

06th May 2025

- Numpy for Matrix
- Matplotlib for Visualization
- Pandas for Data Frame, Data Structure+math+linear algebra
- Data Analysis Libraries
- Numpy - Matrix
- Pandas - DataFrame
- Matplotlib - Visualization
- Seaborn - Advance Visualization
- Scipy - Statstical library

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

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

```
Out[2]: '2.1.2'
```

```
In [3]: stats = pd.read_csv(r'C:\Users\Windows10 Pro\Downloads\DataScience_AI\2025\May2025\stats

# 3 attributes are -> Categorical data
# 2 attributes are -> Numerical data
```

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(stats)`

Out[4]: 195

In [5]: `stats.shape`

Out[5]: (195, 5)

In [6]: `stats.columns`

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

In [7]: `type(stats)`Out[7]: `pandas.core.frame.DataFrame`In [8]: `len(stats.columns)`

Out[8]: 5

In [9]: `stats.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 [10]: stats.columns
```

```
Out[10]: Index(['CountryName', 'CountryCode', 'BirthRate', 'InternetUsers',
               'IncomeGroup'],
              dtype='object')
```

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

Out[11]:

	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]: stats.tail()
```

Out[12]:

	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]: stats.tail(4)
```

Out[13]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
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 [14]:

```
stats.tail(10)
```

Out[14]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
185	Virgin Islands (U.S.)	VIR	10.700	45.3	High income
186	Vietnam	VNM	15.537	43.9	Lower middle income
187	Vanuatu	VUT	26.739	11.3	Lower middle income
188	West Bank and Gaza	PSE	30.394	46.6	Lower middle income
189	Samoa	WSM	26.172	15.3	Lower middle 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

In [15]:

```
stats.head(10)
```

Out[15]:

	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 [16]: stats

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
...
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

Slicing in Pandas

In [17]: stats[::-1]

Out[17]:

	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 [18]: stats[:5]

Out[18]:

	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 [19]: stats[5:]

Out[19]:

	CountryName	CountryCode	BirthRate	InternetUsers	IncomeGroup
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
...
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

190 rows × 5 columns

In [20]:

stats[0:200:10]

Out[20]:

	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 [21]: `stats.describe()` *# descriptive statistics will display only Numerical values*

Out[21]:

	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 [22]: `stats.describe().transpose()` *# transpose convert column into rows*

Out[22]:

	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 [23]: `stats.head(2)`

Out[23]:

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

In [24]: `stats.columns`Out[24]: `Index(['CountryName', 'CountryCode', 'BirthRate', 'InternetUsers', 'IncomeGroup'], dtype='object')`In [25]: `stats.columns = ['a', 'b', 'c', 'd', 'e']` *# column name changed/renaming*In [26]: `stats.head()`

Out[26]:

	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 [27]:

```
stats.columns = ['CountryName', 'CountryCode', 'BirthRate', 'InternetUsers', 'IncomeGroup']
```

In [28]:

```
stats.head()
```

Out[28]:

	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 [29]:

```
stats[:]
```

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
...
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 [30]: stats[0:5]
```

```
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

Missing Values

```
In [31]: stats.isnull()
```

```
Out[31]:
```

	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 [32]: stats.isnull().sum()
```

```
Out[32]: CountryName      0
CountryCode      0
BirthRate        0
InternetUsers     0
IncomeGroup      0
dtype: int64
```

I want to split the data into categorical part & numerical part

In [33]: `stats.dtypes`

Out[33]:

CountryName	object
CountryCode	object
BirthRate	float64
InternetUsers	float64
IncomeGroup	object
dtype:	object

In [34]: `stats['CountryName']`

Out[34]:

0	Aruba
1	Afghanistan
2	Angola
3	Albania
4	United Arab Emirates
...	...
190	Yemen, Rep.
191	South Africa
192	Congo, Dem. Rep.
193	Zambia
194	Zimbabwe

Name: CountryName, Length: 195, dtype: object

In [35]: `stats[['CountryName', 'CountryCode']]`

Out[35]:

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

195 rows × 2 columns

```
In [36]: stats_categorical = stats[['CountryName', 'CountryCode', 'IncomeGroup']]
stats_categorical.head()
```

Out[36]:

	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 [37]: stats_categorical.describe()
```

Out[37]:

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

```
In [38]: stats
```

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
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

07th May 2025

```
In [41]: stats_num = stats[['BirthRate', 'InternetUsers']]
stats_num.head()
```

```
Out[41]:
```

	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 [42]: stats_num.describe().transpose()
```

```
Out[42]:
```

	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 [44]: stats_num.describe()
```

```
Out[44]:
```

	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 [45]: stats.head()
```

Out[45]:

	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]: `stats.BirthRate * stats.InternetUsers`

Out[46]:

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 [47]: `stats.columns`

Out[47]: Index(['CountryName', 'CountryCode', 'BirthRate', 'InternetUsers',
'IncomeGroup'],
dtype='object')

In [48]: `stats['myCalc'] = stats.BirthRate * stats.InternetUsers`In [49]: `stats` *# DML Data Manipulation --> adding myCalc column in dataset*

Out[49]:

	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]: `stats.columns`

Out[50]: Index(['CountryName', 'CountryCode', 'BirthRate', 'InternetUsers', 'IncomeGroup', 'myCalc'], dtype='object')

In [51]: `len(stats.columns)`

Out[51]: 6

In [52]: `stats.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 [55]:

```
stats = stats.drop('myCalc', axis=1, errors='ignore') # axis=0 means rows & axis=
```

In [56]:

```
stats
```

Out[56]:

	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 [57]:

```
stats.InternetUsers<2
```

```
Out[57]: 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 [58]: stats[stats.InternetUsers<2] # below poverty line internet users
```

```
Out[58]:
```

	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 [59]: stats.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]: stats[stats.BirthRate>40]
```

Out[60]:

	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 [61]:

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

Out[61]:

	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 [62]:

stats[stats.IncomeGroup == 'Low income']

Out[62]:

	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 [63]: # How to get unique categories
stats.IncomeGroup.unique()
```

```
Out[63]: array(['High income', 'Low income', 'Upper middle income',
               'Lower middle income'], dtype=object)
```

```
In [64]: stats.IncomeGroup.nunique()
```

```
Out[64]: 4
```

seaborn

```
In [65]: import matplotlib.pyplot as plt # visualization
import seaborn as sns # for advance visualizations -> distribution plot, line pl
```

```
In [67]: plt.rcParams['figure.figsize']=9,5
import warnings
warnings.filterwarnings('ignore') # OS Errors
```

```
In [68]: stats.head()
```

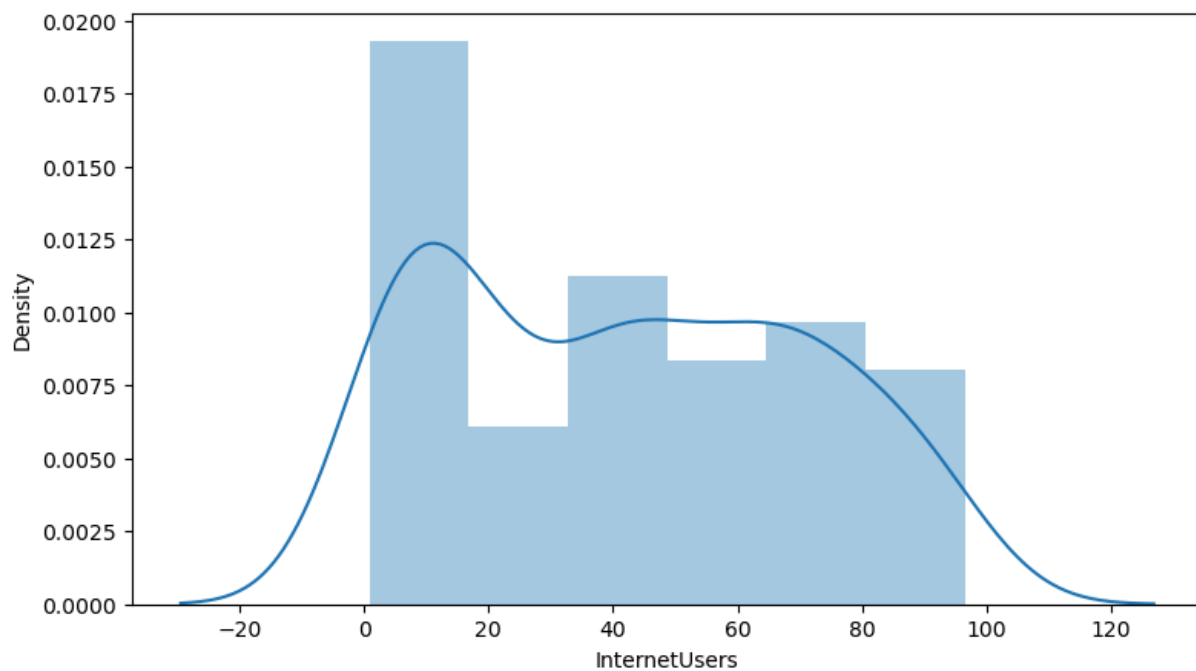
```
Out[68]:
```

	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

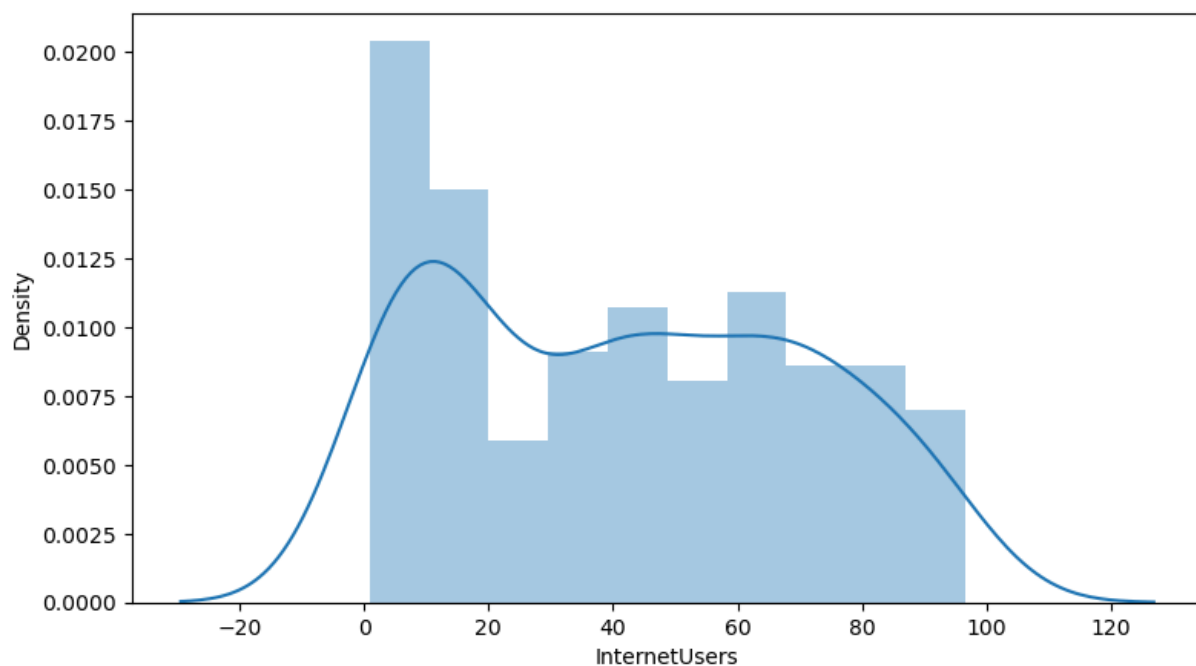
```
In [69]: stats['InternetUsers']
```

```
Out[69]: 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 [75]: # Distributions
plt.rcParams['figure.figsize']=9,5
vis1 = sns.distplot(stats["InternetUsers"]) # Univariate Analysis is Plot the gra
```



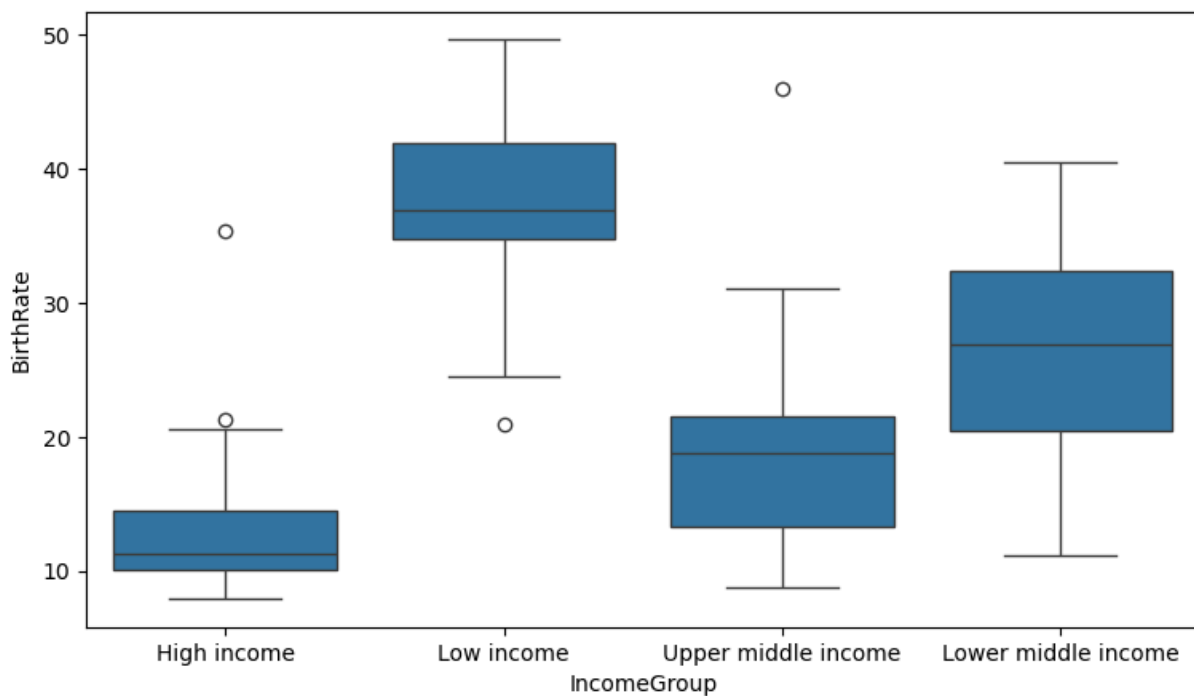
```
In [78]: vis1 = sns.distplot(stats["InternetUsers"], bins=10)
```



```
In [80]: # Box Plots

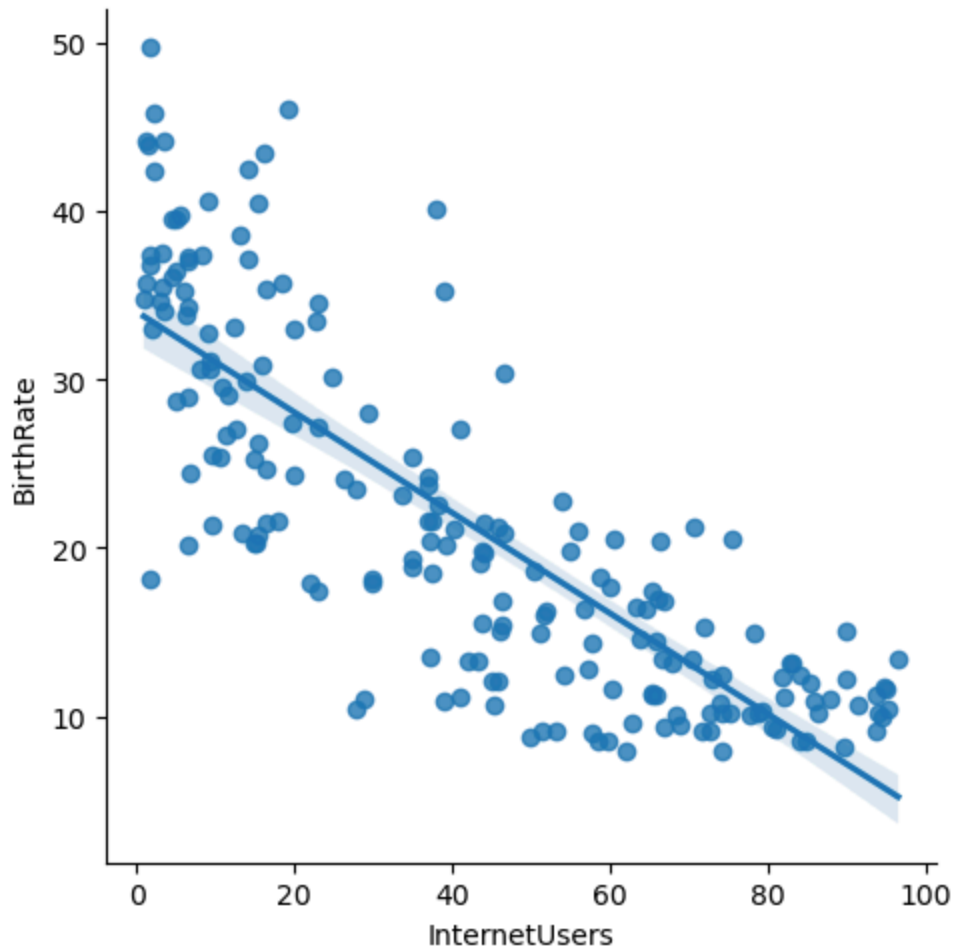
vis2 = sns.boxplot(data = stats, x='IncomeGroup', y='BirthRate') # Bi Varient Anal

# stright line is the Average
# stright line below is Lower Quartile
# stright line above is Upper Quartile
```

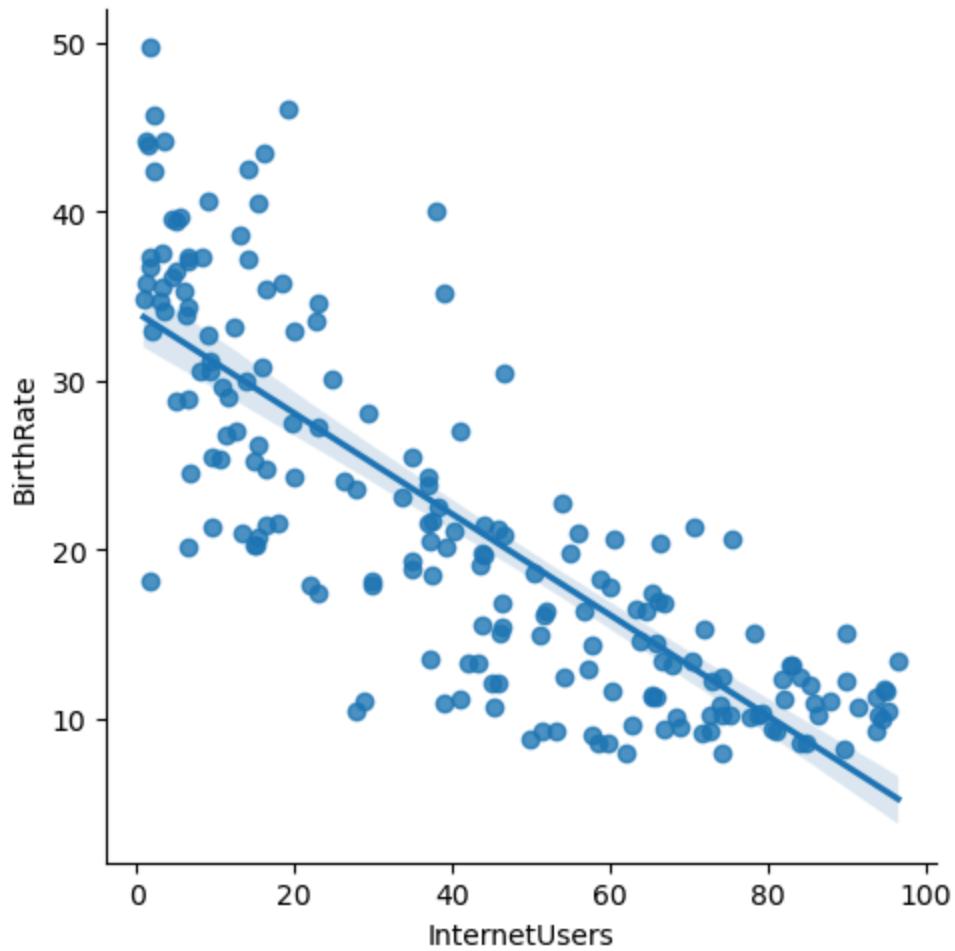


```
In [ ]: # Outliers - Outliers in stats datapoints which are far from other observation oth  
# Outliers also called as Anomaly detection  
# Outliers will impact on Machine Learning Algortihm  
# We adjust Outliers by implementing Probability function == Logistic regression al
```

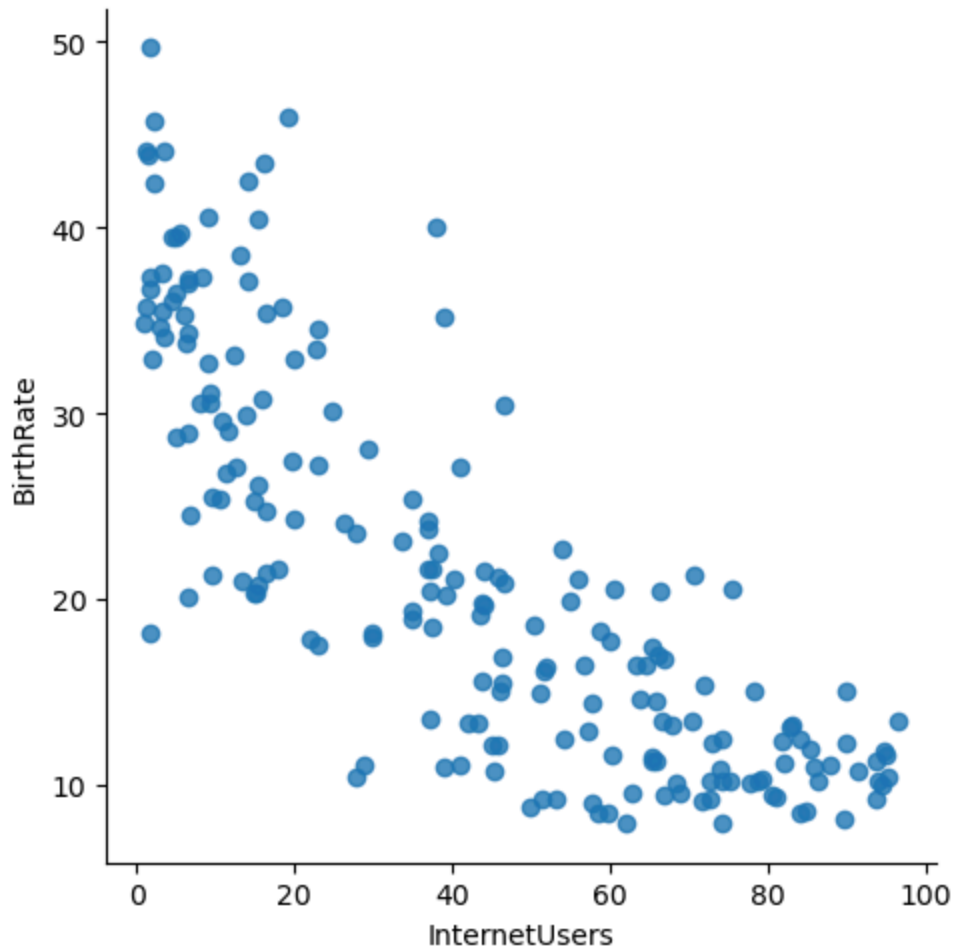
```
In [81]: vis4 = sns.lmplot(data=stats, x='InternetUsers', y='BirthRate')
```



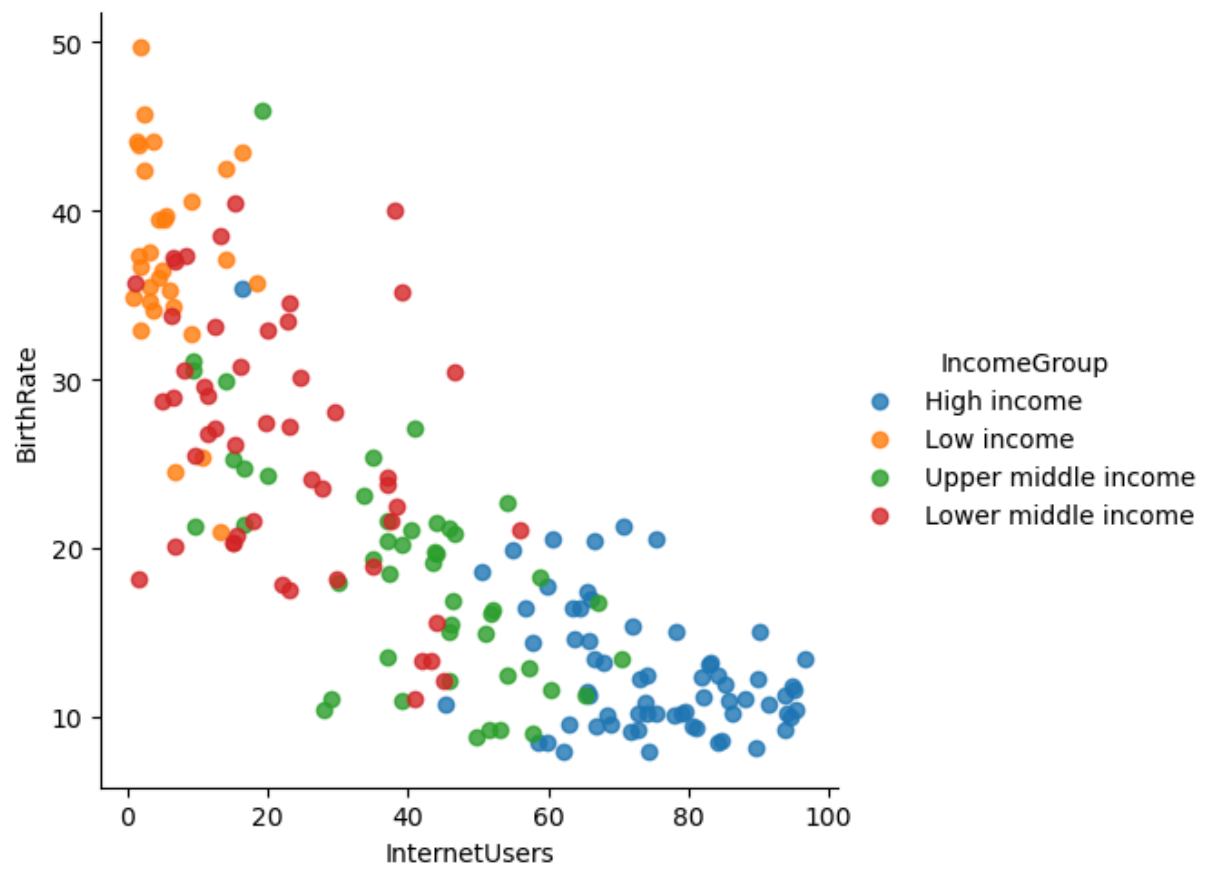
```
In [82]: vis3 = sns.lmplot(data=stats, x='InternetUsers', y='BirthRate', fit_reg=True) # Lm
```

```
In [83]: vis31 = sns.lmplot(data = stats, x = 'InternetUsers', y = 'BirthRate', fit_reg = Fal
```



```
In [84]: vis5 = sns.lmplot(data = stats, x = 'InternetUsers', y = 'BirthRate', fit_reg = False)
```



In []: