In [ ]:

In [3]:
```
import pandas as pd
```

In [5]:
```
df = pd.read_csv(r'E:\One_Drive(Microsoft)\OneDrive\Data_Science_cource\Module_1_Python_29_July\Assignment\data.csv')
```

In [7]:
```
df
```

Out[7]:

	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 [9]:
```
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 [16]:
```
df.shape
```

Out[16]:
```
(195, 5)
```

In [18]:
```
print(len(df))
```

```
195
```

In [22]:
```
df.columns
```

Out[22]:
```
Index(['CountryName', 'CountryCode', 'BirthRate', 'InternetUsers',
       'IncomeGroup'],
      dtype='object')
```

In [26]:
```
df.isnull()
```

Out[26]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
0	False	False	False	False	False
1	False	False	False	False	False
2	False	False	False	False	False
3	False	False	False	False	False
4	False	False	False	False	False
...	...	...	...	...	...
190	False	False	False	False	False
191	False	False	False	False	False
192	False	False	False	False	False
193	False	False	False	False	False
194	False	False	False	False	False

195 rows × 5 columns

In [30]:
```
print(df.isnull().sum())
```

```
CountryName      0
CountryCode      0
BirthRate        0
InternetUsers    0
IncomeGroup      0
dtype: int64
```

In [32]:
```
df.head()
```

Out[32]:

	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.tail()
```

Out[34]:

	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 [36]:
```
df.tail(1)
```

Out[36]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
194	Zimbabwe	ZWE	35.715	18.5	Low income

In [38]:
```
df.head(2)
```

Out[38]:

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

In [40]:
```
df[::-1] # reverce order
```

Out[40]:

	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 [42]:
```
df[:5] # first five
```

Out[42]:

	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 [46]:
```
df[6:] # from 6th row
```

Out[46]:

	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 [48]:
```
df[0:200:10]
```

Out[48]:

	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 [60]:
```
df.describe() # descriptive statestic
```

Out[60]:

	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 [62]:
```
df.describe().transpose() # it will comver rowas become function
```

Out[62]:

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

In [64]:
```
df.columns
```

Out[64]:
```
Index(['CountryName', 'CountryCode', 'BirthRate', 'InternetUsers',
       'IncomeGroup'],
      dtype='object')
```

In [124…
```
# how a rename a columns
df.columns =  ['a','b','c','d','e']
```

In [126…
```
df.columns
```

Out[126…
```
Index(['a', 'b', 'c', 'd', 'e'], dtype='object')
```

In [130…
```
df.columns = ['CountryName', 'CountryCode', 'BirthRate', 'InternetUsers', 'IncomeGroup']
df.columns
```

Out[130…
```
Index(['CountryName', 'CountryCode', 'BirthRate', 'InternetUsers',
       'IncomeGroup'],
      dtype='object')
```

In [82]:
```
df[['CountryName','CountryCode', 'BirthRate']]
```

Out[82]:

	CountryName	CountryCode	BirthRate
0	Aruba	ABW	10.244
1	Afghanistan	AFG	35.253
2	Angola	AGO	45.985
3	Albania	ALB	12.877
4	United Arab Emirates	ARE	11.044
...	...	...	...
190	Yemen, Rep.	YEM	32.947
191	South Africa	ZAF	20.850
192	Congo, Dem. Rep.	COD	42.394
193	Zambia	ZMB	40.471
194	Zimbabwe	ZWE	35.715

195 rows × 3 columns

In [140…

In [144…
```
df[16:27]
```

Out[144...

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
16	Bulgaria	BGR	9.200	53.06150	Upper middle income
17	Bahrain	BHR	15.040	90.00004	High income
18	Bahamas, The	BHS	15.339	72.00000	High income
19	Bosnia and Herzegovina	BIH	9.062	57.79000	Upper middle income
20	Belarus	BLR	12.500	54.17000	Upper middle income
21	Belize	BLZ	23.092	33.60000	Upper middle income
22	Bermuda	BMU	10.400	95.30000	High income
23	Bolivia	BOL	24.236	36.94000	Lower middle income
24	Brazil	BRA	14.931	51.04000	Upper middle income
25	Barbados	BRB	12.188	73.00000	High income
26	Brunei Darussalam	BRN	16.405	64.50000	High income

In [146...

```
df[:]
```

Out[146...

	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 [148...

```
df[:10:2] # from 0th index to 10-1 and 2 steps
```

Out[148...

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
0	Aruba	ABW	10.244	78.9	High income
2	Angola	AGO	45.985	19.1	Upper middle income
4	United Arab Emirates	ARE	11.044	88.0	High income
6	Armenia	ARM	13.308	41.9	Lower middle income
8	Australia	AUS	13.200	83.0	High income

In [52]:

```
# only one columns data
df.CountryName.head() # first way
```

```
df[['CountryName']].head() # by index
```

Out[52]:

	CountryName
0	Aruba
1	Afghanistan
2	Angola
3	Albania
4	United Arab Emirates

In [150…

```
df.head()
```

Out[150…

	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 [158…

```
# to print the data in reverse order
# syntax
# [step:-Stop_p:-Start_p:]

print(df[:-10:-1])
```

```
            CountryName CountryCode  BirthRate  InternetUsers  \
194            Zimbabwe         ZWE     35.715           18.5
193              Zambia         ZMB     40.471           15.4
192   Congo, Dem. Rep.          COD     42.394            2.2
191        South Africa         ZAF     20.850           46.5
190         Yemen, Rep.         YEM     32.947           20.0
189               Samoa         WSM     26.172           15.3
188  West Bank and Gaza         PSE     30.394           46.6
187             Vanuatu         VUT     26.739           11.3
186             Vietnam         VNM     15.537           43.9

             IncomeGroup
194           Low income
193  Lower middle income
192           Low income
191  Upper middle income
190  Lower middle income
189  Lower middle income
188  Lower middle income
187  Lower middle income
186  Lower middle income
```

