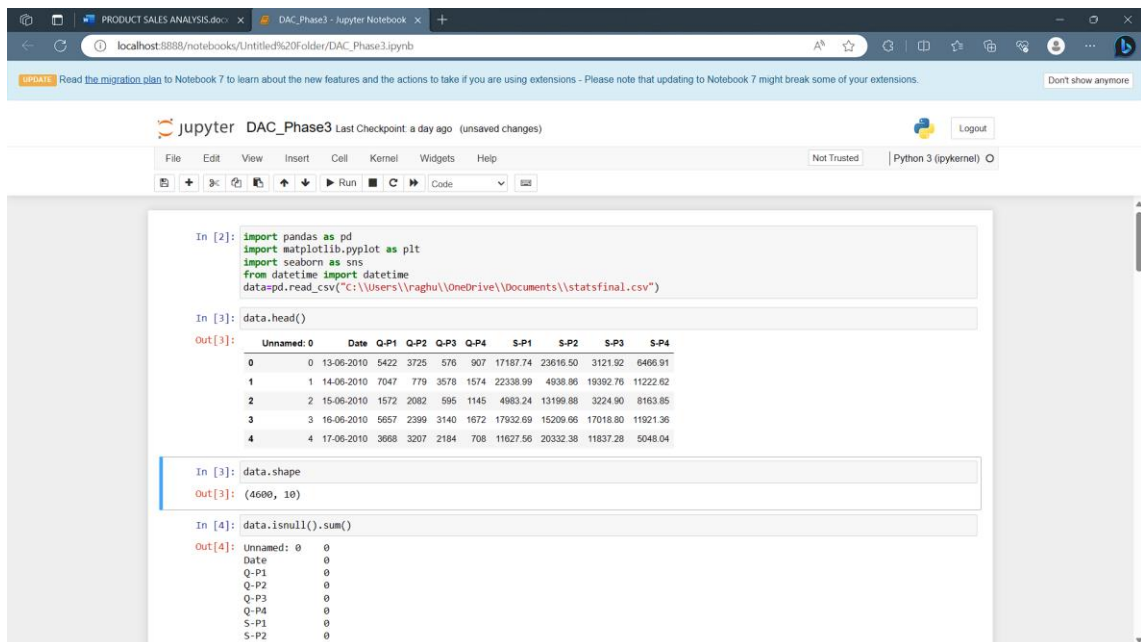


PRODUCT SALES ANALYSIS

Product sales analysis is a systematic examination and evaluation of the performance and results associated with the sales of a particular product or a group of related products. This analysis involves gathering and studying data and information to gain insights into how well a product is selling in the market. It aims to understand various aspects of product sales.

Visualize using python

First, import the CSV file using the function “pd.read_csv()”.



The screenshot displays a Jupyter Notebook window titled "DAC_Phase3 - Jupyter Notebook". The browser address bar shows "localhost:8888/notebooks/Untitled%20of%20older/DAC_Phase3.ipynb". A notification banner at the top mentions a migration plan to Notebook 7. The notebook interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help) and a toolbar with icons for file operations, cell execution, and code editing. The code area contains the following Python code:

```
In [2]: import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from datetime import datetime
data=pd.read_csv("C:\\Users\\raghu\\OneDrive\\Documents\\statsfinal.csv")

In [3]: data.head()
```

The output of the code shows the first five rows of the CSV file:

Unnamed: 0	Date	Q-P1	Q-P2	Q-P3	Q-P4	S-P1	S-P2	S-P3	S-P4
0	13-06-2010	5422	3725	576	907	17187.74	23616.50	3121.92	6466.91
1	14-06-2010	7047	779	3578	1574	22338.90	4938.06	19382.78	11222.62
2	15-06-2010	1572	2082	595	1145	4983.24	13199.88	3224.90	8163.85
3	16-06-2010	5657	2399	3140	1672	17932.69	15209.66	17018.80	11921.36
4	17-06-2010	3668	3207	2184	708	11627.56	20332.38	11837.28	5048.04

Below the head output, the code and its output are shown:

```
In [3]: data.shape
Out[3]: (4600, 10)

In [4]: data.isnull().sum()
Out[4]: Unnamed: 0    0
Date            0
Q-P1            0
Q-P2            0
Q-P3            0
Q-P4            0
S-P1            0
S-P2            0
S-P3            0
S-P4            0
```

By using shape() function, when it comes to the analysis of data and its variants, it is extremely important to realize the volume of data. That is, before we plan to analyze the data and perform synthesis on it, we need to be aware of the dimensions of the data.

This is when the Python shape() method comes into the picture.

With the shape() method, comes the flexibility to obtain the dimensions of any Python object. Yes, it returns a tuple value that indicates the dimensions of a Python object.

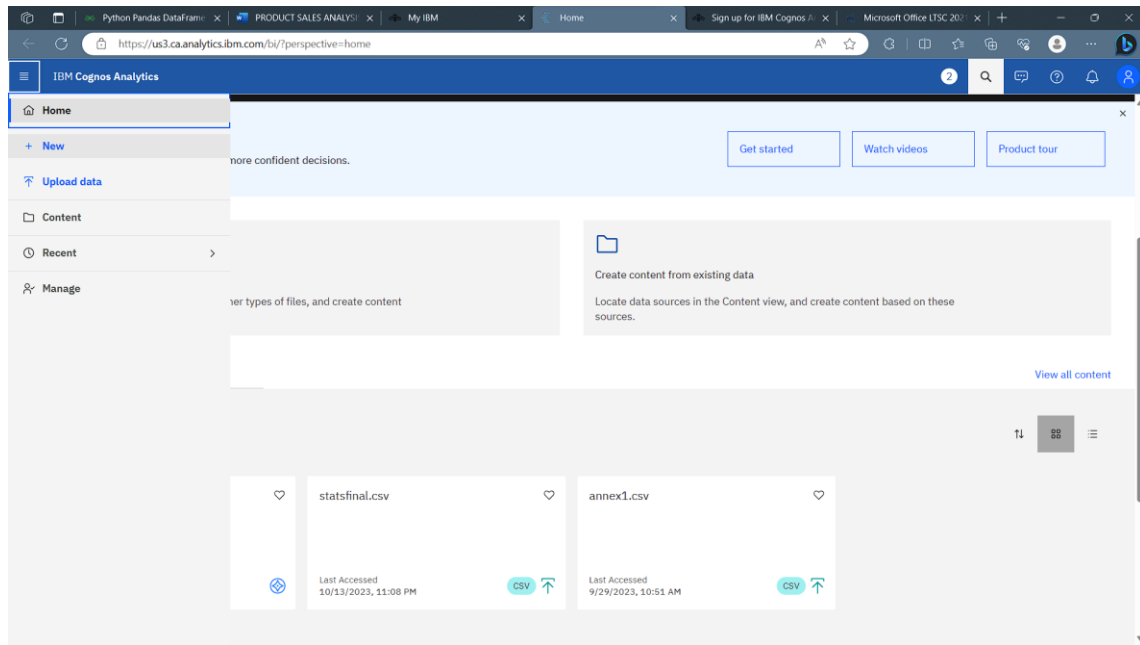
Data Frame in Pandas is two-dimensional size-mutable, potentially heterogeneous tabular data structure with labeled axes (rows and columns). A Data frame is a two-dimensional data structure, i.e., data is aligned in a tabular fashion in rows and columns. Pandas Data Frame consists of three principal components, the **data**, **rows**, and **columns**.

Diagram illustrating the components of a Pandas Data Frame. The table below shows player statistics, with labels for Rows, Columns, and Data.

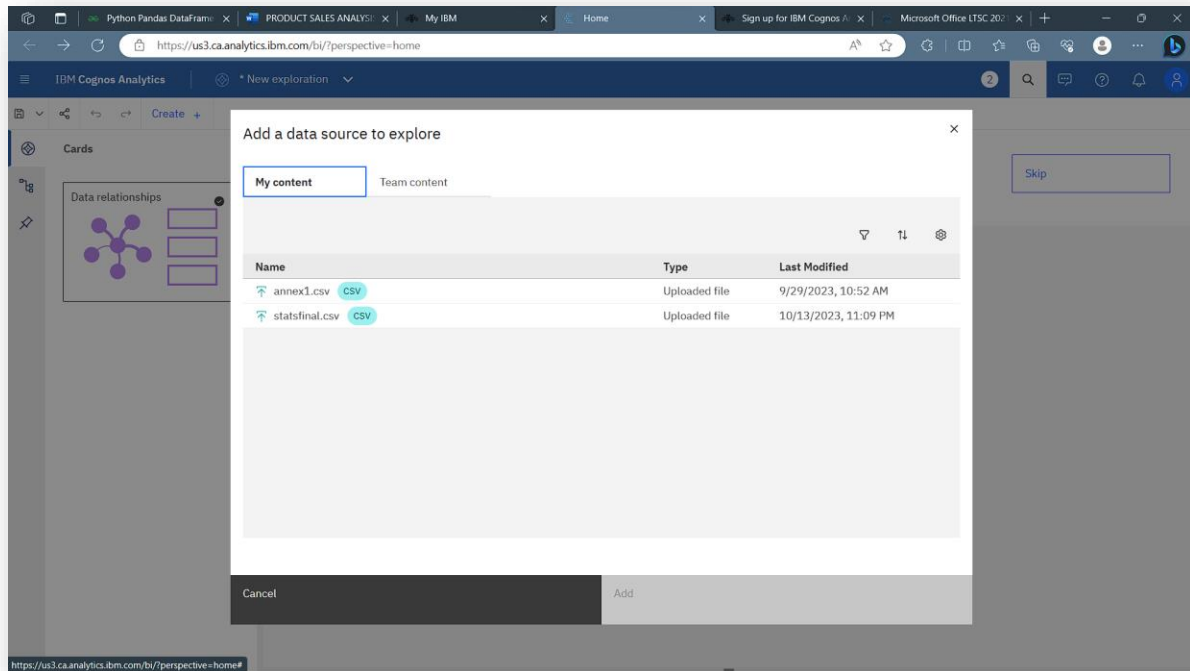
	<i>Name</i>	<i>Team</i>	<i>Number</i>	<i>Position</i>	<i>Age</i>
0	Avery Bradley	Boston Celtics	0.0	PG	25.0
1	John Holland	Boston Celtics	30.0	SG	27.0
2	Jonas Jerebko	Boston Celtics	8.0	PF	29.0
3	Jordan Mickey	Boston Celtics	NaN	PF	21.0
4	Terry Rozier	Boston Celtics	12.0	PG	22.0
5	Jared Sullinger	Boston Celtics	7.0	C	NaN
6	Evan Turner	Boston Celtics	11.0	SG	27.0

Visualize using IBM Cognos Analytics:

First, open IBM Cognos analytic in browser. In the top left corner, click the option button. Then select “New”.



Select the dataset.



After that, choose the Visualization or Exploration techniques and method to analyse your dataset.