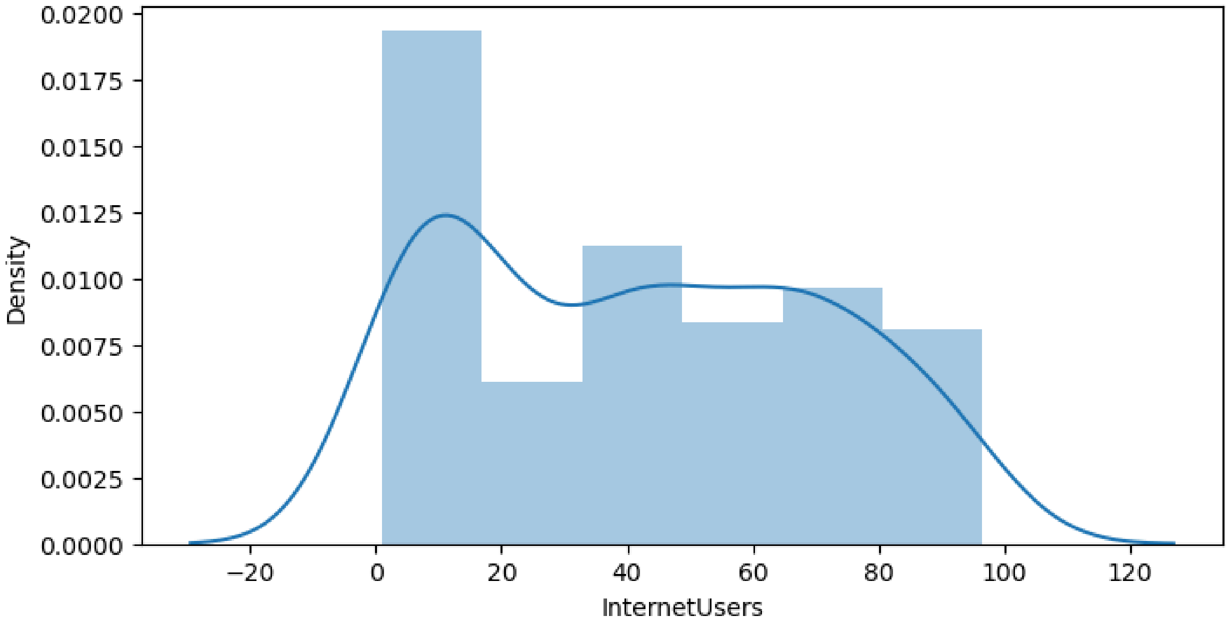
Out[72]:

	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 [73]: df["InternetUsers"]
```

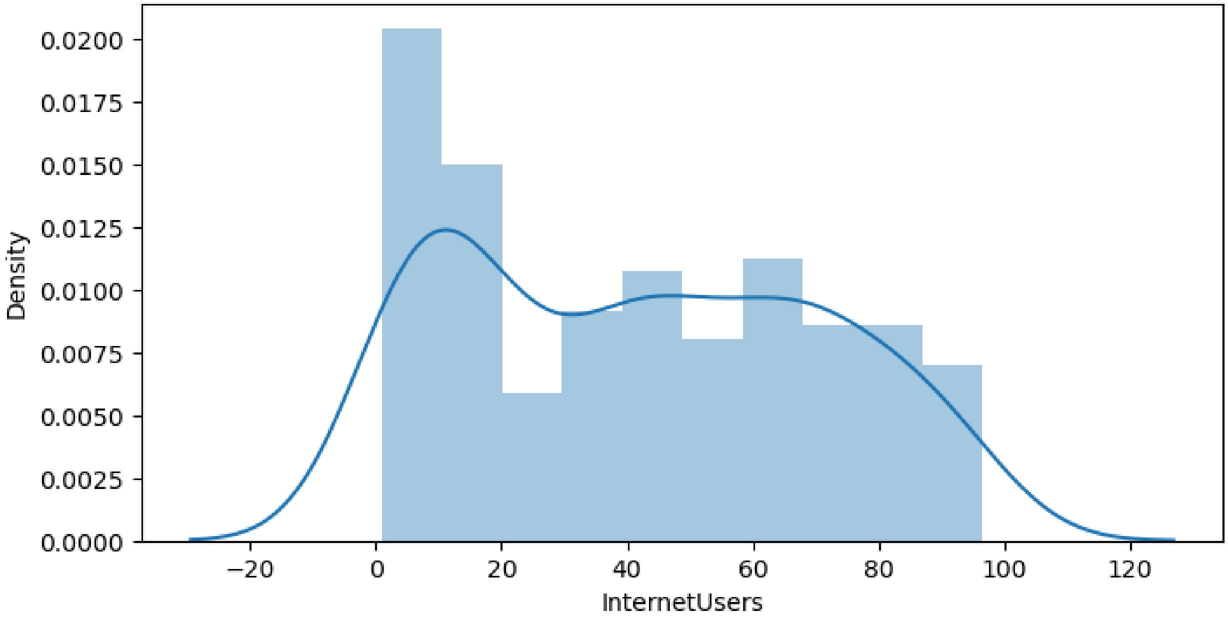
```
Out[73]: 0      78.9
         1       5.9
         2      19.1
         3      57.2
         4      88.0
                ...
