

Data analysis

Data Analysis on online sales dataset

financial mathematics

January 2021

By:

SAIRA KHAN

NABIHA NOOR

RABBIA ASHFAQ

University of karachi	<u>Date:</u> <u>January 2021</u>	<u>Pages:</u>
<u>Degree programme:</u> Financial mathematics		
<u>Name of report:</u> Data Analysis on online sales dataset		
<u>Instructor:</u> Syed Umiad Ahmed		
<u>Purpose of this report:</u> The aim of this thesis is to discover how to analyses data using jupyter with different data sets. The proposal of this report to analyses datasets of sales of product in the whole year by using Python's libraries .Here we investigate data to utilizes logical techniques, procedures, calculations and frameworks to separate information Knowledge from organized and unstructured information which is identified with data mining and big data.		

Index:

	contents	Page no:
1	Introduction of data analytic	<u>06</u>
2	Data analyses	<u>07</u>
2.1	Role of data analysis	<u>08</u>
2.2	Python in data analyses	<u>08</u>
3	Data science	09
3.1	Python for data science	10
3.2	Importance of data science	10-11
3.2.1	Purpose of data science	12

3.3	Data science vs. data analytic and big data	12
4	Data set of online purchasing products	13
4.1	Work task	14
5	Os ,pandas and matplotlib lib pkgs	15-16
5.1	For viewing all folders in a directory	17
5.2	Create file using jupyter	18
5.2.2	How to read file on jupyter	19
5.2.3	To see null records from data:	20
5.2.4	To eliminate all null record	20

6	Value type	21
6.1	Numeric conversion	21
6.2	Conversion into date time form	22
6.3	Data representation	23
7	Plotting of data	24
7.1	Demanding region plot	24
7.2	City selection plot	24
7.3	Performance bar graph	26
7.4	Item sale per hour	27

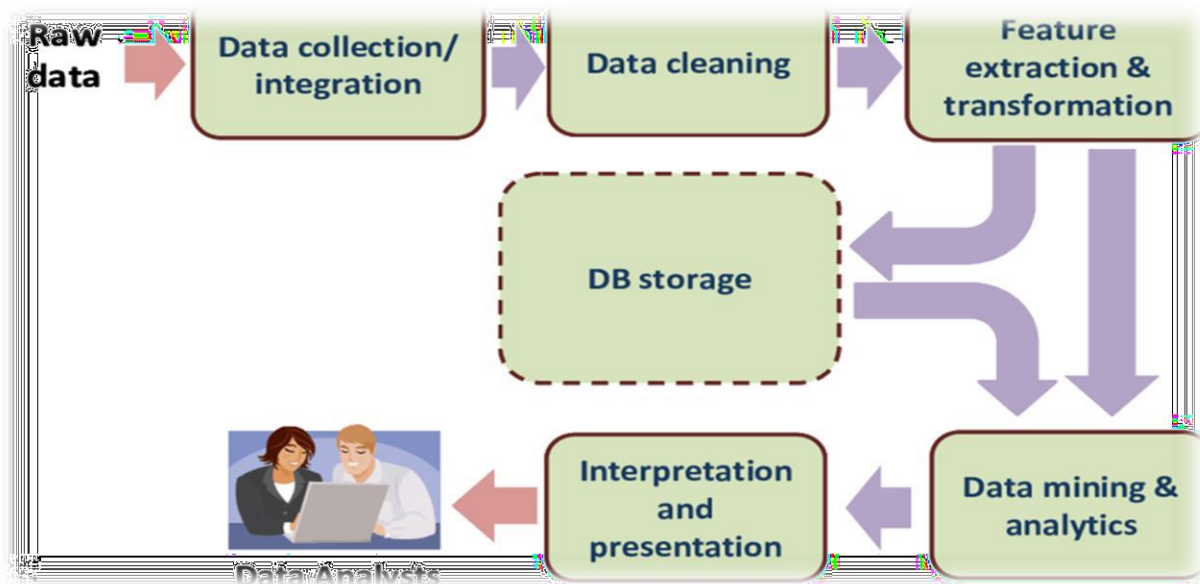
7.5	Item sale per day	27
7.6	Item demand	28
7.7	Item price	29
7. 8	Last working	30

I.INTRODUCTION :

Data analytics is the separating a long data according to our choices. A number of methods has been use in machine learning to refined data for the use of us. This data can be utilized to improve the purchasingpower of a product and it will also vast business. Any kind of data can be exposed to information examination methods to get the knowledge that can be utilized to improve things tasks at hand, so the machines work nearer to crest limit. Data science is an idea to gather all measurements, information analysis. This report will mainly discover how to purify data by utilizing jupyter with python packages. Finally, this report will conclude how to prepare data for analysis, perform simple statistical analysis, create meaningful data visualizations and predict future trends from data.

2. DATA ANALYSIS:

Data Analysis is characterized as a procedure of cleaning, changing, displaying data, examining, gathering, demonstrating, changing and displaying data with the objective of finding helpful and necessary data in establishing basic leadership. In today's business world, data analysis is responsible for making progressive developments. The outcome Are conveyed, proposing ends, and supporting basic Leadership.

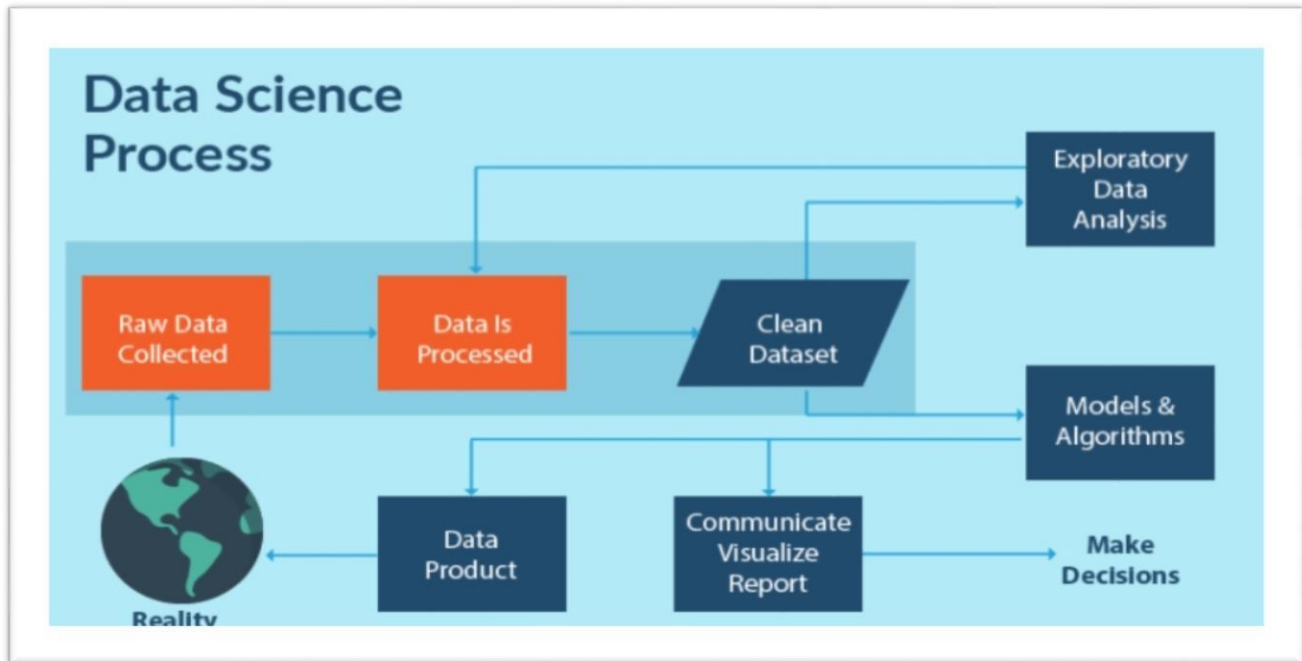


2.1 Role of data analytic:

A Data Analyst interprets data and turns it into information which can offer ways to improve a business, thus affecting business decisions. Data Analysts gather information from various sources and interpret patterns and trends – as such a Data Analyst job description should highlight the analytical nature of the role.

2.2 Python in data analytic:

Python is a useful programming language. It underpins different programming ideal models, including procedural, object-oriented, and practical programming. This current field's important facility is to transfer important data into advertising and business methodologies which enables an organization to develop. Python is easy to use instrument for information investigation. According to the Stack Overflow of 2018, the most standard programming language on world and called the most sensible language for data science mechanical assemblies and applications is Python.



3.Data Science:

Data science is an inter-disciplinary field that uses scientific methods, processes, algorithms and systems to extract knowledge and insights from many structural and unstructured data. Data science is related to data mining, machine learning and big data.

Data science is a data analysis to understand and analyze actual phenomena with data. It uses techniques and theories of mathematics, statistics, computer science, domain knowledge and information science

3.1 Python for data science:

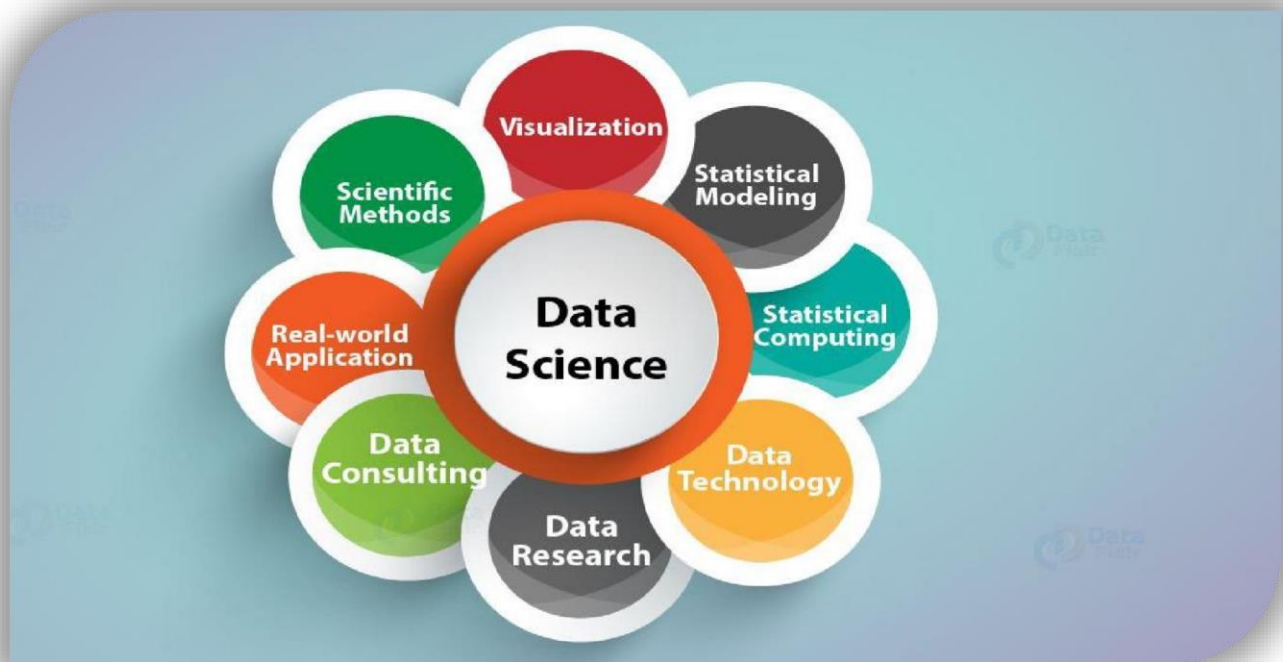
Python is open source, interpreted, high level language and provides great approach for objectoriented programming. It is one of the best language used by data scientist for various data science projects/application. Python provide great functionality to deal with mathematics, statistics and scientific function. It provides great libraries to deals with data science application. There has been a lot of evolution in deep learning Python frameworks and it's rapidly upgrading.

3.2 importance of data science:

Why data science is important?

Data Science churns raw data into meaningful insights. Therefore, industries need data scientist. A Data Scientist is a wizard who knows how to create magic using data. A skilled Data Scientist will know how to dig out meaningful information with whatever data he comes across. He helps the

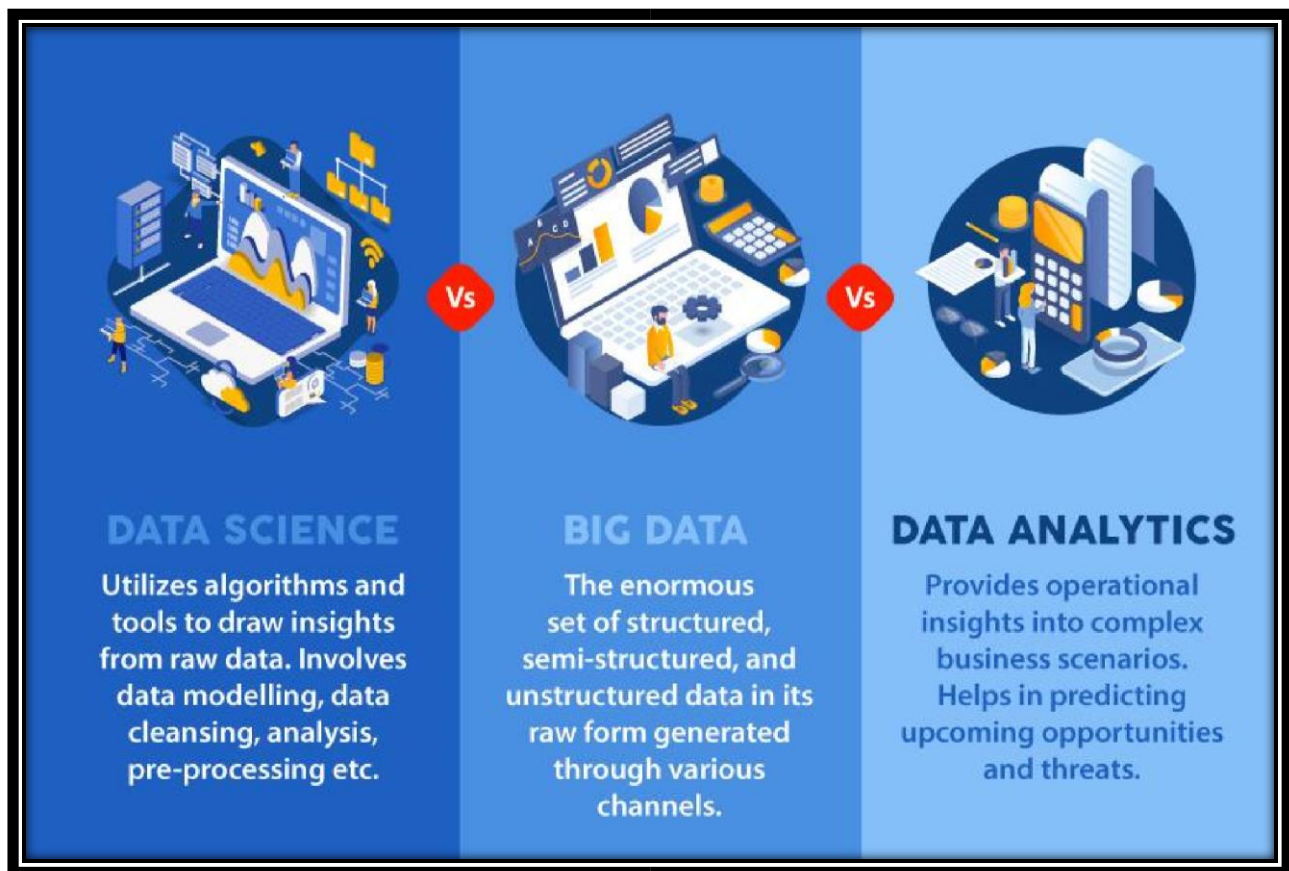
company in the right direction. The company requires strong data-driven decisions at which he's an expert. The Data Scientist is an expert in various underlying fields of Statistics and Computer Science. He uses his analytical aptitude to solve business problems.



Purpose: The purpose of Data Scientists is to extract, pre-process and analyze data. Through this, companies can make better decisions. Various companies have their own requirements and use data













accordingly. In the end, the goal of Data Scientist to make businesses grow better.

Data science vs. data analytics vs. big data:



4. Dataset of online purchasing products:

Complete records of productivity during a whole year:

	Sales-April	1/1/2021 6:48 PM	Microsoft Excel C...	31 KB
	Sales-August	1/1/2021 6:50 PM	Microsoft Excel C...	29 KB
	Sales-December	1/1/2021 6:50 PM	Microsoft Excel C...	66 KB
	Sales-February	1/1/2021 6:50 PM	Microsoft Excel C...	28 KB
	Sales-January	1/1/2021 6:50 PM	Microsoft Excel C...	27 KB
	Sales-July	1/1/2021 6:49 PM	Microsoft Excel C...	33 KB
	Sales-June	1/1/2021 6:49 PM	Microsoft Excel C...	31 KB
	Sales-March	1/1/2021 6:51 PM	Microsoft Excel C...	32 KB
	Sales-May	1/1/2021 6:49 PM	Microsoft Excel C...	29 KB
	Sales-November	1/1/2021 6:50 PM	Microsoft Excel C...	42 KB
	Sales-October	1/1/2021 6:50 PM	Microsoft Excel C...	31 KB
	Sales-September	1/1/2021 6:50 PM	Microsoft Excel C...	30 KB

	A	B	C	D	E	F	G	H	I
1	Sales Rep	Sales Rep	Sales Representa	Sales Representati	Sales Representative	Sales Rep	Sales Representati	Sales Rep	Sales Rep
2	Sara Snyder	Massachu	East	Phillip Young	11/1/2016	Stuff	2	16.32	32.64
3	Sara Snyder	New Jerse	East	Joshua Washington	11/1/2016	Widgets	4	53.35	213.4
4	Randy Wa	New York	East	Frances Campbell	11/1/2016	Junk	9	12.42	111.78
5	Randy Wa	Massachu	East	Lori Shaw	11/1/2016	Widgets	10	53.35	533.5
6	Sara Snyder	New York	East	Roger Freeman	11/1/2016	Junk	9	12.42	111.78
7	Patrick Gr	Nevada	West	Rachel Dunn	11/1/2016	Things	5	17.83	89.15
8	Diane Gor	Oregon	West	Jean Griffin	11/1/2016	Things	2	17.83	35.66
9	Frances W	New York	East	Betty Stewart	11/1/2016	Things	5	17.83	89.15
10	Randy Wa	Massachu	East	Eugene Schmidt	11/1/2016	Things	8	17.83	142.64
11	Patrick Gr	Washingt	West	Gregory Richardsor	11/1/2016	Things	3	17.83	53.49
12	Sara Snyder	New York	East	Debra Palmer	11/1/2016	Stuff	2	16.32	32.64
13	Sara Snyder	New York	East	Evelyn Greene	11/1/2016	Widgets	5	53.35	266.75
14	Randy Wa	Massachu	East	Gary Young	42675	Junk	7	12.42	86.94
15	Sara Snyder	Massachu	East	Harold Rodriguez	42675	Stuff	5	16.32	81.6
16	Sara Snyder	New Jerse	East	Frank Spencer	42675	Widgets	4	53.35	213.4
17	Sara Snyder	Massachu	East	Roy Franklin	42675	Junk	10	12.42	124.2

4.I Work task:

To improve data analysis skills and simplify your decisions, execute these steps in the data analysis process: I. Step-I Collect data

2. Step- processing data

3. Step-Make a folder of these collected datasets by using Jupyter-Python

4. Step-Read file on jupyter

5. Step-cleaning the data

6. Step- analyze data

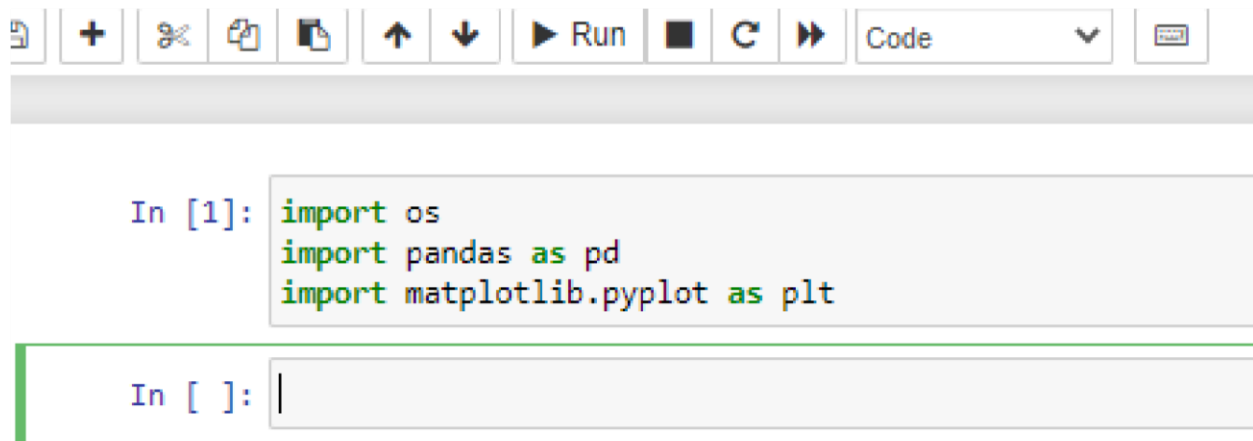
A. Make a new column of month

B. Calculate total sales per month

C. Analyze data by plotting graph

7. Step- examine the result

5 importing packages:



The screenshot shows a Jupyter Notebook interface. At the top is a toolbar with icons for file operations (save, add, delete, copy, paste), navigation (up, down), execution (run, stop, refresh, next), and a dropdown menu currently set to 'Code'. Below the toolbar is a code cell with the following Python code:

```
In [1]: import os
import pandas as pd
import matplotlib.pyplot as plt
```

Below the code cell is an empty input cell with the prompt 'In []: ' and a cursor.

OS package:

The OS module in python provides functions for interacting with the operating system. OS, comes under Python's standard utility modules. This module provides a portable way of using operating system dependent functionality. The `*os*` and `*os.path*` modules include many functions to interact with the file system.

Pandas package:

In computer programming, pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. It is free software released under the three-clause BSD license.

Matplot lib:

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general purpose GUI toolkits like Tkinter, wxPython, .

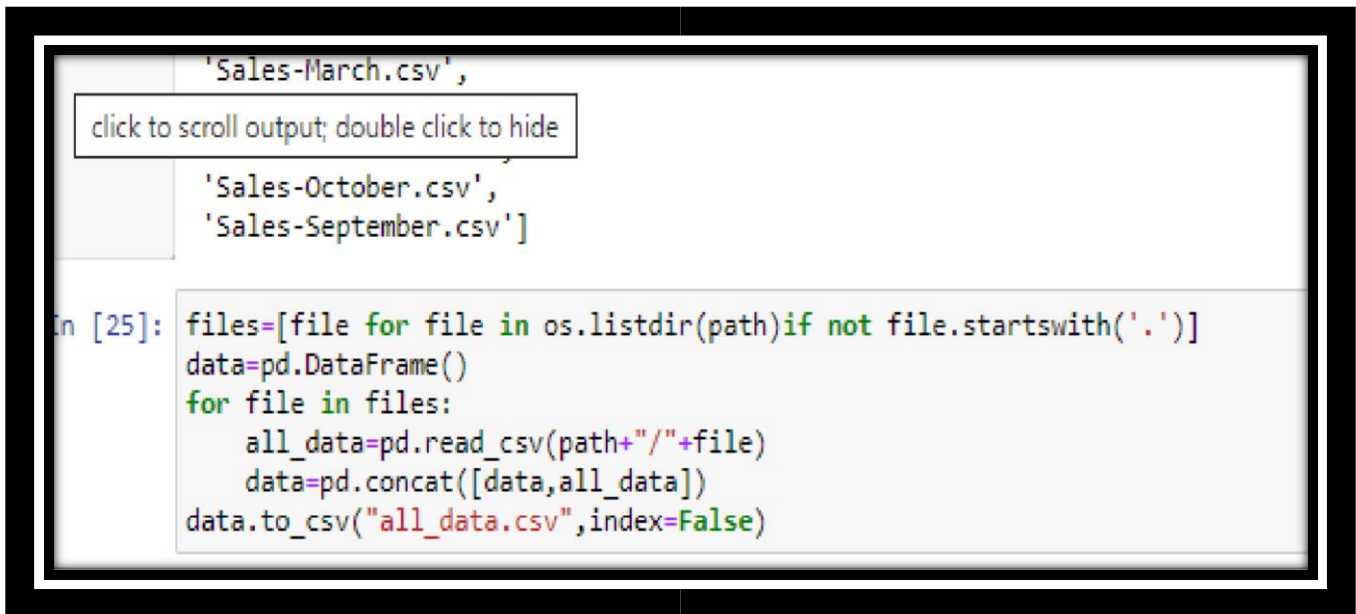
There is also a procedural "pylab" interface based on a state machine (like OpenGL), designed to closely resemble that of MATLAB.

5...I: For viewing all contents in a folders(directory):
Whenever a file appears in any directory, we can assign a variable of path to it, write your file address there or direct copy paste the path in it. All the files that are there this library will kept all folders in front of us.

```
In [19]: path=r"C:\Users\KHAN\Desktop\data sci\sales_data"  
os.listdir(path)
```

```
Out[19]: ['all_data.csv',  
          'Sales-April.csv',  
          'Sales-August.csv',  
          'Sales-December.csv',  
          'Sales-Februrary.csv',  
          'Sales-January.csv',  
          'Sales-July.csv',  
          'Sales-June.csv',  
          'Sales-March.csv',  
          'Sales-May.csv',  
          'Sales-November.csv',  
          'Sales-October.csv',  
          'Sales-September.csv']
```

5.2.1 Create file using jupyter:



```
'Sales-March.csv',  
'Sales-October.csv',  
'Sales-September.csv']  
  
In [25]: files=[file for file in os.listdir(path)if not file.startswith('.')]  
data=pd.DataFrame()  
for file in files:  
    all_data=pd.read_csv(path+"/"+file)  
    data=pd.concat([data,all_data])  
data.to_csv("all_data.csv",index=False)
```

To make one file of whole data we will merged the file.

(path+"/"+file) here jupyter will reached the directory add the files then again do same thing it will complete the loop until it reads everything.

Pd.concat keep the file on one place and then turn it into csv file

5.2.2 How to read file on jupyter:

Data `.head()`..... through this command

```
In [26]: data.head()
```

```
Out[26]:
```

	Sales Representative	Location	Region	Customer	Order Date	Item	Quantity	Price	Total Sale Amount
0	Patrick Graham	Washington	West	Jeremy Baker	2016/04/01	Junk	8	12.42	99.36
1	Frances Warren	New Jersey	East	Albert Dunn	2016/04/01	Widgets	9	53.35	480.15
2	Sara Snyder	New York	East	Robert Hayes	2016/04/01	Junk	4	12.42	49.68
3	Randy Watson	New York	East	Cheryl Riley	2016/04/01	Widgets	5	53.35	266.75
4	Randy Watson	New Jersey	East	Kimberly Coleman	2016/04/01	Widgets	2	53.35	106.70

5.2.3 To see null records from data:

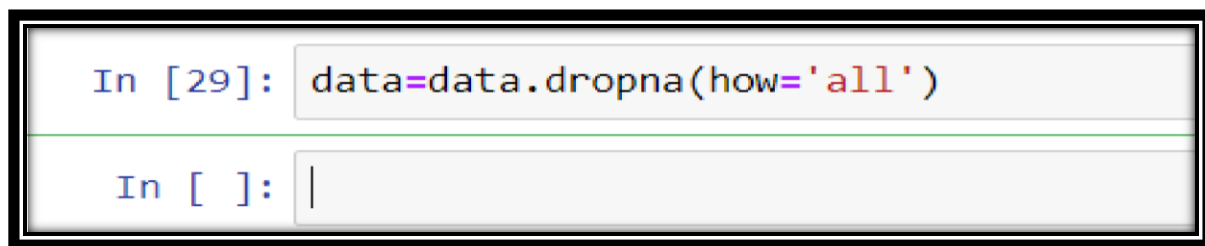
```
In [27]: n_df=data[data.isna().any(axis=1)]  
display(n_df.head())
```

	Sales Representative	Location	Region	Customer	Order Date	Item	Quantity	Price	Total Sale Amount
--	----------------------	----------	--------	----------	------------	------	----------	-------	-------------------

```
In [ ]: |
```

In the process of finding null data in our dataset we use `.isna()` which is Python's command to find null data in the given dataset. When we use argument `.any()` it help to show all null values in data on all axis.

5.2.4 Eliminate null record from data:

A screenshot of a Jupyter Notebook cell. The cell contains two lines of code. The first line is `In [29]: data=data.dropna(how='all')`. The second line is `In []: |`, which is a placeholder for the next command. The cell is highlighted with a thick black border.

```
In [29]: data=data.dropna(how='all')  
  
In [ ]: |
```

`.dropna()` is Python command which use to delete null values on dataset.

6.I conversion into numeric:

Before converting first we have to look out which type of value we have.

There are different types of values like, float numbers, integers, object etcccc

6.find out in which type of value we have in data:

```
all_data.info()

<class 'pandas.core.frame.DataFrame'>
Int64Index: 3888 entries, 0 to 380
Data columns (total 12 columns):
 #   Column                Non-Null Count  Dtype
---  -
 0   Sales Representative    3888 non-null   object
 1   Location                3888 non-null   object
 2   Region                 3888 non-null   object
 3   Customer               3888 non-null   object
 4   Order Date             3888 non-null   datetime64[ns]
 5   Item                   3888 non-null   object
 6   Quantity                3888 non-null   int64
 7   Price                  3888 non-null   float64
 8   Total Sale Amount      3888 non-null   float64
 9   day                    3888 non-null   int64
10  month                  3888 non-null   int64
11  year                   3888 non-null   int64
dtypes: datetime64[ns](1), float64(2), int64(4), object(5)
memory usage: 318.9+ KB
```

Here we have “quantity and total sale amount” in floating type;

6.I pd.to_numeric will change it into numbers.

```
3]: all_data['Price']=pd.to_numeric(all_data['Price'])
    all_data['Total Sale Amount']=pd.to_numeric(all_data['Total Sale Amount'])
```

6.2; what is the purpose of changing value in date time format.?

To get Python to manipulate how a date is formatted, we need to import the native date time module. This module contains all of the methods we need to take care of a majority of the formatting needs we may have. We can import it with a simple import statement. We're using the from here so that we can reference the functions without using dot notation.

```
[36]: all_data['Order Date'] = pd.to_datetime(all_data['Order Date'])  
      all_data['Order Date'].dtype  
      type(all_data['Order Date'])
```

```
[36]: pandas.core.series.Series
```

```
[60]: all_data['day'] = (all_data['Order Date']).dt.day  
      all_data['month'] = (all_data['Order Date']).dt.month  
      all_data['year'] = (all_data['Order Date']).dt.year  
      all_data['minute'] = (all_data['Order Date']).dt.minute  
      all_data['hour'] = (all_data['Order Date']).dt.hour
```

6.3 data presentation into date time format:

Through this conversion we will not face any faculty in date format. This date function separate the each value . datetime helps us identify and process time-related elements like dates, hours, minutes, seconds, days of the week, months, years, etc. It offers various services like managing time zones and daylight savings time. It can work with timestamp data. It can extract the day of the week, day of the month, and other date and time formats from strings.

```
all_data.describe()
```

	Quantity	Price	Total Sale Amount	day	month	year	minute	hour
count	3888.000000	3888.000000	3888.000000	3888.000000	3888.000000	3888.0	3888.0	3888.0
mean	5.566358	22.880005	127.617577	15.584362	3.051955	2016.0	0.0	0.0
std	2.852649	15.997386	120.360891	8.324094	1.382710	0.0	0.0	0.0
min	1.000000	12.420000	12.420000	1.000000	1.000000	2016.0	0.0	0.0
25%	3.000000	12.420000	53.350000	8.000000	2.000000	2016.0	0.0	0.0
50%	6.000000	16.320000	97.920000	16.000000	3.000000	2016.0	0.0	0.0
75%	8.000000	17.830000	142.640000	23.000000	4.000000	2016.0	0.0	0.0
max	10.000000	53.350000	533.500000	31.000000	5.000000	2016.0	0.0	0.0

The next stage is to start to make some comparisons, again using the right dash boarding software this should be just a few clicks, otherwise you will need to put in a few hours work in excel.

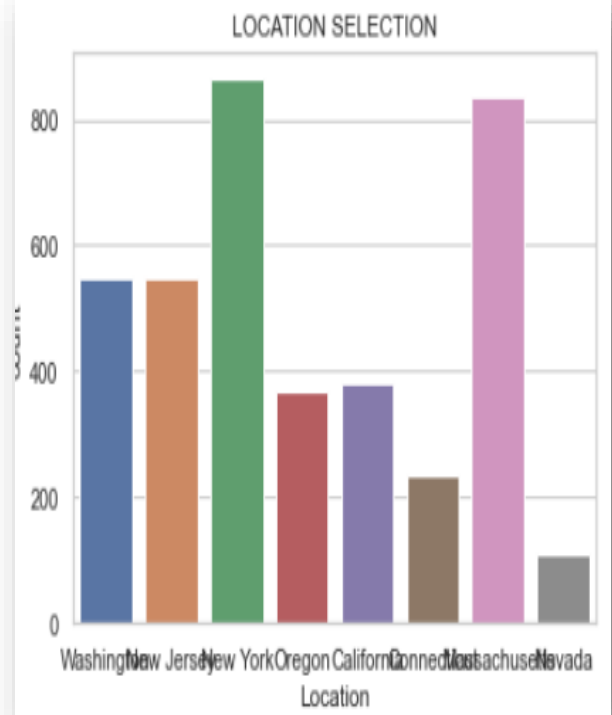
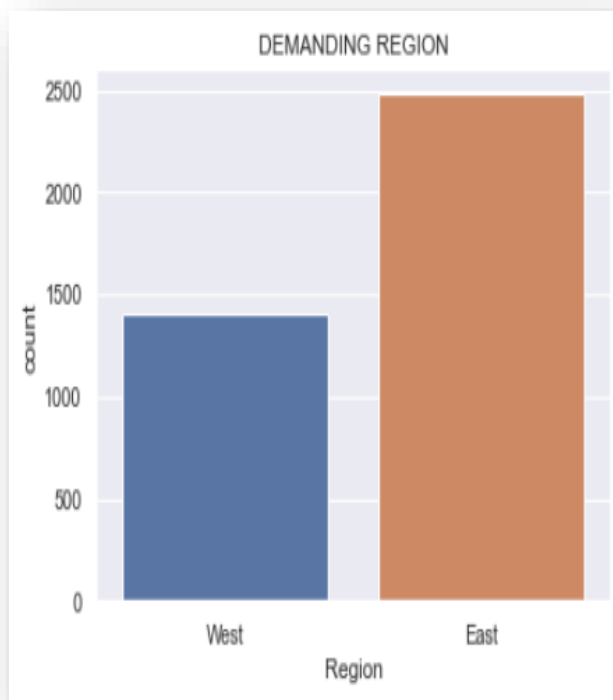
Explore performance quickly by changing between salesperson, location, region, customer type, profit, order value, etc.

7. Plotting:

By using seaborn library we are showing a useful aspects of data. In this analyses of region will help us to know which side of country like our products a lot so we can increase our productivity their more and more to earn a prominent profit.

```
2]: sns.set(style="whitegrid") #wana open branch in Lew Location
NEW_BRANCH = sns.countplot(x="Location", data =all_data).set_title("LOCATION SELECTION")
```

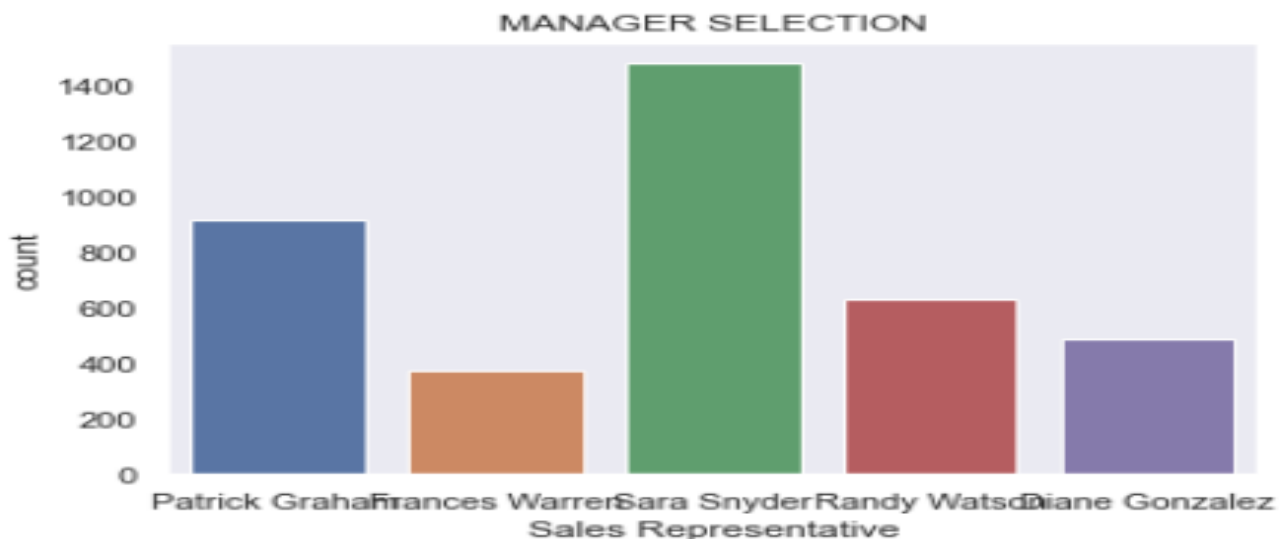
```
: sns.set(style="darkgrid") #which region like our producst most
REGION = sns.countplot(x="Region", data =all_data).set_title("DEMANDING REGION")
```



7.3: Performance bar graph:

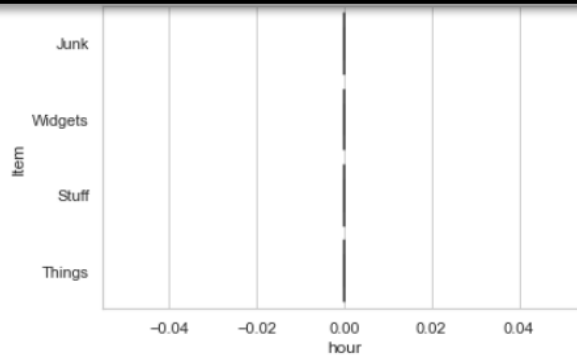
```
[64]: sns.set(style="dark")                                #wana know who is wrokign more efficiently
EMPOLYE= sns.countplot(x="Sales Representative", data =all_data).set_title("MANAGER SELECTION")
```

Get an idea who is working more efficiently, will help to select one person from a huge crowd

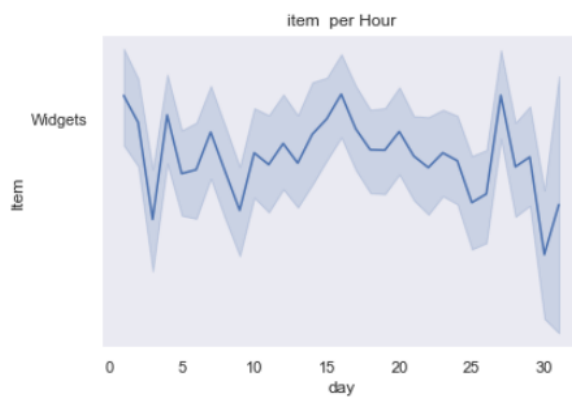


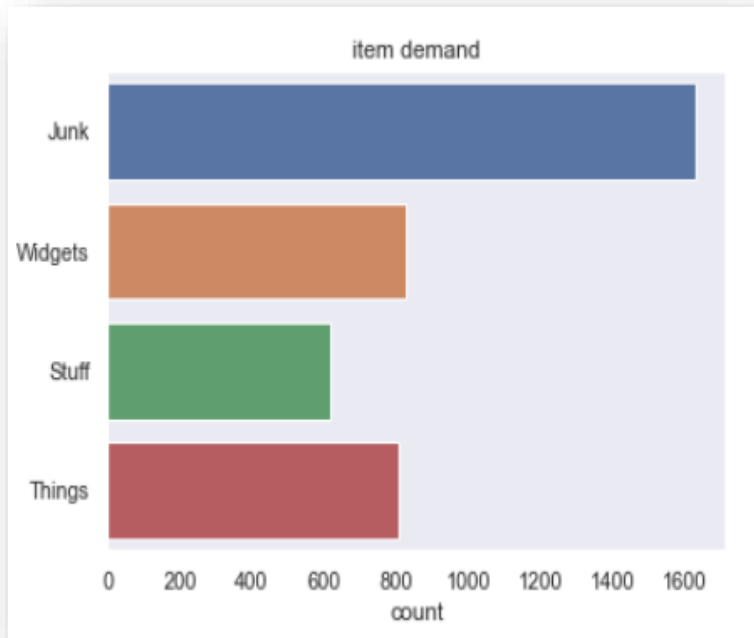
```
In [86]: sns.countplot(y="Item", data=all_data).set_title("item demand")
```

```
Out[86]: Text(0.5, 1.0, 'item demand')
```



```
[73]: genderCount = sns.lineplot(x="day", y = 'Item', data =all_data).set_title("item per Hour")
```





This code find out the demanding item for us, to analyze what is more common among people, what people like to buy fast.

If we know the interest of public it will boost our revenue when we

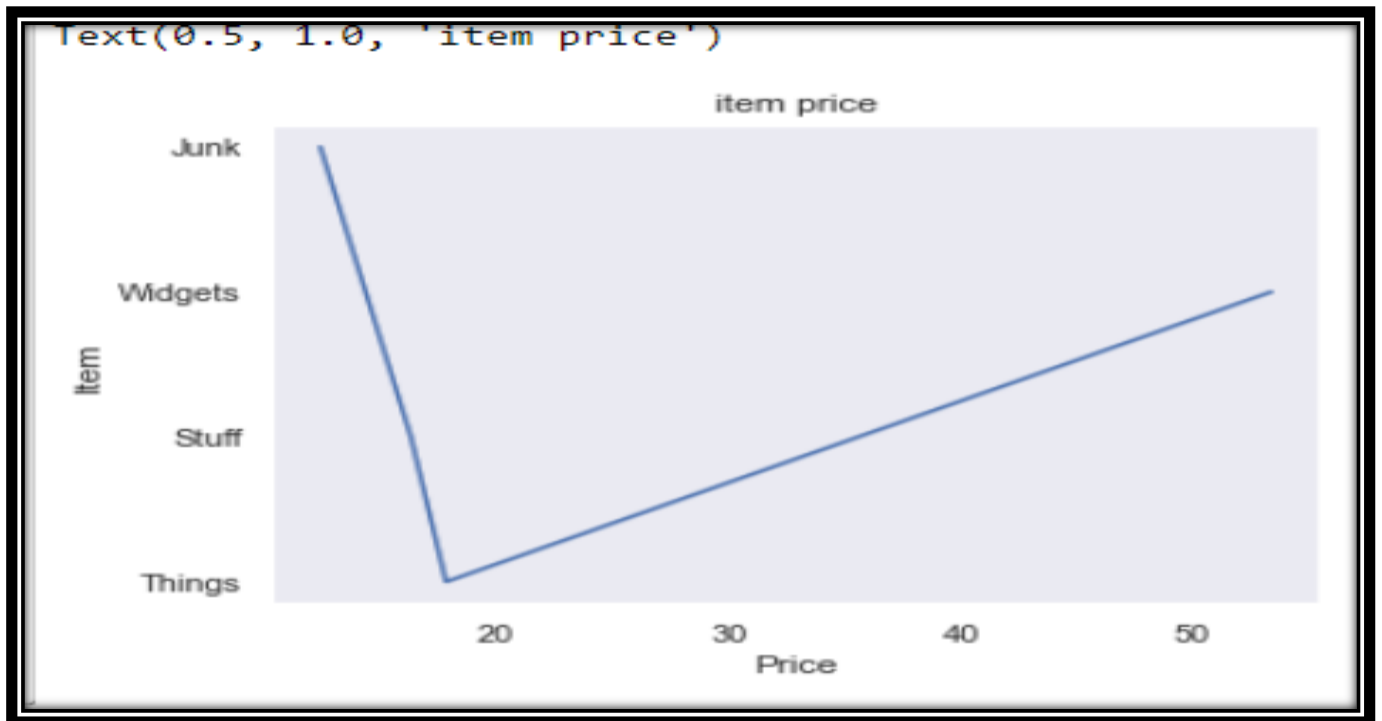
starting generation this product more and more with new variations with attractive offers.

```
In [89]: sns.lineplot(y="Item",x="Price", data=all_data).set_title("item price")  
Out[89]: Text(0.5, 1.0, 'item price')
```

Through this coding and graph we will analyze which

Item has much price

What is the price fluctuation we have in our products?



THE LAST WORKING OF DATA:

```
[In [99]: sns.barplot(x="Customer", y="Total Sale Amount", estimator = sum, data=all_data)
Out[99]: <AxesSubplot:xlabel='Customer', ylabel='Total Sale Amount'>
```

