

INTERNSHIP AT INFOLABZ IT SERVICES

AN INTERNSHIP REPORT

Submitted by

Parth Rajeshbhai Oza

190390107021

In partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

In

Computer Engineering

S.P.B. Patel Engineering College, Mehsana



**S.P.B. PATEL
ENGINEERING COLLEGE
SAFFRONY INSTITUTE OF TECHNOLOGY**



Gujarat Technological University, Ahmedabad

May, 2023



S.P.B. Patel Engineering College

Near Shanku's Water Park, Ahmedabad – Mehsana Highway, Linch, Gujarat

CERTIFICATE

This is to certify that the project report submitted along with the project entitled Internship at INFOLABZ, AHMEDABAD has been carried out by Parth Rajeshbhai Oza under my guidance in partial fulfilment for the degree of Bachelor of Engineering in Computer Engineering, 8th Semester of Gujarat Technological University, Ahmedabad during the academic year 2022-23.

Sign

Prof. Chetan Chauhan

Internal Guide

Sign

Prof. Akshay Kansara

Head of Department

Company Certificate



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



Date: 25 / 04 / 2023

TO WHOM IT MAY CONCERN

This is to certify that Parth Rajeshbhai Oza has successfully completed his internship in the field of Machine Learning from 23 January 2023 to 15 April 2023 (Total number of months: 12) under the guidance of Mr. Chintan Nagrecha.

His internship activities include work of data analysis, data visualization and machine learning. He had given his inputs in company's projects on machine learning and also helped as teaching assistant.

During the period of his internship program with us, he had been exposed to different processes and was found diligent, hardworking and inquisitive.

We wish him ~~success~~ success in his life and career.



Ms. Zarna Shah,
Human Resources Department,
INOLABZ IT SERVICES PVT. LTD.



+91 8866662662
+91 8141236662



contact@infolabz.in
www.infolabz.in



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



S.P.B. PATEL
ENGINEERING COLLEGE
SAFFRONY INSTITUTE OF TECHNOLOGY



S.P.B. Patel Engineering College, Mehsana

Near Shanku's Water Park, Ahmedabad – Mehsana Highway, Linch, Gujarat

DECLARATION

We hereby declare that the Internship report submitted along with the Project Internship at INFOLABZ, AHMEDABAD submitted in partial fulfillment for the degree of Bachelor of Engineering in Computer Engineering to Gujarat Technological University, Ahmedabad, is a bonafide record of original project work carried out by me under the supervision of Prof. Chetan Chauhan & Mr.Chintan Nagrecha (External Guide) and that no part of this report has been directly copied from any students' reports or taken from any other source, without providing due reference.

Name of the Student

Sign of Student

Parth Rajeshbhai Oza

ACKNOWLEDGMENT

I really believe that behind every record and success of student, not only hard work and perfection is there, also there is inspirations and guidance of the teachers and the persons who support them until Training has been done. It gives us great pleasure in submitting this in plant Internship report of my training experience which I gained from 23rd January 2023 to 12th April 2023 at “Infolabz IT Services Pvt Ltd” located at Ahmedabad, Gujarat.

I would like to express our heart full gratitude to Mr. Chintan Nagrecha who granted me the permission for obtaining training at Zenkins Technologies Pvt. Ltd. We are very thankful to Mr. Zarna Shah(HR) for giving us a valuable chance for training at Infolabz IT Services Pvt Ltd.

So, I would like to convey my gratitude towards the all-Technical Staff of Infolabz IT Services Pvt Ltd for giving me technical and practical knowledge of the development and helping me via developing the bridge between theoretical knowledge and its practical application.

Abstract

The word engineering itself means to develop something new by knowing the basis of fundamental theories and principles engineering concepts are always powered by strong theoretical concepts.

Theory scrutinizes all the ideal situations. But it is rightly said that theory is incomplete without practice and without theory. It is very important to relate ideal situation with real ones so as to understand how a real system works. Industrial training is the best platform where the so called Engineering student can sharpen his analytical as well as practical skills so as to become the best of the best promising engineer for the future world. It usually helps to clear all his theoretical concepts and can learn how to apply it.

As a part of our academics, we were placed to train at “Infolabz IT Services Pvt Ltd” located at Ahmedabad, Gujarat.

List of Figures

Fig 1.1 Company Logo.....	1
Fig 1.2 Organizational chart.....	2
Fig 3.1 Opportunities chart.....	10
Fig 4.1.1 Data sat of the isro api.....	11
Fig 4.1.2 ISRO and Foreign satellites.....	11
Fig 4.2 Script for finding website of university.....	12
Fig 4.3 Time vs price action.....	13
Fig 4.4.1 Fetching data of cricket.....	13
Fig 4.4.2 Analysis.....	14
Fig 4.4.3 Bar Visualization.....	15
Fig 4.5.1 Covid data fitted into regression model.....	15
Fig 4.5.2 Covid data analysis.....	16
Fig 4.5.3 Daily confirmed cases.....	16
Fig 4.5.4 Cases state wise.....	17
Fig 4.5.5 case on input date.....	17
Fig 4.6.1 sample data.....	18
Fig 4.6.2 Inner join.....	18
Fig 4.6.3 Left join.....	19
Fig 5.1 Basic functionalities.....	20
Fig 5.2 Shape of data.....	21
Fig 5.3 Visualization of popular products.....	22
Fig 5.4 Ratings.....	22
Fig 5.5 Utility matrix.....	23
Fig 5.6 Utility matrix shape.....	23
Fig 5.7 Transposing matrix.....	24
Fig 5.8 Correlation matrix.....	25
Fig 5.9 Random product.....	25
Fig 5.10 Correlation matrix.....	26
Fig 5.11 Suggestion Products.....	27

Table of Contents

Acknowledgement.....	i
Abstract.....	ii
List of Figures.....	iii
List of Tables	iv
List of Abbreviations	v
Table of Contents.....	vi
Chapter 1 Introduction.....	1
1.1 Company Profile.....	2
1.2 Scope of Work.....	3
1.3 Organization chart.....	3
1.4 Capacity of Company.....	3
Chapter 2 Overview of Department.....	4
2.1 Various Development Departments.....	4
2.2 Work Being Carried out at Company.....	5
Chapter 3 Internship and Project.....	6
3.1 Internship Summary.....	6
3.2 Purpose	6
3.3 Objective	6
3.4 Tools and Technology	6
3.5 Implementation Platform.....	8
3.5.1 Pycharm.....	8
3.5.2 Jupyter.....	8
3.5.3 Tableau.....	8
3.5.4 Powerbi.....	8
3.6 Why Python.....	10
Chapter 4 Assignments.....	15
4.1 Mini Project 1.....	15
4.2 Mini Project 2.....	17
4.3 plot time vs price action.....	17
4.4 Mini Project 3.....	18

4.5 plot time vs price action.....	20
Chapter 5 Project.....	23
5.1 Importing Libraries and loading data.....	23
5.2 Preparing Popular products.....	24
5.3 Visualizing using bar chart.....	25
5.4 Amazon ratings.....	26
5.5 Model based collaborative filtering system.....	27
5.6 Utility Matrix.....	27
5.7 Transposing the matrix.....	28
Chapter 6 Conclusion	
6.1 Summary.....	30
6.2 Date of continuous Evaluation.....	30
Chapter 7 References.....	31
Appendix.....	32

CHAPTER 1. INTRODUCTION

1.1 COMPANY PROFILE:

Company Name: InfoLabz IT Services Pvt. Ltd

Address: 405 Vraj Avenue, Above SAM'S Pizza, Nr Commerce Six Road, Navrangpura Ahmedabad, 380009

Email: info@infolabz.in



Fig 1.1 Company Logo

Fig 1.1 Company Logo

Infolabz is an Indian information technology services, IT consulting, business services, and software development company headquartered in Ahmedabad India.

Since 2018, Infolabz is present on the international market as a supplier of innovative solutions, adding business value through technology implementation at a professional level. Focusing our activities on developing complex software products, Infolabz Technologies provides consistent results in emerging fields such as mobile and web-based custom business applications, e-commerce, e-payment, e-security, e-health, and enterprise & resource planning, entertainment, and the finance industry.

1.2 SCOPE OF WORK:

They offer website development and maintenance, social media marketing, API development, and Artificial Intelligence solutions as well as provides cross-platform mobile application development solutions using different frameworks.

1.3 ORGANIZATION CHART:

Head of Company(CEO) provides the tasks to the Head of Department by showing their work and their flexibility towards the job. Next, the Head of Department assigns the duty to the project managers who will guide the project throughout the time. Then project managers will provide guidance to the employee about how to achieve goal of the task.

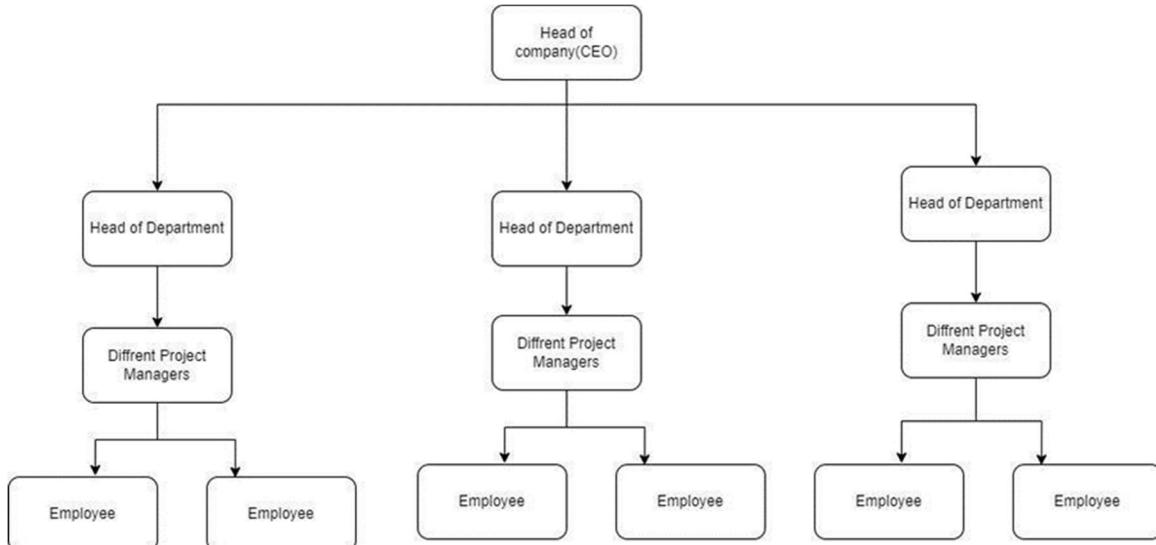


Fig 1.2 Organizational chart

1.4 CAPACITY OF COMPANY:

More than 15 employees work in this company and about 20 interns work in the company. Generally used to hire dedicated software developers and support engineers.

CHAPTER 2: OVERVIEW OF DEPARTMENT

2.1 VARIOUS DEVELOPMENT DEPARTMENTS:

Data Science Development Department:

Data scientists collaborate closely with business leaders and other key players to comprehend company objectives and identify data-driven strategies for achieving those objectives. A data scientist's job is to gather a large amount of data, analyze it, separate out the essential information, and then utilize tools like SAS, R programming, Python, etc. to extract insights that may be used to increase the productivity and efficiency of the business. Depending on an organization's needs, data scientists have a wide range of roles and responsibilities.

Website Development Department:

A web developer is a tech-savvy role that requires expertise in web design and involves translating customer needs into interactive web apps with visual elements that users see and interact with. A front-end developer is responsible for ensuring that the application is optimized both in technology and aesthetics while delivering the best user experience on desktop and mobile. Most frontend frameworks use JavaScript open-source programs. Two popular frameworks, Angular and React are examples. Front-end development might also use extension languages like SASS, which can extend the functionality of an existing CS

Backend development Department:

Backend development refers to the creation and maintenance of the back end of a website.

There are three components to the backend of a website

- 1) Servers share data and resources, distribute work and process computations.
- 2) An application is computer software designed to help the user perform specific tasks.
- 3) A database is used to store and organize data.

Backend development typically relies a little more heavily on frameworks and libraries because they help developers create web applications that can use a programming language to communicate with the database and generate the final content effectively. For backend work, many popular frameworks adopt a model view controller (MVC) pattern that separates application concerns into clearly defined layers. Popular examples of MVC frameworks for backend development are Django, ASP.NET, MVC, and Ruby on the Rails.

2.2 WORK BEING CARRIED OUT AT COMPANY:

Our company has a well-coordinated team and work like a well-oiled machine. In fact, we work on a human scale and build long lasting businesses with our clients. Our firmness affords us the required room and flexibility to manage our future and independence. Apart from this, it also provides us a strong platform to protect our ethics and core values.

Our company works on several domains for Android Apps, iOS Apps as well as Cross Platform Apps. It also works on web development platforms.

CHAPTER 3 : INTERNSHIP AND PROJECT

3.1 INTERNSHIP SUMMARY:

This Project is about how to get the developed recommendation system that will help businesses improve their shopper's experience on websites and result in better customer acquisition and retention.

3.2 PURPOSE:

Industry experience is often an important part of applying for full time positions. Gaining experience through internship can be helpful for our future scope. Similarly, my purpose was to gain experience and decide my career path so that I can have a clear path towards my goals. Also, I wanted to develop new skills and build a network with professionals.

3.3 OBJECTIVE:

A motivated individual with in-depth knowledge of languages and development tools, seeking a position in a growth-oriented company where I can use my skills to the advantage of the company while having the scope to develop my own skills.

To work as a data science engineer applying my knowledge in the field of data science, machine learning, and data analytics to cater to the specific needs of the people. I wish to work in a team of motivated individuals who wish to work towards the advancement of the company.

3.4 TOOLS AND TECHNOLOGY:

I used various Languages and technologies in this Internship which I mentioned below with the functions of each tool and library:

- 1) C: c is an imperative procedural language, supporting structured programming, lexical variable scope and recursion, with a static type system. It was designed to be compiled to provide low-level access to memory and language constructs that map efficiently to machine instructions, all with minimal runtime support.
- 2) Python: Python is a computer programming language often used to build websites and software, automate tasks, and conduct data analysis. Python is a general-purpose language, meaning it can be used to create a variety of different programs and isn't specialized for any specific problems.

Some things include:

- Data analysis and machine learning
- Web development
- Automation or scripting
- Software testing and prototyping
- Everyday tasks

3) Data analysis and machine learning: Python has become a staple in data science, allowing data analysts and other professionals to use the language to conduct complex statistical calculations, create data visualizations, build machine learning algorithms, manipulate and analyze data, and complete other data-related tasks.

Python can build a wide range of different data visualizations, like line and bar graphs, pie charts, histograms, and 3D plots. Python also has a number of libraries that enable coders to write programs for data analysis and machine learning more quickly and efficiently, like TensorFlow and Keras.

4) Automation or scripting : If you find yourself performing a task repeatedly, you could work more efficiently by automating it with Python. Writing code used to build these automated processes is called scripting. In the coding world, automation can be used to check for errors across multiple files, convert files, execute simple math, and remove duplicates in data.

Python can even be used by relative beginners to automate simple tasks on the computer—such as renaming files, finding and downloading online content or sending emails or texts at desired intervals.

3.5 IMPLEMENTATION PLATFORM:

3.5.1 Pycharm:

PyCharm is a dedicated IDE tool focused on providing a complete solution for creating full-fledged packages and software in Python, including classes and graphical user interfaces (GUIs). It also excels in complex environments where multiple scripts interact with each other and need to be managed.

PyCharm's most popular features include a built-in debugger and smart auto-complete as well as DevOps tools, such as version control, which makes it ideal for developers and software engineers.

3.5.2 Jupyter:

Jupyter is a browser-based open-source data science notebook tool that supports Python; Julia; and other dynamic programming languages such as R, Scilab and Octave. Focused on scripts and accompanying documentation, Jupyter is ideal for data scientists who need a way to create quick data visualizations. However, source code is stored as HTML and readable by Jupyter rather than Python.

3.5.3 Tableau:

Tableau is a data visualization and business intelligence tool that enables users to connect, visualize and share data in a highly interactive and intuitive way. It allows users to quickly analyze and explore large and complex datasets using a drag-and-drop interface without requiring coding or programming skills.

Tableau provides a wide range of chart types and visualization options, such as line charts, bar charts, maps, scatter plots, and many more. Tableau can connect to various data sources, including databases, spreadsheets, big data platforms, and cloud services. It also allows users to perform data cleaning and transformation tasks, create custom calculations, and generate insights using advanced analytics features. Tableau provides a suite of products, including Tableau Desktop for creating and publishing visualizations, Tableau Server for sharing and collaborating on data and visualizations across an organization, and Tableau Prep for preparing and cleaning data before analysis. With its user-friendly interface and powerful visualization capabilities, Tableau has become popular for businesses and organizations looking to make data-driven decisions.

3.5.4 Powerbi:

Power BI is a business analytics service by Microsoft. It provides interactive visualizations and business intelligence capabilities with an interface simple enough for end-users to create reports and dashboards. It allows users to connect to multiple data sources, clean and transform data, create custom calculations, and visualize data through charts, graphs, and tables.

3.5.5 Why Python

1. Scalability: Python is a programming language that scales very fast. Among all available languages, Python is a leader in scaling. That means that Python has more and more possibilities. Python flexibility is super useful for any problem in-app development.

Any problem can be solved easily with new updates that are coming. Saying that Python provides the best options for newbies because there are many ways to decide the same issue. Even if you have a team of non-Python programmers, who

knows C++ +design patterns, Python will be better for them in terms of time needed to develop and verify code correctness. It happens fast because you don't spend your time to find memory leaks, work for compilation or segmentation faults.

2. Libraries and Frameworks: Due to its popularity, Python has hundreds of different libraries and frameworks which is a great addition to your development process. They save a lot of manual time and can easily replace the whole solution. As a Data Scientist, you will find that many of these libraries will be focused on Data Analytics and Machine Learning. Also, there is a huge support for Big Data. I suppose there should be a strong pro why you need to learn Python as your first language.

Some of these libraries are given below:

- Pandas

It is great for data analysis and data handling. Pandas provides data manipulation control.

- NumPy

NumPy is a free library for numerical computing. It provides high-level math functions along with data manipulations.

- SciPy

This library is related to scientific and technical computing. SciPy can be used for data optimization and modification, algebra, special functions, etc.

3. Web Development: To make your development process as easy as it is possible only, learn Python. There are a lot of Django and Flask libraries and frameworks that make your coding productive and speed up your work. If you compare PHP and Python, you can find that the same task can be created within a few hours of code via PHP. But with Python, it will take only a few minutes. Just take a look at the Reddit website — it was created with Python.

Here are Python's Full Stack frameworks for web development:

- Django
- Pyramid
- Flask
- Bottle

4. Huge Community: Python has a powerful community. Anyone might think that it shouldn't be one of the main reasons why you need to select Python. But the truth is vice versa.

5. Automation:

Using Python automation frameworks like PYUnit gives you a lot of advantages:

- No additional modules are required to install. They come with the box.
- Even if you don't have a Python background you will find work with Unittest very comfortable. It is derivative and its working principle is similar to other xUnit frameworks.
- You can run singular experiments in a more straightforward way. You should simply indicate the names on the terminal. The output is compact too, making the structure adaptable with regards to executing test cases.
- The test reports are generated within milliseconds.

5. Python Frameworks For Test Automation:

1. Robot Framework
2. UnitTest
3. Pytest
4. Behave
5. Lettuce

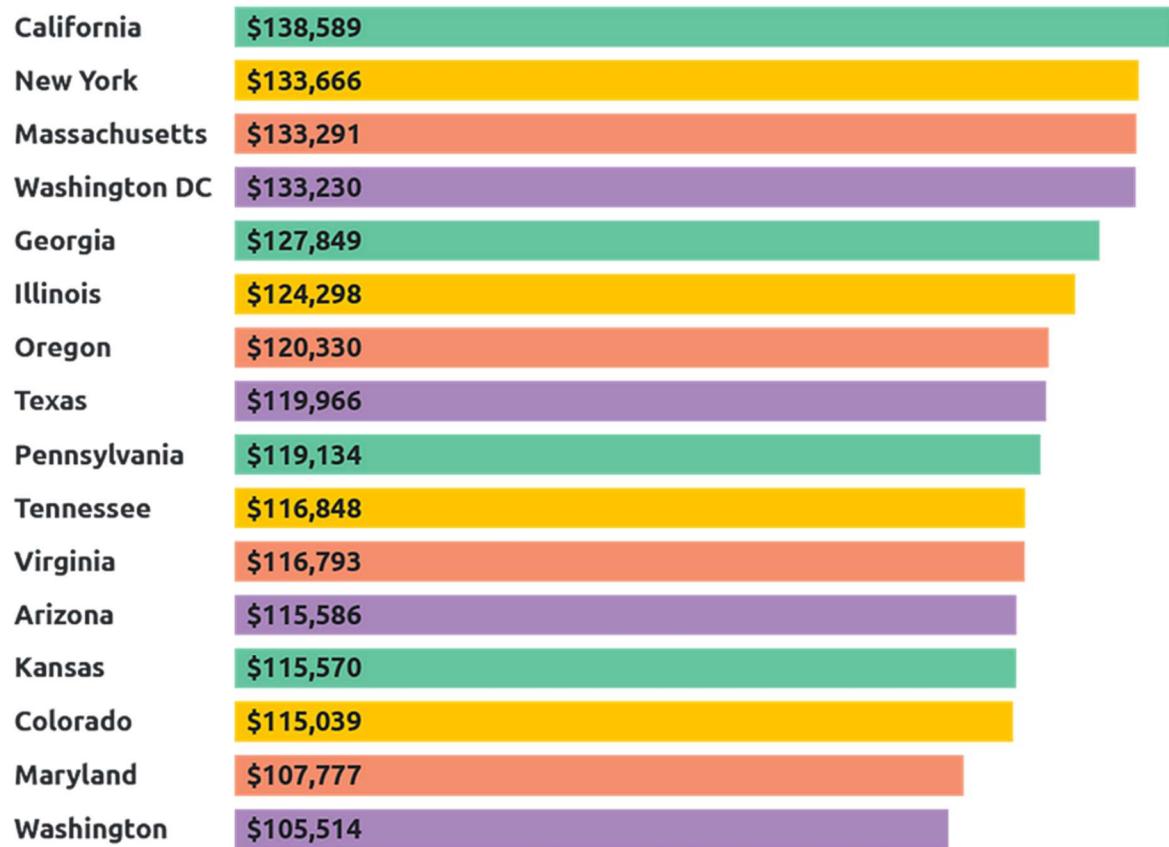
6. Jobs and Growth: Python is a unique language that has powerful growth and opens multiple career opportunities for Data Scientists. If you learn Python you can consider multiple additional jobs you might want to make the switch to in the future:

- Python Developer
- Product Manager
- Educator
- Financial Advisors
- Data Journalist

7. Salary

If you are looking for high paying opportunities, Python has massive options for you. Just check these stats:

AVERAGE PYTHON PROGRAMMER SALARIES BY STATE 2020 | INDEED



Source: daxx.com

Fig 3.1 Opportunities chart

CHAPTER 4: ASSIGNMENTS

4.1 Mini project :(Customer of Isro satellites)

1.From fetching the data from API provided by the ISRO

```
import requests
url = requests.post("https://isro.vercel.app/api/customer_satellites")
data = url.json()
print(data)

{'customer_satellites': [{"id': 'DLR-TUBSAT', 'country': 'Germany', 'launch_date': '26-05-1999', 'mass': '45', 'launcher': 'PSL V-C2'}, {"id": "KITSAT-3", "country": "REPUBLIC OF KOREA", "launch_date": "26-05-1999", "mass": "110", "launcher": "PSLV-C2"}, {"id": "BIRD", "country": "GERMANY", "launch_date": "22-10-2001", "mass": "92", "launcher": "PSLV-C3"}, {"id": "PROBA", "country": "BELGIUM", "launch_date": "22-10-2001", "mass": "94", "launcher": "PSLV-C3"}, {"id": "LAPAN-TUBSAT", "country": "INDONESIA", "launch_date": "10-07-2007", "mass": "56", "launcher": "PSLV-C7"}, {"id": "PEHUENSAT-1", "country": "ARGENTINA", "launch_date": "10-07-2007", "mass": "6", "launcher": "PSLV-C7"}, {"id": "AGILE", "country": "ITALY", "launch_date": "23-04-2007", "mass": "350", "launcher": "PSLV-C8"}, {"id": "TESCAR", "country": "ISRAEL", "launch_date": "21-01-2008", "mass": "300", "launcher": "PSLV-C10"}, {"id": "CAN-X2", "country": "CANADA", "launch_date": "28-04-2008", "mass": "7", "launcher": "PSLV-C9"}, {"id": "CUTE-1.7", "country": "JAPAN", "launch_date": "28-04-2008", "mass": "5", "launcher": "PSLV-C9"}, {"id": "DELFI-C3", "country": "CUTE-1.7"}]}
```

Fig 4.1.1 Data sat of the isro api

The process begins with calling a request function that interacts with a specific API. This function is responsible for sending a GET request to the API's designated URL and retrieving the desired data.

Once the data is fetched, it is displayed using the print function. The data contains various details related to customer satellites, including the name of the customer, the country where the satellite is located, the launch date, the mass of the satellite, the launcher used for the satellite's deployment, and possibly many other attributes.

By utilizing the print function, the fetched data is presented in a readable format. This allows users to view and analyze the information pertaining to customer satellites, such as their origins, launch dates, and other relevant characteristics.

2. To analyze and visualize the distribution of domestic and foreign customers, the matplotlib library can be used along with its functionalities such as pie charts. The data is segregated based on the customers' origin, distinguishing between domestic customers (ISRO) and foreign customers.

Using the pie chart, the distribution of customers can be represented visually. The size of each slice in the pie chart corresponds to the proportion of customers from each category. The Legend function is employed to clearly differentiate between ISRO (domestic customer) and foreign customers.

The resulting visualization provides an overview of the customer distribution, making it easy to identify the relative proportions of domestic and foreign customers in the dataset. This analysis helps in understanding the extent of involvement of ISRO as a domestic customer compared to foreign customers.

By utilizing the matplotlib library and its pie chart functionality, along with the Legend function to distinguish between ISRO and foreign customers, a clear and visually appealing visualization is generated. This enables users to gain insights into the composition of domestic and foreign customers and their respective contributions in the analyzed dataset.

```
url1 = requests.get("https://isro.vercel.app/api/spacecrafts")
data1 = url1.json()

import matplotlib.pyplot as plt
customer = len(data['customer_satellites'])
isro = len(data1['spacecrafts'])

label = ['ISRO', 'Foreign Customer']
dataaa = [isro,customer]
plt.pie(dataaa, labels=label, autopct='%1.1f%%')
plt.title('ANALYSIS OF DOMESTIC VS FOREIGN CUSTOMER INVOLVEMENT')
plt.axis('equal')
plt.legend()
plt.show()
```

ANALYSIS OF DOMESTIC VS FOREIGN CUSTOMER INVOLVEMENT

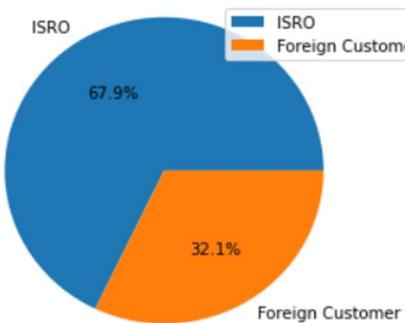


Fig 4.1.2 Visualization of ISRO and Foreign satellites

4.2 Mini project :(Finding the university)

Collecting website name from giving input of university from anywhere over the world:

```
import requests
url = requests.get('http://universities.hipolabs.com/search?country')
data = url.json()
print(data[0])

{'domains': ['marywood.edu'], 'web_pages': ['http://www.marywood.edu'], 'state-province': None, 'name': 'Marywood University', 'country': 'United States', 'alpha_two_code': 'US'}

u = input('Enter University name:')
for i in range(len(data)):
    x = data[i]['name']
    if u == x:
        print('Country name:', data[i]['country'])
        print('Website:', data[i]['web_pages'][0])
        break
    else: ('Not found')
else:
    print('Wrong input')

Enter University name:Gujarat Technological University Ahmedabad
Country name: India
Website: http://www.gtu.ac.in/
```

4.2 Script for finding website of university.

In the code, the requests library is utilized to call a JSON API. The API URL is provided, and a GET request is sent to retrieve the data. Using an if..else condition, the code executes different actions based on the response received.

If the response status code is 200, indicating a successful request, the data is extracted from the response using the json() method. The data is typically in JSON format, which can be easily parsed and processed.

Next, the code prompts the user to provide the input, specifically the college name. This input serves as a search query to find relevant information about the college from the retrieved data.

The code then proceeds to search for the provided college name within the data. It iterates through the data and compares each college's name with the input using an if statement. If a match is found, the code proceeds to extract additional details such as the country name and the website of the college's organization.

By displaying the country name and website, the code generates an output that provides information related to the college based on the user's input. This output helps users to easily obtain relevant details about a specific college, such as the country in which it is located and the website of its organization.

This process, involving the utilization of the requests library, calling a JSON URL, and utilizing if..else conditions, enables users to interactively retrieve and display information about colleges based on their input.

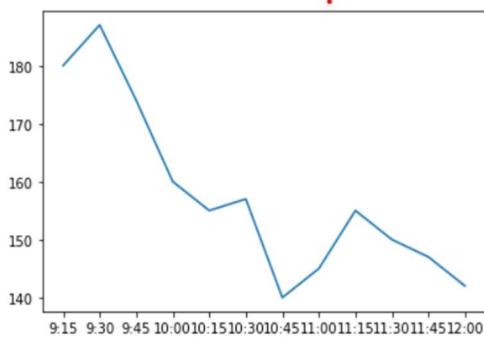
4.3 plot of time vs price action:

Using the matplotlib library the given time vs price chart was executed by plt.show in plt.plot(x,y) function. Where x is time and y is price.

```
In [1]: from matplotlib import pyplot as plt
```

```
time =['9:15' , '9:30', '9:45', '10:00', '10:15', '10:30', '10:45', '11:00', '11:15', '11:30', '11:45', '12:00']
price = [180,187,174,160,155,157,140,145,155,150,147,142]
```

```
plt.plot(time,price)
plt.show()
```



4.3 Time vs price action

The matplotlib library is a powerful tool for data visualization in Python. In this case, it was utilized to create a time vs price chart. The chart serves as a visual representation of the relationship between time and corresponding price values.

To generate the chart, the plt.plot(x, y) function is called, where x represents the time values and y represents the corresponding price values. This function connects the data points in a line plot, creating a visual representation of the relationship between time and price.

After plotting the chart, the plt.show() function is used to display the chart on the screen. This function renders the chart and opens a window to showcase the plot.

The time vs price chart provides valuable insights into how the price fluctuates over time. By examining the chart, one can observe trends, patterns, and any significant changes in price values. This visualization aids in understanding the historical behavior of prices and can be useful for various purposes such as financial analysis, stock market trends, or any other domain where time-series data analysis is relevant.

4.4 Mini project (Visualization of scores):

1.Cricket data fetched from the online kaggle website

```
In [1]: import numpy as np
import pandas as pd
from matplotlib import pyplot as plt

In [3]: file = (r"C:\Users\Ozapa\Dropbox\PC\Downloads\cricket (1).xlsx")
filedata = pd.read_excel(file)

In [4]: print(filedata)
print(type(filedata))

      CRICKET  VIRAT  DHAWAN  SHARMA
0            1      55      25      45
1            2      23      34      65
2            3      15      65      78
3            4      65      21     123
4            5      97      75      49
<class 'pandas.core.frame.DataFrame'>
```

4.4.1 Fetching data of cricket

Libraries such as numpy, pandas, and matplotlib are commonly used in data analysis and visualization tasks in Python. In this scenario, these libraries have been imported to assist with data manipulation, analysis, and plotting.

To begin, a dataset in the form of an Excel file is accessed. This file was obtained from the Kaggle website, a popular platform for data science and machine learning competitions. The specific file is in the XLSX format.

By utilizing pandas, the file is loaded into a DataFrame, a tabular data structure that allows for efficient manipulation and analysis. The DataFrame provides a convenient way to work with structured data, such as records of cricket players' performance.

After loading the data, it can be printed to the console using the print() function or by directly calling the DataFrame object. This allows us to inspect the dataset and verify its contents, ensuring that the file was loaded correctly.

Moving on to the analysis, the objective is to calculate the maximum, minimum, and average runs scored by three cricketers: Virat, Sharma, and Dhawan. By utilizing the appropriate pandas DataFrame operations, such as grouping and aggregation, these statistics can be derived easily.

By applying the appropriate filters, the runs scored by each player can be isolated and the required calculations can be performed. The maximum, minimum, and average runs of Virat, Sharma, and Dhawan are then obtained.

This analysis provides insights into the performance of these players based on their run scores. It allows for a comparison of their maximum, minimum, and average performances, which can be useful for various purposes, including evaluating their consistency, identifying standout performances, or comparing their overall contributions to the team.

By leveraging libraries such as numpy, pandas, and matplotlib, the analysis of cricket player performances becomes more streamlined, enabling efficient data manipulation, calculation of statistics, and visualization of the results.

```
In [6]: print("Virat DATA : " ,filedata[ "VIRAT"])
print("Virat max run : " , np.max(filedata[ "VIRAT"]))
print("Sharma min run : " ,np.min(filedata[ "SHARMA"]))
print("Dhawan average run : " ,np.average(filedata[ "DHAWAN"]))

Virat DATA :  0      55
1    23
2    15
3    65
4    97
Name: VIRAT, dtype: int64
Virat max run :  97
Sharma min run :  45
Dhawan average run :  44.0

In [7]: print("Dhawan run at 4th match : " ,filedata[ "DHAWAN"] [3])

Dhawan run at 4th match :  21

In [8]: matches = list(filedata[ "CRICKET"])
kohli = list(filedata[ "VIRAT"])
dhawan = list(filedata[ "DHAWAN"])
rohit = list(filedata[ "SHARMA"])

In [9]: print(matches)
barwidth = 0.15

[1, 2, 3, 4, 5]
```

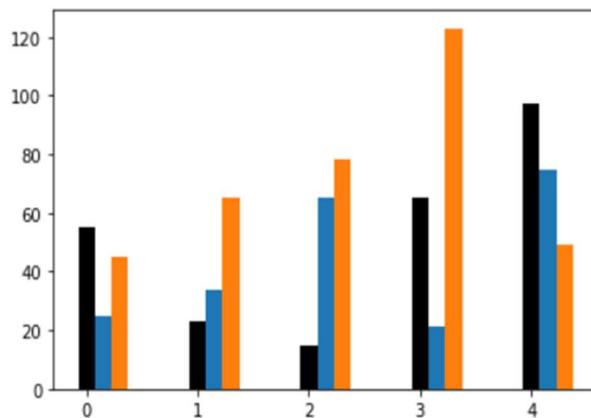
4.4.2 Analysis

Analysing the run of 4th match of Dhawan's. After That the matches are printed.

3. Bar graph of 3 main cricketers

```
In [10]: plt.bar(np.arange(len(matches)),kohli,width=barwidth,label="VIRAT",color="black")
plt.bar(np.arange(len(matches))+0.15,dhawan,width=barwidth,label="DHAWAN")
plt.bar(np.arange(len(matches))+0.30,rohit,width=barwidth,label="SHARMA")
```

Out[10]: <BarContainer object of 5 artists>



4.4.3 Bar Visualization

To differentiate the runs of the three cricketers (Virat, Sharma, and Dhawan), a bar chart can be created using the matplotlib library. The bar chart visually represents the runs scored by each player, allowing for easy comparison.

By utilizing the plt.bar() function from the matplotlib library, separate bars are created for each player's runs. To differentiate the bars, an indentation of 0.15 is applied between each player.

The data for each player's runs can be obtained from the analysis performed earlier. This data is then passed as inputs to the plt.bar() function, along with appropriate formatting parameters such as bar width, color, labels, and legends. By customizing these parameters, the bar chart can be visually appealing and informative.

By displaying the bar chart, the runs scored by each cricketer are clearly differentiated. The indentation of 0.15 between each player's bar ensures distinct visual separation, making it easier to compare their performances.

4.5 Mini Project(covid):

- Covid data from 2020 dataset by fitting the data into linear regression model

```

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

df = pd.read_csv('prices.csv')
print(df)
plt.xlabel('area')
plt.ylabel('prices')
plt.scatter(df.area, df.price, color='red')
plt.show()

reg = linear_model.LinearRegression()
reg.fit(df[['area']], df[['price']])
print(reg.predict([[3300]]))

print(reg.coef_)
print(reg.intercept_)
print(reg.intercept_ + reg.coef_* 3300)

{'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', 'date': '31 January 2020', 'dateymd': '2020-01-31', 'totalconfirmed': '1', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '0', 'dailydeceased': '0', 'date': '1 February 2020', 'dateymd': '2020-02-01', 'totalconfirmed': '1', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '1', 'dailydeceased': '0', 'date': '2 February 2020', 'dateymd': '2020-02-02', 'totalconfirmed': '2', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '1', 'dailydeceased': '0', 'date': '3 February 2020', 'dateymd': '2020-02-03', 'totalconfirmed': '3', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '1', 'dailydeceased': '0', 'date': '4 February 2020', 'dateymd': '2020-02-04', 'totalconfirmed': '4', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '1', 'dailydeceased': '0', 'date': '5 February 2020', 'dateymd': '2020-02-05', 'totalconfirmed': '5', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '1', 'dailydeceased': '0', 'date': '6 February 2020', 'dateymd': '2020-02-06', 'totalconfirmed': '6', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '1', 'dailydeceased': '0', 'date': '7 February 2020', 'dateymd': '2020-02-07', 'totalconfirmed': '7', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '1', 'dailydeceased': '0', 'date': '8 February 2020', 'dateymd': '2020-02-08', 'totalconfirmed': '8', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '1', 'dailydeceased': '0', 'date': '9 February 2020', 'dateymd': '2020-02-09', 'totalconfirmed': '9', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '1', 'dailydeceased': '0', 'date': '10 February 2020', 'dateymd': '2020-02-10', 'totalconfirmed': '10', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '1', 'dailydeceased': '0', 'date': '11 February 2020', 'dateymd': '2020-02-11', 'totalconfirmed': '11', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '1', 'dailydeceased': '0', 'date': '12 February 2020', 'dateymd': '2020-02-12', 'totalconfirmed': '12', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '1', 'dailydeceased': '0', 'date': '13 February 2020', 'dateymd': '2020-02-13', 'totalconfirmed': '13', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '1', 'dailydeceased': '0', 'date': '14 February 2020', 'dateymd': '2020-02-14', 'totalconfirmed': '14', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '1', 'dailydeceased': '0', 'date': '15 February 2020', 'dateymd': '2020-02-15', 'totalconfirmed': '15', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '1', 'dailydeceased': '0', 'date': '16 February 2020', 'dateymd': '2020-02-16', 'totalconfirmed': '16', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '1', 'dailydeceased': '0', 'date': '17 February 2020', 'dateymd': '2020-02-17', 'totalconfirmed': '17', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '1', 'dailydeceased': '0', 'date': '18 February 2020', 'dateymd': '2020-02-18', 'totalconfirmed': '18', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '1', 'dailydeceased': '0', 'date': '19 February 2020', 'dateymd': '2020-02-19', 'totalconfirmed': '19', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '1', 'dailydeceased': '0', 'date': '20 February 2020', 'dateymd': '2020-02-20', 'totalconfirmed': '20', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '1', 'dailydeceased': '0', 'date': '21 February 2020', 'dateymd': '2020-02-21', 'totalconfirmed': '21', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '1', 'dailydeceased': '0', 'date': '22 February 2020', 'dateymd': '2020-02-22', 'totalconfirmed': '22', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '1', 'dailydeceased': '0', 'date': '23 February 2020', 'dateymd': '2020-02-23', 'totalconfirmed': '23', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '1', 'dailydeceased': '0', 'date': '24 February 2020', 'dateymd': '2020-02-24', 'totalconfirmed': '24', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '1', 'dailydeceased': '0', 'date': '25 February 2020', 'dateymd': '2020-02-25', 'totalconfirmed': '25', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '1', 'dailydeceased': '0', 'date': '26 February 2020', 'dateymd': '2020-02-26', 'totalconfirmed': '26', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '1', 'dailydeceased': '0', 'date': '27 February 2020', 'dateymd': '2020-02-27', 'totalconfirmed': '27', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '1', 'dailydeceased': '0', 'date': '28 February 2020', 'dateymd': '2020-02-28', 'totalconfirmed': '28', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '1', 'dailydeceased': '0', 'date': '29 February 2020', 'dateymd': '2020-02-29', 'totalconfirmed': '29', 'totaldeceased': '0', 'totalrecovered': '0'}, {'dailyconfirmed': '1', 'dailydeceased': '0', 'date': '1 March 2020', 'dateymd': '2020-03-01', 'totalconfirmed': '30', 'totaldeceased': '0', 'totalrecovered': '0'}]}

```

4.5.1 Covid data fitted into regression model

- By getting data of rows and columns

```

for i in data:
    print(i)

cases_time_series
statewise
tested

print(data["cases_time_series"][0]['date'])

30 January 2020

for i in data.keys():
    print(i)

cases_time_series
statewise
tested

print(len(data["cases_time_series"]))

```

565

4.5.2 Covid data analysis

In first and third, for loop all the columns are printed . In second the first row is 30 january 2020.

3.Print the all confirmed cases since day 1.

```
In [37]: for i in range(0,len(data["cases_time_series"])):
    print('DAY',i+1,":", data["cases_time_series"][i]["dailyconfirmed"])

    DAY 1 : 1
    DAY 2 : 0
    DAY 3 : 0
    DAY 4 : 1
    DAY 5 : 1
    DAY 6 : 0
    DAY 7 : 0
    DAY 8 : 0
    DAY 9 : 0
    DAY 10 : 0
    DAY 11 : 0
    DAY 12 : 0
```

Fig 4.5.3 Daily confirmed cases

Using for loop all the confirmed cases are printed in day wise.

4.Print the cases state wise

```
In [38]: #total cases in gujarat
print(len(data["statewise"]))

38

In [39]: for i in range(1,len(data["statewise"])):
    print("Cases in",data["statewise"][i]['state'], "are :", data["statewise"][i]['confirmed'])

Cases in Andaman and Nicobar Islands are : 7549
Cases in Andhra Pradesh are : 1994606
Cases in Arunachal Pradesh are : 51513
Cases in Assam are : 580657
Cases in Bihar are : 725497
Cases in Chandigarh are : 62031
Cases in Chhattisgarh are : 1003814
Cases in Dadra and Nagar Haveli and Daman and Diu are : 10660
Cases in Delhi are : 1437118
Cases in Goa are : 172568
Cases in Gujarat are : 825196
Cases in Haryana are : 770230
Cases in Himachal Pradesh are : 210419
```

4.5.4 Cases state wise

All the counts of cases of states are shown in state wises. All the data are fetched by for loop.

5. By asking the input date and give output of covid case of input date

```
a = input("Enter date for Example 10 July 2020:")
for i in range(1,len(data["cases_time_series"])):
    #    print(data["cases_time_series"][i]['date'])
    if a == data["cases_time_series"][i]['date']:
        print(data["cases_time_series"][i])
        break
else:
    print('no')

Enter date for Example 10 July 2020:10 July 2020
{'dailyconfirmed': '27762', 'dailydeceased': '520', 'dailyrecovered': '20289', 'date': '10 July 2020', 'dateymd': '2020-07-10', 'totalconfirmed': '822609', 'totaldeceased': '22146', 'totalrecovered': '516251'}
```

4.5.5 case on input date

To obtain daily confirmed cases, daily deceased, daily recovered, and the total confirmed cases on a particular date, you can implement a script that takes a date as input and provides the desired output.

Using a dataset or data source that contains daily COVID-19 statistics, you can filter the data based on the input date. By applying the appropriate filters and calculations, you can derive the desired information.

To obtain the daily confirmed cases, you can subtract the total confirmed cases of the previous day from the total confirmed cases of the input date.

Similarly, you can calculate the daily deceased and daily recovered cases by subtracting the respective cumulative values of the previous day from the corresponding cumulative values of the input date.

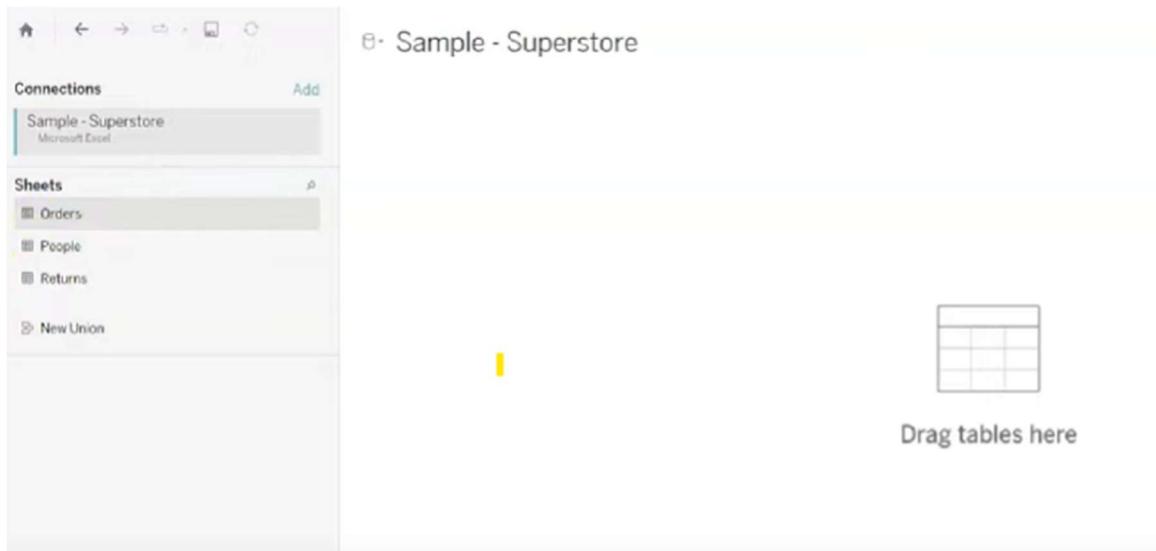
Additionally, the total confirmed cases on the particular date can be obtained directly from the dataset or by summing up the daily confirmed cases until the input date.

By executing the script with the input date, the output will provide the daily confirmed cases, daily deceased cases, daily recovered cases, and the total confirmed cases on that particular date.

This approach allows for the extraction of specific COVID-19 statistics based on the input date, providing valuable insights into the daily changes and overall status of the pandemic on a given day.

4.6 Tableau overview(Visualization of orders):

1.The datasheet of orders,people and returns of the orders are imported in tableau by drag.



4.6.1 sample data

Here the input data sets are imported using drag and drop.

2.The datasheet of Orders and people are attached with the relationship using Inner join.\

#	Abc Orders	Orders	Orders	Abc Orders	Abc Orders	Abc Orders	Abc Orders
Row ID	Order ID	Order Date	Ship Date	Ship Mode	Customer ID	Customer Name	Segment
1	CA-2016-152156	08-11-2016	11-11-2016	Second Class	CG-12520	Claire Gute	Consumer
2	CA-2016-152156	08-11-2016	11-11-2016	Second Class	CG-12520	Claire Gute	Consumer

4.6.2 Inner join

Here the Orders and people are related using inner join and the whole datasets are combined.

3.The datasheet of Returns and people are join by left join.

#	Abc Orders	Abc Orders	Abc Orders	Abc People	Abc People	Abc Returns	Abc Returns
Sales	Quantity	Discount	Profit	Person	Region (People)	Returned	Order ID (Returns)
2.54	3	0.800000	-3.82	Kelly Williams	Central	null	null
665.88	6	0.000000	13.32	Kelly Williams	Central	null	null
55.50	2	0.000000	9.99	Anna Andreadi	West	null	null
8.56	2	0.000000	2.48	Anna Andreadi	West	Yes	CA-2014-143336

4.6.3 Left join

By using a specific function, the script can identify and display the order IDs that were returned by customers. For orders that were not returned, the output will show "null" to indicate the absence of a return.

When this function is applied, it affects a total of 25 columns and 9994 rows in the dataset. This implies that the function scans and processes the data across all columns and rows, identifying the relevant information related to customer returns.

The data being affected by this function refers to the fact that the function filters and modifies the dataset, specifically highlighting the order IDs associated with customer returns and replacing non-returned orders with "null" values. This allows for easy identification and tracking of returned orders within the dataset.

The affected data provides valuable insights into customer behavior and the rate of returns within a specific state or context. By identifying and analyzing returned orders, businesses can gain insights into product satisfaction, quality control issues, or any other factors that may contribute to customer returns.

By utilizing this function and processing the data, businesses can better understand customer preferences and make data-driven decisions to improve customer satisfaction, reduce returns, and optimize their operations.

It is worth noting that the exact implementation of the function may vary depending on the specific dataset and the programming language or tools being used. However, the concept remains consistent in terms of identifying returned order IDs and indicating non-returned orders with "null" values, ultimately affecting a significant portion of the dataset across multiple columns and rows.

CHAPTER 5: PROJECT

Title: Behavioural and Sentimental analysis based recommendation system for e-commerce, magazines and other data sets

Abstract: The Recommendation System is a data science-based project that focuses on utilizing advanced algorithms to provide users with personalized recommendations for products or services. By analysing user behaviour and preferences, the system aims to enhance the user experience and drive sales and revenue for the business.

At the core of the project are machine learning techniques that process and analyse user data to generate personalized recommendations. These techniques leverage historical user behaviour, such as past purchases, browsing history, ratings, or interactions with the platform, to understand user preferences and make accurate predictions.

The first step in building a recommendation system involves collecting and pre-processing the relevant user data. This may include information such as user profiles, transaction history, product details, and user ratings or reviews. This data serves as the foundation for training the machine learning models.

Once the data is collected, it is used to train machine learning models that can predict user preferences and generate personalized recommendations. Various algorithms can be employed, such as collaborative filtering, content-based filtering, or hybrid approaches that combine multiple techniques. These models learn patterns from the data and use them to make accurate predictions about the products or services that a user is likely to be interested in.

The ultimate goal of the Recommendation System is to improve the user experience by presenting users with relevant and personalized recommendations. By offering tailored suggestions, users are more likely to discover products or services that match their interests, leading to increased engagement, customer satisfaction, and potentially higher sales and revenue for the business.

Furthermore, the Recommendation System is an ongoing project that requires continuous refinement and improvement. As more data is collected and user preferences evolve, the models need to be updated and fine-tuned to ensure the recommendations remain accurate and effective.

In summary, the Recommendation System leverages machine learning techniques to analyze user behavior and preferences and generate personalized recommendations. By providing users with relevant suggestions, the system aims to enhance the user experience and drive sales and revenue for the business.

Keywords: Business Intelligence, Sentimental Analysis , Machine Learning , Data Science , E-Commerce.

5.1 Importing libraries and loading the data set using pandas

The data is collected from the kaggle. matplotlib library and ggplot functionality are called.

```
# importing libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

plt.style.use("ggplot")

import sklearn
from sklearn.decomposition import TruncatedSVD

# Loading the dataset
amazon_ratings = pd.read_csv(r"C:\Users\Ozapa\AppData\Local\Temp\Temp1_archive (1).zip\ratings_Beauty.csv")
amazon_ratings = amazon_ratings.dropna()
amazon_ratings.head()
```

	Userid	ProductId	Rating	Timestamp
0	A39HTATAQ9V7YF	0205616461	5.0	1369699200
1	A3JM6GV9MNOF9X	0558925278	3.0	1355443200
2	A1Z513UWSAA00F	0558925278	5.0	1404691200
3	A1WMRR494NWEWV	0733001998	4.0	1382572800
4	A3IAAVS479H7M7	0737104473	1.0	1274227200

5.1 Basic functionalities

To perform the required tasks, the code imports the necessary libraries, including sklearn and truncatedSVD. The sklearn library provides a wide range of machine learning algorithms and tools, while truncatedSVD is a dimensionality reduction technique often used in recommendation systems.

After importing the libraries, the code proceeds to read the data. This could involve loading the data from a file, database, or API, depending on the specific source.

Once the data is loaded, the code performs a preprocessing step by dropping all the empty cells. This ensures that only the relevant and complete data is considered for further analysis. Dropping the empty cells helps to clean the data and avoid potential issues or inaccuracies in subsequent computations.

Following the preprocessing step, the code then proceeds to print the data. This can be done using the appropriate print function or by utilizing data visualization libraries such as pandas or matplotlib.

Printing the data allows for a visual inspection and verification of the loaded and cleaned dataset. It helps to ensure that the data has been correctly processed and that any missing or incomplete information has been appropriately handled.

By printing the data, users can review the dataset's structure, observe the available features, and gain a better understanding of the information contained within it. This step is crucial for data exploration and serves as a starting point for further analysis and modeling tasks.

In summary, the code snippet imports the necessary libraries, including sklearn and truncatedSVD. It then reads the data and performs preprocessing by dropping empty cells.

Finally, it prints the cleaned dataset, allowing users to verify the data and proceed with subsequent analysis or modeling tasks.

5.2 preparing popular products in descending order

The shape of data is observed and it turns out that it has 2023070 rows and 4 columns.

```
amazon_ratings.shape
```

```
(2023070, 4)
```

```
popular_products = pd.DataFrame(amazon_ratings.groupby('ProductId')['Rating'].count())
most_popular = popular_products.sort_values('Rating', ascending=False)
most_popular.head(10)
```

ProductId	Rating
B001MA0QY2	7533
B0009V1YR8	2869
B0043OYFKU	2477
B0000YUXI0	2143
B003V265QW	2088
B000ZMBSPE	2041
B003BQ6QXK	1918
B004OHQR1Q	1885
B00121UVU0	1838
B000FS05VG	1589

5.2 Shape of data

To fetch the most popular products, the code utilizes the groupby functionality. This allows the data to be grouped based on product IDs or any other relevant criteria.

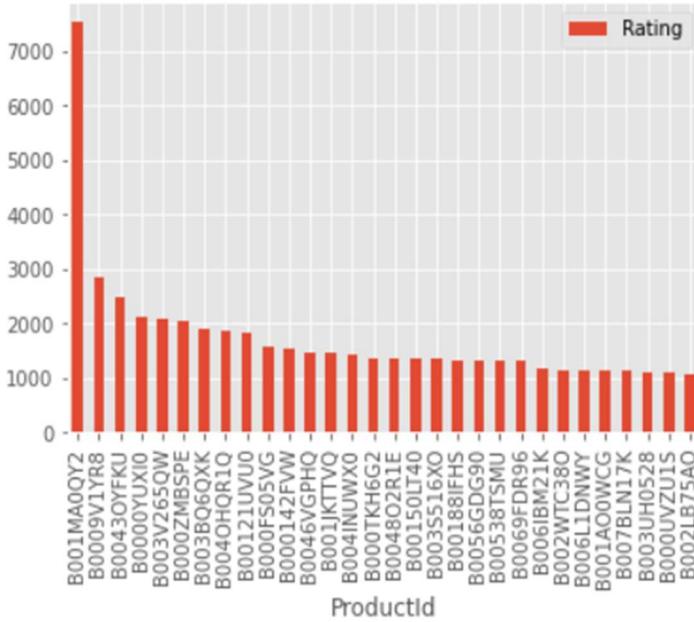
Once the data is grouped, the code calculates the popularity of each product. This can be done by counting the occurrences of each product or by using any other suitable metric that indicates product popularity.

The code then fetches the most popular products in descending order. By sorting the products based on their popularity, the top 10 most popular products can be identified.

Finally, the code prints and displays the top 10 most popular products according to their IDs. This allows for observation and analysis of the most popular products based on the provided dataset.

5.3 Visualising top 30 most popular product using bar chart

```
most_popular.head(30).plot(kind = "bar")
<AxesSubplot:xlabel='ProductId'>
```



5.3 Visualization of popular products

Analysis:

The graph presented above illustrates the popularity of products sold by the business, arranged in descending order. The popularity is determined based on the sales volume of each product.

Upon analyzing the graph, we can observe that product with ID # B001MA0QY2 has recorded sales of over 7000 units, making it the most popular product sold by the business. Following closely behind is product ID # B0009V1YR8, which has sales of 3000 units. The graph further highlights other popular products, showcasing their respective sales figures.

This information provides valuable insights into the products that have gained significant traction among customers. By understanding the popularity of different products, businesses can make informed decisions regarding inventory management, marketing strategies, and potential expansion opportunities.

Overall, the graph serves as a visual representation of the most popular products, showcasing their sales figures in descending order. This allows businesses to identify trends and patterns in customer preferences, enabling them to cater to the demands of their target audience effectively.

5.4 Amazon ratings which has top 10000 of ratings

```
In [29]: amazon_ratings1 = amazon_ratings.head(10000)
```

5.4 Ratings

5.5 Model-based collaborative filtering system

Recommendation systems play a crucial role in suggesting relevant items to users based on their preferences and behaviors. Collaborative filtering is one of the widely used techniques in recommendation systems, particularly in making predictions by identifying patterns from multiple user data.

Here's an overview of how a model-based collaborative filtering approach can be utilized to recommend items to users based on their purchase history and similarity of ratings provided by other users:

- Data Collection: Gather information about users, items, and their interactions. This data includes the purchase history of users, the items they have bought, and the ratings they have assigned to those items. Ensure that the data is structured and organized for further processing.
- Matrix Representation: Create a matrix representation of the data, commonly known as the user-item matrix. Each row in the matrix represents a user, and each column represents an item. The cells of the matrix contain the ratings provided by users for the corresponding items they have interacted with. Missing values can be represented as NaN or 0.
- Model Training: Utilize a model-based collaborative filtering technique, such as matrix factorization or singular value decomposition (SVD), to learn the underlying patterns and relationships within the user-item matrix. These techniques decompose the matrix into lower-rank matrices and capture latent factors that represent user preferences and item characteristics.
- Similarity Calculation: Calculate the similarity between users based on their rating patterns. Common similarity metrics include cosine similarity or Pearson correlation coefficient. The similarity score measures how closely the ratings of two users align with each other.
- User-Item Recommendations: For a particular user, identify similar users based on their rating patterns and purchase history. Aggregate the ratings of those similar users for items the target user has not interacted with. Generate a recommendation list by selecting the items with the highest aggregated ratings.

- Refinement and Personalization: Enhance the recommendations by incorporating additional factors such as item popularity, user demographics, or contextual information. Apply techniques like item-based filtering or content-based filtering to provide more personalized recommendations.
- Evaluation and Iteration: Assess the performance of the recommendation system using appropriate evaluation metrics, such as precision, recall, or mean average precision. Continuously iterate and refine the model and algorithms to improve the accuracy and relevance of the recommendations.

```
In [31]: ratings_utility_matrix = amazon_ratings1.pivot_table(values='Rating', index='UserId', columns='ProductId', fill_value=0)
ratings_utility_matrix.head()
```

Out[31]:

UserId	0205616461	0558925278	0733001998	0737104473	0762451459	1304139212	1304139220	130414089X	130414643X	1304146537	...
A00205921JHJK5X9LNP42	0	0	0	0	0	0	0	0	0	0	...
A024581134CV80ZBLIZTZ	0	0	0	0	0	0	0	0	0	0	...
A03056581JJIOL5FSKJY7	0	0	0	0	0	0	0	0	0	0	...
A03099101ZRK4K607JVHH	0	0	0	0	0	0	0	0	0	0	...
A0505229A7NSH3FRXRR4	0	0	0	0	0	0	0	0	0	0	...

5 rows × 886 columns

5.5 Utility matrix

5.6 Utility Matrix based on products sold and user reviews.

An utility matrix is a representation of all possible user-item preferences or ratings, organized in the form of a matrix. Each row of the matrix corresponds to a user, and each column represents an item. The cells of the matrix contain the ratings provided by users for the corresponding items they have expressed a preference for.

The utility matrix is typically sparse because it is highly unlikely for any user to have interacted with or rated all the items in the list. Therefore, most of the values in the utility matrix are unknown or missing. This sparsity is a common characteristic of recommendation systems, where the focus is on predicting and filling in the missing values to provide personalized recommendations for users.

By analyzing the available ratings and patterns in the utility matrix, recommendation algorithms can identify similarities between users or items, and make predictions or recommendations based on those patterns. Various techniques, such as collaborative filtering or matrix factorization, can be applied to infer missing ratings and suggest relevant items to users based on their preferences and the preferences of similar users.

The utility matrix serves as a foundational data structure in recommendation systems, enabling the application of algorithms and methodologies to generate accurate and useful recommendations, even in the presence of sparse data.

```
In [33]: ratings_utility_matrix.shape
```

Out[33]: (9697, 886)

5.6 Utility matrix shape

So, the utility matrix you have mentioned has dimensions of 9697 rows and 886 columns. This means that there are 9697 users in your dataset, and each user has expressed their preferences or ratings for 886 items.

The utility matrix serves as a fundamental data structure in recommendation systems. Each cell in the matrix represents a user's rating or preference for a particular item. However, since the matrix is typically sparse, it implies that most of the cells will be empty or unknown, as not all users have rated all items.

With the dimensions of the utility matrix known, you can apply various techniques to analyze the data, such as collaborative filtering, matrix factorization, or other recommendation algorithms. These methods leverage the available ratings to make predictions and generate personalized recommendations for users based on their preferences and the preferences of similar users.

By utilizing the utility matrix, recommendation systems can effectively handle large datasets and provide valuable insights into user-item interactions, enabling accurate and relevant recommendations to enhance the user experience.

5.7 Transposing the matrix

X = ratings_utility_matrix.T
X.head()
Userid A00205921JHJK5X9LNP42 A024581134CV80ZBLIZTZ A03056581JJ10L5FSKJY7 A03099101ZRK4K607JVHH A0505229A7NSH3FRXRR4 A05492663T95K1
ProductId
0205616461 0 0 0 0 0
0558925278 0 0 0 0 0
0733001998 0 0 0 0 0
0737104473 0 0 0 0 0
0762451459 0 0 0 0 0
5 rows × 9697 columns
X.shape
(886, 9697)

5.7 Transposing matrix

In linear algebra, the transpose of a matrix involves interchanging its rows and columns. The resulting matrix, denoted as X' , is obtained by placing the element at the i th row and j th column of the original matrix X in the j th row and i th column of X' .

Let's consider a matrix X with dimensions $m \times n$, meaning it has m rows and n columns. The transpose of X , denoted as X' , will have dimensions $n \times m$, where the number of rows in X' is equal to the number of columns in X , and the number of columns in X' is equal to the number of rows in X .

5.8 Correlation matrix

```
X1 = X

# Decomposing the Matrix

SVD = TruncatedSVD(n_components=10)
decomposed_matrix = SVD.fit_transform(X)
decomposed_matrix.shape

(886, 10)

# Correlation Matrix

correlation_matrix = np.corrcoef(decomposed_matrix)
correlation_matrix.shape

(886, 886)
```

5.8 Correlation matrix

5.9 Isolating Product ID # 6117036094 from the Correlation Matrix

Assuming the customer buys Product ID # 6117036094 (randomly chosen)

```
In [46]: i = "6117036094"

product_names = list(X.index)
product_ID = product_names.index(i)
product_ID

Out[46]: 99
```

5.9 Random product

5.10 Correlation

```
In [48]: correlation_product_ID = correlation_matrix[product_ID]
correlation_product_ID.shape

Out[48]: (886,)
```

5.10 correlation

In the recommendation system, all items that were purchased by a particular customer are identified based on the items rated by other customers. By analyzing the preferences and ratings of other customers, the system determines which items are commonly purchased alongside the item of interest.

To illustrate this, the system represents the customers who bought the same product as the customer in question using a shape, specifically in this case, a shape with 886 points. Each point in the shape represents a customer who has purchased the same product.

By examining the shape and the customers represented within it, patterns and trends can be identified. This information can be used to make recommendations to the customer, suggesting items that are frequently purchased together with their selected product.

This approach leverages the power of collaborative filtering techniques, where recommendations are based on the collective preferences and behaviors of a group of customers. By considering the purchasing patterns of similar customers, the system can provide personalized and relevant recommendations to the customer.

Overall, the recommendation system analyzes the items rated by other customers and identifies those that are frequently purchased alongside the item of interest. The customers who bought the same product are visually represented in the shape of 886, providing insights for generating personalized recommendations.

5.11 Product Id # Here are the top 10 products to be displayed by the recommendation system to the above customer based on the purchase history.

```
Recommend = list(X.index[correlation_product_ID > 0.90])

# Removes the item already bought by the customer
Recommend.remove(i)

Recommend[0:9]

['1304139212',
 '130414643X',
 '1304174867',
 '1304174905',
 '1304196070',
 '1304511111',
 '1304511154',
 '1304622452',
 '1304622657']
```

5.11 Suggestion Products

CHAPTER 6: CONCLUSION

6.1 Summary of Internship

During my internship, I had the opportunity to work as a Python Developer, where I gained practical experience in developing machine learning models. This experience provided me with a deep understanding of software development principles, data analysis techniques, database management, and version control systems like Git.

Throughout the internship, I had the chance to work on various tasks and projects, completing more than 40 different assignments. These tasks allowed me to further enhance my skills in programming languages such as Python, Ruby, and Java.

Working closely with my mentor and senior colleagues proved to be immensely helpful. Their guidance and expertise helped me navigate challenges and ensure that projects were completed within the given time frame. Collaborating with experienced professionals allowed me to learn from their insights and acquire valuable knowledge in the field.

By actively participating in the development of machine learning models, I gained hands-on experience in applying various algorithms and techniques to analyze data and make predictions. This practical exposure significantly contributed to my understanding of machine learning concepts and their real-world applications.

Furthermore, working on diverse projects allowed me to broaden my knowledge of different domains and industries. This exposure helped me understand how to tailor solutions to specific business needs and deliver effective results.

Overall, my internship as a Python Developer provided me with a well-rounded understanding of software development, data analysis, and machine learning. The completion of numerous tasks and the guidance of mentors and colleagues were instrumental in my growth and development during this period. I look forward to leveraging these experiences and skills in future endeavors.

6.2 Date of Continuous Evaluation

During our college institution's internship program, a continuous evaluation procedure was implemented to monitor the performance of students throughout the 12-week internship. The evaluation process consisted of two parts, with the first evaluation conducted on 15th March, 2023, and the second evaluation on 10th May, 2023.

As part of the evaluation procedure, students were required to present their work and submit weekly reports to an internal mentor. These reports served as a means to track the progress of students and provide feedback on their performance.

The weekly reports played a crucial role in documenting the students' activities, highlighting the tasks accomplished, challenges faced, and lessons learned during the

internship. By submitting these reports, students were able to showcase their work and reflect on their experiences throughout the program.

The internal mentor assigned to each student provided guidance and support throughout the internship. They reviewed the weekly reports, monitored the students' progress, and offered valuable feedback and suggestions for improvement. This mentorship system ensured that students received regular guidance and had a point of contact for any questions or concerns.

The continuous evaluation procedure, including the presentations, weekly reports, and mentorship, aimed to foster a culture of accountability, learning, and growth. It provided students with a structured framework to demonstrate their progress and receive guidance to enhance their overall learning and professional development.

In summary, the continuous evaluation procedure in our college institution's internship program involved two evaluations conducted on specific dates. Students were required to present their work and submit weekly reports to an internal mentor. This process allowed for the monitoring of students' performance and provided a platform for feedback and guidance throughout the internship.

CHAPTER 7: REFERENCES

- [1] For data set: "<https://www.kaggle.com/datasets/skillsmuggler/amazon-ratings?resource=download>"
- [2] For Analysis of basic ggplots: "<https://github.com/RudrenduPaul/Python-Ecommerce-recommendation-system-using-machine-learning/blob/master/Recommendation%20System%20-%20Paul.ipynb>"
- [3] For reference of pie plots: "<https://github.com/Vatshayan/Final-Year-Machine-Learning-Stock-Price-Prediction-Project>"
- [4] For Matplotlib libraries: "<https://ieeexplore.ieee.org/document/9660854>"
- [5] For correlation matrix: "<https://machinelearningmastery.com>"
- [6] For latest popular products: "<https://corporatefinanceinstitute.com>"
- [7] Book: "Hands-on ML with Scikit-Learn, Keras & TensorFlow" by Aurelien Geron
- [8] Book: "Mathematics for Machine Learning" by Marc Peter Deisenroth
- [9] Book: "Fundamentals of Machine Learning for Predictive Data Analytics" by John D. Kelleher, Brian Mac Namee, and Aoife D'Arcy.

Appendix



GUJARAT TECHNOLOGICAL UNIVERSITY
(Established under Gujarat Act No. 20 of 2007)
ગુજરાતટેકનોલોજીકલ યુનિવર્સિટી
(ગુજરાત અધિનિયમ ક્રમાંક: ૨૦/૨૦૦૭ દ્વારા સ્થાપિત)

Annexure 2

Feedback Form by Industry expert

Student Name: Pooja Rajeshbhai Oza Date: 17-4-23
 Work Supervisor: Chintan Nagdecha Title: Machine learning
 Company/Organization: Infolabz IT Services Pvt LTD.
 Enrollment No: 190390107021

Internship Address: 405, Vasai Avenue, Nr commerce six Rd, Navrangpura
 Dates of Internship: From 23-1-23 to 16-4-23

Please evaluate your intern by indicating the frequency with which you observed the following behaviors:

Parameters	Needs improvement	Satisfactory	Good	Excellent
Shows interest in work and his/her initiatives				✓
Produces high quality work and accepts responsibility				✓
Uses technical knowledge and expertise			✓	
Analyzes problems effectively		✓		
Communicates well and writes effectively				✓

Overall performance of student intern: (Needs improvement/ Satisfactory/Good/Excellent):

Additional comments, if any:

Signature of Industry person with name and Stamp:

Twinkle Shah



Signature of the Faculty Mentor



GUJARAT TECHNOLOGICAL UNIVERSITY
(Established under Gujarat Act No. 20 of 2007)

ગુજરાત ટેકનોલોજીકલ યુનિવર્સિટી

(ગુજરાત અધિનિયમ ક્રમાંક: ૨૦/૨૦૦૭ ગ્રાન્ટ સ્થાપિત)

Annexxure I

Enrollment no:

190390107021

STUDENT'S WEEKLY RECORD OF INTERNSHIP

NAME OF STUDENT: PARTH RAJESH RHAZ 02A

DIARY OF THE WEEK: Dt: 23/01/2023 TO 27/01/2023

DEPARTMENT: COMPUTER ENGINEERING SEM: 08th

NAME OF THE ORGANISATION: INFOLABZ IT SERVICES PVT LTD.

NAME OF THE PLANT/SECTION/DEPARTMENT: DATA ANALYST DEPARTMENT

NAME OF OFFICER INCHARGE OF THE PLANT/SECTION/DEPARTMENT: CHINTAN NAGRECHA

DESCRIPTION OF THE WORK DONE IN BRIEF

(1) Road map of Internship

(2) Introduction of Data science and Machine Learning

(3) Why we learn them.

(4) Compared different languages like Python and Java

(5) Basics of Python.



GUJARAT TECHNOLOGICAL UNIVERSITY
(Established under Gujarat Act No. 20 of 2007)

ગુજરાત ટેકનોલોજીકલ યુનિવર્સિટી
 (ગુજરાત અધિનિયમ ક્રમાંક: ૨૦/૨૦૦૭ દ્વારા સ્થાપિત)

Annexure I

Enrollment no:

190390107021

STUDENT'S WEEKLY RECORD OF INTERNSHIP

NAME OF STUDENT: PARTH RAJESHBHAI OZA

DIARY OF THE WEEK: Dt: 30/1/2023 TO 03/02/2023

DEPARTMENT: COMPUTER ENGINEERING SEM: 08th

NAME OF THE ORGANISATION: INFOLABZ IT SERVICES PVT LTD.

NAME OF THE PLANT/SECTION/DEPARTMENT: DATA ANALYST DEPARTMENT

NAME OF OFFICER INCHARGE OF THE PLANT/SECTION/DEPARTMENT: CHINTAN NAGRECHA

DESCRIPTION OF THE WORK DONE IN BRIEF

- (1) Getting started with Python and installation
- (2) Why to use Python and top companies why use them like google, Netflix, IBM, metu....
- (3) Tried different types of Python versions and installed it on different operating systems.
- (4) Installation of Jupyter Notebook, pycharm.
- (5) Explored different lists, tuples, Dictionary.
- (6) Learnt command like pip install library-name.



GUJARAT TECHNOLOGICAL UNIVERSITY
(Established under Gujarat Act No. 20 of 2007)

ગુજરાત ટેકનોલોજીકલ યુનિવર્સિટી
 (ગુજરાત અધિનિયમ ક્રમાંક: ૨૦/૨૦૦૭ દ્વારા સ્થાપિત)

Annexure I

Enrollment no:

190390107021**STUDENT'S WEEKLY RECORD OF INTERNSHIP**NAME OF STUDENT: PARTH RAJESHBHAI OZADIARY OF THE WEEK: Dt: 06/02/2023 TO 10/02/2023DEPARTMENT: COMPUTER ENGINEERING SEM: O8thNAME OF THE ORGANISATION: INFOLABZ IT SERVICES PVT LTD.NAME OF THE PLANT/SECTION/DEPARTMENT: DATA ANALYST DEPARTMENTNAME OF OFFICER INCHARGE OF THE PLANT/SECTION/DEPARTMENT: CHINTAN NAGRECHA

DESCRIPTION OF THE WORK DONE IN BRIEF

- (1) Completed different programs in basics of Python.
- (2) Implemented oops concept in different programs
- (3) Journey of new Inbuilt libraries like os, time.
- (4) Applied different types of loops like for, while, continue statement.
- (5) Started learning about libraries like numpy, pandas, matplotlib and seaborn.



GUJARAT TECHNOLOGICAL UNIVERSITY
(Established under Gujarat Act No. 20 of 2007)

ગુજરાત ટેકનોલોજીકલ યુનિવર્સિટી
 (ગુજરાત અધિનિયમ ક્રમાંક: ૨૦/૨૦૦૭ દ્વારા સ્થાપિત)

Annexure 1

Enrollment no:

190390107021

STUDENT'S WEEKLY RECORD OF INTERNSHIP

NAME OF STUDENT: PARTH RATESHBHAI ORA

DIARY OF THE WEEK: Dt: 13/12/2023 TO 17/12/2023

DEPARTMENT: COMPUTER ENGINEERING SEM: 08TH

NAME OF THE ORGANISATION: INFOLABZ IT SERVICES PVT LTD.

NAME OF THE PLANT/SECTION/DEPARTMENT: DATA ANALYST DEPARTMENT

NAME OF OFFICER INCHARGE OF THE PLANT/SECTION/DEPARTMENT: CHINTAN NAGRECHA

DESCRIPTION OF THE WORK DONE IN BRIEF

- (1) Started working on Pandas library.
- (2) Completed remaining previous week's task and submitted it for review purpose.
- (3) Learnt practical and detailed knowledge about dataframes.
- (4) How we deal with data using Pandas.
- (5) Started research on Matplotlib library.



GUJARAT TECHNOLOGICAL UNIVERSITY
(Established under Gujarat Act No. 20 of 2007)
ગુજરાત ટેકનોલોજીકલ યુનિવર્સિટી
(ગુજરાત અધિનિયમ ક્રમાંક: ૨૦/૨૦૦૭ દ્વારા સ્થાપિત)

Annexure 1

Enrollment no:
190390107021

STUDENT'S WEEKLY RECORD OF INTERNSHIP

NAME OF STUDENT: Prauth R. Oza

DIARY OF THE WEEK: Dt: 20-2-23 **TO** 24-2-23

DEPARTMENT: Computer **SEM:** 8

NAME OF THE ORGANISATION: Intellecube

NAME OF THE PLANT/SECTION/DEPARTMENT: Machine learning

NAME OF OFFICER INCHARGE OF THE PLANT/SECTION/DEPARTMENT: Chintan Nagrocha

DESCRIPTION OF THE WORK DONE IN BRIEF

- getting knowledge of matplotlib python library.
- using this library we made different graphs like bar graph, histogram, pie chart, scatter plot etc..
- we create static and animated interactive visualized methods ..

Scanned with OKEN Scanner



GUJARAT TECHNOLOGICAL UNIVERSITY
(Established under Gujarat Act No. 20 of 2007)

ગુજરાત ટેકનોલોજીકલ યુનિવર્સિટી
 (ગુજરાત અધિનિયમ ક્રમાંક: ૨૦/૨૦૦૭ દ્વારા સ્થાપિત)

Annexure 1

Enrollment no:
190390107021

STUDENT'S WEEKLY RECORD OF INTERNSHIP

NAME OF STUDENT: Purath R. Oza.

DIARY OF THE WEEK: Dt: 27-2-23 TO 3-3-23

DEPARTMENT: Computer SEM: 8

NAME OF THE ORGANISATION: Infelabz

NAME OF THE PLANT/SECTION/DEPARTMENT: Machine learning

NAME OF OFFICER INCHARGE OF THE PLANT/SECTION/DEPARTMENT: Chintan Negandhi

DESCRIPTION OF THE WORK DONE IN BRIEF

- We start working on SQL.
- Understand the basics of SQL and their different types, also basics of relational databases.
- we have also done our different task and understand different queries.
- Also explore advance concept like triggers, procedure etc.



GUJARAT TECHNOLOGICAL UNIVERSITY
(Established under Gujarat Act No. 20 of 2007)

ગુજરાત ટેકનોલોજીકલ યુનિવર્સિટી
 (ગુજરાત અધિનિયમ ક્રમાંક: ૨૦/૨૦૦૭ હારા સ્થાપિત)

Annexure 1

Enrollment no:

190390107021

STUDENT'S WEEKLY RECORD OF INTERNSHIP

NAME OF STUDENT: PARTH R. OZA

DIARY OF THE WEEK: DT: 6-3-23 TO 10-3-23

DEPARTMENT: Computer **SEM:** 8th

NAME OF THE ORGANISATION: Infelisys IT services PVT. LTD.

NAME OF THE PLANT/SECTION/DEPARTMENT: Machine learning.

NAME OF OFFICER INCHARGE OF THE PLANT/SECTION/DEPARTMENT: Chintan Naigrocha

DESCRIPTION OF THE WORK DONE IN BRIEF

- Makes changes in previous tasks according to the suggestions.
- Work on visualization tools.
- In market different type of tools are available like PowerBI, tableau, excel, etc..
- Work on Power query related tasks.

Scanned with OKEN Scanner



GUJARAT TECHNOLOGICAL UNIVERSITY
(Established under Gujarat Act No. 20 of 2007)

ગુજરાત ટેકનોલોજીકલ યુનિવર્સિટી
 (ગુજરાત અધિનિયમ ક્રમાંક: ૨૦/૨૦૦૭ દ્વારા સ્થાપિત)

Annexure 1

Enrollment no:

190390107021

STUDENT'S WEEKLY RECORD OF INTERNSHIP

NAME OF STUDENT: Parm R. Oza

DIARY OF THE WEEK: Dt: 13-3-23 TO 17-3-23

DEPARTMENT: Computer **SEM:** 8

NAME OF THE ORGANISATION: Infolab, IT service

NAME OF THE PLANT/SECTION/DEPARTMENT: Machine Learning

NAME OF OFFICER INCHARGE OF THE PLANT/SECTION/DEPARTMENT: Chintan Mehta

DESCRIPTION OF THE WORK DONE IN BRIEF

- completed weekly assignments on tablet.
- Started making mini project of diabetes model prediction.
- find accuracy of model.
- different datasets and using API we managed to fetch the data using multiple data fetching techniques.



GUJARAT TECHNOLOGICAL UNIVERSITY
(Established under Gujarat Act No. 20 of 2007)

ગુજરાત ટેકનોલોજીકલ યુનિવર્સિટી

(ગુજરાત અધિનિયમ ક્રમાંક: ૨૦/૨૦૦૭ દ્વારા સ્થાપિત)

Annexure 1

Enrollment no:

190390107021

STUDENT'S WEEKLY RECORD OF INTERNSHIP

NAME OF STUDENT: PARTH R. 02A

DIARY OF THE WEEK: Dt: 20-3-23 TO 24-3-23

DEPARTMENT: computer **SEM:** 8th

NAME OF THE ORGANISATION: Infabiz

NAME OF THE PLANT/SECTION/DEPARTMENT: Machine learning

NAME OF OFFICER INCHARGE OF THE PLANT/SECTION/DEPARTMENT: chirantan narodecha

DESCRIPTION OF THE WORK DONE IN BRIEF

- Explored dashboard view in power bi.
- Understand about how to make effective reports which can impact more on our work
- Explored different feature like flexible fixes, navigation pane, natural language, Q & A, question box, informative reports etc.
- Work on different task related this

Scanned with OKEN Scanner



GUJARAT TECHNOLOGICAL UNIVERSITY
(Established under Gujarat Act No. 20 of 2007)

ગુજરાત ટેકનોલોજીકલ યુનિવર્સિટી
 (ગુજરાત અધિનિયમ ક્રમાંક: ૨૦/૨૦૦૭ હારા સ્થાપિત)

Annexure 1

Enrollment no: _____

STUDENT'S WEEKLY RECORD OF INTERNSHIP

NAME OF STUDENT: PARTH R. OZA

DIARY OF THE WEEK: Dt: 27-3-23 TO 31-3-23

DEPARTMENT: Computer **SEM:** 8

NAME OF THE ORGANISATION: Intelubz

NAME OF THE PLANT/SECTION/DEPARTMENT: Machine learning.

NAME OF OFFICER IN CHARGE OF THE PLANT/SECTION/DEPARTMENT: Chintan Nugroho

DESCRIPTION OF THE WORK DONE IN BRIEF

→ Practicing more on making dashboard and also explore mobile view presentation report view, online dashboard view etc...

→ Explored tableau, it has also drag and drop feature like powerBI, working on some examine project to getting better knowledge.

Scanned with OKEN Scanner



GUJARAT TECHNOLOGICAL UNIVERSITY
(Established under Gujarat Act No. 20 of 2007)

ગુજરાત ટેકનોલોજીકલ યુનિવર્સિટી

(ગુજરાત અધિનિયમ ક્રમાંક: ૨૦/૨૦૦૭ દ્વારા સ્થાપિત)

Annexure 1

Enrollment no:

190390107021

STUDENT'S WEEKLY RECORD OF INTERNSHIP

NAME OF STUDENT: PARTH R. OZA

DIARY OF THE WEEK: Dt: 3-4-23 TO 7-4-23

DEPARTMENT: Computer **SEM:** 8th

NAME OF THE ORGANISATION: Infalabs

NAME OF THE PLANT/SECTION/DEPARTMENT: Machine learning

NAME OF OFFICER INCHARGE OF THE PLANT/SECTION/DEPARTMENT: Chintan Mehta

DESCRIPTION OF THE WORK DONE IN BRIEF

- Start working on Machine learning.
- Understanding about what is machine learning and how it can be useful in our daily life.
- different types of machine learning like, supervised, unsupervised, reinforcement learning etc.
- Also exposed different types of algorithms.

Scanned with OKEN Scanner



GUJARAT TECHNOLOGICAL UNIVERSITY
(Established under Gujarat Act No. 20 of 2007)
ગુજરાત ટેકનોલોજીકલ યુનિવર્સિટી
(ગુજરાત અધિનિયમ ક્રમાંક: ૨૦/૨૦૦૭ દ્વારા સ્થાપિત)

Annexure 1**Enrollment no:**190390107021**STUDENT'S WEEKLY RECORD OF INTERNSHIP****NAME OF STUDENT:** PARTH R. OZA**DIARY OF THE WEEK: Dt:** 10-4-23 **TO** 14-4-23**DEPARTMENT:** Computer **SEM:** 8th**NAME OF THE ORGANISATION:** Infelabz IT Services PVT. LTD**NAME OF THE PLANT/SECTION/DEPARTMENT:** Machine learning**NAME OF OFFICER INCHARGE OF THE PLANT/SECTION/DEPARTMENT:** Chintan Nagrecha**DESCRIPTION OF THE WORK DONE IN BRIEF**

- Explored different algorithms like linear regression, PCA, SVM. Designed tree algorithms. Logistic regression etc.
- Complete remain task related this.
- Another talk related Power BI. Done for refreshing our knowledge.
- Working and complete machine learning task.