         190    20.0
         191    46.5
         192     2.2
         193    15.4
         194    18.5
         Name: InternetUsers, Length: 195, dtype: float64
```

```
In [77]: # Distributions:
         vis1 = sns.distplot(df["InternetUsers"])  #UNIVERIATE Analysis
         plt.show(vis1)
```



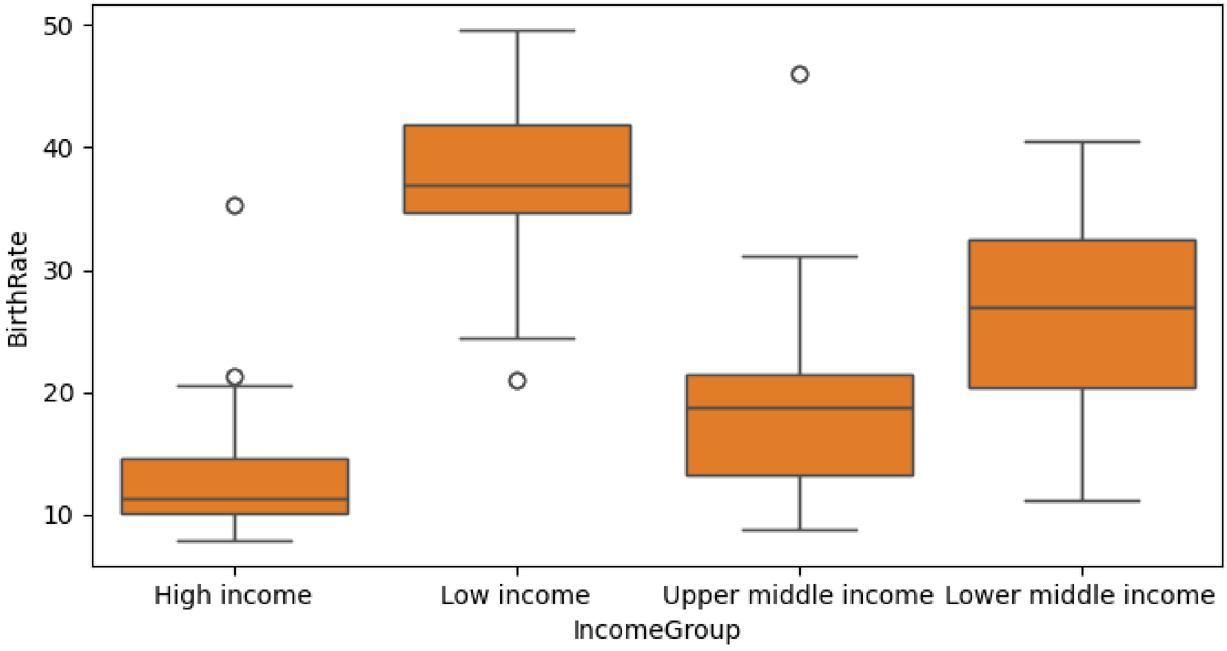
In [78]:

```
vis1 = sns.distplot(df["InternetUsers"], bins=10)
plt.show(vis1)
```



In [80]:

```
#BOX PLOTS:
vis2 = sns.boxplot(data = df, x="IncomeGroup", y='BirthRate')
plt.show(vis2)
```



In [81]:

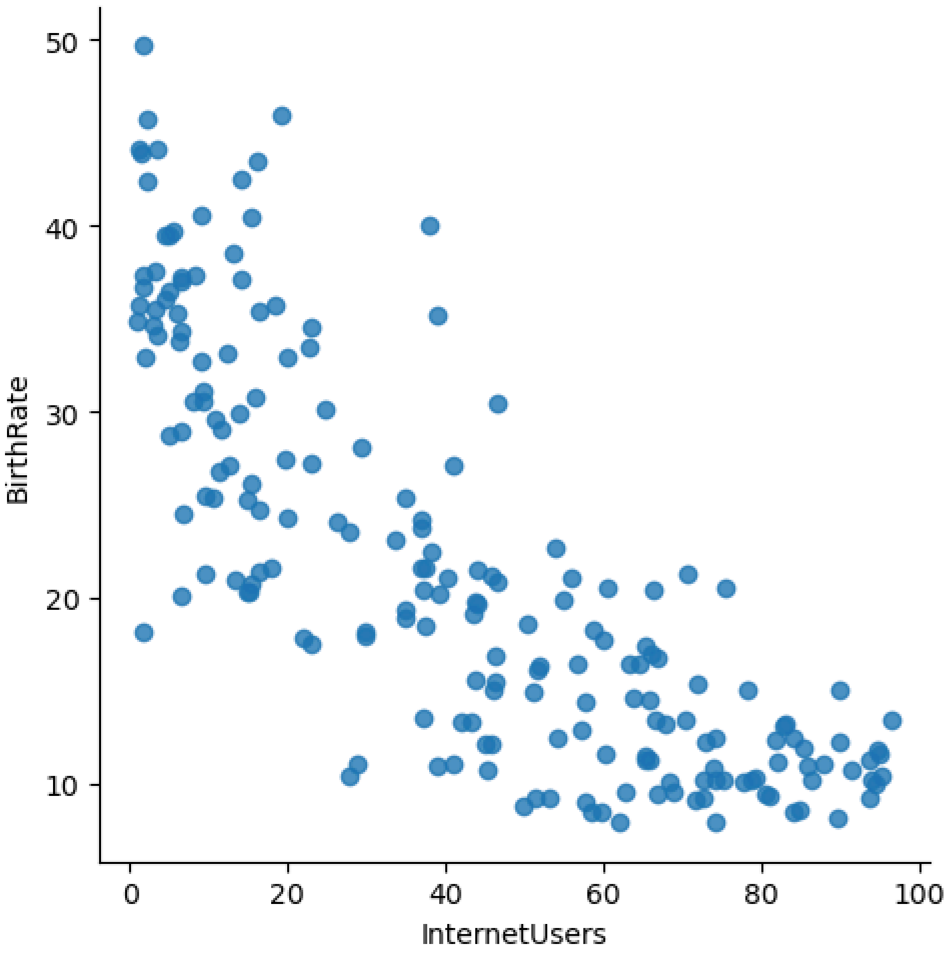
```
# refer to seaborn gallary
```

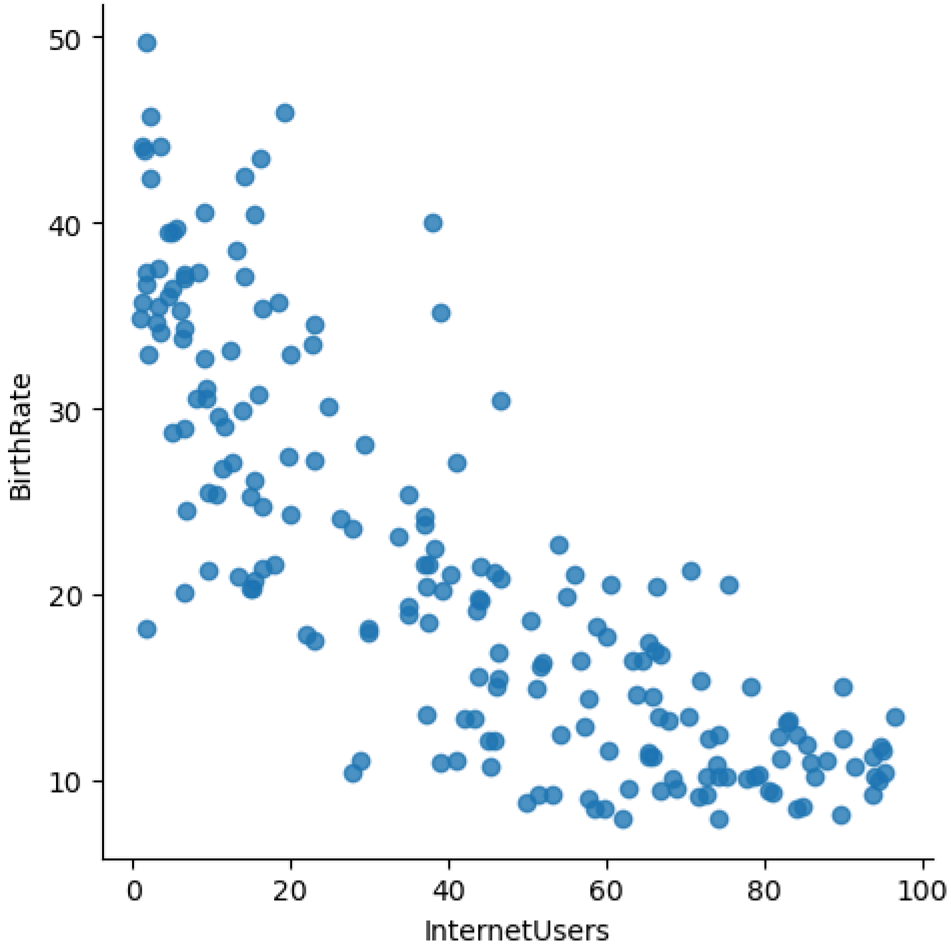
In [82]:

```
# visualizing with seaborn
```

In [84]:

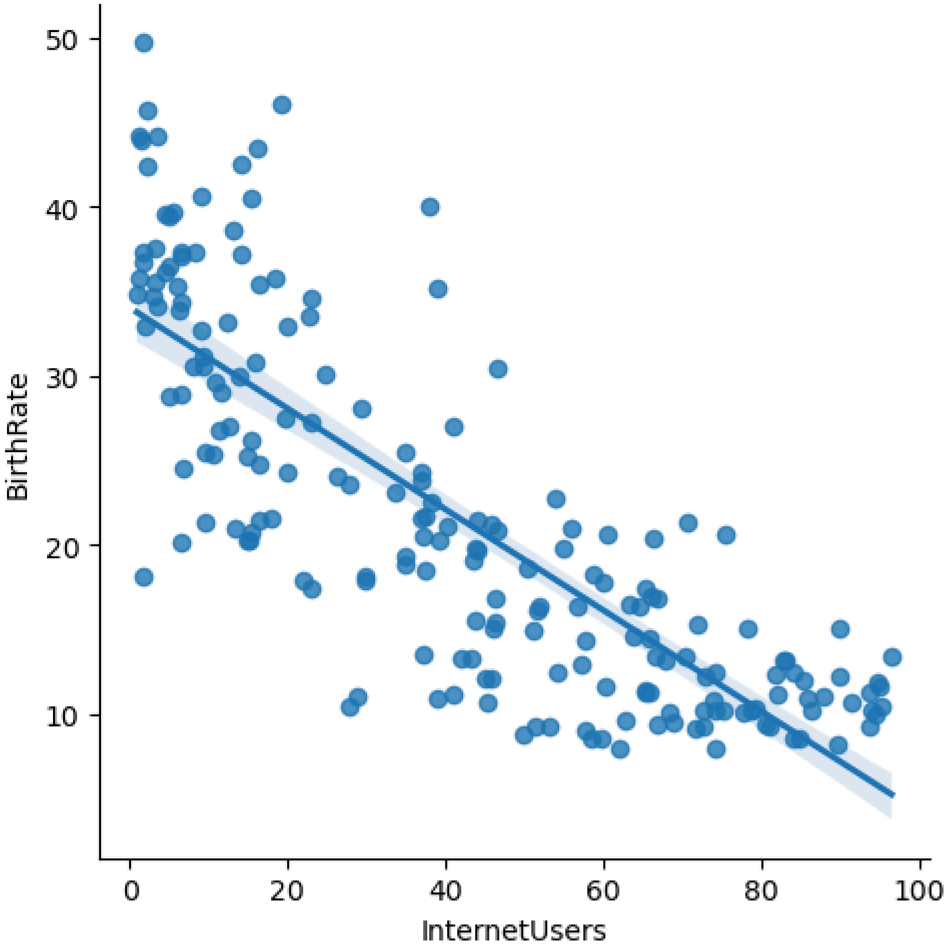
```
vis3 = sns.lmplot(data = df,x = 'InternetUsers', y = 'BirthRate', fit_reg = Fa
plt.show(vis3)
```



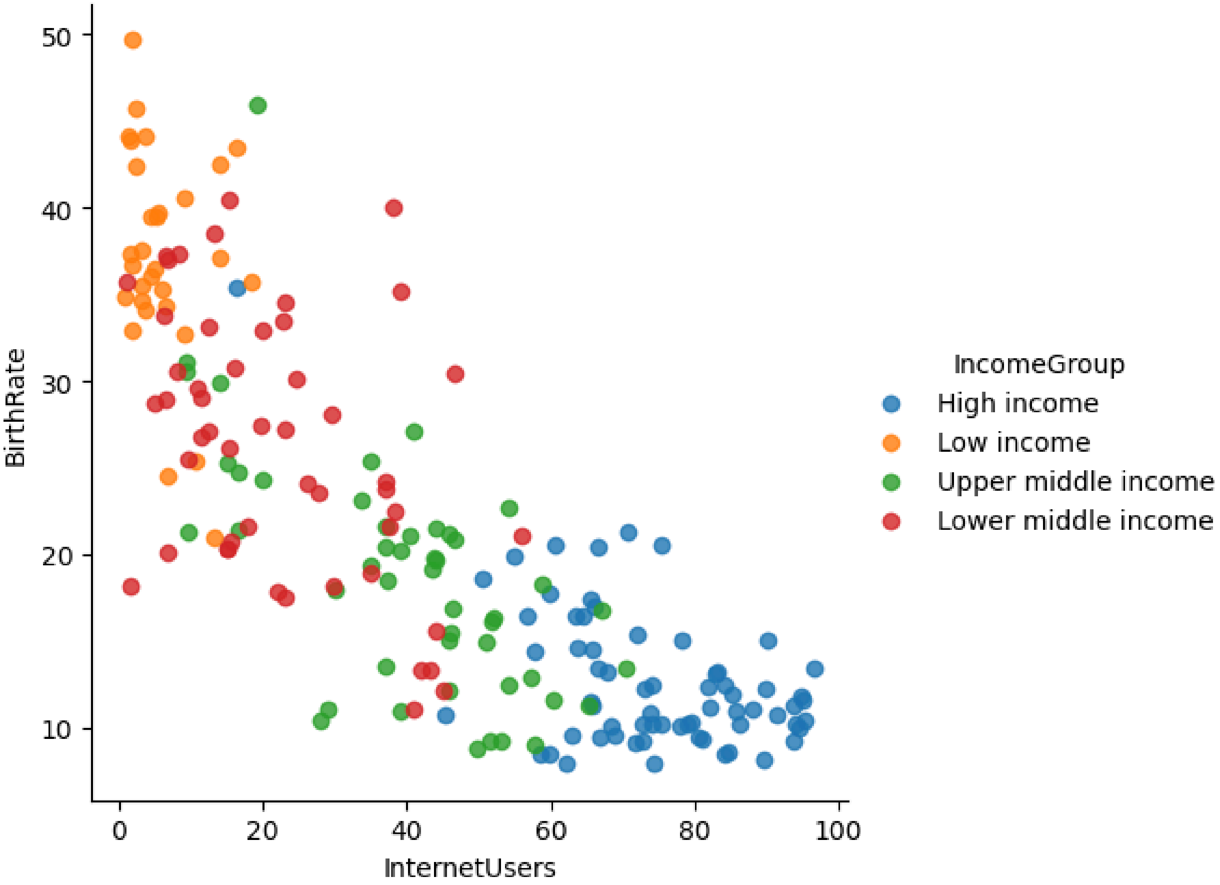


In [87]:

```
vis4 = sns.lmplot(data = df,x = 'InternetUsers', y = 'BirthRate')
plt.show(vis4)
```

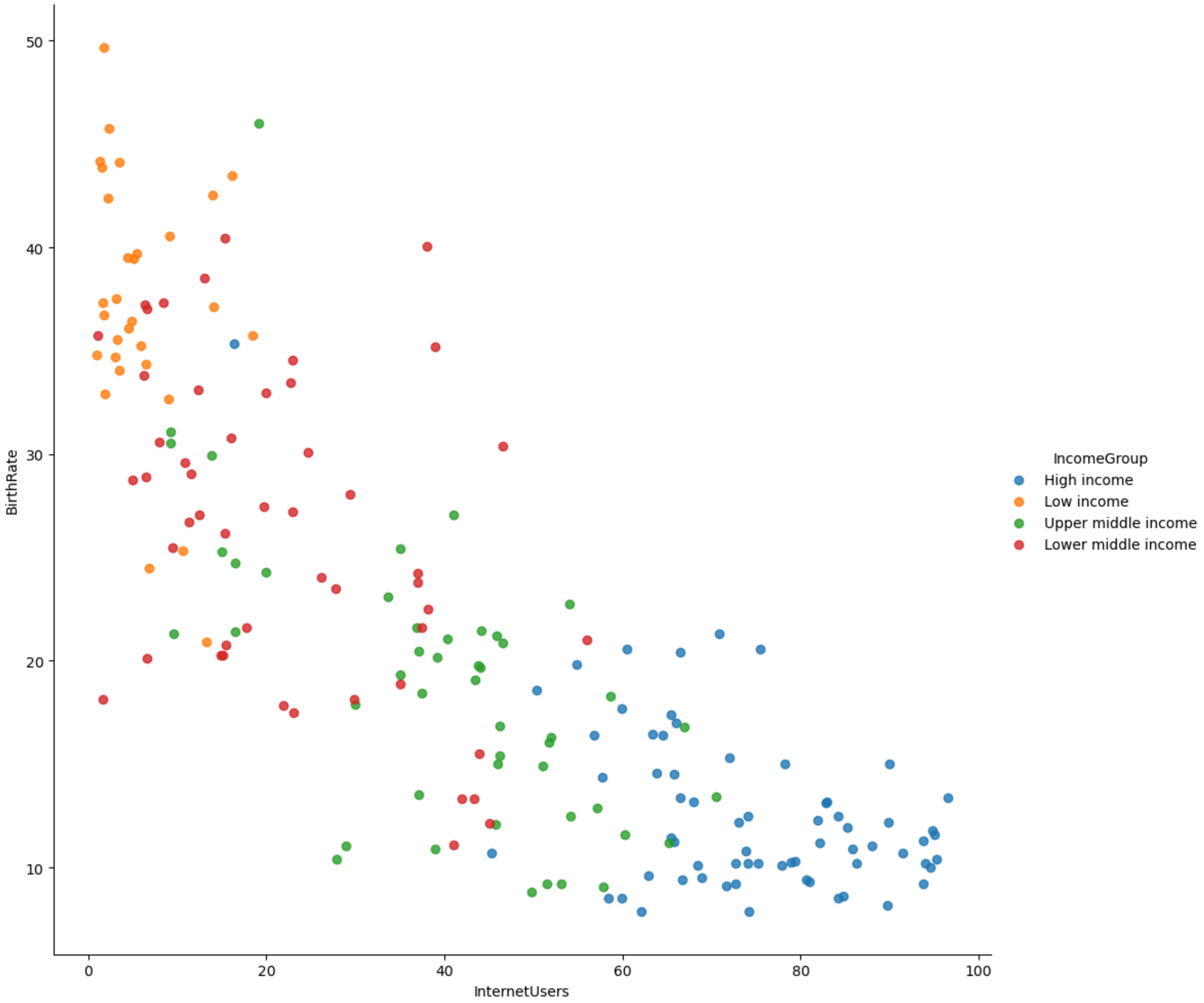


```
In [88]: vis5 = sns.lmplot(data = df,x = 'InternetUsers', y = 'BirthRate',
                           fit_reg = False,hue = 'IncomeGroup') #hue - parameter for cc
         plt.show(vis5)
```



In [95]:

```
vis5 = sns.lmplot(data = df,x = 'InternetUsers', y = 'BirthRate',
                  fit_reg = False,hue = 'IncomeGroup', height = 10) #hue - par
plt.show(vis5)
```



# In this section we learned 1> importing data into python 2> Dataframe via panda 3> exploring datasets: head()tail()info()describe() 4> Renaming columns 5> subsetting dataframes 6> Basic operations with dataframe 8> filtering data frames 9> seaborn introduction

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: