Applied Data Science and Machine Learning with Python

PHOSPHENE AI

Today's contents

- Introduction to Data Science
- Real world example
- Lightning tour of Python and Jupyter notebooks

Q&A every 30 minutes.

What you will not learn in this series of lectures

Let's dive in

- **Question of the day:**
  - **What is data science and what do we do?**

What is Data Science?

- Data science is the field of study that combines domain expertise, programming skills, and knowledge of mathematics and statistics to extract meaningful insights from data.
- It is an end to end process that involves acquisition of data to identification of patterns and insights.

Country	Population	GDP	Surface Area
Canada	35.467	1785387	9984670.0
France	63.951	2833687	640679.0
Germany	80.94	3874437	357114.0
Italy	60.665	2167744	301336.0
Japan	127.061	4602367	377930.0
United Kingdom	64.511	2950039	242495.0

Tabular Data

1	Twitter Own and Retweeted Tweets Table						
2							
3	pageName	createTime	message	link	retweets	favs	replies
4	@realDonaldTrump	09/02/2016 19	#AmericaFirst #ImWithYou I	https://twitter.com/r	4267	11671	150
5	@realDonaldTrump	09/02/2016 12	Great new poll Iowa - thank	https://twitter.com/r	6918	17422	271
6	@realDonaldTrump	09/02/2016 08	I visited our Trump Tower ca	https://twitter.com/r	5603	21556	1011
7	@realDonaldTrump	09/02/2016 08	People will be very surprise	https://twitter.com/r	7172	21719	420
8	@realDonaldTrump	09/02/2016 08	Just heard that crazy and ve	https://twitter.com/r	4877	16742	983
9	@realDonaldTrump	09/01/2016 18	I will be interviewed by @e	https://twitter.com/r	3611	13986	349
10	@realDonaldTrump	09/01/2016 13	I am promising you a new le	https://twitter.com/r	8680	26267	260
11	@realDonaldTrump	09/01/2016 10	Thank you for having me thi	https://twitter.com/r	5780	18576	156
12	@realDonaldTrump	09/01/2016 06	Poll numbers way up - maki	https://twitter.com/r	9351	35537	446
13	@realDonaldTrump	09/01/2016 06	Thank you to @foxandfrien	https://twitter.com/r	6641	26977	220
14	@realDonaldTrump	09/01/2016 06	Mexico will pay for the wall	https://twitter.com/r	27781	53159	1793
15	@realDonaldTrump	09/01/2016 01	Under a Trump administratio	https://twitter.com/r	7991	21867	377
16	@realDonaldTrump	09/01/2016 01	Hillary Clinton doesn't have	https://twitter.com/r	6338	18987	292

Text Data

Image Data

Kinds of Data to Explore

The Data Science Process

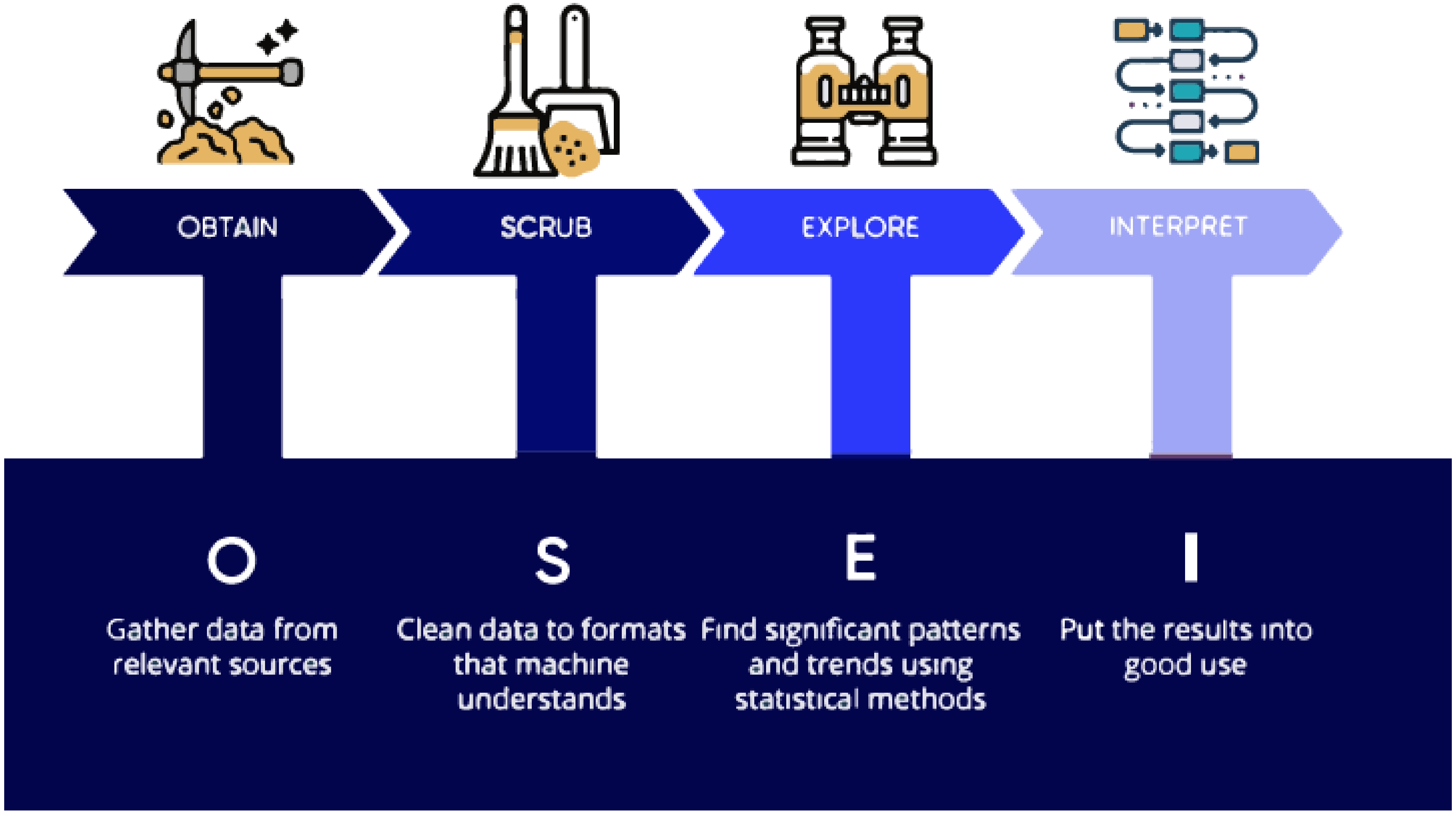


Image Source: Hillary Mason and Chris Wigins

Two ways to get insights out of data

Exploratory Data Analysis

Machine Learning / Pattern Recognition

Data Science vs Data Mining

- Data Mining is one of the many subprocesses in Data Science.
- In Data Mining, we just try to infer patterns out of data, while Data Science involves a plethora of processes like Data Collection, Cleaning, Wrangling, Visualization, etc.

Exploratory Data Analysis (EDA)

- Involves visualization of data into charts and using descriptive statistics such as mean, quantiles, quartiles, correlation, etc. to understand the data.
- We'll deal with those terms later, but those are the terms it is all about

Data Science Tools

Auto Managed Closed Tools

Programming Languages

Auto Managed Closed Tools Vs Programming Languages

Auto Managed Closed Tools

- Closed Source
- Expensive
- Limited Tooling
- Easy to Learn

Programming Languages

- Open Source
- Free (Mostly)
- Extremely Powerful
- Steep Learning Curve

Why Python

- Python will be our goto language in this series of lectures.
- It is:
  - Simple and intuitive
  - Powerful libraries
  - Amazing community
  - Free and Open source

Libraries we'll be using

- **Pandas** : Data manipulation and Pre-processing
- **Matplotlib/Seaborn** : Visualization Library
- **Altair** : Visualization library and helps with creating dashboards
- **NumPy**: A scientific computing tool

Let's dive into a Sample Project

Lightning tour of Python and Jupyter notebooks

Understanding the Python environment

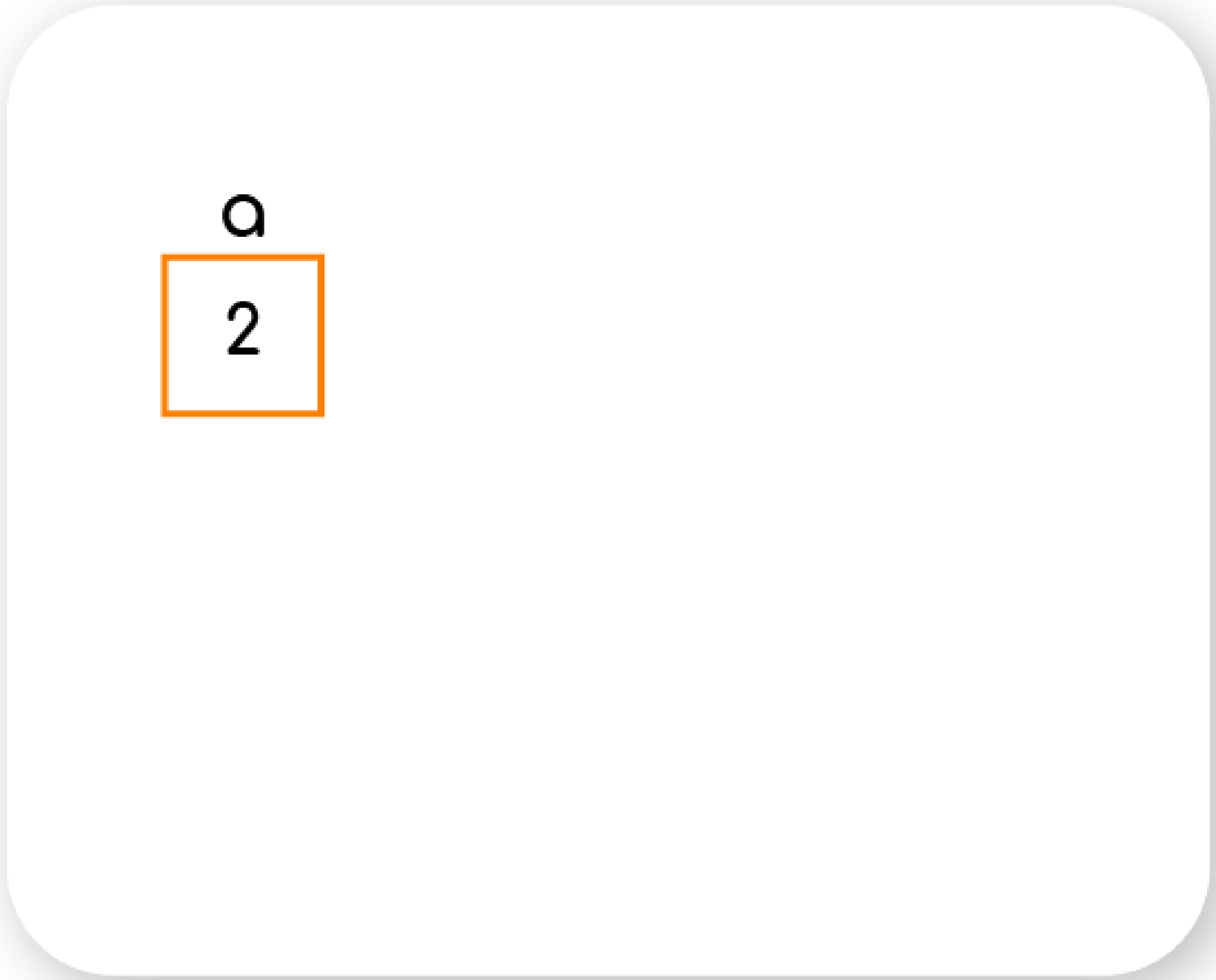
Environment

>>

```
>> print ( " Hello World " )
```

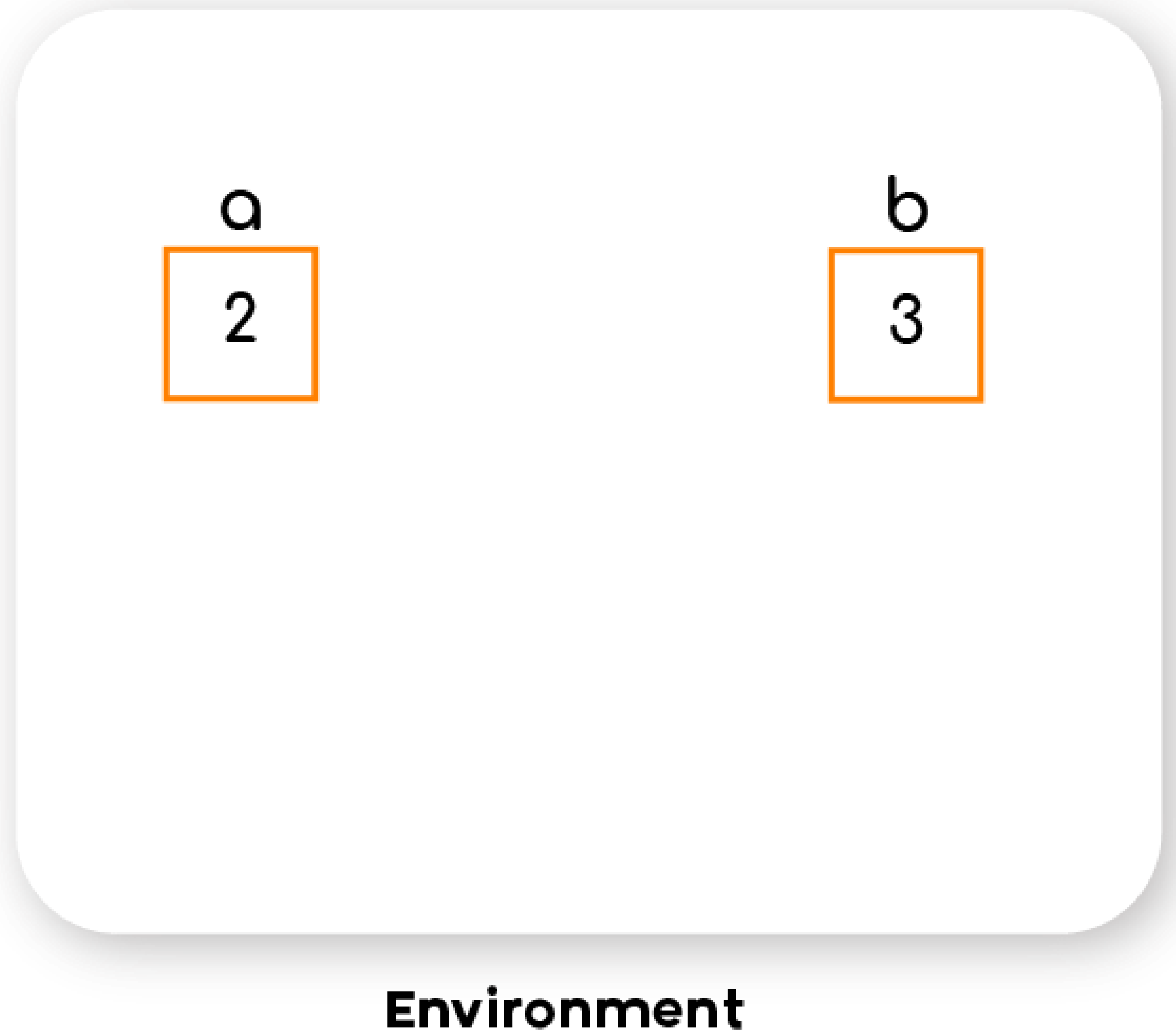
Output : Hello World

Environment

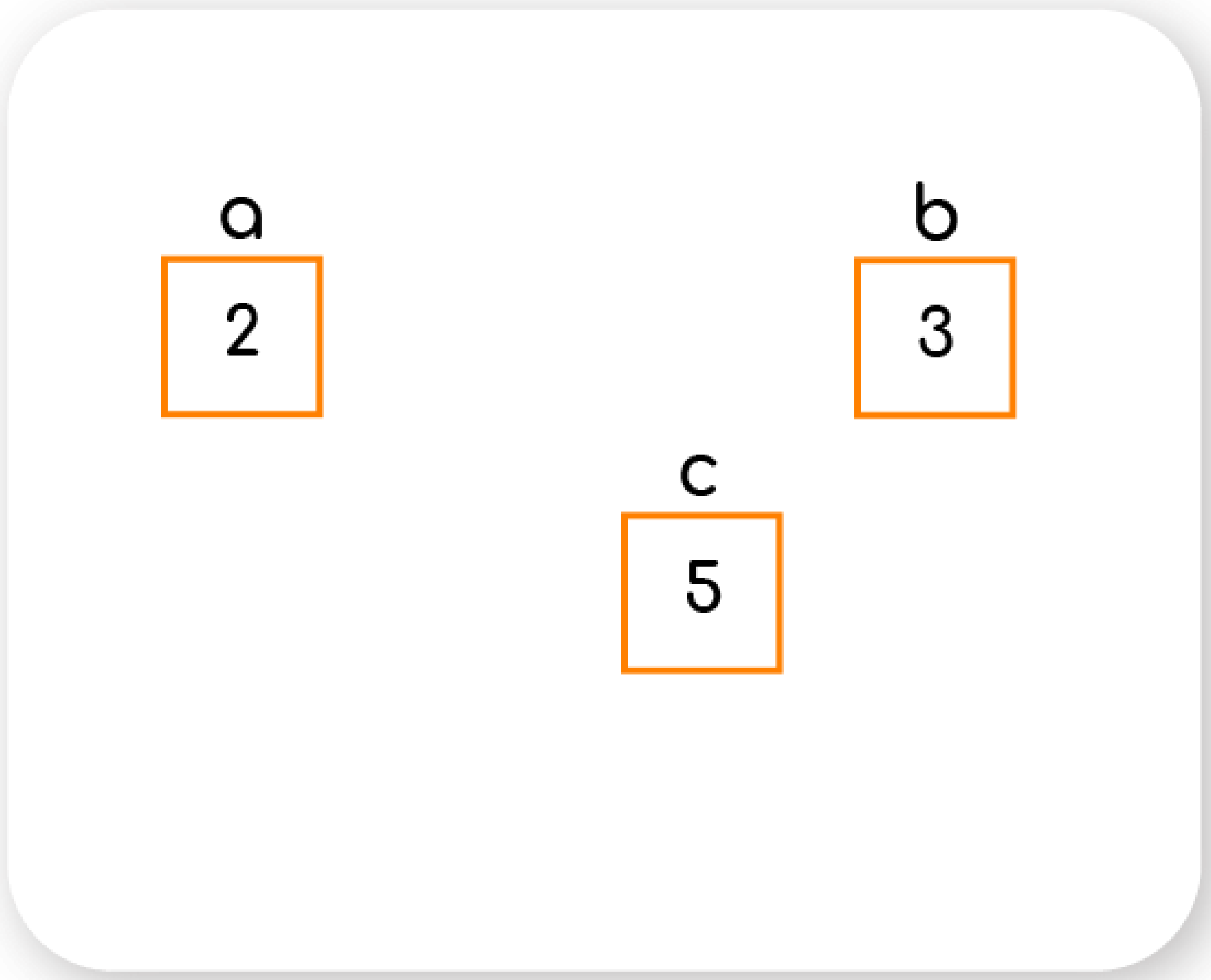


Environment

```
>> a = 2
```

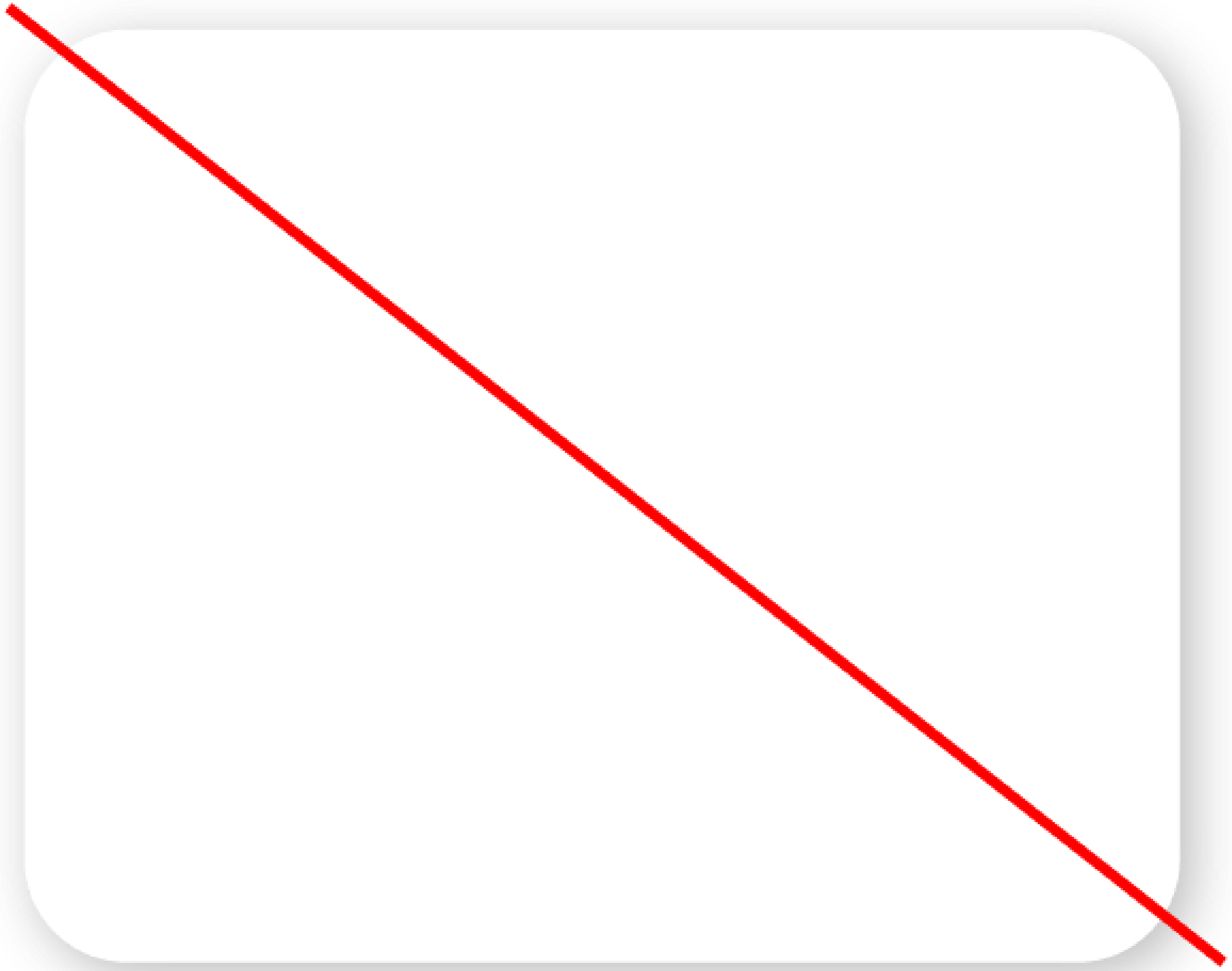


>> b = 3



Environment

>> c = b + a



Environment

>> quit ( )

Python Lists

0	1	2	3	4	5	6
0	7	8	9	10	14	20

a[0] = 0
a[2] = 8
a[2:5] = [8, 9, 10]
a[-1] = a[6] = 20

$$a = \begin{array}{cc} & \begin{array}{ccc} 0 & 1 & 2 \end{array} \\ \begin{array}{c} 0 \\ 1 \end{array} & \left[\begin{array}{ccc} [0 & 2 & 6] \\ [4 & 10 & 17] \end{array} \right] \end{array}$$

$$a[0] = \begin{array}{c} \begin{array}{ccc} 0 & 1 & 2 \end{array} \\ \left[\begin{array}{ccc} 0 & 2 & 6 \end{array} \right] \end{array}$$

$$a[0][2] = 6$$

Pandas Dataframes

Pandas
Selecting Rows

	Country	Population	GDP	Surface Area	HDI	Continent
0	Canada	35.467	1785387	9984670.0	0.913	America
1	France	63.951	2833687	640679.0	0.888	Europe
2	Germany	80.940	3874437	357114.0	0.916	Europe
3	Italy	60.665	2167744	301336.0	0.873	Europe
4	Japan	127.061	4602367	377930.0	0.891	Asia
5	United Kingdom	64.511	2950039	242495.0	0.907	Europe
6	United States	318.523	17348075	9525067.0	0.915	America
7	Western Sahara	NaN	908900	NaN	NaN	Africa
8	North Korea	NaN	32000000	120538.0	NaN	Asia

data.loc [n : k]

Starting index

Ending Index

Pandas
Selecting Rows

	Country	Population	GDP	Surface Area	HDI	Continent
0	Canada	35.467	1785387	9984670.0	0.913	America
1	France	63.951	2833687	640679.0	0.888	Europe
2	Germany	80.940	3874437	357114.0	0.916	Europe
3	Italy	60.665	2167744	301336.0	0.873	Europe
4	Japan	127.061	4602367	377930.0	0.891	Asia
5	United Kingdom	64.511	2950039	242495.0	0.907	Europe
6	United States	318.523	17348075	9525067.0	0.915	America
7	Western Sahara	NaN	908900	NaN	NaN	Africa
8	North Korea	NaN	32000000	120538.0	NaN	Asia

data.loc [2 : 5] =

	Country	Population	GDP	Surface Area	HDI	Continent
2	Germany	80.940	3874437	357114.0	0.916	Europe
3	Italy	60.665	2167744	301336.0	0.873	Europe
4	Japan	127.061	4602367	377930.0	0.891	Asia
5	United Kingdom	64.511	2950039	242495.0	0.907	Europe

Pandas
Selecting Rows

	Country	Population	GDP	Surface Area	HDI	Continent
0	Canada	35.467	1785387	9984670.0	0.913	America
1	France	63.951	2833687	640679.0	0.888	Europe
2	Germany	80.940	3874437	357114.0	0.916	Europe
3	Italy	60.665	2167744	301336.0	0.873	Europe
4	Japan	127.061	4602367	377930.0	0.891	Asia
5	United Kingdom	64.511	2950039	242495.0	0.907	Europe
6	United States	318.523	17348075	9525067.0	0.915	America
7	Western Sahara	NaN	908900	NaN	NaN	Africa
8	North Korea	NaN	32000000	120538.0	NaN	Asia

```
data[ 'column-name' ]
```

Pandas Selecting Rows

	Country	Population	GDP	Surface Area	HDI	Continent
0	Canada	35.467	1785387	9984670.0	0.913	America
1	France	63.951	2833687	640679.0	0.888	Europe
2	Germany	80.940	3874437	357114.0	0.916	Europe
3	Italy	60.665	2167744	301336.0	0.873	Europe
4	Japan	127.061	4602367	377930.0	0.891	Asia
5	United Kingdom	64.511	2950039	242495.0	0.907	Europe
6	United States	318.523	17348075	9525067.0	0.915	America
7	Western Sahara	NaN	908900	NaN	NaN	Africa
8	North Korea	NaN	32000000	120538.0	NaN	Asia

data['Country'] =

```
0            Canada
1            France
2           Germany
3             Italy
4             Japan
5    United Kingdom
6     United States
7    Western Sahara
8       North Korea
Name: Country, dtype: object
```

Pandas
Selecting Rows

	Country	Population	GDP	Surface Area	HDI	Continent
0	Canada	35.467	1785387	9984670.0	0.913	America
1	France	63.951	2833687	640679.0	0.888	Europe
2	Germany	80.940	3874437	357114.0	0.916	Europe
3	Italy	60.665	2167744	301336.0	0.873	Europe
4	Japan	127.061	4602367	377930.0	0.891	Asia
5	United Kingdom	64.511	2950039	242495.0	0.907	Europe
6	United States	318.523	17348075	9525067.0	0.915	America
7	Western Sahara	NaN	908900	NaN	NaN	Africa
8	North Korea	NaN	32000000	120538.0	NaN	Asia

data[['Country', 'Population']] =

	Country	Population
0	Canada	35.467
1	France	63.951
2	Germany	80.940
3	Italy	60.665
4	Japan	127.061
5	United Kingdom	64.511
6	United States	318.523
7	Western Sahara	NaN
8	North Korea	NaN

Pandas indexing

Selecting rows and columns at the same time

	Country	Population	GDP	Surface Area	HDI	Continent
0	Canada	35.467	1785387	9984670.0	0.913	America
1	France	63.951	2833687	640679.0	0.888	Europe
2	Germany	80.940	3874437	357114.0	0.916	Europe
3	Italy	60.665	2167744	301336.0	0.873	Europe
4	Japan	127.061	4602367	377930.0	0.891	Asia
5	United Kingdom	64.511	2950039	242495.0	0.907	Europe
6	United States	318.523	17348075	9525067.0	0.915	America
7	Western Sahara	NaN	908900	NaN	NaN	Africa
8	North Korea	NaN	32000000	120538.0	NaN	Asia

data.loc [n , k]

Row Index

Column Index

Pandas indexing

	Country	Population	GDP	Surface Area	HDI	Continent
0	Canada	35.467	1785387	9984670.0	0.913	America
1	France	63.951	2833687	640679.0	0.888	Europe
2	Germany	80.940	3874437	357114.0	0.916	Europe
3	Italy	60.665	2167744	301336.0	0.873	Europe
4	Japan	127.061	4602367	377930.0	0.891	Asia
5	United Kingdom	64.511	2950039	242495.0	0.907	Europe
6	United States	318.523	17348075	9525067.0	0.915	America
7	Western Sahara	NaN	908900	NaN	NaN	Africa
8	North Korea	NaN	32000000	120538.0	NaN	Asia

data.loc[1:4, ['Country', 'GDP]] =

	Country	GDP
1	France	2833687
2	Germany	3874437
3	Italy	2167744
4	Japan	4602367

Deleting columns and rows in Pandas

Axis = 1

Axis = 0 (default)

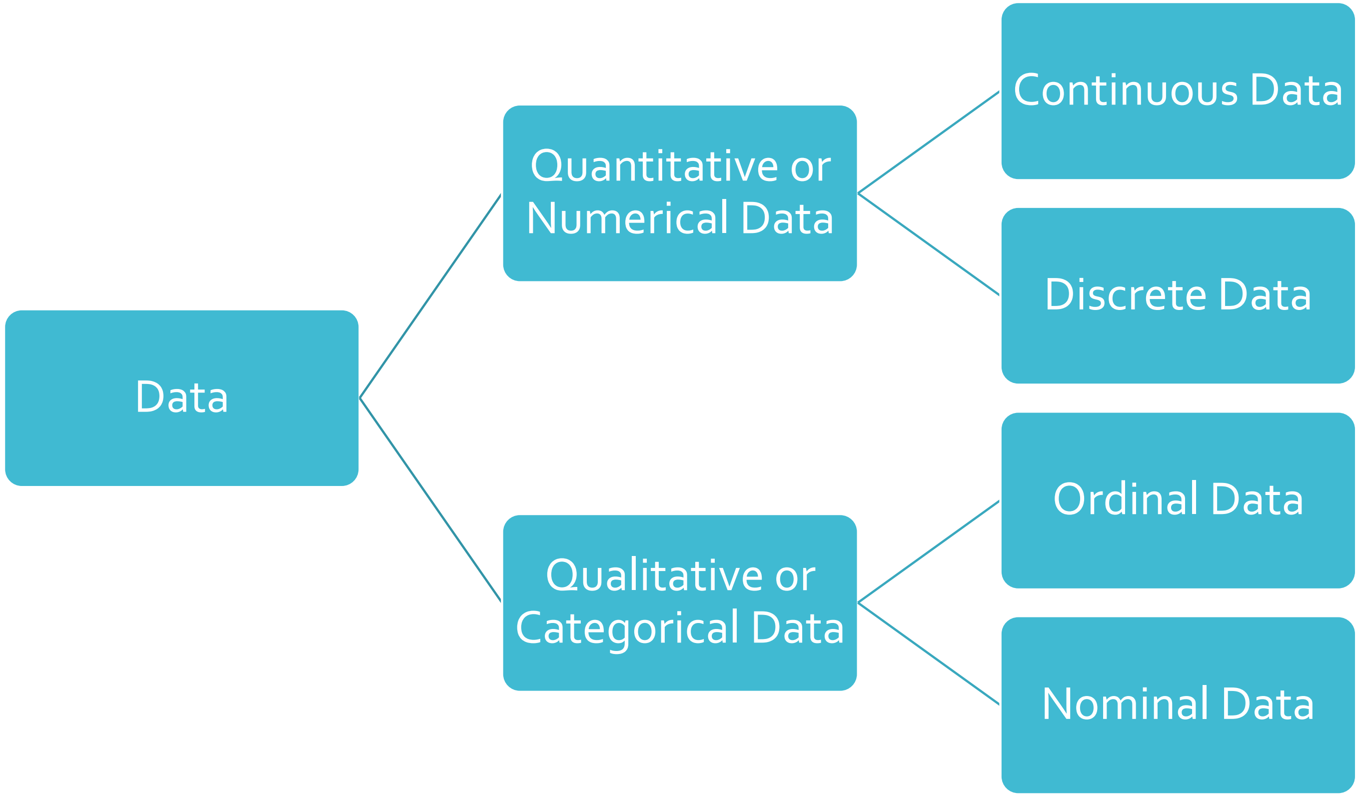
	Country	Population	GDP	Surface Area	HDI	Continent
0	Canada	35.467	1785387	9984670.0	0.913	America
1	France	63.951	2833687	640679.0	0.888	Europe
2	Germany	80.940	3874437	357114.0	0.916	Europe
3	Italy	60.665	2167744	301336.0	0.873	Europe
4	Japan	127.061	4602367	377930.0	0.891	Asia
5	United Kingdom	64.511	2950039	242495.0	0.907	Europe
6	United States	318.523	17348075	9525067.0	0.915	America
7	Western Sahara	NaN	908900	NaN	NaN	Africa
8	North Korea	NaN	32000000	120538.0	NaN	Asia

To remove rows – data.drop([list of row numbers])
Eg. data.drop([2, 4, 5])

To remove columns – data.drop([list of columns], axis=1)
Eg. data.drop(['Population', 'GDP'], axis = 1)

Basic Descriptive Statistics

Types of Data



Quantitative Data

- Continuous Data represents measurements and therefore their values **can’t be counted but they can be measured**.

Eg. Height of someone. You can't count it, but you can measure it using a scale. Another eg is satisfaction level of someone in a company. You can have a range like to zero to 1, but you can't count it, but you can measure it using the opinion of the individuals.

- Discrete Data represents values that can be counted.

Eg, Number of people in a room or number of employees in a company.

Nominal Data

What is your Gender?

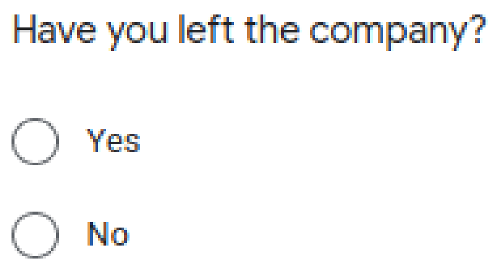
- Male
- Female
- Prefer not to say

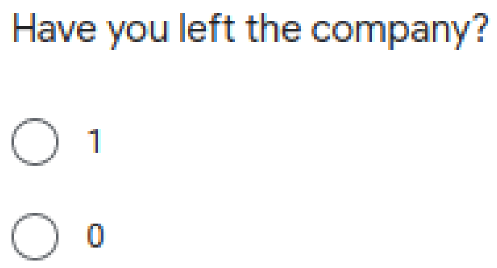
Which of the below languages can you speak?

- English
- French
- Spanish
- Latin

Nominal Data

Value can be a number, but the data could still fall into the Nominal Data category





Here, the option with value '1' means that the person has left the company, while the option '0' means the person hasn't left the company. Even though numerically 1 is greater than 0, here, 1 means the person has left the company and 0 means the person has stayed. Since, we've assigned a labelled meaning, even though the values are numerical in nature, they are still to be assumed as labels.

Ordinal Data

In which age category do you fall in?

- Child
- Teenager
- Youth
- Middle Aged
- Old

What is the size of the shirt you are wearing?

- S
- M
- L
- XL
- XXL

Mean

Marks of students:

A - 70
B - 79
C - 91
D - 85
E - 34
F - 62
G - 56
H - 84
I - 92
J - 78
K - 45
L - 68
M - 71
N - 52
O – 31
P - 62

Mean = sum of all values/number of values

Mean = 1060/16 = 66.25

Median

Marks of students:

A - 70
B - 79
C - 91
D - 85
E - 34
F - 62
G - 56
H - 84
I - 92
J - 78
K - 45
L - 68
M - 71
N - 52
O – 31
P - 62

Sorted Order:

31, 45, 34, 52, 56, 62, 62, 68, 70, 71, 78, 79, 84, 85, 91, 92

Median

n is odd,

Median = $\left(\frac{n+1}{2}\right)^{th}$ observation

n is even,

Median = $\dfrac{\left(\frac{n}{2}\right)^{th} + \left(\frac{n}{2}+1\right)^{th} \text{ observation}}{2}$

Median

Marks of students:

A - 70
B - 79
C - 91
D - 85
E - 34
F - 62
G - 56
H - 84
I - 92
J - 78
K - 45
L - 68
M - 71
N - 52
O – 31
P - 62

Sorted Order:

31, 45, 34, 52, 56, 62, 62, 68, 70, 71, 78, 79, 84, 85, 91, 92

Median

n is odd,

Median = $\left(\frac{n+1}{2}\right)^{th}$ observation

n is even,

Median = $\dfrac{\left(\frac{n}{2}\right)^{th} + \left(\frac{n}{2}+1\right)^{th} \text{ observation}}{2}$

Median = 69

Quartiles

$$\textbf{Lower Quartile (Q1)} = (N+1) \times \frac{1}{4}$$

$$\textbf{Middle Quartile (Q2)} = (N+1) \times \frac{2}{4}$$

$$\textbf{Upper Quartile (Q3)} = (N+1) \times \frac{3}{4}$$

Quartiles

Lower Quartile (Q1) $= (N+1) \times \frac{1}{4}$

Middle Quartile (Q2) $= (N+1) \times \frac{2}{4}$

Upper Quartile (Q3) $= (N+1) \times \frac{3}{4}$

Sorted Order:

31, 45, 34, 52, 56, 62, 62, 68, 70, 71, 78, 79, 84, 85, 91, 92

Q1 = (52 + 56)/2 = 54

Q3 = (79 + 84)/2 = 81.5

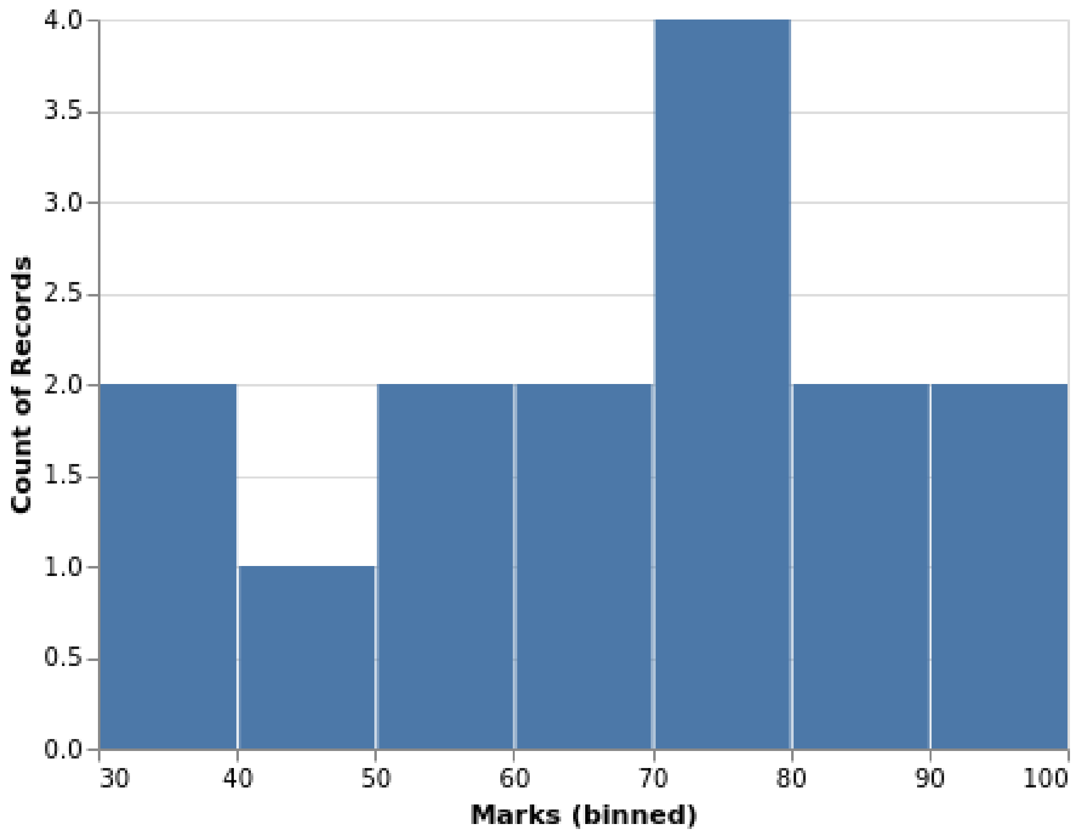
Q2 = Median

Visualizations

Scatterplots

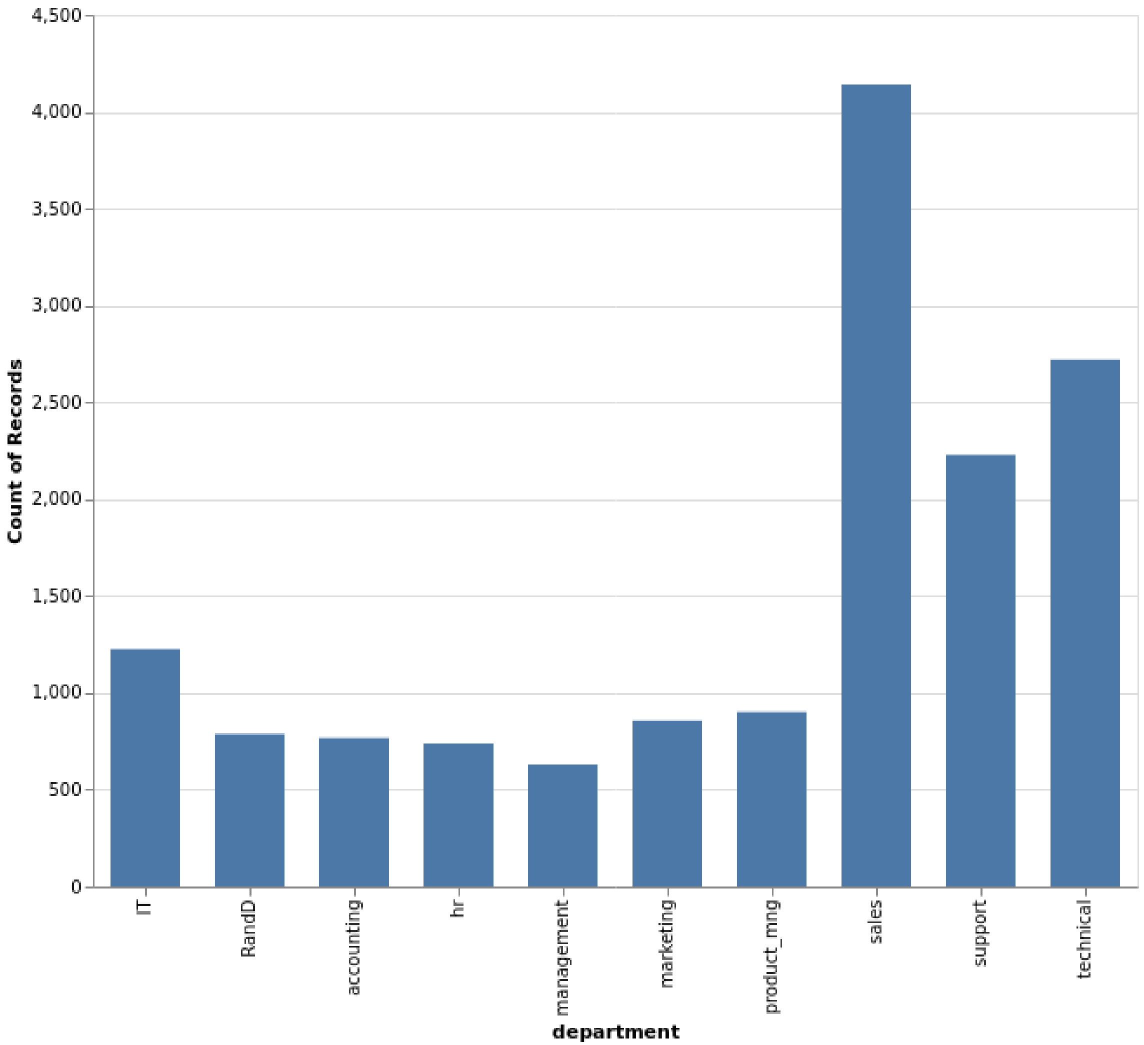
Valid only if both variables are quantitative in nature

Histograms – Single Variable



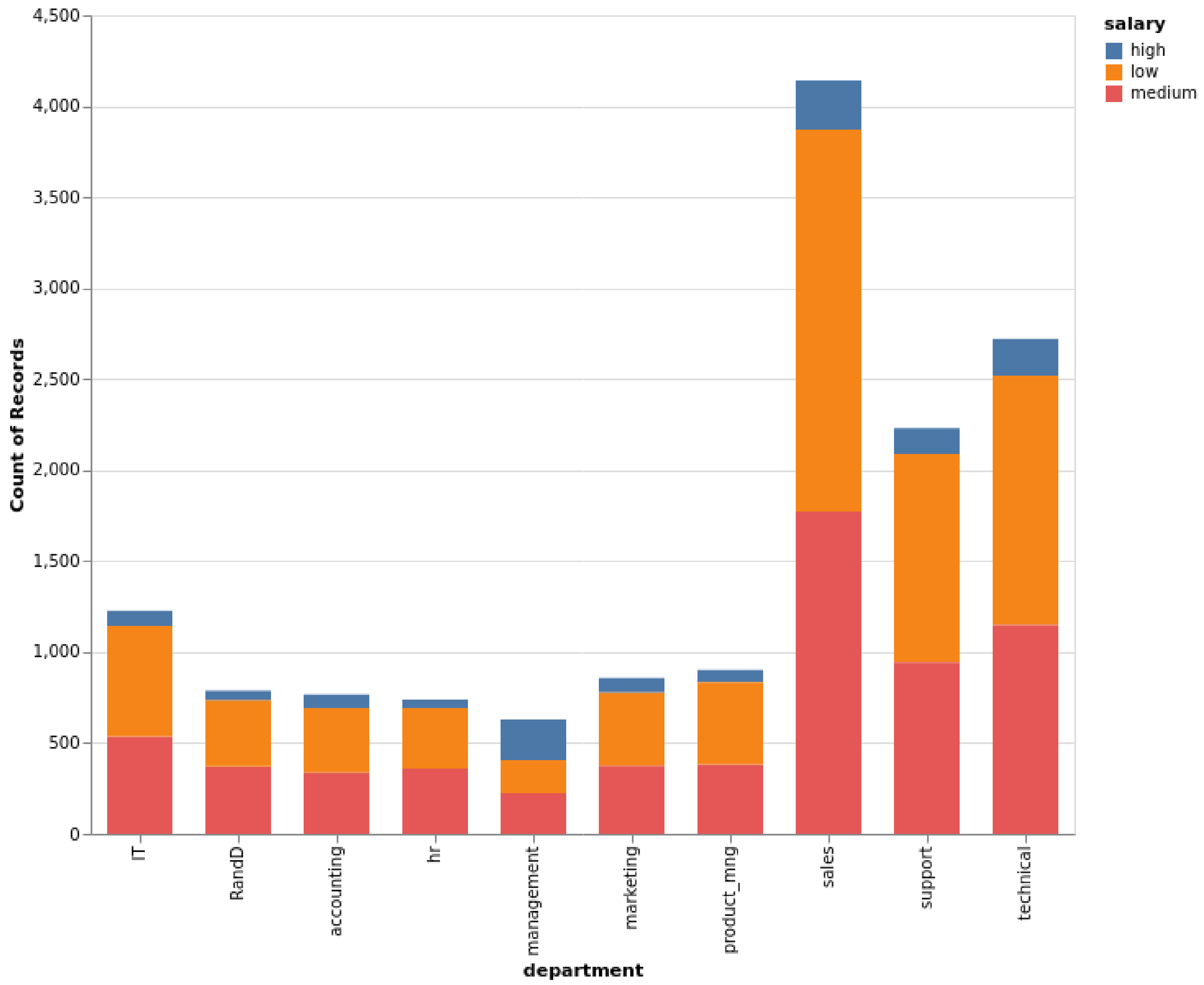
Valid only for Quantitative Variables

Bar Charts – For one variable



Valid only for Categorical/Qualitative Variables

Bar Charts -
Stacked Bar Charts



Boxplot

Five Number Summary For Data Set:

1,2,3,4,5,11,11,12,14,20,20

1,2.5,8,13,20

Minimum

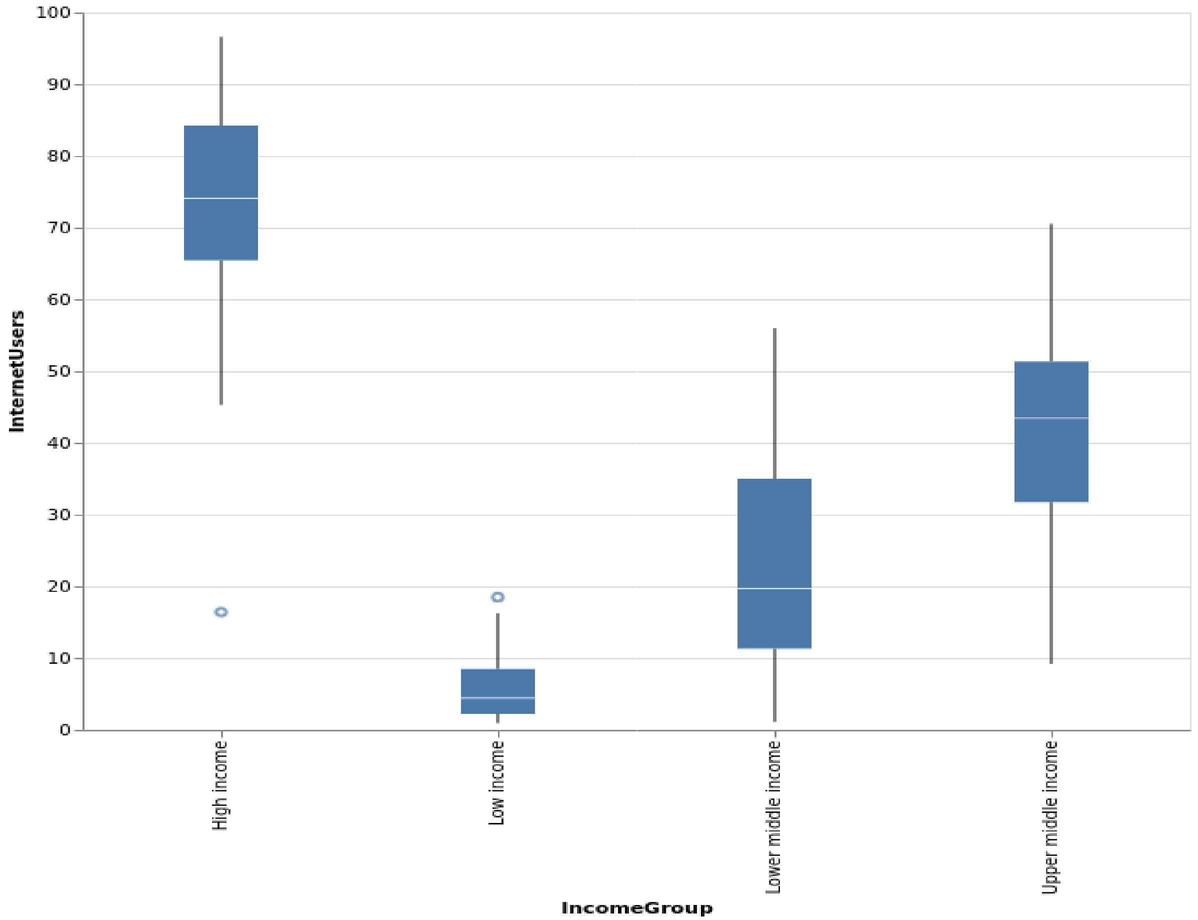
1st Quartile

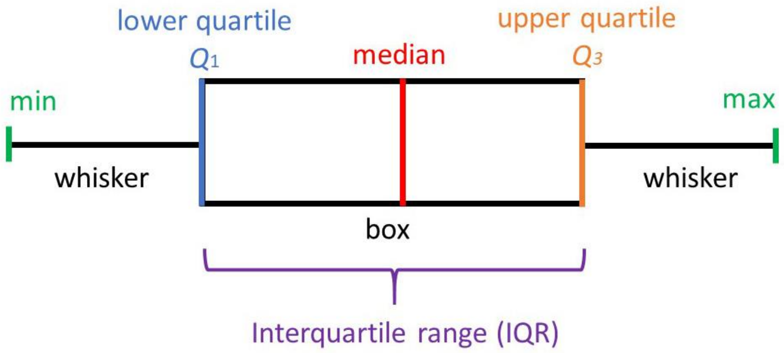
Median

3rd Quartile

Maximum

Boxplot


100
90
80
70
60
50
40
30
20
10
0
InternetUsers
High income
Low income
Lower middle income
Upper middle income
IncomeGroup


lower quartile
Q1
median
upper quartile
Q3
min
max
whisker
whisker
box
Interquartile range (IQR)

	CountryName	CountryCode	Region	BirthRate	InternetUsers	FertilityRate	IncomeGroup
0	Aruba	ABW	The Americas	10.244	78.9	1.669	High income
1	Afghanistan	AFG	Asia	35.253	5.9	5.050	Low income
2	Angola	AGO	Africa	45.985	19.1	6.165	Upper middle income
3	Albania	ALB	Europe	12.877	57.2	1.771	Upper middle income
4	United Arab Emirates	ARE	Middle East	11.044	88.0	1.801	High income

	CountryName	CountryCode	Region	BirthRate	InternetUsers	FertilityRate	IncomeGroup
0	Aruba	ABW	The Americas	10.244	78.9	1.669	High income
4	United Arab Emirates	ARE	Middle East	11.044	88.0	1.801	High income
5	Argentina	ARG	The Americas	17.716	59.9	2.335	High income
7	Antigua and Barbuda	ATG	The Americas	16.447	63.4	2.088	High income
8	Australia	AUS	Oceania	13.200	83.0	1.921	High income

	CountryName	CountryCode	Region	BirthRate	InternetUsers	FertilityRate	IncomeGroup
2	Angola	AGO	Africa	45.985	19.1000	6.165	Upper middle income
3	Albania	ALB	Europe	12.877	57.2000	1.771	Upper middle income
10	Azerbaijan	AZE	Asia	18.300	58.7000	2.000	Upper middle income
16	Bulgaria	BGR	Europe	9.200	53.0615	1.500	Upper middle income
19	Bosnia and Herzegovina	BIH	Europe	9.062	57.7900	1.272	Upper middle income

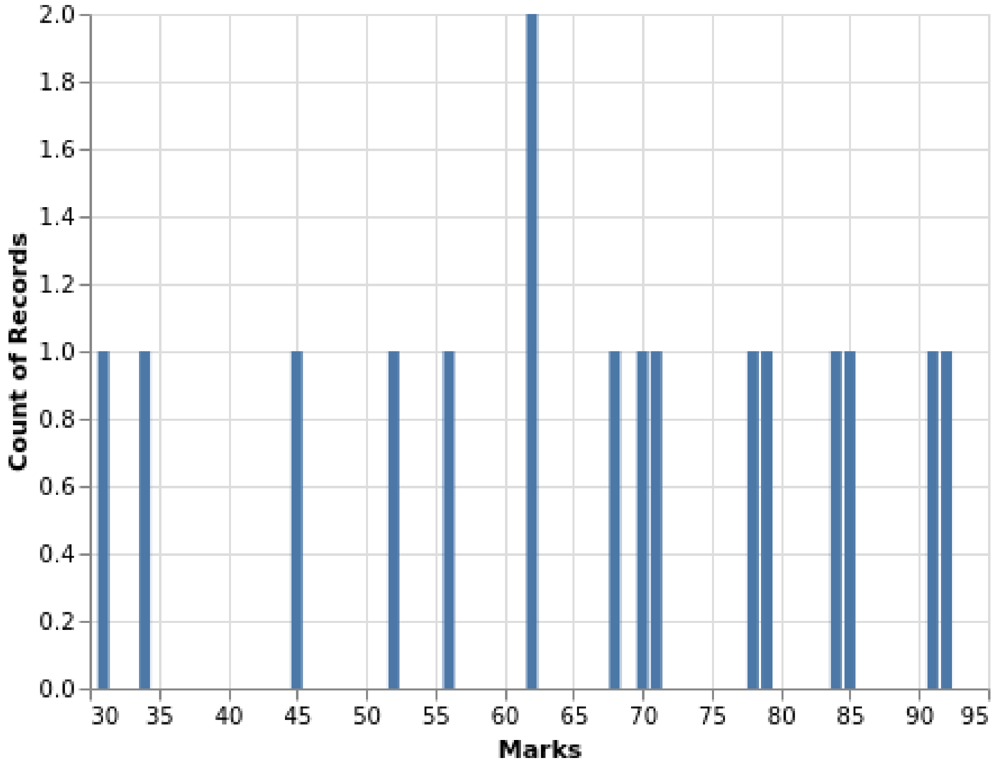
	CountryName	CountryCode	Region	BirthRate	InternetUsers	FertilityRate	IncomeGroup
1	Afghanistan	AFG	Asia	35.253	5.9	5.050	Low income
11	Burundi	BDI	Africa	44.151	1.3	6.035	Low income
13	Benin	BEN	Africa	36.440	4.9	4.846	Low income
14	Burkina Faso	BFA	Africa	40.551	9.1	5.607	Low income
28	Central African Republic	CAF	Africa	34.076	3.5	4.368	Low income

	CountryName	CountryCode	Region	BirthRate	InternetUsers	FertilityRate	IncomeGroup
6	Armenia	ARM	Asia	13.308	41.90	1.553	Lower middle income
15	Bangladesh	BGD	Asia	20.142	6.63	2.209	Lower middle income
22	Bolivia	BOL	The Americas	24.236	36.94	3.017	Lower middle income
26	Bhutan	BTN	Asia	18.134	29.90	2.082	Lower middle income
33	Cote d'Ivoire	CIV	Africa	37.320	8.40	5.063	Lower middle income

Understanding Binning

Marks of students:

A - 70
B - 79
C - 91
D - 85
E - 34
F - 62
G - 56
H - 84
I - 92
J - 78
K - 45
L - 68
M - 71
N - 52
O – 31
P - 62



Bins

Understanding Binning

Marks of students:
A - 70
B - 79
C - 91
D - 85
E - 34
F - 62
G - 56
H - 84
I - 92
J - 78
K - 45
L - 68
M - 71
N - 52
O – 31
P - 62

Bins:

30-40 - { }

40-50 - {}

50-60 - {}

60-70 - {}

70-80 - {}

80-90 - {}

90-100 - {}

Understanding Binning

Marks of students:
A - 70
B - 79
C - 91
D - 85
E - 34
F - 62
G - 56
H - 84
I - 92
J - 78
K - 45
L - 68
M - 71
N - 52
O – 31
P - 62

Bins:

30-40 - { }

40-50 - {}

50-60 - {}

60-70 - {}

70-80 - {70}

80-90 - {}

90-100 - {}

Understanding Binning

Marks of students:

A - 70
B - 79
C - 91
D - 85
E - 34
F - 62
G - 56
H - 84
I - 92
J - 78
K - 45
L - 68
M - 71
N - 52
O – 31
P - 62

Bins:

30-40 - { }

40-50 - {}

50-60 - {}

60-70 - {}

70-80 - {70, 79}

80-90 - {}

90-100 - {}

Understanding Binning

Marks of students:

A - 70
B - 79
C - 91
D - 85
E - 34
F - 62
G - 56
H - 84
I - 92
J - 78
K - 45
L - 68
M - 71
N - 52
O – 31
P - 62

Bins:

30-40 - { }

40-50 - {}

50-60 - {}

60-70 - {}

70-80 - {70, 79}

80-90 - {}

90-100 - {91}

Understanding Binning

Marks of students:

A - 70
B - 79
C - 91
D - 85
E - 34
F - 62
G - 56
H - 84
I - 92
J - 78
K - 45
L - 68
M - 71
N - 52
O – 31
P - 62

Bins:

30-40 - { }

40-50 - {}

50-60 - {}

60-70 - {}

70-80 - {70, 79}

80-90 - {85 }

90-100 - {91}

Understanding Binning

Marks of students:

A - 70
B - 79
C - 91
D - 85
E - 34
F - 62
G - 56
H - 84
I - 92
J - 78
K - 45
L - 68
M - 71
N - 52
O – 31
P - 62

Bins:

30-40 - { 34 }

40-50 - {}

50-60 - {}

60-70 - {}

70-80 - {70, 79}

80-90 - {85, }

90-100 - {91}

Understanding Binning

Marks of students:

A - 70
B - 79
C - 91
D - 85
E - 34
F - 62
G - 56
H - 84
I - 92
J - 78
K - 45
L - 68
M - 71
N - 52
O – 31
P - 62

Bins:

30-40 - { 34 }

40-50 - {}

50-60 - {}

60-70 - {62}

70-80 - {70, 79}

80-90 - {85, }

90-100 - {91}

Understanding Binning

Marks of students:

A - 70
B - 79
C - 91
D - 85
E - 34
F - 62
G - 56
H - 84
I - 92
J - 78
K - 45
L - 68
M - 71
N - 52
O – 31
P - 62

Bins:

30-40 - { 34 }

40-50 - {}

50-60 - { 56}

60-70 - {62}

70-80 - {70, 79}

80-90 - {85, }

90-100 - {91}

Understanding Binning

Marks of students:

A - 70
B - 79
C - 91
D - 85
E - 34
F - 62
G - 56
H - 84
I - 92
J - 78
K - 45
L - 68
M - 71
N - 52
O – 31
P - 62

Bins:

30-40 - { 34 }

40-50 - {}

50-60 - { 56}

60-70 - {62}

70-80 - {70, 79}

80-90 - {85, 84}

90-100 - {91}

Understanding Binning

Marks of students:

A - 70
B - 79
C - 91
D - 85
E - 34
F - 62
G - 56
H - 84
I - 92
J - 78
K - 45
L - 68
M - 71
N - 52
O – 31
P - 62

Bins:

30-40 - { 34 }

40-50 - {}

50-60 - {56}

60-70 - {62}

70-80 - {70, 79}

80-90 - {85, 84}

90-100 - {91, 92}

Understanding Binning

Marks of students:

A - 70
B - 79
C - 91
D - 85
E - 34
F - 62
G - 56
H - 84
I - 92
J - 78
K - 45
L - 68
M - 71
N - 52
O – 31
P - 62

Bins:

30-40 - { 34 }

40-50 - {}

50-60 - {56}

60-70 - {62}

70-80 - {70, 79, 78}

80-90 - {85, 84}

90-100 - {91, 92}

Understanding Binning

Marks of students:

A - 70
B - 79
C - 91
D - 85
E - 34
F - 62
G - 56
H - 84
I - 92
J - 78
K - 45
L - 68
M - 71
N - 52
O – 31
P - 62

Bins:

30-40 - { 34 }

40-50 - {45}

50-60 - {56}

60-70 - {62}

70-80 - {70, 79, 78}

80-90 - {85, 84}

90-100 - {91, 92}

Understanding Binning

Marks of students:

A - 70
B - 79
C - 91
D - 85
E - 34
F - 62
G - 56
H - 84
I - 92
J - 78
K - 45
L - 68
M - 71
N - 52
O – 31
P - 62

Bins:

30-40 - { 34 }

40-50 - {45}

50-60 - {56}

60-70 - {62, 68}

70-80 - {70, 79, 78}

80-90 - {85, 84}

90-100 - {91, 92}

Understanding Binning

Marks of students:

A - 70
B - 79
C - 91
D - 85
E - 34
F - 62
G - 56
H - 84
I - 92
J - 78
K - 45
L - 68
M - 71
N - 52
O – 31
P - 62

Bins:

30-40 - { 34 }

40-50 - {45}

50-60 - {56}

60-70 - {62, 68}

70-80 - {70, 79, 78, 71}

80-90 - {85, 84}

90-100 - {91, 92}

Understanding Binning

Marks of students:

A - 70
B - 79
C - 91
D - 85
E - 34
F - 62
G - 56
H - 84
I - 92
J - 78
K - 45
L - 68
M - 71
N - 52
O – 31
P - 62

Bins:

30-40 - { 34 }

40-50 - {45}

50-60 - {56, 52}

60-70 - {62, 68}

70-80 - {70, 79, 78, 71}

80-90 - {85, 84}

90-100 - {91, 92}

Understanding Binning

Marks of students:

A - 70
B - 79
C - 91
D - 85
E - 34
F - 62
G - 56
H - 84
I - 92
J - 78
K - 45
L - 68
M - 71
N - 52
O – 31
P - 62

Bins:

30-40 - { 34, 31 }

40-50 - {45}

50-60 - {56, 52}

60-70 - {62, 68}

70-80 - {70, 79, 78, 71}

80-90 - {85, 84}

90-100 - {91, 92}

Understanding Binning

Marks of students:

A - 70
B - 79
C - 91
D - 85
E - 34
F - 62
G - 56
H - 84
I - 92
J - 78
K - 45
L - 68
M - 71
N - 52
O – 31
P - 62

Bins:

30-40 - { 34, 31 }

40-50 - {45}

50-60 - {56, 52}

60-70 - {62, 68, 62}

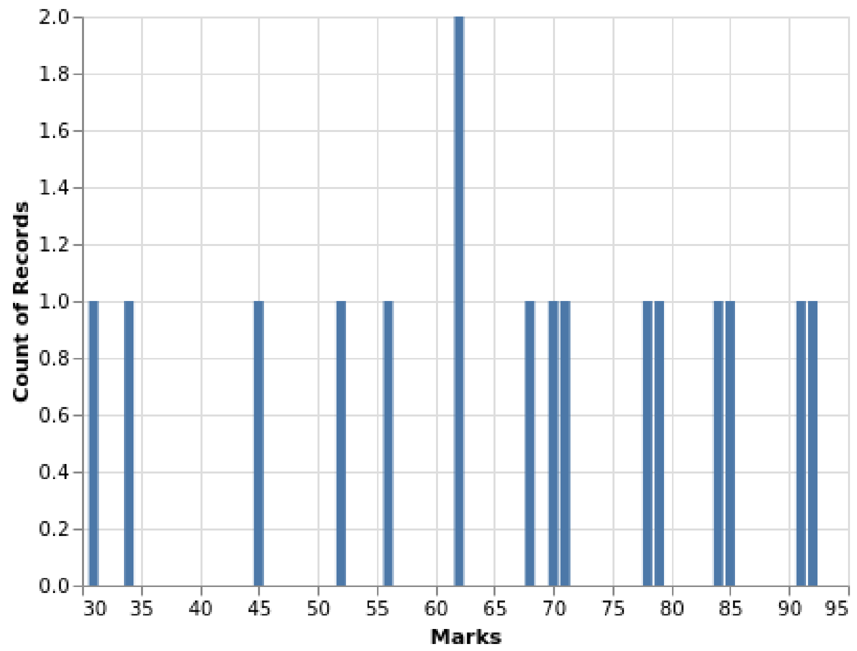
70-80 - {70, 79, 78, 71}

80-90 - {85, 84}

90-100 - {91, 92}

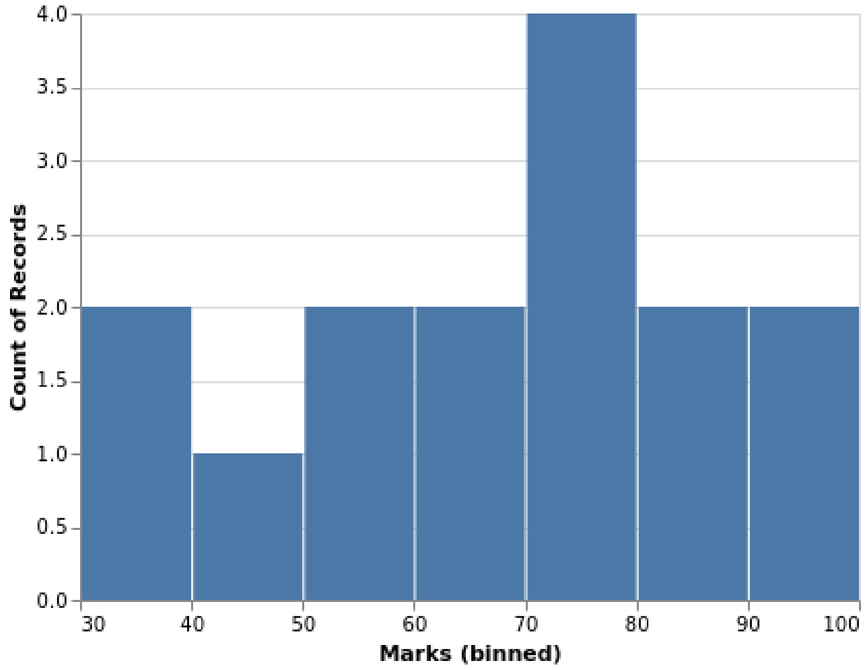
Histogram for previous bins

Before Binning



After Binning

Binned



Merging Datasets

data1

Country Name	Region	Year	Fertility Rate	Country Code
Aruba	The Americas	1960	4.82	ABW
Afghanistan	Asia	1960	7.45	AFG
Angola	Africa	1960	7.379	AGO
Albania	Europe	1960	6.186	ALB
United Arab Emirates	Middle East	1960	6.928	ARE
Argentina	The Americas	1960	3.109	ARG

data2

Country Code	Country Name	Birth rate	Internet users	Income Group
ABW	Aruba	10.244	78.9	High income
AFG	Afghanistan	35.253	5.9	Low income
AGO	Angola	45.985	19.1	Upper middle income
ALB	Albania	12.877	57.2	Upper middle income
ARE	United Arab Emirates	11.044	88	High income
ARG	Argentina	17.716	59.9	High income

```
pd.merge(data1, data2, on='Country Code')
```

Country Name_X	Country Code	Birth rate	Internet users	Income Group	Country Name_Y	Region	Year	Fertility Rate
Aruba	ABW	10.244	78.9	High income	Aruba	The Americas	1960	4.82
Afghanistan	AFG	35.253	5.9	Low income	Afghanistan	Asia	1960	7.45
Angola	AGO	45.985	19.1	Upper middle income	Angola	Africa	1960	7.379
Albania	ALB	12.877	57.2	Upper middle income	Albania	Europe	1960	6.186
United Arab Emirates	ARE	11.044	88	High income	United Arab Emirates	Middle East	1960	6.928
Argentina	ARG	17.716	59.9	High income	Argentina	The Americas	1960	3.109

Pandas GroupBy and Aggregation

	CountryName	CountryCode	Region	BirthRate	InternetUsers	FertilityRate	IncomeGroup
0	Aruba	ABW	The Americas	10.244	78.9	1.669	High income
1	Afghanistan	AFG	Asia	35.253	5.9	5.050	Low income
2	Angola	AGO	Africa	45.985	19.1	6.165	Upper middle income
3	Albania	ALB	Europe	12.877	57.2	1.771	Upper middle income
4	United Arab Emirates	ARE	Middle East	11.044	88.0	1.801	High income

	CountryName	CountryCode	Region	BirthRate	InternetUsers	FertilityRate	IncomeGroup
0	Aruba	ABW	The Americas	10.244	78.9	1.669	High income
4	United Arab Emirates	ARE	Middle East	11.044	88.0	1.801	High income
5	Argentina	ARG	The Americas	17.716	59.9	2.335	High income
7	Antigua and Barbuda	ATG	The Americas	16.447	63.4	2.088	High income
8	Australia	AUS	Oceania	13.200	83.0	1.921	High income

	CountryName	CountryCode	Region	BirthRate	InternetUsers	FertilityRate	IncomeGroup
2	Angola	AGO	Africa	45.985	19.1000	6.165	Upper middle income
3	Albania	ALB	Europe	12.877	57.2000	1.771	Upper middle income
10	Azerbaijan	AZE	Asia	18.300	58.7000	2.000	Upper middle income
16	Bulgaria	BGR	Europe	9.200	53.0615	1.500	Upper middle income
19	Bosnia and Herzegovina	BIH	Europe	9.062	57.7900	1.272	Upper middle income

	CountryName	CountryCode	Region	BirthRate	InternetUsers	FertilityRate	IncomeGroup
1	Afghanistan	AFG	Asia	35.253	5.9	5.050	Low income
11	Burundi	BDI	Africa	44.151	1.3	6.035	Low income
13	Benin	BEN	Africa	36.440	4.9	4.846	Low income
14	Burkina Faso	BFA	Africa	40.551	9.1	5.607	Low income
28	Central African Republic	CAF	Africa	34.076	3.5	4.368	Low income

	CountryName	CountryCode	Region	BirthRate	InternetUsers	FertilityRate	IncomeGroup
6	Armenia	ARM	Asia	13.308	41.90	1.553	Lower middle income
15	Bangladesh	BGD	Asia	20.142	6.63	2.209	Lower middle income
22	Bolivia	BOL	The Americas	24.236	36.94	3.017	Lower middle income
26	Bhutan	BTN	Asia	18.134	29.90	2.082	Lower middle income
33	Cote d'Ivoire	CIV	Africa	37.320	8.40	5.063	Lower middle income

```
df.groupby('IncomeGroup').mean()
```

IncomeGroup	BirthRate	InternetUsers	FertilityRate
High income	12.589836	74.152833	1.804615
Low income	37.238267	5.988333	4.984000
Lower middle income	26.225776	21.871822	3.314306
Upper middle income	18.943638	40.040844	2.342851

We're Funding R&D!

Apply at ai_research@phospheneai.com