In [162…
```
df.columns
```

Out[162…
```
Index(['CountryName', 'CountryCode', 'BirthRate', 'InternetUsers',
       'IncomeGroup'],
      dtype='object')
```

In [168…
```
# to access a atribues first 5 columns
df[['CountryName', 'CountryCode']].head()
```

Out[168…

	CountryName	CountryCode
0	Aruba	ABW
1	Afghanistan	AFG
2	Angola	AGO
3	Albania	ALB
4	United Arab Emirates	ARE

In [174…
```
df[['CountryName', 'CountryCode', 'BirthRate', 'InternetUsers']].tail(4)
print(df[['BirthRate','InternetUsers']].head(3))
```

```
   BirthRate  InternetUsers
0     10.244           78.9
1     35.253            5.9
2     45.985           19.1
```

In [132…
```
df.isnull()
```

Out[132…

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
0	False	False	False	False	False
1	False	False	False	False	False
2	False	False	False	False	False
3	False	False	False	False	False
4	False	False	False	False	False
…	…	…	…	…	…
190	False	False	False	False	False
191	False	False	False	False	False
192	False	False	False	False	False
193	False	False	False	False	False
194	False	False	False	False	False

195 rows × 5 columns

In [176…
```
df.columns
```

Out[176…
```
Index(['CountryName', 'CountryCode', 'BirthRate', 'InternetUsers',
       'IncomeGroup'],
      dtype='object')
```

In [188…

```
# suseting a framework in pandas
# row
# column
# combining 2


# Combining the two columns
df[20:35][['BirthRate', 'InternetUsers','IncomeGroup']]
# also
df[['BirthRate', 'InternetUsers']][30:45]
```

Out[188…

	BirthRate	InternetUsers
30	10.900	85.8000
31	10.200	86.3400
32	13.385	66.5000
33	12.100	45.8000
34	37.320	8.4000
35	37.236	6.4000
36	37.011	6.6000
37	16.076	51.7000
38	34.326	6.5000
39	21.625	37.5000
40	15.022	45.9600
41	10.400	27.9300
42	12.500	74.1000
43	11.436	65.4548
44	10.200	74.1104

In [ ]:
```
# we can also write the
```

In [194…
```
df1 = df[['BirthRate', 'InternetUsers']][20:44]
print(df1)
```

```
    BirthRate  InternetUsers
20     12.500        54.1700
21     23.092        33.6000
22     10.400        95.3000
23     24.236        36.9400
24     14.931        51.0400
25     12.188        73.0000
26     16.405        64.5000
27     18.134        29.9000
28     25.267        15.0000
29     34.076         3.5000
30     10.900        85.8000
31     10.200        86.3400
32     13.385        66.5000
33     12.100        45.8000
34     37.320         8.4000
35     37.236         6.4000
36     37.011         6.6000
37     16.076        51.7000
38     34.326         6.5000
39     21.625        37.5000
40     15.022        45.9600
41     10.400        27.9300
42     12.500        74.1000
43     11.436        65.4548
```

In [198…
```
df2 = df[44:122]
df2
```

Out[198…

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
44	Czech Republic	CZE	10.200	74.1104	High income
45	Germany	DEU	8.500	84.1700	High income
46	Djibouti	DJI	25.486	9.5000	Lower middle income
47	Denmark	DNK	10.000	94.6297	High income
48	Dominican Republic	DOM	21.198	45.9000	Upper middle income
...	...	...	...	...	...
117	Myanmar	MMR	18.119	1.6000	Lower middle income
118	Montenegro	MNE	11.616	60.3100	Upper middle income
119	Mongolia	MNG	24.275	20.0000	Upper middle income
120	Mozambique	MOZ	39.705	5.4000	Low income
121	Mauritania	MRT	33.801	6.2000	Lower middle income

78 rows × 5 columns

In [200…
```
df.columns
```

Out[200…
```
Index(['CountryName', 'CountryCode', 'BirthRate', 'InternetUsers',
       'IncomeGroup'],
      dtype='object')
```

In [ ]:
```

```

In [92]:
```
# mathematical calculation on columns
new_cl = df.BirthRate * df.InternetUsers

df['MyCal'] = new_cl
print(df.head(3))
```

```
   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

      MyCal
0  808.2516
1  207.9927
2  878.3135
```

In [94]:
```
print(df[['MyCal']])
```

```
       MyCal
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

[195 rows x 1 columns]
```

In [96]:
```
# adding a new column in data set
df['MaCal2'] = df.BirthRate * df.InternetUsers
print(df.head(3))
```

```
   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

      MyCal    MaCal2
0  808.2516  808.2516
1  207.9927  207.9927
2  878.3135  878.3135
```

In [98]:
```
# Removing newly added columns
# using df.drop() function
```

```
# syntax: obj.drop('column_name')


# drop() method can take three arguments
# column_name: The name of the column you want to remove.
# axis=1: Specifies that you are dropping a column (use axis=0 for rows).
# inplace=True: Modifies the DataFrame directly. If you want to create a new DataFrame with the column removed,
# set inplace=False or omit this parameter.

# => column_name, axis=1, inplace = True

print(df.columns)
# drop
df.drop('MaCal2',axis=1,inplace= True) # remove/ dropping one column
df.drop('MyCal',axis = 1,inplace=True)
print(df.columns)

#  both the columns are removed
#
```

```
Index(['CountryName', 'CountryCode', 'BirthRate', 'InternetUsers',
       'IncomeGroup', 'MyCal', 'MaCal2'],
      dtype='object')
Index(['CountryName', 'CountryCode', 'BirthRate', 'InternetUsers',
       'IncomeGroup'],
      dtype='object')
```

In [104…

```
# accessing columns
print(df.columns[0])
print(df.columns[1])
print(df.columns[2])
```

```
CountryName
CountryCode
BirthRate
```

In [ ]:

In [86]:

```
df.dtypes
```

Out[86]:
```
CountryName       object
CountryCode       object
BirthRate        float64
InternetUsers    float64
IncomeGroup       object
dtype: object
```

In [94]:
```
df_text = df[['CountryName', 'CountryCode','IncomeGroup']]
df_text
```

Out[94]:

	CountryName	CountryCode	IncomeGroup
0	Aruba	ABW	High income
1	Afghanistan	AFG	Low income
2	Angola	AGO	Upper middle income
3	Albania	ALB	Upper middle income
4	United Arab Emirates	ARE	High income
...	...	...	...
190	Yemen, Rep.	YEM	Lower middle income
191	South Africa	ZAF	Upper middle income
192	Congo, Dem. Rep.	COD	Low income
193	Zambia	ZMB	Lower middle income
194	Zimbabwe	ZWE	Low income

195 rows × 3 columns

In [110…
```
#
df_text.describe()
```

Out[110…

	CountryName	CountryCode	IncomeGroup
count	195	195	195
unique	195	195	4
top	Aruba	ABW	High income
freq	1	1	67

In [108…

```
print(df[0:100:10])
```

```
   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
```

In [142…

```
# conditional and relational operators with df
df[(df.InternetUsers>= 50) & (df.BirthRate <= 20 )]
```

Out[142...

	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
...	...	...	...	...	...
165	Seychelles	SYC	18.600	50.40	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

65 rows × 5 columns

In [144...

```
df[(df.IncomeGroup =='High income') & (df.InternetUsers >= 50 )]
```

Out[144...

	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
...	...	...	...	...	...
165	Seychelles	SYC	18.600	50.40	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

65 rows × 5 columns

In [156...

```
df[['CountryName','BirthRate']][(df.InternetUsers >= 55) & (df.IncomeGroup == 'High income')]
```

Out[156…

	CountryName	BirthRate
0	Aruba	10.244
4	United Arab Emirates	11.044
5	Argentina	17.716
7	Antigua and Barbuda	16.447
8	Australia	13.200
...	...	...
162	Slovenia	10.200
163	Sweden	11.800
174	Trinidad and Tobago	14.590
180	Uruguay	14.374
181	United States	12.500

63 rows × 2 columns

In [170…

```
df[(df.BirthRate >= 40)][['IncomeGroup','CountryName']]
```

Out[170…

	IncomeGroup	CountryName
2	Upper middle income	Angola
11	Low income	Burundi
14	Low income	Burkina Faso
65	Low income	Gambia, The
115	Low income	Mali
127	Low income	Niger
128	Lower middle income	Nigeria
156	Low income	Somalia
167	Low income	Chad
178	Low income	Uganda
192	Low income	Congo, Dem. Rep.
193	Lower middle income	Zambia

In [65]:
```
df
```

Out[65]:

	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 [113…

```
# filters with single and multiple conditions
######filter_1 = df[(df.InternetUsers >= 50) & (df.IncomeGroup == 'High income')] # total result copied to filter_1 =
# filter_1 = (df['InternetUsers'] >=50 ) & (df['IncomeGroup'] == 'High income')
# finter 2 => internet users less than and equal to 50
# filter_2 = (df.InternetUsers <= 50)
# filter 3
filter_3 = (df.BirthRate >= 30) & (df.IncomeGroup == 'High income')



# anther way to write the filter condition

filter_1 = (df['BirthRate'] < 40) & (df['IncomeGroup'] == 'Low income')
# filter_2
filter_2 = (df.BirthRate > 42)
```

```
# filter_3
filter_3 = (df.IncomeGroup == 'Upper middle income')
```

In [91]:
```
df[filter_1] # filter_1
```

Out[91]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
1	Afghanistan	AFG	35.253	5.90	Low income
13	Benin	BEN	36.440	4.90	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
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
120	Mozambique	MOZ	39.705	5.40	Low income
123	Malawi	MWI	39.459	5.05	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
158	South Sudan	SSD	37.126	14.10	Low income
168	Togo	TGO	36.080	4.50	Low income
177	Tanzania	TZA	39.518	4.40	Low income
194	Zimbabwe	ZWE	35.715	18.50	Low income

In [95]:
```
df[filter_2]
```

Out[95]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
2	Angola	AGO	45.985	19.1	Upper middle income
11	Burundi	BDI	44.151	1.3	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
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

In [107…

```
df[filter_3]
df[filter_3]
```

Out[107...

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
2	Angola	AGO	45.985	19.100000	Upper middle income
3	Albania	ALB	12.877	57.200000	Upper middle income
10	Azerbaijan	AZE	18.300	58.700000	Upper middle income
16	Bulgaria	BGR	9.200	53.061500	Upper middle income
19	Bosnia and Herzegovina	BIH	9.062	57.790000	Upper middle income
20	Belarus	BLR	12.500	54.170000	Upper middle income
21	Belize	BLZ	23.092	33.600000	Upper middle income
24	Brazil	BRA	14.931	51.040000	Upper middle income
28	Botswana	BWA	25.267	15.000000	Upper middle income
33	China	CHN	12.100	45.800000	Upper middle income
37	Colombia	COL	16.076	51.700000	Upper middle income
40	Costa Rica	CRI	15.022	45.960000	Upper middle income
41	Cuba	CUB	10.400	27.930000	Upper middle income
48	Dominican Republic	DOM	21.198	45.900000	Upper middle income
49	Algeria	DZA	24.738	16.500000	Upper middle income
50	Ecuador	ECU	21.070	40.353684	Upper middle income
57	Fiji	FJI	20.463	37.100000	Upper middle income
60	Gabon	GAB	30.555	9.200000	Upper middle income
69	Grenada	GRD	19.334	35.000000	Upper middle income
82	Iran, Islamic Rep.	IRN	17.900	29.950000	Upper middle income
83	Iraq	IRQ	31.093	9.200000	Upper middle income
87	Jamaica	JAM	13.540	37.100000	Upper middle income

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
88	Jordan	JOR	27.046	41.000000	Upper middle income
90	Kazakhstan	KAZ	22.730	54.000000	Upper middle income
98	Lebanon	LBN	13.426	70.500000	Upper middle income
100	Libya	LBY	21.425	16.500000	Upper middle income
101	St. Lucia	LCA	15.430	46.200000	Upper middle income
112	Maldives	MDV	21.447	44.100000	Upper middle income
113	Mexico	MEX	19.104	43.460000	Upper middle income
114	Macedonia, FYR	MKD	11.222	65.240000	Upper middle income
118	Montenegro	MNE	11.616	60.310000	Upper middle income
119	Mongolia	MNG	24.275	20.000000	Upper middle income
122	Mauritius	MUS	10.900	39.000000	Upper middle income
124	Malaysia	MYS	16.805	66.970000	Upper middle income
125	Namibia	NAM	29.937	13.900000	Upper middle income
136	Panama	PAN	19.680	44.030000	Upper middle income
137	Peru	PER	20.198	39.200000	Upper middle income
143	Paraguay	PRY	21.588	36.900000	Upper middle income
146	Romania	ROU	8.800	49.764500	Upper middle income
157	Serbia	SRB	9.200	51.500000	Upper middle income
160	Suriname	SUR	18.455	37.400000	Upper middle income
169	Thailand	THA	11.041	28.940000	Upper middle income
171	Turkmenistan	TKM	21.322	9.600000	Upper middle income
173	Tonga	TON	25.409	35.000000	Upper middle income

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
175	Tunisia	TUN	19.800	43.800000	Upper middle income
176	Turkey	TUR	16.836	46.250000	Upper middle income
183	St. Vincent and the Grenadines	VCT	16.306	52.000000	Upper middle income
191	South Africa	ZAF	20.850	46.500000	Upper middle income

In [119…
```
# applying the two filters at ones
df[filter_1 & filter_2]
```

Out[119…

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup

In [123…
```
# applying to three filters
df[filter_1 & filter_2 & filter_3]
```

Out[123…

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup

In [125…
```
df[filter_2 | filter_1]
```

Out[125...

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
1	Afghanistan	AFG	35.253	5.90	Low income
2	Angola	AGO	45.985	19.10	Upper middle income
11	Burundi	BDI	44.151	1.30	Low income
13	Benin	BEN	36.440	4.90	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

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
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 [130…
```
len(df[filter_1])
```

Out[130…
```
21
```

In [132…
```
len(df[filter_1 | filter_2])
```

Out[132…
```
30
```

In [136…
```
len(df[filter_2 & filter_3])
```

Out[136…
```
1
```

In [178…
```
filter_4 = (df[['IncomeGroup']][df.IncomeGroup == 'Upper middle income'])
```

In [195…
```
# df[filter_4] filter created as object
filter_4 # only upper-middle income column values will print
print("in data frame there are total = ",len(filter_4)," upper middle incomes ")
```

```
in data frame there are total =  48  upper middle incomes
```

In [199…
```
print("\nIn Data frame there are total ",len(df.IncomeGroup == 'Low income')," Low income groups")
```

```
In Data frame there are total  195  Low income groups
```

In [201…
```
# unique and nunique
df.IncomeGroup.unique()
```

Out[201…
```
array(['High income', 'Low income', 'Upper middle income',
       'Lower middle income'], dtype=object)
```

In [203…
```
df.IncomeGroup.nunique()
```

Out[203…
```
4
```

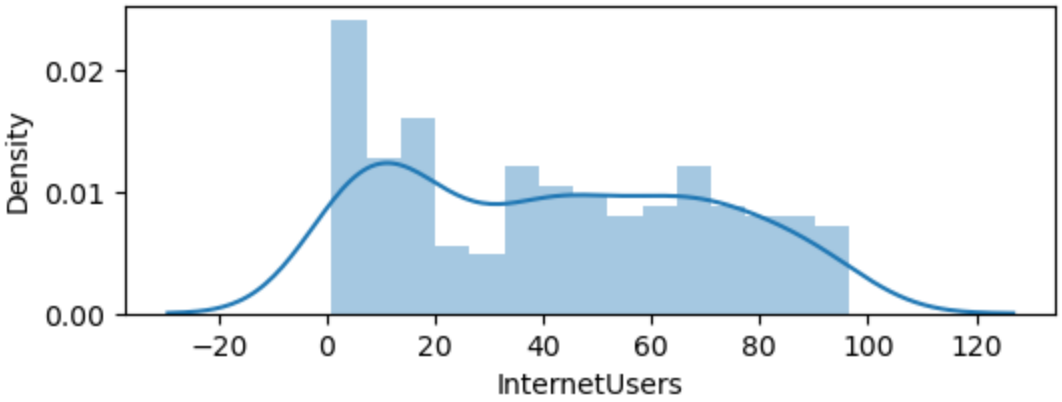
In [ ]:

In [ ]:

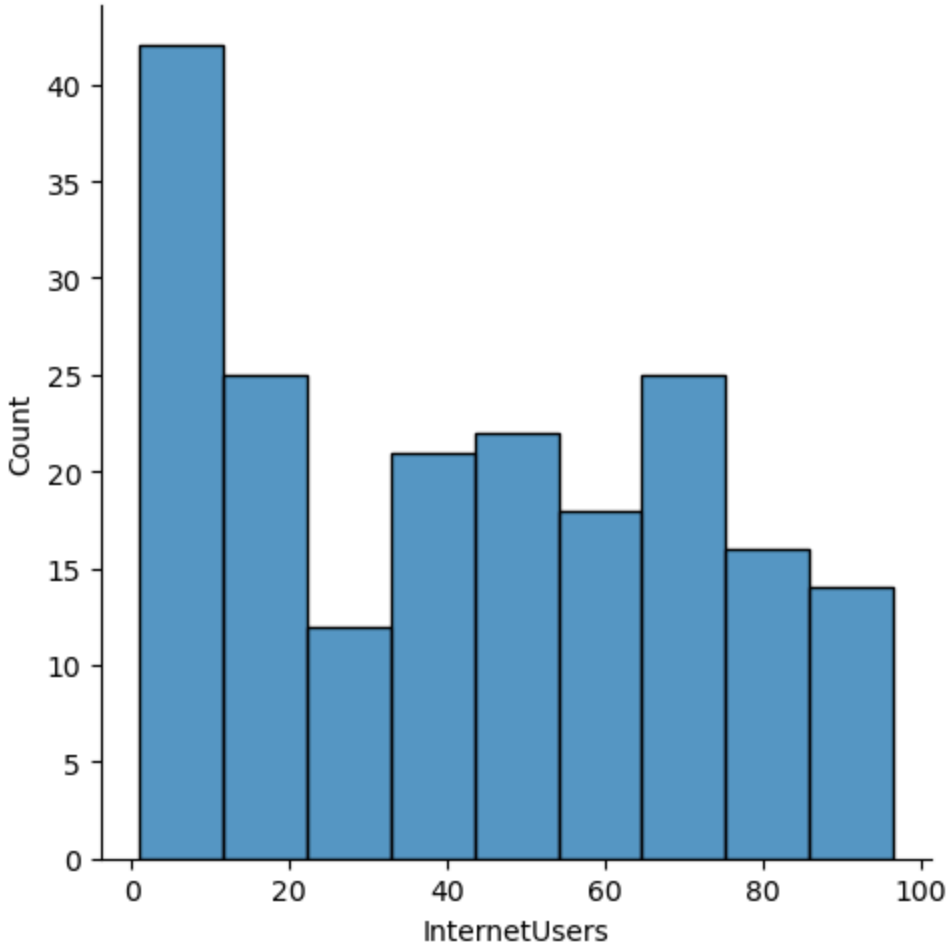
In [112…
```
# df.[:8][]
```

In [11]:
```
# import matplotlib as plt
# import seaborn as sns
# %%matplotlib inline
# plt .rcParams['figure.figuresize'
# Introduction to seaborn # seaborn is very powerfull visualizatio(STATISTIC VISULAIZATION) pkg in python

import matplotlib.pyplot as plt # visulaiztion
import seaborn as sns # distribution visualtion
# seaborn are used for advance visualization e.x --> distribution plot, line plot

%matplotlib inline
plt.rcParams['figure.figsize'] = 6,2

import warnings
warnings.filterwarnings('ignore') # os error
```

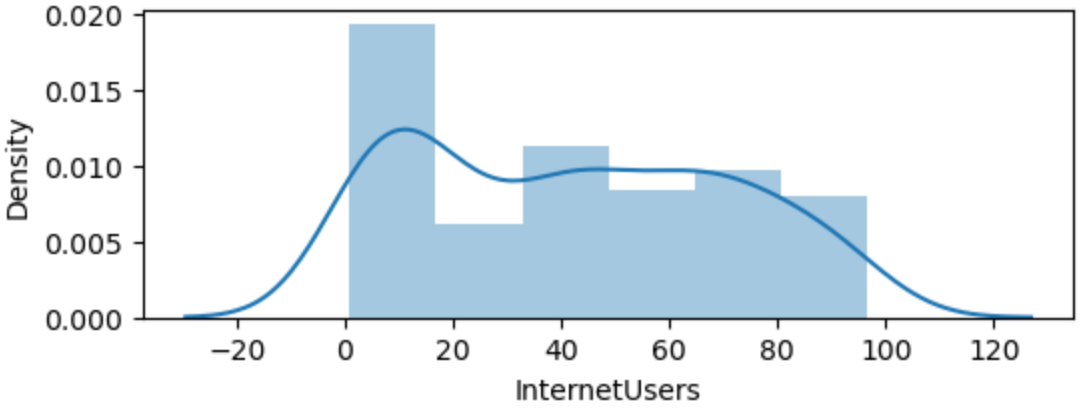
In [25]:
```
vis1 = sns.distplot(df["InternetUsers"],bins = 15)
```



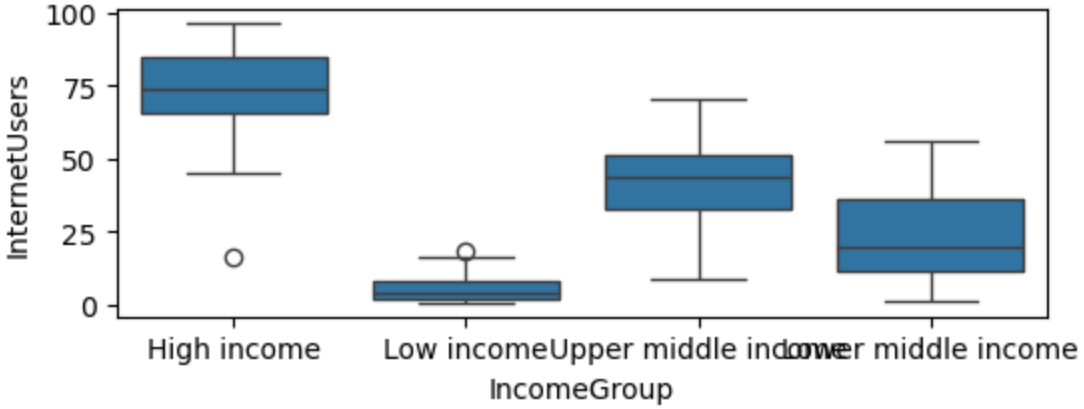
In [29]:
```
vis1 = sns.displot(df["InternetUsers"])
```



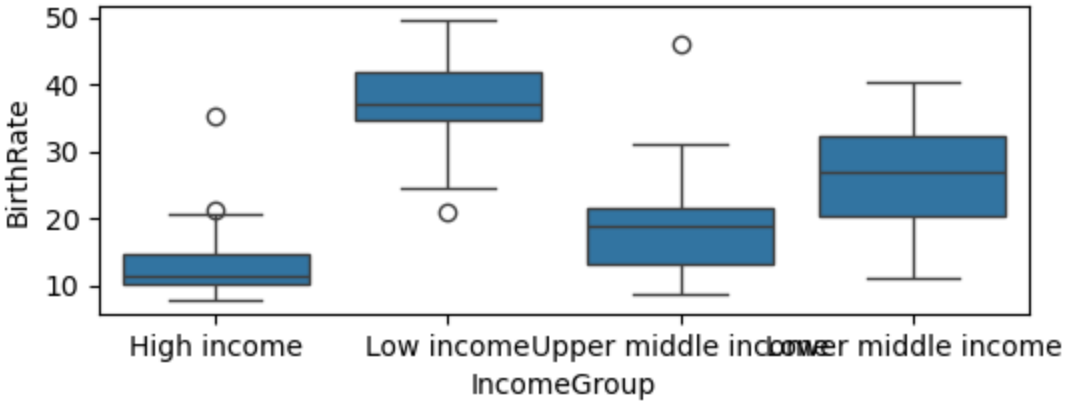
In [37]:
```
vis1 = sns.distplot(df["InternetUsers"])
```



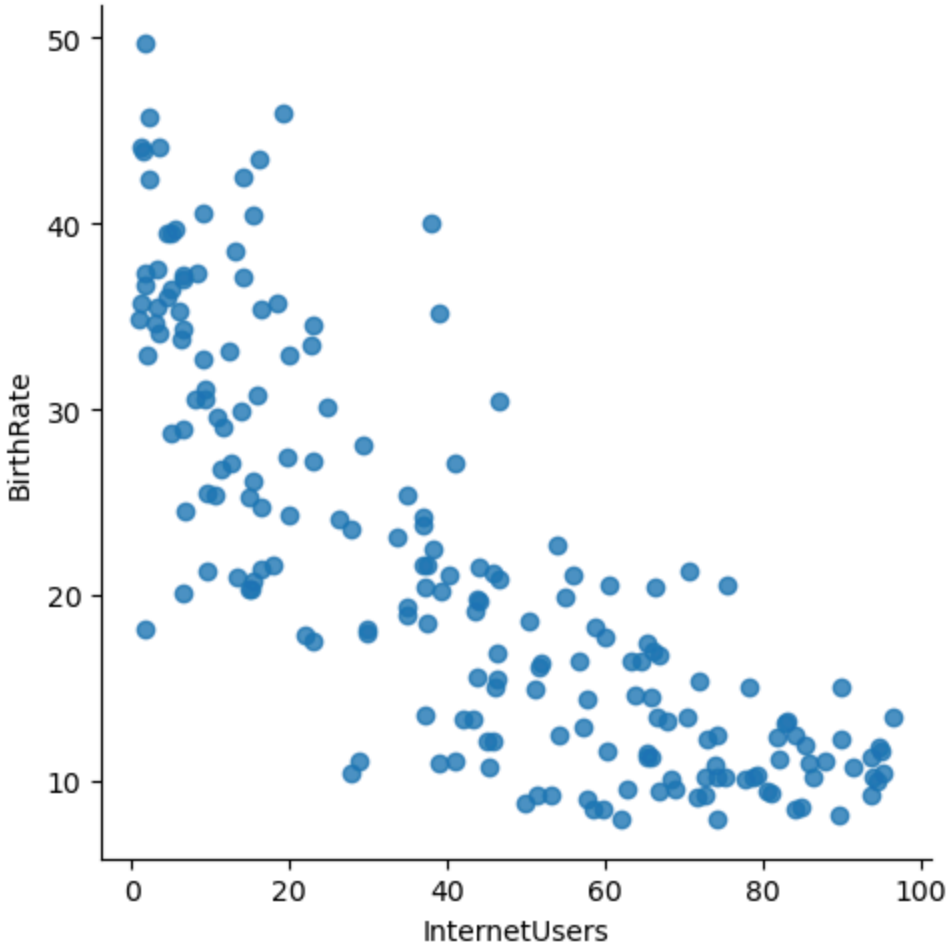
In [41]:
```
# seaborn boxplot
vis2 = sns.boxplot(data = df, x = "IncomeGroup", y = "InternetUsers")
```



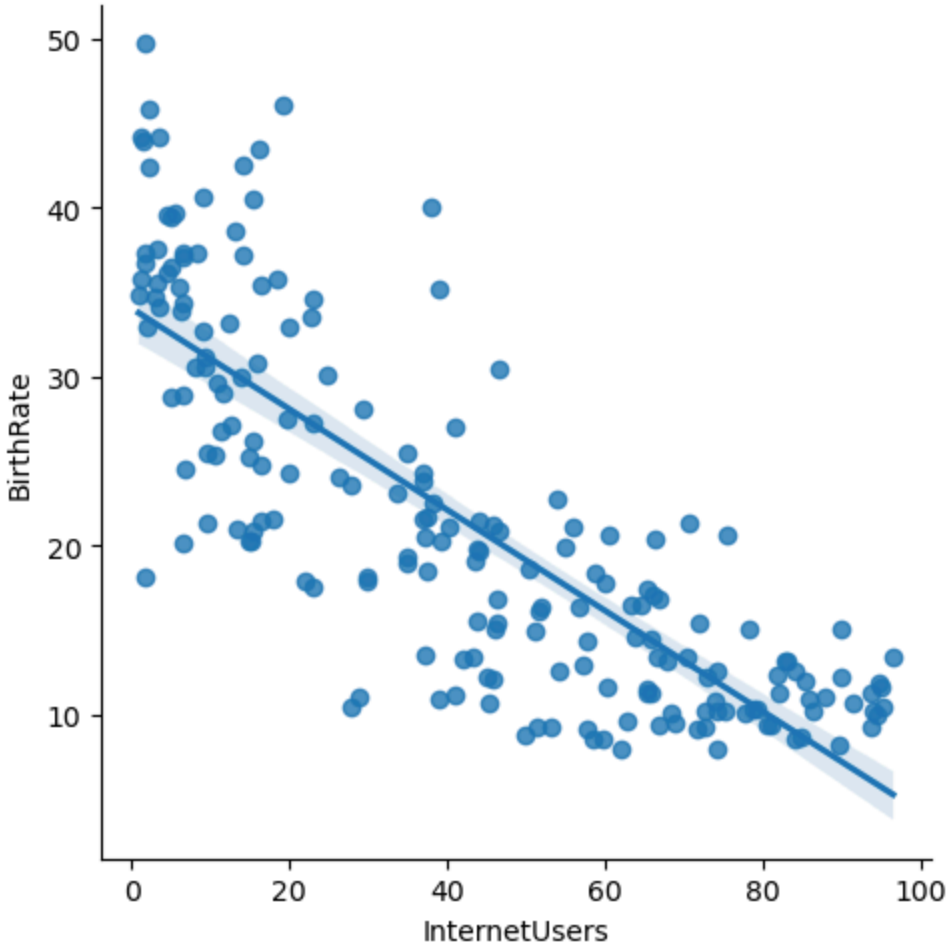
In [45]:
```
vis3 = sns.boxplot(data = df, x = "IncomeGroup", y = "BirthRate")
```



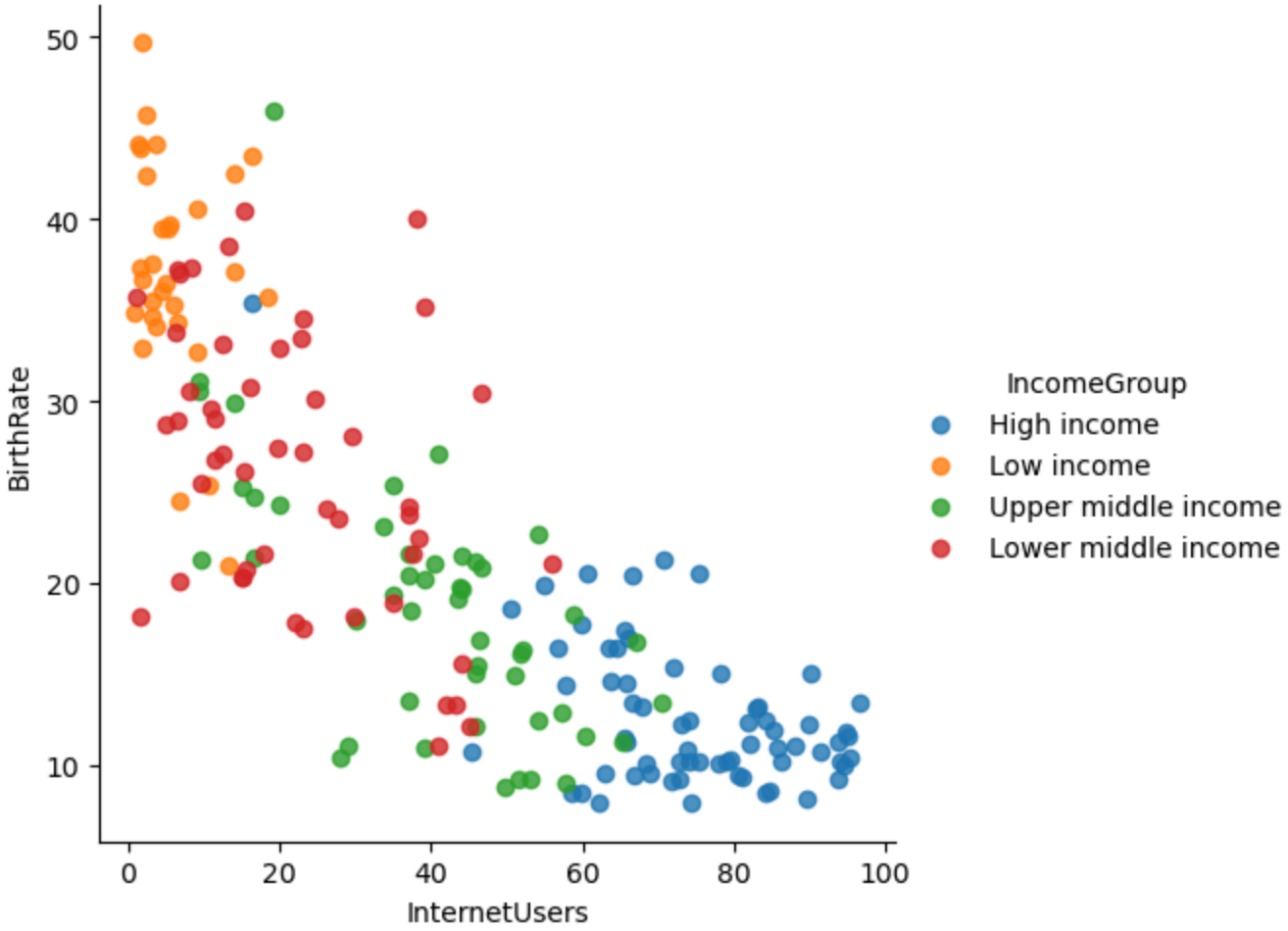
In [55]:
```
vis4 = sns.lmplot(data = df, x = "InternetUsers", y = "BirthRate", fit_reg = False)
```



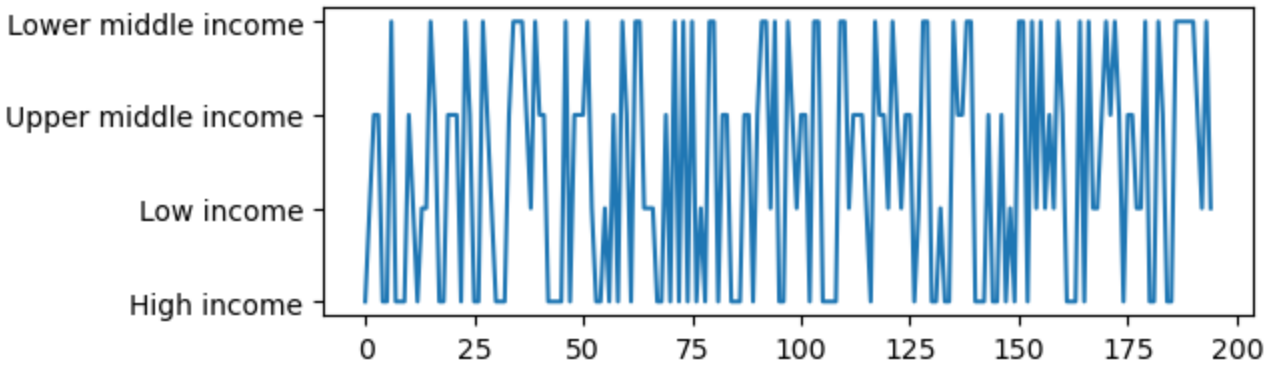
In [67]:
```
vis3 = sns.lmplot(data = df, x = "InternetUsers", y = "BirthRate",  fit_reg = True) # bydefault fig_reg = True
```



In [75]:
```
vis5 = sns.lmplot(data = df, x = "InternetUsers",  y = "BirthRate", fit_reg = False, hue = "IncomeGroup")
```



In [84]:
```
plt.plot(df["IncomeGroup"])
plt.show()

# plt.plot()
# plt.show()
```


Lower middle income
Upper middle income
Low income
High income
0
25
50
75
100
125
150
175
200

In [ ]: