

DATA ANALYTICS & MACHINE LEARNING WITH PYTHON

A PROJECT REPORT

Submitted by

MODI JAY PANKAJKUMAR

190130107067

In partial fulfilment of the degree of

BACHELORS OF ENGINEERING

In

Computer Engineering

Government Engineering College , Gandhinagar



Gujarat Technological University, Ahmedabad

August,2022



Government Engineering College Sec-28 Gandhinagar

Academic Year

[2022-23]

CERTIFICATE

This is to certify that the work entitled [Summer Internship] “Data Analytics with Python ” has been carried out by Modi Jay Pankajkumar (190130107067) under my guidance in partial fulfillment of the degree of Bachelor of Engineering in Computer Engineering (7th Semester) at Government Engineering College, Sec 28, Gandhinagar affiliated to Gujarat Technological University during the academic year 2022-23.

Guided By:

Prof. Dhawal Khem

Head of the department :

Prof. D A. Parikh

COMPANY PROFILE



Company Name : INFO LABZ IT SERVICES PVT. LTD.

Address : 405, Vraj Avenue, Above Sam's pizza, Nr. Commerce
Six Road, Navrangpura, Ahmedabad, 380009

Contact No : 8866662662

About the company

Established in 2016, incorporation with our parent IT company, INFO LABZ IT SERVICES PVT. LTD. has managed to make its own position in the IT Sector. We are involved in Web Development, App Development, Progressive Web Application Development, IOT solutions, Graphics & Designing, Digital Marketing, Domain & Hosting services, SMS services etc.

In the span of six years we have managed to deliver all projects on time with utmost accuracy to our clients across the globe. We have dedicated teams of experienced and hard working developers. Our developers who are always willing to take on new challenges and looking forward to learning new things, are the heart of this company.

Our objective is to sustain exponential growth in the IT industry. Our mission is to deliver the best with top notch quality every quarter and our vision is to develop a product with one of its kind concepts which could be used by millions of people.

JOINING LETTER



INFOLABZ IT SERVICES PVT. LTD.
WEB DEVELOPMENT | APP DEVELOPMENT | DATA SCIENCE | IOT



Date: 11-06-2022

Subject: Regarding Summer Internship at INFOLABZ IT SERVICES PVT. LTD.

Internship Domain: Data Analytics and Machine Learning

Dear Modi Jay Pankajkumar,

In reference to your application we would like to congratulate you on being selected for internship with InfoLabz IT Services Pvt. Ltd. based at Ahmedabad. Your internship is scheduled to start from 20 June 2022 for a period of 15 days.

During this internship, the concentration will be on helping you understand logical concepts with their practicality and implications to help you connect your classroom knowledge to industry standards. Your internship will include orientation and focus primarily on learning and developing new skills and gaining a deeper understanding of concepts through hands-on application of the knowledge.

We will be happy to guide you to learn new skills which are extremely helpful in professional standard.



Ms. Twinkle Shah
Internship Coordinator
InfoLabz, Ahmedabad



+91 8866662662
+91 8141236662



info@infolabz.in
www.infolabz.in



405 Vraj Avenue, Above SAM'S Pizza
Nr, Commerce Six Rd, Navrangpura,
Ahmedabad, Gujarat 380009



COMPLETION LETTER



INFOLABZ IT SERVICES PVT. LTD.
WEB DEVELOPMENT | APP DEVELOPMENT | DATA SCIENCE | IOT

COMPLETION CERTIFICATE OF SUMMER INTERNSHIP

Date: 05-07-2022

Enrollment: 190130107067

Semester: 7th, Computer Engineering

Government Engineering College Gandhinagar

To whom it may concern,

We are delighted to provide this certificate for the successful completion of the requirements and work performed during the two-week free internship (20 June 2022 to 4 July 2022) by Modi Jay Pankajkumar.

In this internship tenure, we have covered the fundamentals of Data Analytics and Machine Learning. In the data analytics part, we have worked on API data and covered the basics of analysis using pandas and data visualization using matplotlib. In machine learning, we have implemented elementary regression models.

We wish Modi Jay Pankajkumar all the best for future endeavors.



Ms. Twinkle Shah

Internship Coordinator

InfoLabz, Ahmedabad



+91 8866662662
+91 8141236662



info@infolabz.in
www.infolabz.in



405 Vraj Avenue, Above SAM'S Pizza
Nr, Commerce Six Rd, Navrangpura,
Ahmedabad, Gujarat 380009

ACKNOWLEDGEMENT

It is my pleasure to present this Summer internship report of “**Data Analytics with Python**”. First I would like to thank **Twinkle Shah** , the Internship Coordinator of Infolabz for giving me this opportunity to do an internship within the organization. Other special thanks goes to Chintan Nagarecha sir,the Director of Infolabz who has taught each concept in a very beautiful manner.

The satisfaction that accompanies the successful completion of any task would be incomplete without the mention of people whose ceaseless corporation made it possible, whose constant guidance and encouragement grown all effort with success. My sincere thanks to our internal guide **Prof. Dhawal Khem** and Our Head of the Department **Prof. D.A . Parikh** for providing valuable guidance throughout our project work. Whenever I Ire confused about anything, they Ire always there to help us.

I am thankful to **Gujarat Technological University** to giving the opportunity for assigning this **Summer Internship** work.I are also thankful to our Parents, Friends and Others for theiropinions and support in the system.

Yours Thankfully

Jay Modi

190130107067

ABSTRACT

Currently the world is running after the Artificial Intelligence and Machine Learning is the building block of Artificial Intelligence. Data Analytics is a building block of Machine Learning. And Python language is ruling over these field with its awesome features. Data Analytics is one of the most important and helpful factors for business growth. One can do analysis on any data using python libraries like NumPy and pandas. Analysed data can be visualized by matplotlib. Data Analysis is useful for decision nmaking and machine learning model development. A machine learning model can be developed using data to predict essential parameters.

LIST OF FIGURES

Figure No.	Figure Description	Page No.
1.1	Try and Catch block	10
2.1	NumPy Example	12
2.2	Example of Library functions	13
3.1	Json Structure	14
3.2	Mutual Fund API	17
3.3	ISRO Pie chart	21
3.4	Excel file of IPL data	22
3.5	Result of IPL data	22
3.6	University Acquisitor Bar chart	24
3.7	University Acquisitor	25
4.1	Price.csv file	28
4.2	Scatter plot	29
4.3	Linear Regression output	29
4.4	Ordersdata.csv file	30
4.5	Multiple Linear model output	30
4.6	Image processing example(1)	31
4.7	Image processing example(2)	31
4.8	Polynomial regression excel data file	32
4.9	Polynomial regression line chart	33
4.10	Polynomial regression Result	33
4.11	Machine learning through API result	35

TABLE OF CONTENTS

Content	Page No.
Certificate	1
Company Profile	2
Joining Letter	3
Completion Letter	4
Acknowledgement	5
Abstract	6
List of figure	7
1. Getting started with Python & OOPS	9
2. Python Libraries	11
3. Data Extraction & API Handling	14
4. Basics of Machine Learning	26
Conclusion	36
References	36

CHAPTER : 1 GETTING STARTED WITH PYTHON AND OOPS

Before getting direct dive towards Data Analytics or Machine Learning it is necessary to get a very good command on the language which we are going to use for implementing these technologies. And Python language is having its own beautiful features and functionalities which are significantly helpful for analytics.

1.1 OOPS Concepts :

1.1.1 Class:

A class is a collection of objects. A class contains the blueprints or the prototype from which the objects are being created. It is a logical entity that contains some attributes and methods.

1.1.2 Objects:

The object is an entity that has a state and behavior associated with it. It may be any real-world object like a mouse, keyboard, chair, table, pen, etc.

Class has `__init__` method. I can call that method as constructor. It is run as soon as an object of a class is instantiated. The method is useful to do any initialization you want to do with your object.

1.1.3 Inheritance:

Inheritance is the capability of one class to derive or inherit the properties from another class. The class that derives properties is called the derived class or child class and the class from which the properties are being derived is called the base class or parent class.

I understand the types of inheritance, like, Single level inheritance, Multilevel inheritance, Hierarchical inheritance, Multiple inheritance.

1.1.4 Polymorphism:

Polymorphism simply means having many forms. For example, I need to determine if the given species of birds fly or not, using polymorphism I can do this using a single function.

1.1.5 Encapsulation:

Encapsulation is one of the fundamental concepts in object-oriented programming (OOP). It describes the idea of wrapping data and the methods that work on data within one unit.

1.1.6 Data Abstraction:

It hides the unnecessary code details from the user. Also, when you do not want to give out sensitive parts of our code implementation and this is where data abstraction came.

1.2 Python : Real World Development

Python is dynamically typed language with great support of In-built libraries and this gives power to developer for building something incredible. One constraint using python is that you have to take care about time complexity because it is an interpreted language and already slow compare to compiled languages like C, C++, JAVA so you have to handle this while programming by making the least possible time complexity. And the best pros using Python is that you have to make your very less focus towards syntax compare to other languages.

In the industry ready development developer has to ensure that in case of inappropriate input or in case of any other type of case software has to return proper kind of error message which can be understood by a non-technical person and for that it has to be compiled through there is invalid input. To creating such kind of logic you can use try and except keywords.

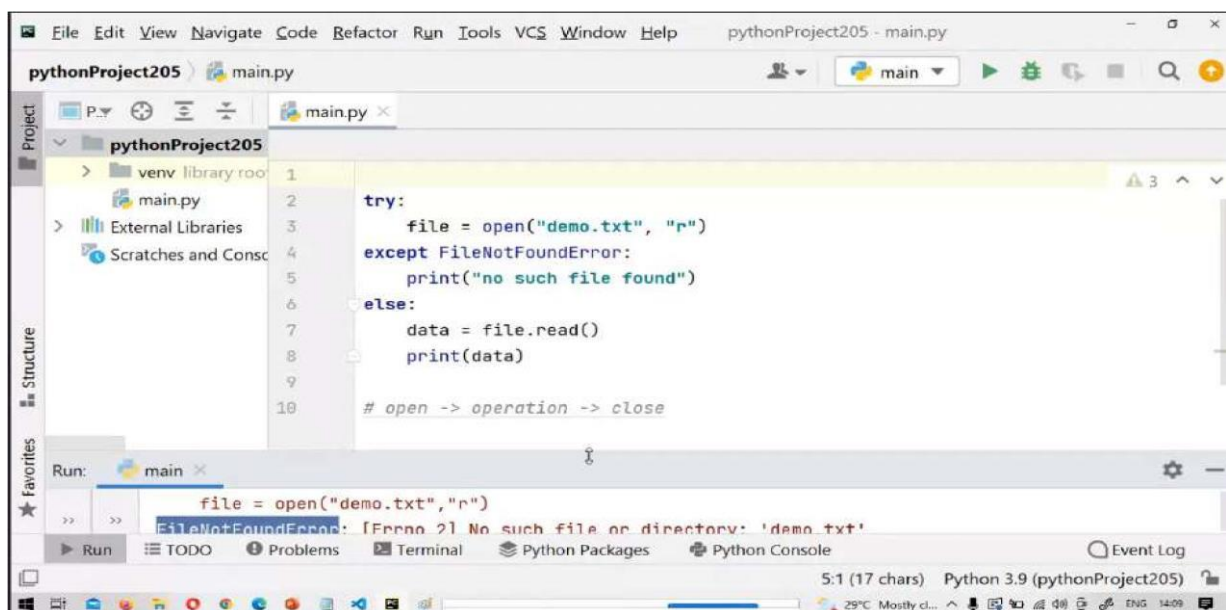


Figure 1.0.1 try catch block

CHAPTER : 2**PYTHON LIBRARIES****2.1 PANDAS:**

Pandas stands for “Python Data Analysis Library”. Pandas is a Python package providing fast, flexible, and expressive data structures designed to make working with “relational” or “labelled” data both easy and intuitive. It aims to be the fundamental high-level building block for doing practical, real-world data analysis in Python. Pandas generally provide two data structures for manipulating data, They are: **i. Series ii. DataFrame**

i. Series:

Pandas Series is a one-dimensional labelled array capable of holding data of any type (integer, string, float, python objects, etc.). The axis labels are collectively called indexes. Pandas Series is nothing but a column in an excel sheet. Labels need not be unique but must be a hashable type. The object supports both integer and label-based indexing and provides a host of methods for performing operations involving the index.

ii. DataFrame:

Pandas DataFrame is a two-dimensional size-mutable, potentially heterogeneous tabular data structure with labelled 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 DataFrame consists of three principal components, the data, rows, and columns.

import pandas as pd

2.2 NUMPY:

NumPy is a library for the Python programming language, adding support for large, multidimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays. It is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays. It is the fundamental package for scientific computing with Python. Its obvious scientific uses, Numpy can also be used as an efficient multi-dimensional container of generic data.

2.2.1 Creating a Numpy Array:

Arrays in Numpy can be created by multiple ways, with various number of Ranks, defining the size of the Array. Arrays can also be created with the use of various data types

253248

such as lists, tuples, etc. The type of the resultant array is deduced from the type of the elements in the sequences.

```
import numpy as np  
np.array()
```

2.2.2 Example

```
In [2]: from collections import Counter  
import numpy as np  
list=[1,2,3,4,["python","java","c++",(10,20,30)],5,6,7,["apple","banana","orange"]]  
print("1)--\n ",list[4][3][2])  
print(" ",list[8][2].capitalize(),"->",list[8][2][5])  
print(" ",list[4][0].capitalize(),"->",list[4][0][4])  
t1=(1,2,3,['hello',10,20,30,"world"],4,5,6,("one","two",100,200,300))  
print("2)--\n ",t1[7][1],"->",t1[7][1][2],"\\n ",t1[3][4].capitalize(),"->",t1[3][4][2],"\\n ",t1[5])  
s='this is string'  
print("3)--\n ",dict(Counter(s)))
```

1)--
30
Orange -> e
Python -> o
2)--
two -> o
World -> r
5
3)--
{'t': 2, 'h': 1, 'i': 3, 's': 3, ' ': 2, 'r': 1, 'n': 1, 'g': 1}

Figure 2.1 NumPy Example

2.3 Matplotlib :

It is an amazing visualization library in Python for 2D plots of arrays. Matplotlib is a multiplatform data visualization library built on NumPy arrays and designed to work with the broader SciPy stack. One of the greatest benefits of visualization is that it allows us visual access to huge amounts of data in easily digestible visuals. Matplotlib consists of several plots like line, bar, scatter, histogram etc.

2.4 Seaborn:

Seaborn is a library mostly used for statistical plotting in Python. It is built on top of Matplotlib and provides beautiful default styles and color palettes to make statistical plots more attractive.

2.5 Scikit-Learn:

Scikit-learn is a free machine learning library for Python. It features various algorithms like support vector machine, random forests, and k-neighbours, and it also supports Python numerical and scientific libraries like NumPy and SciPy.

2.6 Tensorflow:

TensorFlow is an awesome library for ML and AI. It can be used across a range of tasks but has a particular focus on [training](#) and [inference](#) of [deep neural networks](#). TensorFlow can be used in a wide variety of programming languages, most notably Python, as well as Javascript, C++, and Java. This flexibility lends itself to a range of applications in many different sectors.

2.7 OpenCV:

OpenCV is a programming library/package that has been created especially for allowing programmers to enter the world of Computer Vision. OpenCV stands for Open-Source Computer Vision (Library). It is the most commonly used, popular, and well-documented Computer Vision library. It is open-source, which means that one does not require a license to utilize the software. As one may know, most Machine Learning Algorithms require inputs to be quantitative in nature, i.e., numerical. OpenCV allows us to apply Machine Learning techniques to images, however, oftentimes we are required to pre-process and prepare the raw images for them to be transformed into features (columns of data) that are useful and usable by our Machine Learning Algorithms.

2.8 Examples:

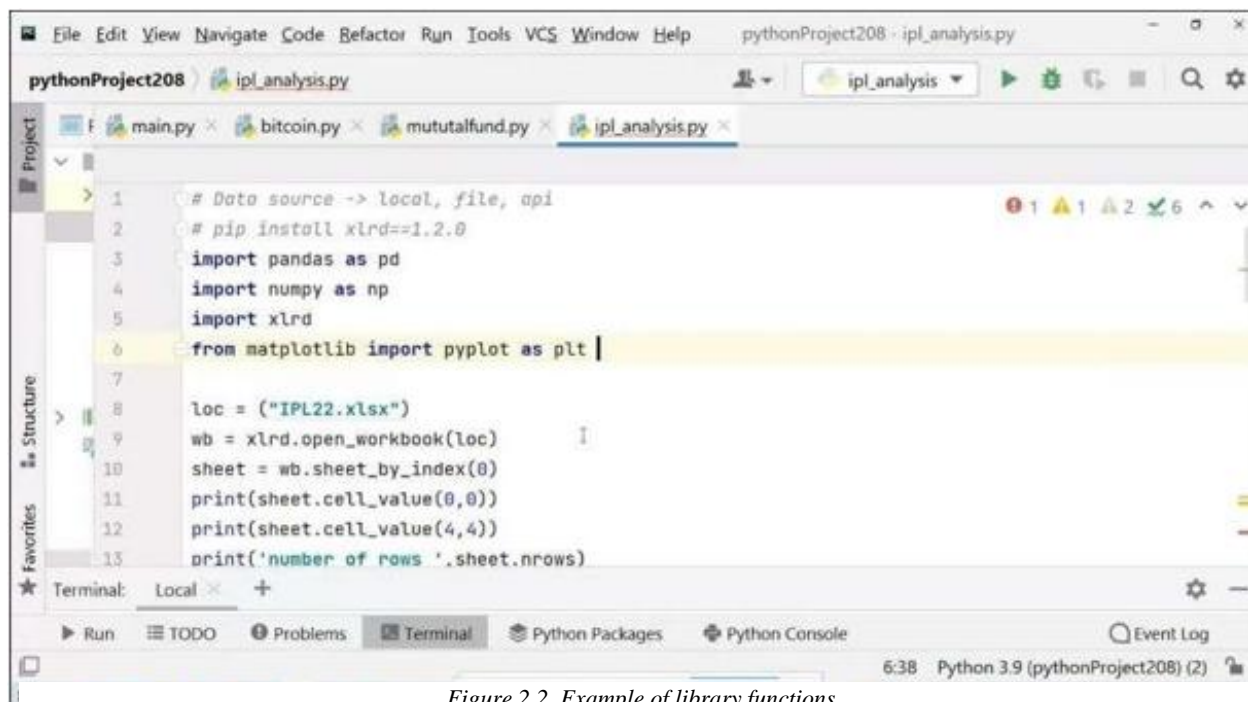


Figure 2.2 Example of library functions

To make use of libraries first you have to import required libraries by 'import' keyword. And you can set alias name for smoothly and frequently use that library using keyword 'as'. Then you can make use of methods just like `xlrd.open_workbook()`, `sheet_by_index()`, `cell_value(x pos,y pos)` and as per the definition of particular method.

CHAPTER : 3 DATA EXTRACTION & API HANDLING

Before diving into data analytics first understanding JSON structure is essential.

3.1 Intro to JSON

- JSON stands for **JavaScript Object Notation**.
- JSON is a text format for storing and transporting data.
- JSON is "self-describing" and easy to understand.

JSON Example:

```
{
  "name" : "Jay",
  "age" : 20,
  "car" : "Alto"
}
```

Here, the basic syntax is **KEY : VALUE**.

Here, name, age and car are the keys and Jay, 20 and Alto are the values,

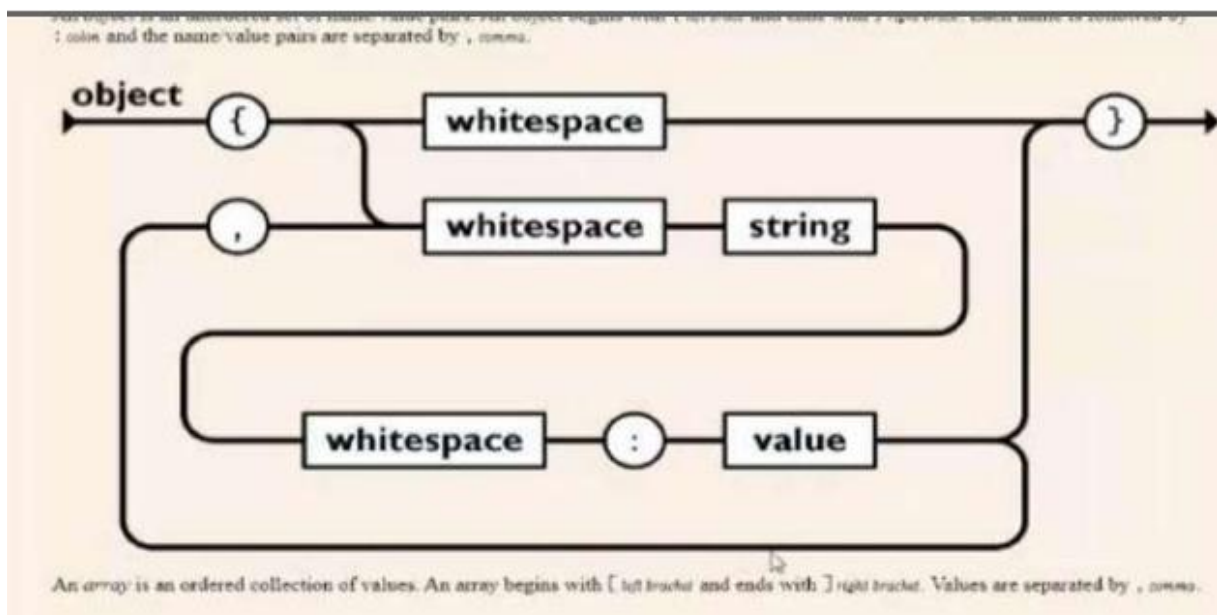


Figure 3.1 Json Structure

Json structure initiates and terminates by '{' and '}' and string and value is separated by ':'. Each pair is separated by ','. In this json data the outer object has only two elements and value of second object has nested data objects which is contained by square brackets.

3.2 Intro to API

API is the acronym for **Application Programming Interface**, which is a software intermediary that allows two applications to communicate with each other in real time.

How Do API works?

An API is the messenger that will deliver client's request to the source and then delivers the response back to client.

3.3 Covid data analysis:

3.3.1 JSON formatted data:

```
"cases_time_series": [
  {
    "dailyconfirmed": "1",
    "dailydeceased": "0",
    "dailyrecovered": "0",
    "date": "30 January 2020",
    "dateymd": "2020-01-30",
    "totalconfirmed": "1",
    "totaldeceased": "0",
    "totalrecovered": "0"
  },
  {
    "dailyconfirmed": "0",
    "dailydeceased": "0",
    "dailyrecovered": "0",
    "date": "31 January 2020",
    "dateymd": "2020-01-31",
    "totalconfirmed": "1",
    "totaldeceased": "0",
    "totalrecovered": "0"
  },
  {.....} ]
```

First we have to import request library in which url handling related methods are included. `get()` method will get the data from inserted url address and that will be stored in url object. Then we have applied `json()` method to convert back that data into Json structure and stored in data variable and it can be printed so on. We can get all the keys of given data as per shown below .

3.3.2 Implementation


```

import requests as r
url=r.get("https://data.covid19india.org/data.json")
d=url.json()
# print(d)
count=1
m = 0
for i in d:
    y = 1
    print("key :- ",count,":",i)
    for x in d['cases_time_series']:
        a = int(x['dailyconfirmed'])
        # print(a)
        print(f"{y,x}")
        if a>m:
            m=a
            max=d['cases_time_series'][y-1]['date']
        y+=1
    print("-----")
    if(count==1):
        print(f"max number of cases = {m}")
        print(f"maximum cases on : {max}")
        print(f"Total number of keys :{y-1}")
    print("-----")
    count+=1

```

Output :

- (1, {'dailyconfirmed': '1', 'dailydeceased': '0', 'dailyrecovered': '0', 'date': '30 January 2020', 'dateymd': '2020-01-30', 'totalconfirmed': '1', 'totaldeceased': '0', 'totalrecovered': '0'})
- (2, {'dailyconfirmed': '0', 'dailydeceased': '0', 'dailyrecovered': '0', 'date': '31 January 2020', 'dateymd': '2020-01-31', 'totalconfirmed': '1', 'totaldeceased': '0', 'totalrecovered': '0'})
- (3, {'dailyconfirmed': '0', 'dailydeceased': '0', 'dailyrecovered': '0', 'date': '1 February 2020', 'dateymd': '2020-02-01', 'totalconfirmed': '1', 'totaldeceased': '0', 'totalrecovered': '0'})
- (4, {'dailyconfirmed': '1', 'dailydeceased': '0', 'dailyrecovered': '0', 'date': '2 February 2020', 'dateymd': '2020-02-02', 'totalconfirmed': '2', 'totaldeceased': '0', 'totalrecovered': '0'})
- (5, {'dailyconfirmed': '1', 'dailydeceased': '0', 'dailyrecovered': '0', 'date': '3 February 2020', 'dateymd': '2020-02-03', 'totalconfirmed': '3', 'totaldeceased': '0', 'totalrecovered': '0'})
- (6, {'dailyconfirmed': '0', 'dailydeceased': '0', 'dailyrecovered': '0', 'date': '4 February 2020', 'dateymd': '2020-02-04', 'totalconfirmed': '3', 'totaldeceased': '0', 'totalrecovered': '0'})
- (7, {'dailyconfirmed': '0', 'dailydeceased': '0', 'dailyrecovered': '0', 'date': '5 February 2020', 'dateymd': '2020-02-05', 'totalconfirmed': '3', 'totaldeceased': '0', 'totalrecovered': '0'})
- (8, {'dailyconfirmed': '0', 'dailydeceased': '0', 'dailyrecovered': '0', 'date': '6 February 2020', 'dateymd': '2020-02-06', 'totalconfirmed': '3', 'totaldeceased': '0', 'totalrecovered': '0'})
- (9, {'dailyconfirmed': '0', 'dailydeceased': '0', 'dailyrecovered': '0', 'date': '7 February 2020', 'dateymd': '2020-02-07', 'totalconfirmed': '3', 'totaldeceased': '0', 'totalrecovered': '0'})

3.4 Mutual Fund data analysis :

```

Jupyter Untitled1 Last Checkpoint: 10 minutes ago (autosaved)
File Edit View Insert Cell Kernel Widgets Help Notebook saved Trusted Python 3 (ipykernel)

In [3]: import requests as rq

In [4]: url = rq.get("https://api.coindesk.com/v1/bpi/currentprice.json")
data = url.json()

In [5]: print(data)

{'time': {'updated': 'Jun 21, 2022 07:10:00 UTC', 'updatedISO': '2022-06-21T07:10:00+00:00', 'updateduk': 'Jun 21, 2022 at 08:10 BST'}, 'disclaimer': 'This data was produced from the coinDesk Bitcoin Price Index (USD). Non-USD currency data converted using hourly conversion rate from openexchangerates.org', 'chartName': 'Bitcoin', 'bpi': {'USD': {'code': 'USD', 'symbol': '$', 'rate': '21,072.3787', 'description': 'United States Dollar', 'rate_float': 21072.3787}, 'GBP': {'code': 'GBP', 'symbol': 'pound', 'rate': '17,109.0646', 'description': 'British Pound Sterling', 'rate_float': 17109.0646}, 'EUR': {'code': 'EUR', 'symbol': 'euro', 'rate': '20,030.7710', 'description': 'Euro', 'rate_float': 20030.7710}}}

In [6]: for k in data:
print(k)

time
disclaimer
chartName
bpi

In [7]: print(f"Current Bitcoin Price : {data['bpi']['USD']['rate']}")

Current Bitcoin Price : 21,072.3787

In [ ]:

```

Figure 3.2 Mutual fund api task

You can use the Jupyter code editor to fetch the Crypto data from the URL link. In line 4, we make the request to get the URL of the Crypto data using JSON. In line 5, we print our data which looks like (key, value) pairs. In line 6, we get the specific data we want by making use of the key.

```

import requests as r
url=r.get("https://api.mfapi.in/mf")
d=url.json()
print(len(d))
for i in range(0,len(d)):
    print(f"scheme code:{d[i]['schemeCode']}")

find=int(input ("Enter shceme code "))
flag=0
for i in range(0, len(d)) :
    if find == d[i]["schemeCode"] :
        flag=1
        print(d[i]["schemeName"])
        break
if flag==0:
    print("Can't find")

```

Output :

```

Total Records : 43658
scheme code:100027
scheme code:100028
scheme code:100029
scheme code:100030

```

253248

scheme code:100031

scheme code:100032

scheme code:100033

scheme code:100034

scheme code:100034

scheme code:100035

.....

.....

scheme code:150549

scheme code:150550

scheme code:150551

Enter scheme code 150550

HDFC FMP 1406D August 2022 - IDCW Option

3.5 Inshorts News data analysis :

API link: inshortsapi.vercel.app

- How many main keys are there in this API? Extract and print all keys.
- How much news is available in this API?
- Print all news below format.
 1. News content, Author: AUTHOR NAME, DATE: Date of news
 2. News content, Author: AUTHOR NAME, DATE: Date of news
 3. News content, Author: AUTHOR NAME, DATE: Date of news

• Implementation :

```
import requests as r
url=r.get("https://inshortsapi.vercel.app/news?category=sports")
data=url.json()
cnt=1
print(f"Total number of main keys : {len(data)}")
for i in data:
    print(f"{cnt}. {i}")
    cnt+=1
print(f"Total News available : {len(data['data'])}\n")
cnt=1
for i in range(0,len(data['data'])):
    print(f"{cnt}. {data['data'][i]['content']}, Author: {data['data'][i]['author']}, DATE: {data['data'][i]['date']}")
    cnt+=1
```

Output :

Total number of main keys : 3

1. category
2. data
3. success

Total News available : 25

1. India and Pakistan had five fielders inside the 30-yard circle for their last two overs and three overs respectively in their Asia Cup match. Both sides failed to be in a position to bowl first ball of their final over by the scheduled end of innings time, and were penalised by the added restriction for slow over-rate, the ICC said., Author: Anmol Sharma, DATE: 29 Aug 2022,Monday
2. After India defeated Pakistan by five wickets in their Asia Cup 2022 match on Sunday, Virat Kohli gifted his signed jersey to Pakistani fast bowler Haris Rauf. Sharing a video, the BCCI wrote, "The match may be over but moments like these shine bright...A heartwarming gesture by @imVkohli as he hands over a signed jersey to Pakistan's Haris Rauf.", Author: Anmol Sharma, DATE: 29 Aug 2022,Monday
3. A video of Pakistan's 19-year-old pacer Naseem Shah crying and refusing to drink water while returning to dressing room following his spell against India in Asia Cup 2022 went viral. Naseem completed his spell against India despite being in pain due to cramps. He registered figures of 4-0-27-2, dismissing KL Rahul for a golden duck and Suryakumar Yadav for 18(18)., Author: Anmol Sharma, DATE: 29 Aug 2022,Monday
4. New Zealand cricketer Glenn Phillips, who represented Rajasthan Royals in 2021, got engaged to his girlfriend Kate Thoroughgood at the Hooker Valley Track, which is a popular walking trail in New Zealand. Sharing pictures on Instagram, Phillips wrote, "She said YES!...looking forward to forever Mouse @katevictoria21." Phillips has represented New Zealand 50 times across formats., Author: Anmol Sharma, DATE: 29 Aug 2022,Monday
-
-
23. BCCI President Sourav Ganguly praised Rohit Sharma-led India for showing a lot of composure in a tight situation against Pakistan at Asia Cup 2022. "Good result for India to start off," he tweeted. After India bowled out Pakistan for 147 in 19.5 overs, they chased the 148-run target in 19.4 overs. India won the group stage match by five wickets., Author: Anant Kaur, DATE: 29 Aug 2022,Monday
24. Talking about dismissing Babar Azam in the third over of Pakistan's innings, India pacer Bhuvneshwar Kumar said they knew that the dismissal doesn't mean that half of the opposition batters were out. "But yes, once he was out, we knew their plans will be disturbed as the batter who plays the role of the anchor was gone," he added., Author: Ankur Taliyan, DATE: 29 Aug 2022,Monday
25. Commentator Harsha Bhogle has said all-rounder Hardik Pandya has impressed him more with his bowling in recent matches. "Hardik was always going to be a very gifted, natural stroke-player...[but] he's squeezing every bit of the bowler out of him and that's what makes it so admirable," tweeted Bhogle. Hardik picked three wickets while conceding 25 runs against Pakistan. , Author: Ankur Taliyan, DATE: 29 Aug 2022,Monday

Process finished with exit code 0

3.6 ISRO data analysis :

API link(1): isro.vercel.app/spacecrafts

- Print all main keys and total number of main keys as well.
- Print all spacecraft names as per the below output format.
- Allow the user to enter the name of the spacecraft and print spacecraft is found or spacecraft not found.

● Implementation :

```
import requests as r
url = r.get("https://isro.vercel.app/api/spacecrafts")
data = url.json()
cnt = 1
print(f"Total Number of main keys : {len(data)}")
for i in data:
    print(f"{cnt}. {i}")
    cnt=cnt+1
cnt = 1

for i in range(0,len(data['spacecrafts'])):
    print(f"{cnt}. {data['spacecrafts'][i]['name']}")
    cnt=cnt+1
print("\n")
ip = input("Enter name of Spacecraft: ")
cnt = 1
fnd = 0
for i in range(0,len(data['spacecrafts'])):
    if data['spacecrafts'][i]['name'] == ip:
        fnd=1
        break
    else:
        fnd = 0
    cnt=cnt+1
if fnd == 1:
    print("spacecraft found")
else:
    print("spacecraft does not found!")
```

Output :

Total Number of main keys : 1

1. spacecrafts

1. Aryabhata

2. Bhaskara-I

3. Rohini Technology Payload (RTP)

.....

.....

110. EOS-01

111. CMS-01

Enter name of Spacecraft: EMISAT

spacecraft found

API link: isro.vercel.app/customer_satellites

- Try to generate a Pie chart indicating the percentage of ISRO's own API vs. Customer satellites from the above API(s).
- For analysis of domestic vs. foreign customer involvement. (Data and APIs given in the above questions are demo purpose only, generated and maintained by respective authorities)
- **Implementation :**

```
import requests as r
from matplotlib import pyplot as plt
own = r.get("https://isro.vercel.app/api/spacecrafts")
cust = r.get("https://isro.vercel.app/api/customer_satellites")
owndata = own.json()
custdata = cust.json()
total_spacecrafts = len(owndata['spacecrafts']) +
len(custdata['customer_satellites'])
ownpercentage =
round((len(owndata['spacecrafts'])*100)/(total_spacecrafts),2)
custpercentage =
round((len(custdata['customer_satellites'])*100)/(total_spacecrafts),2)
percentages = [ownpercentage, custpercentage]
labels= [f"ISRO'S own
spacecrafts({ownpercentage}%)",f"Customer_Satellites
({custpercentage}%)"]
colors = ['#eb6e15', '#8a84cc']
plt.pie(percentages, labels=labels, startangle=90, shadow=True, colors =
colors)
plt.show()
```

- **OUTPUT:**

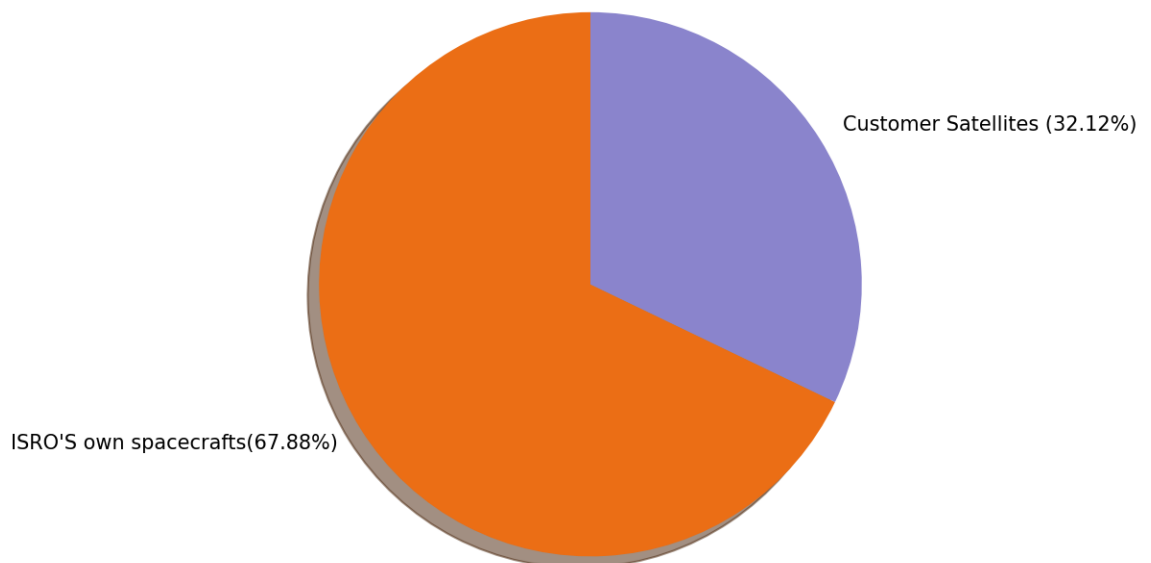


Figure 3.3 ISRO data pie chart

This was all about how we can handle data and extract meaningful insights from it. Now let see how data from Excel file can be extracted and how analytics can be implemented. The xlrd library is very useful while working with excel files.

3.7 IPL data excel file :

This excel file contains IPL2022 data. Suppose we want to get an idea of highest runs.

	A	B	C	D	E	F
1	CRICKET	BUTTLER	RAHUL	DEKOCK	HARDIK	MILLER
2		1	35	0	7	33
3		2	100	40	61	31
4		3	71	68	1	27
5		4	12	24	80	50
6		5	54	0	39	87
7		6	103	103	24	67
8		7	116	30	3	10
9		8	8	103	10	3
10		9	67	6	46	1
11		10	22	77	23	24
12		11	30	0	50	11
13		12	7	8	11	7
14		13	2	10	7	62
15		14	2	68	140	40

Figure 3.4 Excel file of IPL data



Figure 3.5 Code snippet of ipl data

For this, we can first store the file to a local variable and open it through the workbook and print the cell value data. The cell values data is present in the form of columns and rows. In this way, you can print a data of particular column. With the help of NumPy and Pandas library data can be sliced, aggregate functions can be performed and many operations can be done.

3.8 University Acquisitor

- API links:

- universities.hipolabs.com/uk
- universities.hipolabs.com/us

- universities.hipolabs.com/india
- universities.hipolabs.com/pakistan

- Generate Bar graph of Countries VS Total Universities.
- Develop a search mechanism :
 - Allow user to insert a country name
 - Allow user to insert a university name. If university is found - display the official link of that university to user. Otherwise not found

3.8.1 Implementation

```
import requests as rq
from matplotlib import pyplot as plt

uk = rq.get("http://universities.hipolabs.com/search?country=United+Kingdom")
us = rq.get("http://universities.hipolabs.com/search?country=United+States")
ind = rq.get("http://universities.hipolabs.com/search?country=india")
pk = rq.get("http://universities.hipolabs.com/search?country=pakistan")
ukdata = uk.json()
usdata = us.json()
inddata = ind.json()
pkdata = pk.json()

countries = ["United Kingdom", "United States", "India", "Pakistan"]
universities = [len(ukdata), len(usdata), len(inddata), len(pkdata)]

plt.bar(countries, universities, color='maroon',
        width = 0.5)
plt.show()

country = input("Enter Country name : ")
university = input("Enter University name : ")

if country.lower() == "united kingdom":
    fnd = 0
    for i in range(0, len(ukdata)):
        if ukdata[i]['name'].lower() == university.lower():
            print(f"University found. Visit {ukdata[i]['web_pages'][0]}")
            fnd = 1
            break
    else:
        fnd = 0
    if fnd == 0:
        print("Not found!")

elif country.lower() == "united states":
    fnd = 0
    for i in range(0, len(usdata)):
        if usdata[i]['name'].lower() == university.lower():
            print(f"University found. Visit {usdata[i]['web_pages'][0]}")
            fnd = 1
            break
    else:
        fnd = 0
    if fnd == 0:
        print("Not found!")
```



```

elif country.lower() == "india":
    fnd = 0
    for i in range(0, len(inddata)):
        if inddata[i]['name'].lower() == university.lower():
            print(f"University found. Visit
{inddata[i]['web_pages'][0]}")
            fnd = 1
            break
        else:
            fnd = 0
    if fnd == 0:
        print("Not found!")

elif country.lower() == "pakistan":
    fnd = 0
    for i in range(0, len(pkdata)):
        if pkdata[i]['name'].lower() == university.lower():
            print(f"University found. Visit
{pkdata[i]['web_pages'][0]}")
            fnd = 1
            break
        else:
            fnd = 0
    if fnd == 0:
        print("Not found!")

else:
    print('Country Not Found!')

```

Output :

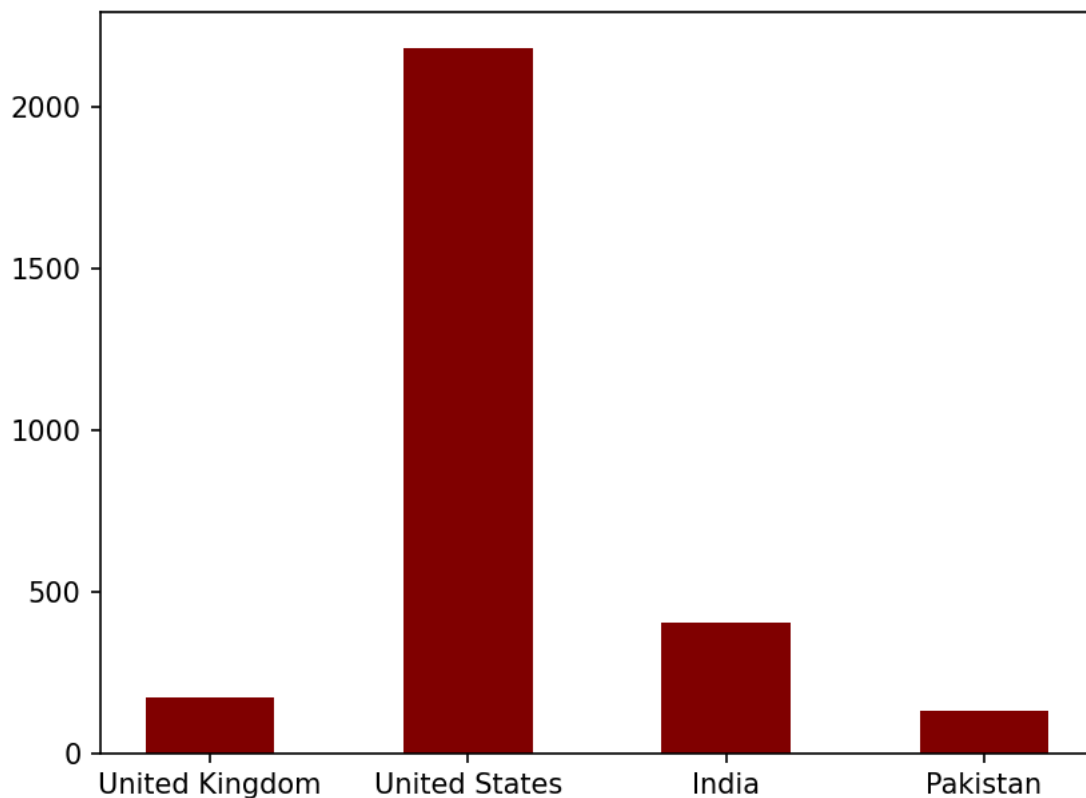


Figure 3.6 University acquiritor bar graph

```
C:\python39\python.exe "D:/internship report/Infoc
Enter Country name : India
Enter University name : Amravati University
University found. Visit http://www.amtuni.com/
```

Figure 3.7 University Acquisitor Output

CHAPTER : 4 BASICS OF MACHINE LEARNING

Machine learning is a branch of artificial intelligence (AI) and computer science which focuses on the use of data and algorithms to imitate the way that humans learn, gradually improving its accuracy.

In machine learning, datasets are split into two subsets. The first subset is known as the training data - it's a portion of our actual dataset that is fed into the machine learning model to discover and learn patterns. In this way, it trains our model. The other subset is known as the testing data. Generally we split in way so that the portion of testing data is more than of training data.

- These are three types of machine learning:
 - i. Supervised learning
 - ii. Unsupervised learning
 - iii. Reinforcement learning

4.1 Commonly used machine learning algorithms:

- ◆ Linear Regression
- ◆ Logistic Regression
- ◆ Decision Tree
- ◆ SVM Algorithm
- ◆ Naive Bayes Algorithm
- ◆ KNN Algorithm
- ◆ K-Means Algorithm
- ◆ Random Forest Algorithm

4.2 SUPERVISED MACHINE LEARNING:

Supervised machine learning, is a subcategory of machine learning and artificial intelligence. It is defined by its use of labelled datasets to train algorithms that classify data or predict outcomes accurately.

There are two types of Supervised Learning:

- ◆Regression
- ◆Classification

4.3 UNSUPERVISED MACHINE LEARNING:

Unsupervised learning is a type of machine learning in which models are trained using an unlabeled dataset and are allowed to act on that data without any supervision.

The goal of unsupervised learning is to find the underlying structure of the dataset, group that data according to similarities, and represent that dataset in a compressed format.

Types of Unsupervised Learning Algorithms:

- ◆Clustering
- ◆Association

4.4 REINFORCEMENT LEARNING:

It is about taking suitable action to maximize reward in a particular situation. It is employed by various software and machines to find the best possible behaviour or path it should take in a specific situation.

4.5 Terminologies of Machine Learning:

4.5.1Model: A model is a specific representation learned from data by applying some machine learning algorithm. A model is also called hypothesis.

4.5.2Feature: A feature is an individual measurable property of our data. A set of numeric features can be conveniently described by a feature vector. Feature vectors are fed as input to the model. For example, in order to predict a fruit, there may be features like color, smell, taste, etc. Choosing informative, discriminating and independent features is a crucial step for effective algorithms. We generally employ a feature extractor to extract the relevant features from the raw data.

4.5.3Target (Label): A target variable or label is the value to be predicted by our model. For the fruit example discussed in the features section, the label with each set of input would be the name of the fruit like apple, orange, banana, etc.

4.5.4Training: The idea is to give a set of inputs(features) and it's expected outputs(labels), so after training, we will have a model (hypothesis) that will then map new data to one of the categories trained on.

4.5.5 Prediction: Once our model is ready, it can be fed a set of inputs to which it will provide a predicted output(label). But make sure if the machine performs well on unseen data, then only we can say the machine performs well.

4.6 SCIKIT-LEARN:

Scikit-learn is a free machine learning library for Python. It features various algorithms like support vector machine, random forests, and k-neighbours, and it also supports Python numerical and scientific libraries like NumPy and SciPy.

4.7 Linear Regression

	A	B	C
1	area	price	
2	800	5180000	
3	1100	7190000	
4	1500	9820000	
5	1800	12100000	
6	2200	14900000	
7			

Figure 4.1 price.csv file

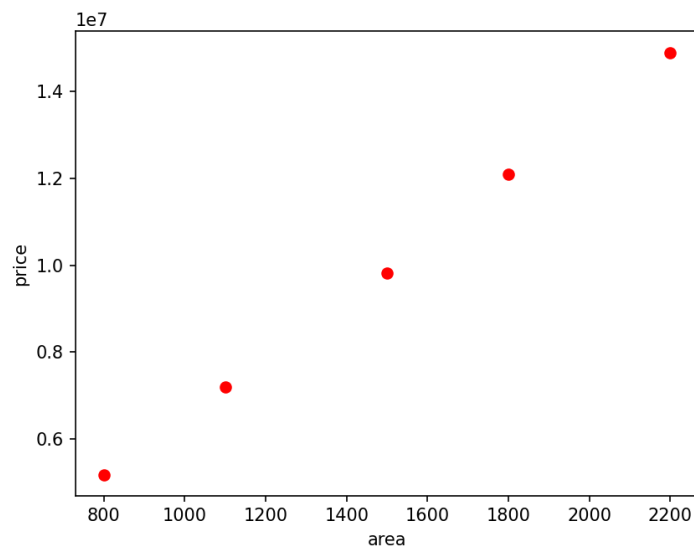
The data provided is about the area and price of the land. We will find the relationship between the area and the price. Here, we will assume that price is linearly dependent on area. So, area is independent variable and price is dependent one in this data. We will fit this data into Linear Regression model and our model will be trained after observing these five entries and then we will get prediction of price for any unknown area value. We will also plot a scatter plot taking area on X axis and price on Y axis.

4.7.1 Implementation

```
import pandas as pd
import matplotlib.pyplot as plt
from sklearn import linear_model

df = pd.read_csv('C:\\Users\\JAY
MODI\\PycharmProjects\\infolabz_intern\\prices.csv')
print(df)
plt.xlabel('area')
plt.ylabel('price')
plt.scatter(df.area,df.price,color="red")
plt.show()

reg = linear_model.LinearRegression()
reg.fit(df[['area']],df.price)
print(f"Predicted value of price: ",reg.predict([[int(input("Enter the
value of area: "))]]))
#y=b0+b1x
print(f"Value of coefficient: ",reg.coef_)      # b1
print(f"value of intercept: ",reg.intercept_)  # b0
print("value of Y: ",reg.intercept_ + reg.coef_ * 3300)
```

Output :*Figure 4.2 Scatter Plot*

```
C:\python39\python.exe "C:/Users/JAY MODI/Pyth
area    price
0    800    5180000
1   1100    7190000
2   1500    9820000
3   1800   12100000
4   2200   14900000
Enter the value of area: 2000
Predicted value of price: [13455048.85993485]
Value of coefficient: [6955.86319218]
value of intercept: -456677.5244299695
value of Y: [22497671.00977199]
```

*Figure 4.3 Linear Regression output***4.8 Multiple Linear Regression**

In the given excel sheet, the data contains multiple variables such as: users, orders, age and amount. The data is about x_1 users has total x_2 orders which are x_3 aged and total amount is x_4 . The variables users, orders and age are independent variables and amount is dependent on users, orders, age. So, other three variables are feature set of the amount variable in this case.

	A	B	C	D	E
1	users	orders	age	amount	
2	2000	4500	35	450000	
3	2865	5705	32	713125	
4	3175	6475	30	971250	
5	3500	7750	28	1240000	
6	4200	9257	26	1943970	
7	5000	13450	24	3497000	

Figure 4.4 ordersdata.csv file

We will analyse the above data through the multiple linear regression method. We will first import linear model from sklearn library. Then we will apply LinearRegression() method from that. We have fitted our three independent variable in X label combined and dependent variable in Y label with fit() method. We can make to model to predict the value using predict() method. And that's it .

4.8.1 Implementation:

```
import pandas as pd
from sklearn import linear_model

df = pd.read_csv('orderdata.csv')
print(df)

reg = linear_model.LinearRegression()
reg.fit(df[['users', 'orders', 'age']], df.amount)
print(f"Predicted amount is :", reg.predict([[int(input("Enter number of
users: ")), int(input("enter number of orders: ")), int(input("Enter age :
"))]]))
```

OUTPUT :

```
C:\python39\python.exe "C:/Users/JAY MOD
users  orders  age  amount
0   2000   4500   35   450000
1   2865   5705   32   713125
2   3175   6475   30   971250
3   3500   7750   28  1240000
4   4200   9257   26  1943970
5   5000  13450   24  3497000
Enter number of users: 4300
enter number of orders: 3200
Enter age : 32
Predicted amount is : [219702.71558204]
```

Figure 4.5 Multiple linear model output

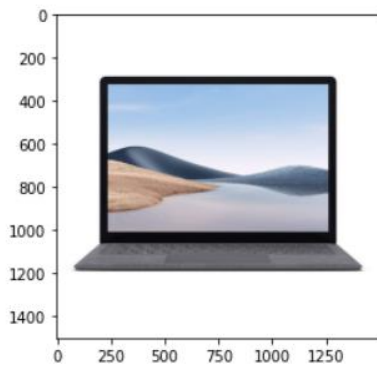
4.9 IMAGE PROCESSING

Now , let see how we can use Tensorflow and OpenCv library in Machine Learning. Both are used to process the image data. They can be used to detect static images and live webcam videos such as monitoring the webcam of the candidate giving the online exam. The data obtained by OpenCV can be shown by graphs using matplotlib.

```
In [1]: import tensorflow as tf
from tensorflow.keras.preprocessing import image
import matplotlib.pyplot as plt
```

```
In [2]: img = image.load_img("training/laptop/1.jpg")
plt.imshow(img)
```

Out[2]: <matplotlib.image.AxesImage at 0x1e23ec768e0>



```
In [4]: import cv2
cv2.imread("training/laptop/1.jpg").shape
```

Out[4]: (1500, 1500, 3)

Figure 4.6 Image processing example(1)

```
In [6]: import os
dir_path = "training/laptop"
for i in os.listdir(dir_path):
    img = image.load_img(dir_path+"/"+i, target_size=(250,250))
    plt.imshow(img)
    plt.show()
```

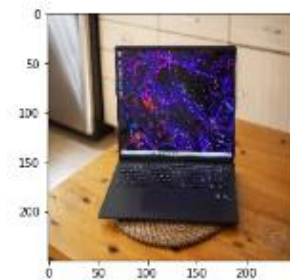
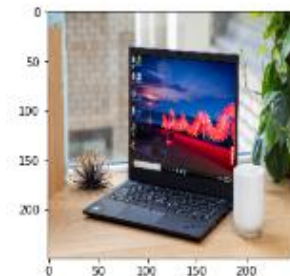
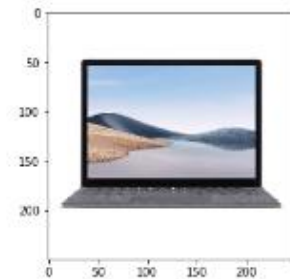


Figure 4.7 Image processing example(2)

4.10 Polynomial Regression

	A	B	C
1	orders	id	amount
2	daily_orders	1	44500
3	half weekly orders	2	127000
4	weekly orders	3	304500
5	half monthly orders	4	667500
6	monthly orders	5	1350000
7	half quarterly order	6	2047500
8	quarterly orders	7	4041000

Figure 4.8 Polynomial Regression Excel Data File

- Learn polynomial regression and implement polynomial regression for orders data (csv sheet given - orders.csv)
- Generate scatterplot of id and amount.
- Predict half yearly amount (id=8) and Yearly amount (id=9) of business using polynomial regression.

4.10.1 Implementation

```
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
from sklearn import linear_model, preprocessing

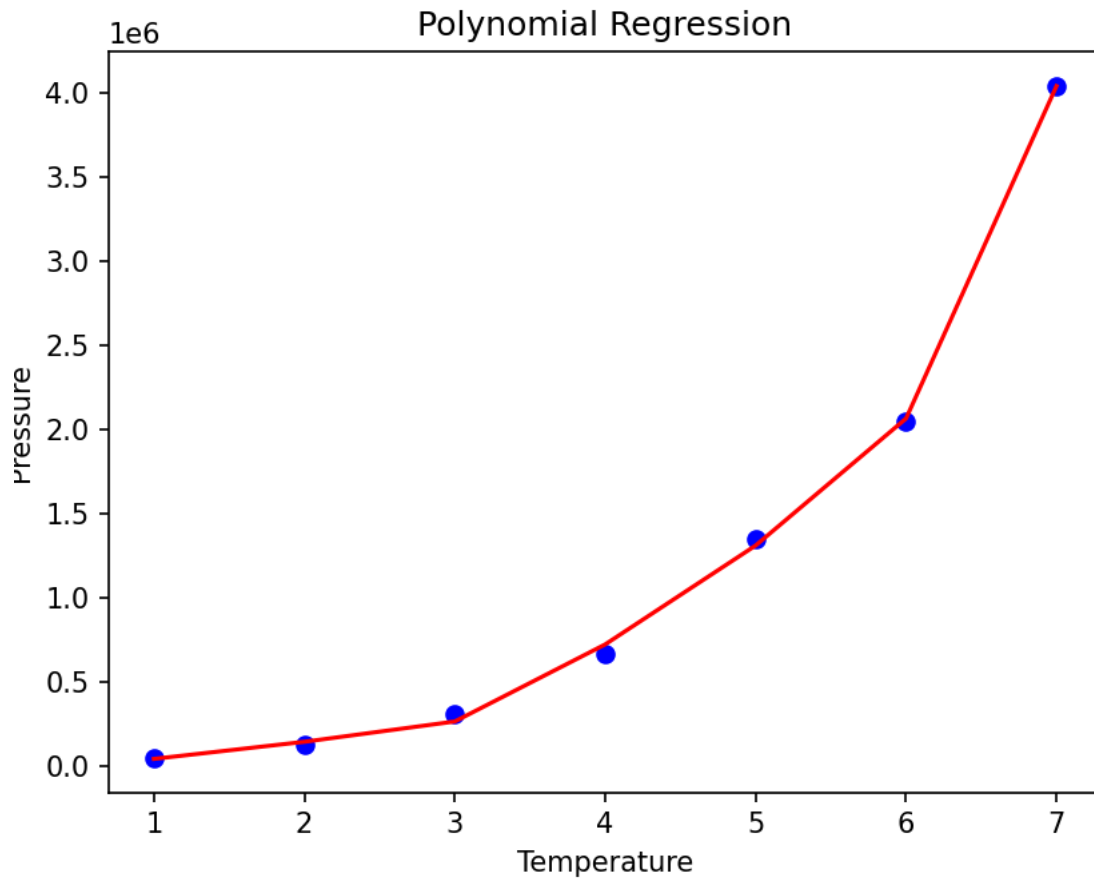
Dataset = pd.read_csv("orders.csv")

x = Dataset.iloc[:, 1].values
X = x[:, np.newaxis]
Y = Dataset.iloc[:, 2].values

poly = preprocessing.PolynomialFeatures(degree=5)
x_poly= poly.fit_transform(X)
linear = linear_model.LinearRegression()
linear.fit(x_poly, Y)
pred = linear.predict(x_poly)

plt.scatter(X, Y, color='blue')
plt.plot(X, pred, color='red')
plt.title('Polynomial Regression')
plt.xlabel('Temperature')
plt.ylabel('Pressure')
plt.show()

print(f"Predicted Amount for id=8(half yearly orders) :
{linear.predict(poly.fit_transform([[8]]))}")
print(f"Predicted Amount for id=9(yearly orders) :
{linear.predict(poly.fit_transform([[9]]))}")
```

Output :*Figure 4.9 Polynomial Regression Line chart*

```
C:\python39\python.exe "D:/internship report/InfoLabz/InfoLabz/assignment_2/polynomial.py"  
Predicted Amount for id=8(half yearly orders) : [10073000.00003654]  
Predicted Amount for id=9(yearly orders) : [25566000.00014979]
```

*Figure 4.10 Polynomial Regression Result***4.11 Machine through API**

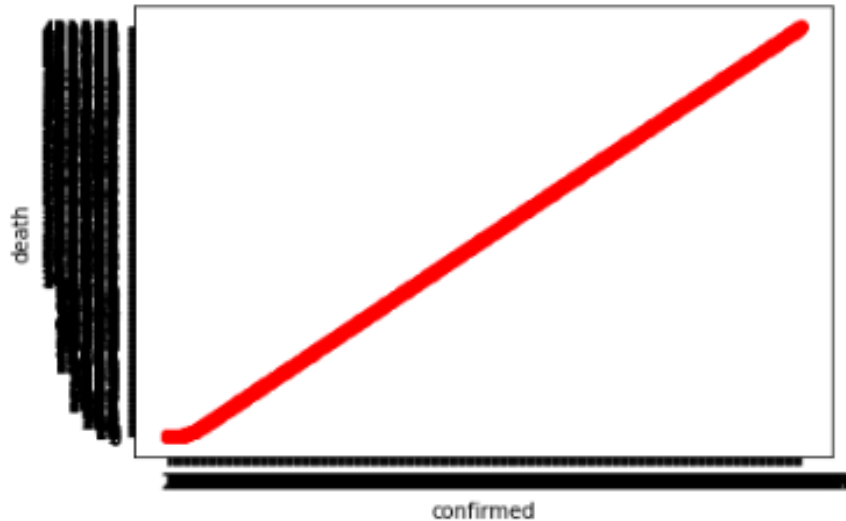
34

GEC Gandhinagar

Figure 4.8 API to Machine Learning

```
In [6]: import matplotlib.pyplot as plt
from sklearn import linear_model

plt.xlabel('confirmed')
plt.ylabel('death')
plt.scatter(df.totalconfirmed,df.totaldeceased,color="red")
plt.show()
```



```
In [11]: reg = linear_model.LinearRegression()
reg.fit(df[['totalconfirmed']],df[['totaldeceased']])
print(reg.predict([[32520545]]))

[[415140.8805548]]
```

Figure 4.11 Machine Learning through API result

CONCLUSION:

Data science and Machine learning are the scenario changing fields of the market in present era. Currently, data is the power and all the market leading companies are making their next business plans according to that. But, raw data is meaningless and well analysed , processed data is scenario changing information. And data analytics is all about finding out fabulous patterns from the marvellous amount of data. Further, imposing these pattern and insights into a model which can be trained by it and can make prediction of any unknown situation is all about Machine Learning. In current situation ML has awesome applications in various spectrum of fields. Like: Health Care, Finance, Real Estate, Education, Retail Market. In this way ML can be a significantly environment changing technology.

REFERENCES:

- [GeeksforGeeks](#)
- [AnalyticsVidhya](#) [medium](#)
- [kaggle](#)
- [YouTube](#)
- [Pandas](#)
- [Scikit-Learn](#)
- [Python](#)