In [1]:
```
import pandas as pd  #use for dataframes
```

In [151…
```
df = pd.read_csv(r'C:\Users\kavya\OneDrive\Documents\data.csv')
```

In [3]:
```
df
```

Out[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
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 [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]:
```
len(df.columns) #number of columns
```

Out[9]: 5

In [10]:
```
df.head()
```

Out[10]:

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

Out[11]:

	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 [12]:
```
df.tail(2)
```

Out[12]:

	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 [13]:
```
df.head(2)
```

Out[13]:

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

In [14]:
```
df[::-1]
```

Out[14]:

	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 [15]:
```
df[6:]
```

Out[15]:

	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 [16]:
```
df[0:200:10]
```

Out[16]:

	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 [17]:
```
df.describe() #describe give information about stats, and numerical information
```

Out[17]:

	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 [18]:
```
df.describe().transpose() #transpose means rows convert into columnss
```

Out[18]:

	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 [19]:
```
df.columns=['a','b','c','d','e']
```

In [39]:
```
df.head(1)
```

Out[39]:

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

In [41]:
```
df.columns = ['CountryName', 'CountryCode', 'BirthRate', 'InternetUsers',
       'IncomeGroup']
```

In [43]:
```
df.head(1)
```

Out[43]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
0	Aruba	ABW	10.244	78.9	High income

In [45]:
```
df[['CountryName', 'CountryCode', 'BirthRate']]
```

Out[45]:

	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 [47]:
```
df.isnull()
```

Out[47]:

	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 [49]:
```
df.isnull().sum()
```

Out[49]:
```
CountryName      0
CountryCode      0
BirthRate        0
InternetUsers    0
IncomeGroup      0
dtype: int64
```

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

Out[51]:
```
Index(['CountryName', 'CountryCode', 'BirthRate', 'InternetUsers',
       'IncomeGroup'],
      dtype='object')
```

In [53]:
```
df_categoical = df[['CountryName', 'CountryCode','IncomeGroup']]
df_categoical.head()
```

Out[53]:

	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

In [55]:
```
df.describe()
```

Out[55]:

	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 [57]:
```
df_categoical.describe()
```

Out[57]:

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

In [59]:
```
df_categoical.describe().transpose()
```

Out[59]:

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

In [61]:
```
df_num = df[['BirthRate', 'InternetUsers']]
df_num.head()
```

Out[61]:

	BirthRate	InternetUsers
0	10.244	78.9
1	35.253	5.9
2	45.985	19.1
3	12.877	57.2
4	11.044	88.0

In [63]:
```
df.head()
```

Out[63]:

	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 [65]:
```
#mathematical operation =
df.BirthRate =  df.InternetUsers
```

In [153…
```
df['mycalc'] = df.BirthRate * df.InternetUsers
```

In [69]:
```
df.columns
```

Out[69]:
```
Index(['CountryName', 'CountryCode', 'BirthRate', 'InternetUsers',
       'IncomeGroup', 'mycalc'],
      dtype='object')
```

In [71]:
```
df
```

Out[71]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup	mycalc
0	Aruba	ABW	78.9	78.9	High income	6225.21
1	Afghanistan	AFG	5.9	5.9	Low income	34.81
2	Angola	AGO	19.1	19.1	Upper middle income	364.81
3	Albania	ALB	57.2	57.2	Upper middle income	3271.84
4	United Arab Emirates	ARE	88.0	88.0	High income	7744.00
...	...	...	...	...	...	...
190	Yemen, Rep.	YEM	20.0	20.0	Lower middle income	400.00
191	South Africa	ZAF	46.5	46.5	Upper middle income	2162.25
192	Congo, Dem. Rep.	COD	2.2	2.2	Low income	4.84
193	Zambia	ZMB	15.4	15.4	Lower middle income	237.16
194	Zimbabwe	ZWE	18.5	18.5	Low income	342.25

195 rows × 6 columns

In [73]:
```
df = df.drop('mycalc',axis =1)
```

In [75]:
```
df.columns
```

Out[75]:
```
Index(['CountryName', 'CountryCode', 'BirthRate', 'InternetUsers',
       'IncomeGroup'],
      dtype='object')
```

In [77]:
```
df['InternetUsers']
```

Out[77]:
```
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 [79]:
```
df.InternetUsers<2 # checking which country has less than 2 internet users
```

Out[79]:
```
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 [81]:
```
df.IncomeGroup.unique()
```

Out[81]:
```
array(['High income', 'Low income', 'Upper middle income',
       'Lower middle income'], dtype=object)
```

In [83]:
```
df.IncomeGroup.nunique() #display no of unique categories
```

Out[83]:
```
4
```

In [103…
```
Filter = df.InternetUsers<2
```

In [105…
```
Filter
```

Out[105…
```
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 [107…
```
df[3:7]
```

Out[107…

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

In [109…
```
df[Filter]
```

Out[109…

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

In [111…
```
df.BirthRate>40
```

Out[111…

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

In [113…

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

Out[113...

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

In [133...

```
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

```
plt.rcParams['figure.figsize']=6,2
import warnings
warnings.filterwarnings('ignore')
```

In [135…
```
df.head()
```

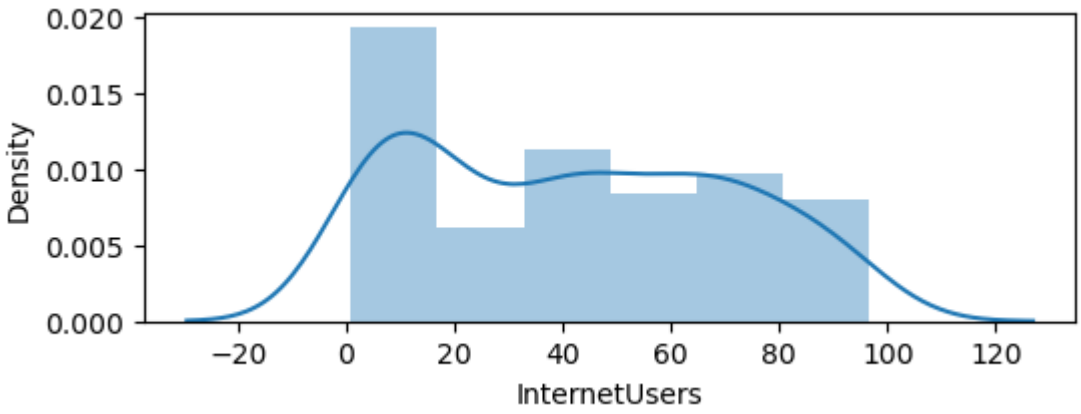
Out[135…

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
0	Aruba	ABW	78.9	78.9	High income
1	Afghanistan	AFG	5.9	5.9	Low income
2	Angola	AGO	19.1	19.1	Upper middle income
3	Albania	ALB	57.2	57.2	Upper middle income
4	United Arab Emirates	ARE	88.0	88.0	High income

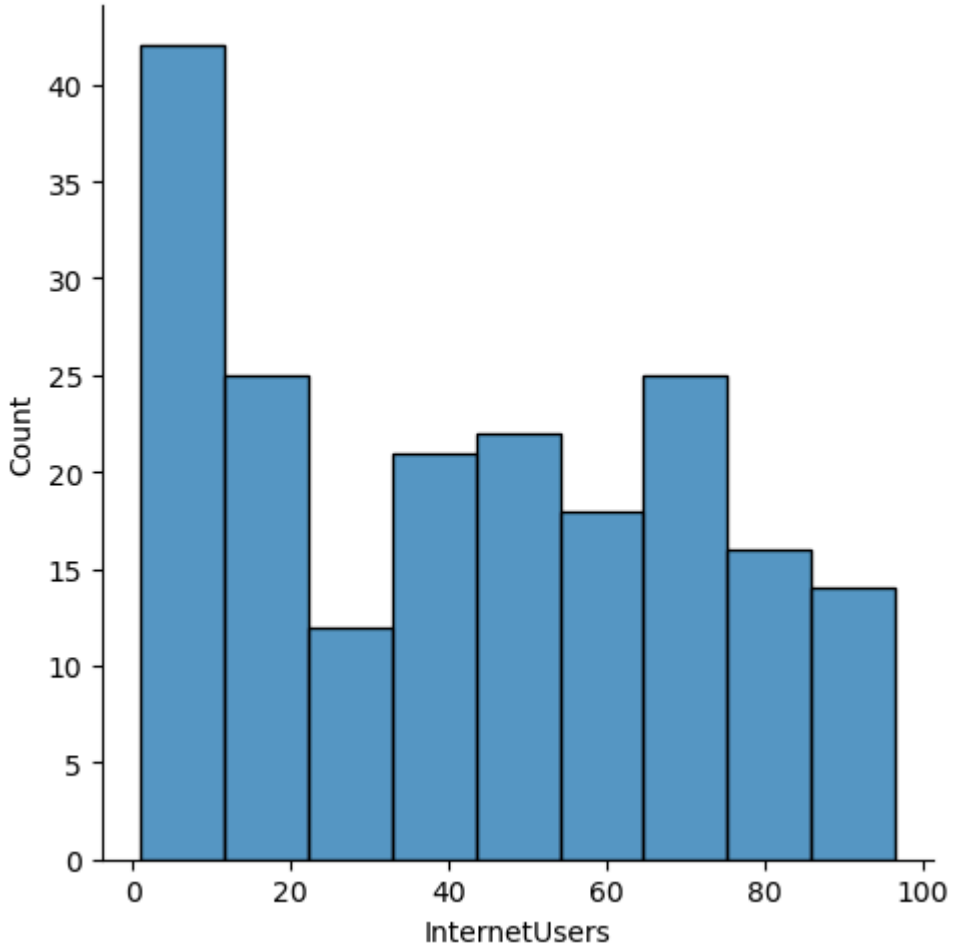
In [137…
```
df["InternetUsers"]
```

Out[137…
```
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
```

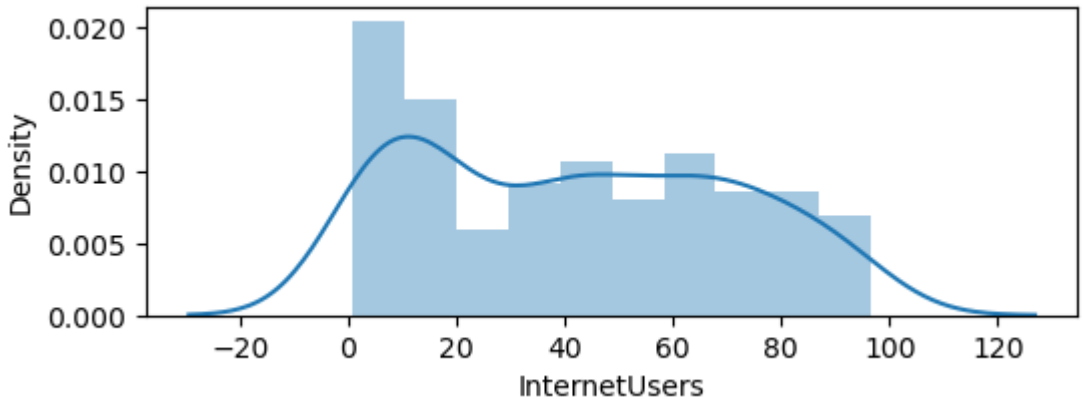
In [169…
```
vis1 = sns.distplot(df["InternetUsers"])
```



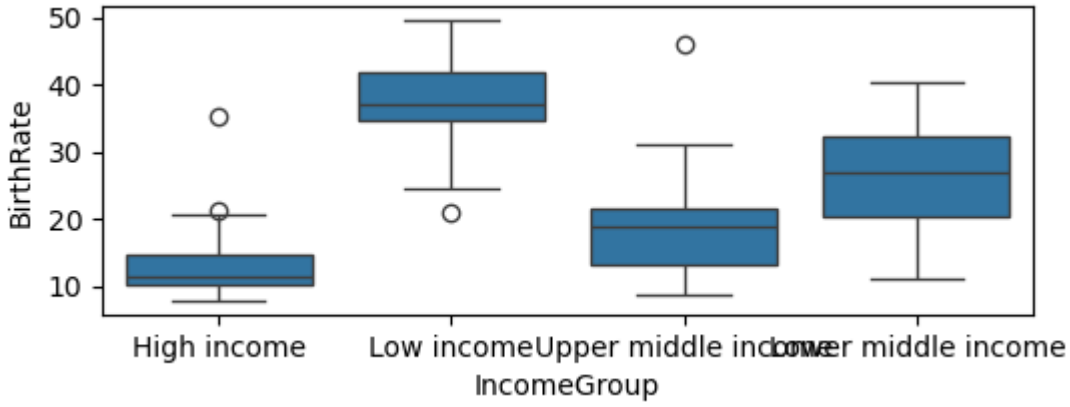
In [171…
```
vis2 = sns.displot(df["InternetUsers"])
```



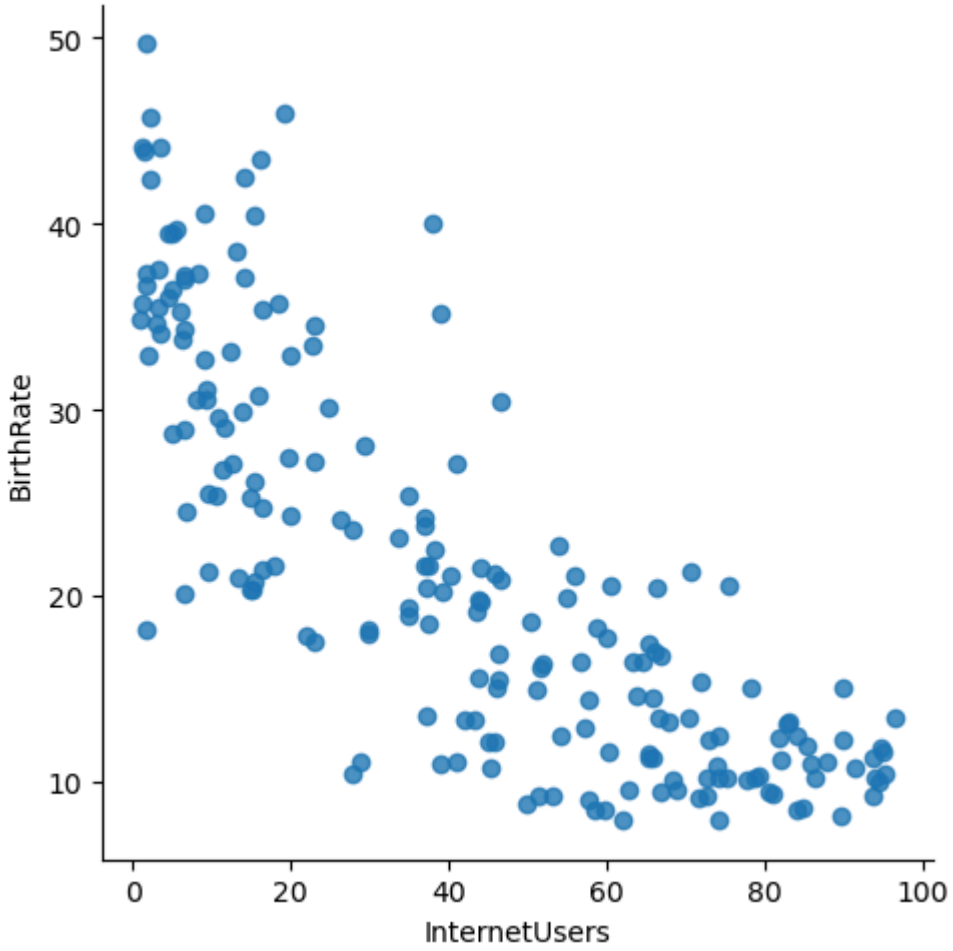
In [173…
```
vis3 = sns.distplot(df["InternetUsers"], bins=10)
```



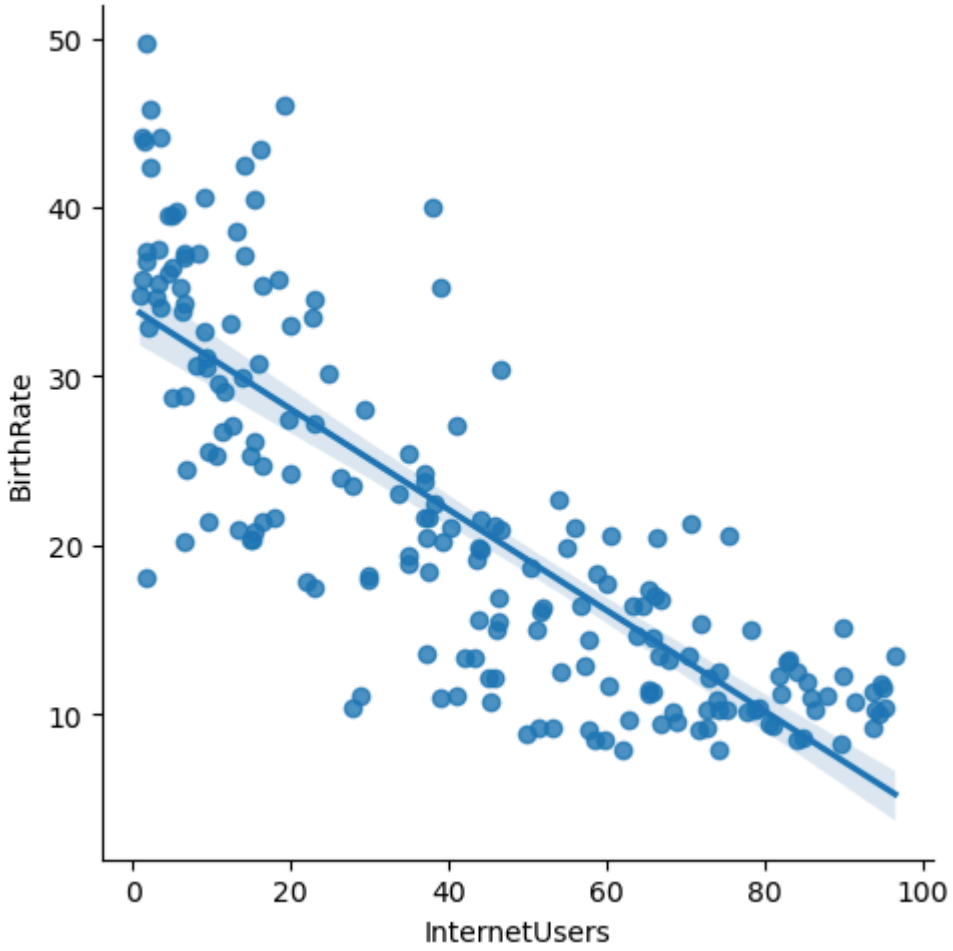
In [175…
```
vis4 = sns.boxplot(data = df, x="IncomeGroup", y='BirthRate')
```



In [177…
```
vis5 = sns.lmplot(data = df,x = 'InternetUsers', y = 'BirthRate', fit_reg = Fals
```

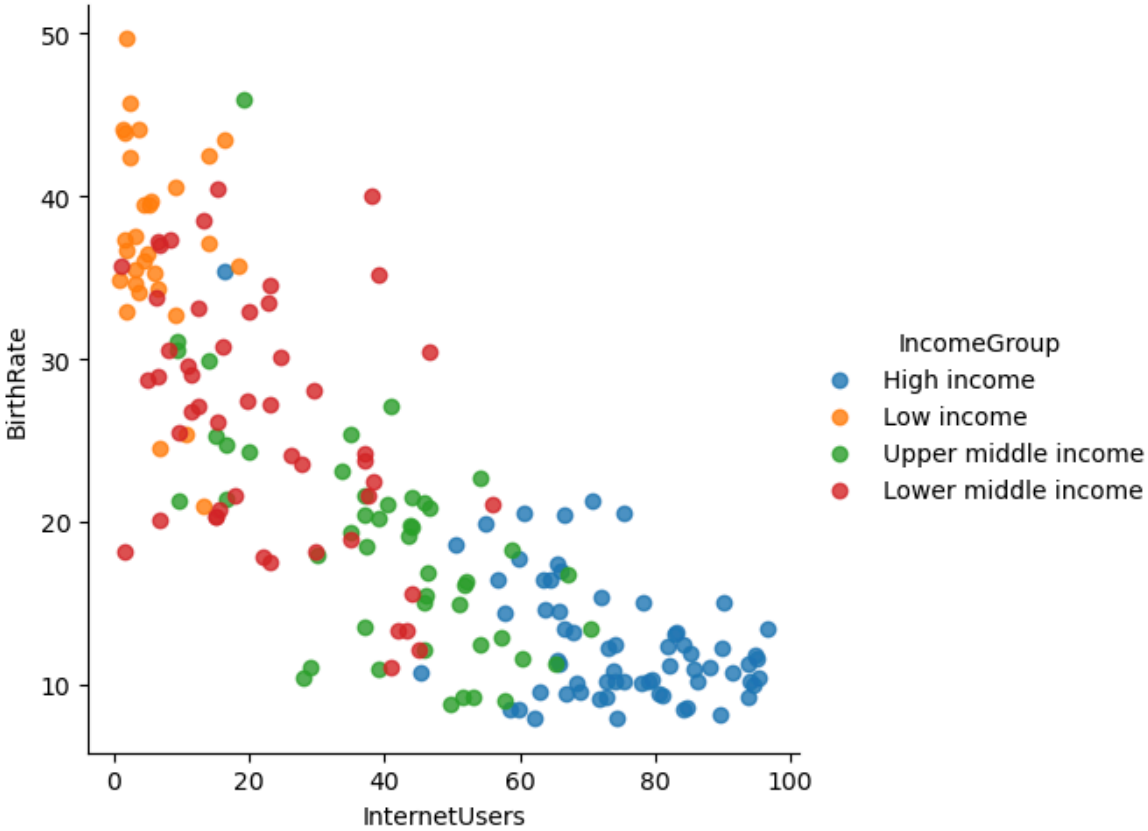


In [179…
```
vis6 = sns.lmplot(data = df,x = 'InternetUsers', y = 'BirthRate')
```



In [181…

```
vis7 = sns.lmplot(data = df,x="InternetUsers", y = "BirthRate",
                  fit_reg = False,hue = 'IncomeGroup')
```



In [ ]:

In [ ]:

In [ ]: