

Project Report

**ON**

**Python-Diwali-Sales-Analysis**

**Master** **of Computer Application** **(MCA 2022-2024)**

****

Submitted To - Submitted By-

Mr. Rahul Agnihotri **Mohd Faraj** Head of Department Roll no. 2200470140032

**(M.C.A) Semester - IV**

CONTENTS

|  |  |  |
| --- | --- | --- |
| **SERIAL**  **NO.** | **DESCRIPTION** | **PAGE NO.** |
|  | Acknowledgement | 3 |
|  | Training Certificate | 4 |
|  | Plagiarism Report | 5 |
|  | Title & Introduction & Objective of Project | 6-8 |
|  | Project Category, Tools & Platform used & Technical Details | 8-9 |
|  | Hardware and Software Requirement | 10-11 |
|  | Feasibility Study | 12-13 |
|  | System Design,Data Flow Diagram | 14-16 |
|  | Tools and Technologies | 17 |
|  | Module Description | 18 |
|  | Coding | 19-175 |
|  | Screenshots | 176-184 |
|  | Future Scope of the Project | 185 |
|  | Glossary | 186 |
|  | Bibliography | 187 |

**Acknowledgment**

I would like to express my heartfelt gratitude to all those who have contributed to the completion of this project. Their support and encouragement have been invaluable throughout this journey.

First and foremost, I am deeply indebted to my HOD, Mr. Rahul Agnihotri sir for their unwavering guidance, patience, and encouragement. Their expertise and insights have played a pivotal role in shaping the direction of this project and refining its outcomes. I am truly grateful for their mentorship and support.

I extend my sincere thanks to the faculty members of the MCA department for their continuous support and encouragement. Their dedication to fostering academic excellence has been a constant source of inspiration.

I would like to acknowledge the assistance and cooperation of my peers and friends. Their valuable feedback, discussions, and moral support have been immensely helpful in overcoming challenges and staying motivated throughout this endeavour.

In conclusion, I am profoundly grateful to everyone who has contributed to this project, directly or indirectly. Your support has been instrumental in its success, and I am truly thankful for the opportunity to undertake this endeavour.

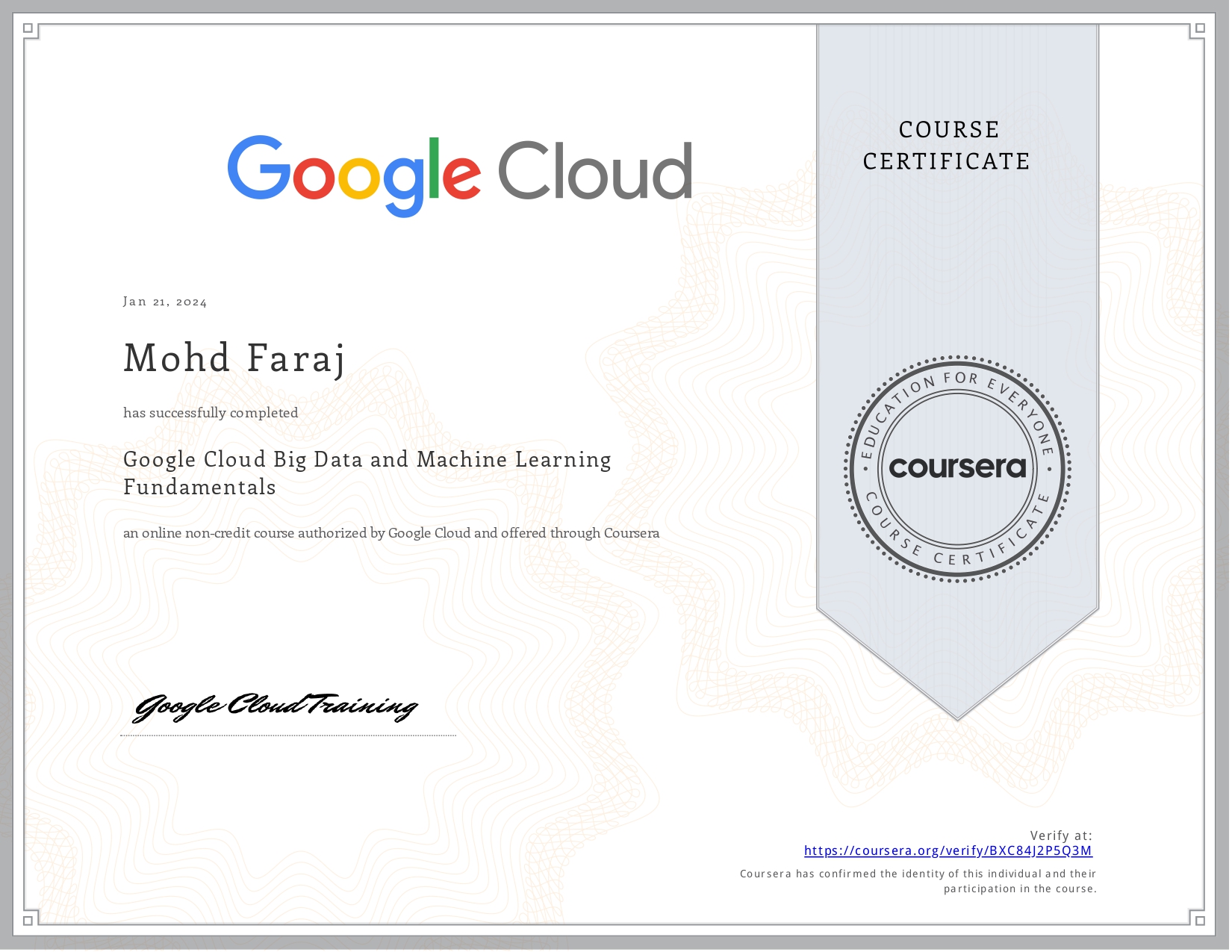
Mr. Rahul Agnihotri Mr.Kamlesh C. Sharma

[HOD-MCA] [Project Coordinator]

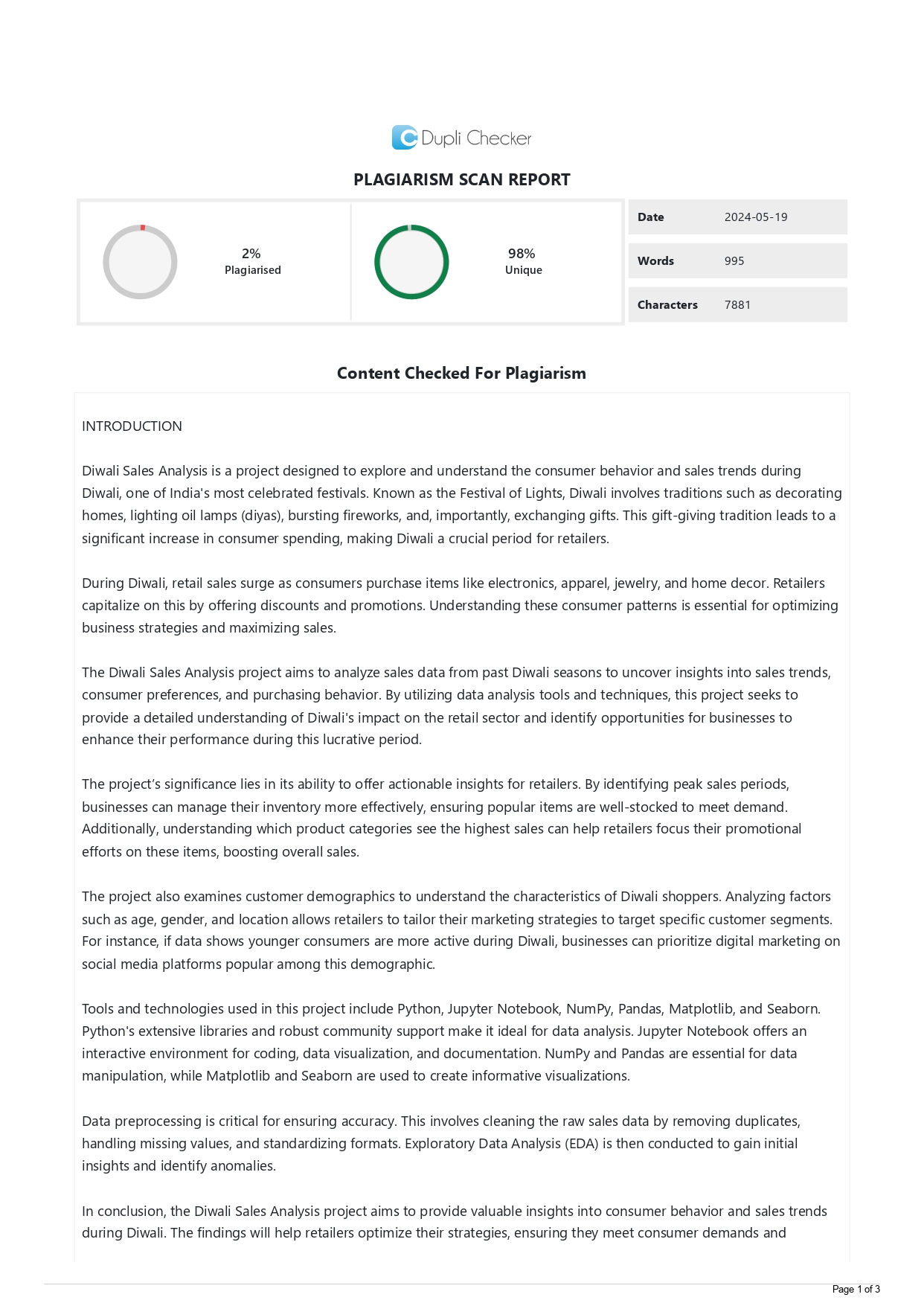
............................. ....................................

# 

# TRAINING CERTIFICATE



# PLAGIARISM REPORT



# TITLE OF THE PROJECT

**“Python-Diwali-Sales-Analysis”**

# INTRODUCTION

**Diwali Sales Analysis** is a project designed to explore and understand the consumer behavior and sales trends during Diwali, one of India's most celebrated festivals. Known as the Festival of Lights, Diwali involves traditions such as decorating homes, lighting oil lamps (diyas), bursting fireworks, and, importantly, exchanging gifts. This gift-giving tradition leads to a significant increase in consumer spending, making Diwali a crucial period for retailers.

During Diwali, retail sales surge as consumers purchase items like electronics, apparel, jewelry, and home decor. Retailers capitalize on this by offering discounts and promotions. Understanding these consumer patterns is essential for optimizing business strategies and maximizing sales.

The Diwali Sales Analysis project aims to analyze sales data from past Diwali seasons to uncover insights into sales trends, consumer preferences, and purchasing behavior. By utilizing data analysis tools and techniques, this project seeks to provide a detailed understanding of Diwali's impact on the retail sector and identify opportunities for businesses to enhance their performance during this lucrative period.

The project’s significance lies in its ability to offer actionable insights for retailers. By identifying peak sales periods, businesses can manage their inventory more effectively, ensuring popular items are well-stocked to meet demand. Additionally, understanding which product categories see the highest sales can help retailers focus their promotional efforts on these items, boosting overall sales.

The project also examines customer demographics to understand the characteristics of Diwali shoppers. Analyzing factors such as age, gender, and location allows retailers to tailor their marketing strategies to target specific customer segments. For instance, if data shows younger consumers are more active during Diwali, businesses can prioritize digital marketing on social media platforms popular among this demographic.

Tools and technologies used in this project include Python, Jupyter Notebook, NumPy, Pandas, Matplotlib, and Seaborn. Python's extensive libraries and robust community support make it ideal for data analysis. Jupyter Notebook offers an interactive environment for coding, data visualization, and documentation. NumPy and Pandas are essential for data manipulation, while Matplotlib and Seaborn are used to create informative visualizations.

Data preprocessing is critical for ensuring accuracy. This involves cleaning the raw sales data by removing duplicates, handling missing values, and standardizing formats. Exploratory Data Analysis (EDA) is then conducted to gain initial insights and identify anomalies.

In conclusion, the Diwali Sales Analysis project aims to provide valuable insights into consumer behavior and sales trends during Diwali. The findings will help retailers optimize their strategies, ensuring they meet consumer demands and capitalize on the economic opportunities presented by this festive period.

# 

# OBJECTIVES

The primary objectives of this Diwali sales analysis project are:

* Identify Peak Sales Periods: Determine the specific times during the Diwali season when sales volumes are at their highest, allowing businesses to focus their efforts on these critical periods.
* Analyze Product Category Trends: Examine which product categories experience the most significant sales boosts during Diwali, helping retailers stock and promote these items effectively.
* Understand Customer Demographics: Analyze purchasing patterns based on customer demographics such as age, gender, and location to tailor marketing strategies accordingly.
* Evaluate Sales Performance: Assess the overall sales performance during the Diwali season compared to non-festive periods to understand the festival's impact on retail.
* Provide Actionable Insights: Offer recommendations for businesses to enhance their sales strategies, promotional efforts, and customer engagement during future Diwali seasons.

By achieving these objectives, the project aims to provide a comprehensive understanding of the dynamics at play during the Diwali sales period and offer practical solutions for businesses to maximize their sales potential.

# Project Category

**Data Analysis and Visualization**

The Diwali Sales Analysis project is categorized under Data Analysis and Visualization. This field focuses on extracting, processing, and interpreting large datasets to uncover meaningful insights. It also involves creating visual representations to illustrate data trends and patterns effectively. The goal is to transform raw data into actionable insights that inform strategic business decisions, particularly during high-impact periods like the Diwali festive season.

Visualizations were created to support the analysis, including line charts, bar graphs, pie charts, and heatmaps. Key findings include:

* Peak sales occurred in the week leading up to Diwali.
* Electronics and clothing were the top-selling categories.
* Younger customers (ages 18-35) were the most active buyers during the Diwali season.

# Tools and Platform Used

## Tools:

Python: The primary programming language used for data manipulation, analysis, and visualization.

Jupyter Notebook: An interactive environment for writing and running code, visualizing data, and documenting the analysis process.

NumPy: A library for numerical operations, essential for efficient data handling and calculations.

Pandas: A powerful data manipulation library used for data cleaning, transformation, and analysis.

Matplotlib: A plotting library used to create static, animated, and interactive visualizations.

Seaborn: A statistical data visualization library built on Matplotlib, used to create attractive and informative graphics.

## Platform:

**Jupyter Notebook**: This platform allows for an interactive coding environment where analysis and visualizations can be seamlessly integrated with narrative text. It supports data cleaning, transformation, and visualization, making it an ideal choice for data analysis projects.

# Technical Details

1. Data Collection and Preprocessing:

Data Sources: Sales data collected from various retail stores over past Diwali seasons, including details on product categories, sales volumes, transaction dates, customer demographics, and revenue.

Data Cleaning: Handling missing values, removing duplicates, and correcting inconsistencies to ensure data accuracy and reliability.

Data Transformation: Converting data types, standardizing date formats, and creating new features such as sales week and day for more detailed analysis.

2. Data Analysis:

Exploratory Data Analysis (EDA): Initial analysis to understand the distribution of data, identify trends, and detect any anomalies or outliers.

Time Series Analysis: Identifying sales trends over the Diwali season to pinpoint peak sales periods.

Category-wise Analysis: Examining sales across different product categories to determine the most popular items.

Customer Demographics Analysis: Analyzing purchasing behavior based on age, gender, and location to tailor marketing strategies.

Correlation Analysis: Exploring relationships between various variables to uncover factors influencing sales.

3. Data Visualization:

Line Charts: Used for time series analysis to visualize sales trends over time.

Bar Graphs: Used to compare sales volumes across different product categories.

Pie Charts: Used to show the distribution of sales among different customer demographics.

Heatmaps: Used to visualize correlations between different variables.

4. Insights and Recommendations:

Actionable Insights: Providing recommendations based on analysis findings, such as optimal stock management, targeted marketing strategies, and effective promotional tactics.

Report Generation: Compiling analysis results and visualizations into a comprehensive report that outlines key findings and business recommendations.

By employing these tools and techniques, the Diwali Sales Analysis project aims to offer a deep understanding of consumer behavior and sales trends during the Diwali season. The insights gained will help retailers optimize their strategies, enhance customer engagement, and improve overall sales performance during this critical period.

# Hardware & Software Requirements

**Hardware Requirements**

To efficiently run the Diwali Sales Analysis project, the following hardware specifications are recommended:

**Processor:**

Minimum: Intel Core i5 or AMD Ryzen 5

Recommended: Intel Core i7 or AMD Ryzen 7 for faster data processing and smoother performance.

**Memory (RAM):**

Minimum: 8 GB

Recommended: 16 GB or higher to handle large datasets and perform complex data manipulations without performance issues.

**Storage:**

Minimum: 256 GB SSD

Recommended: 512 GB SSD or higher to ensure quick access to data and faster read/write operations.

**Graphics Card:**

Optional but recommended for data visualization tasks that can benefit from GPU acceleration.

**Display:**

A Full HD (1920x1080) monitor or higher resolution for better visualization of data plots and graphs.

**Internet Connection:**

A stable internet connection for downloading necessary libraries, datasets, and for any cloud-based operations.

Software Requirements

The following software tools and libraries are necessary for the development and execution of the Diwali Sales Analysis project:

**Operating System:**

Windows 10/11, macOS, or Linux.

**Python:**

Version: Python 3.7 or higher.

**Integrated Development Environment (IDE):**

**Jupyter Notebook:** For an interactive coding and documentation environment.

Anaconda Distribution (optional): For easy installation of Python and Jupyter Notebook along with the required libraries.

**Python Libraries:**

NumPy: For numerical operations and handling arrays.

Pandas: For data manipulation and analysis.

Matplotlib: For creating static, animated, and interactive visualizations.

Seaborn: For statistical data visualization built on top of Matplotlib.

**Additional Tools:**

scikit-learn: For potential advanced data analysis and machine learning models.

Plotly: For interactive and web-based visualizations (optional).

**Version Control:**

Git: For version control and collaboration.

**Web Browser:**

Google Chrome, Mozilla Firefox, or any modern web browser for running Jupyter Notebook and accessing online resources.

Installation Steps

**Python and Anaconda:**

Download and install Anaconda Distribution from Anaconda's official website, which includes Python, Jupyter Notebook, and essential libraries.

Libraries Installation:

Open Anaconda Prompt (or command line if not using Anaconda) and install the required libraries using pip:

bash

Copy code

pip install numpy pandas matplotlib seaborn scikit-learn plotly

Setting Up Jupyter Notebook:

Launch Jupyter Notebook from Anaconda Navigator or by running the following command in the Anaconda Prompt:

bash

Copy code

jupyter notebook

By ensuring these hardware and software requirements are met, you will be well-equipped to conduct a thorough and efficient analysis of Diwali sales, gaining valuable insights into consumer behavior and sales trends.

# FEASIBILITY STUDY

## Technical Feasibility

* 1. **Hardware Requirements:** The hardware requirements for the Diwali Sales Analysis project are modest and within the reach of most modern computing systems. An Intel Core i5 or AMD Ryzen 5 processor with at least 8 GB of RAM and 256 GB of SSD storage is sufficient for handling the data and performing the necessary analyses. These specifications are commonly available in standard desktop and laptop computers, ensuring that the project can be executed without significant hardware upgrades.

**1.2. Software Requirements:** The software requirements include Python, Jupyter Notebook, and various Python libraries such as NumPy, Pandas, Matplotlib, and Seaborn. These tools are all open-source and freely available, reducing the cost burden. Additionally, the installation and setup process for these tools is straightforward, especially with the use of the Anaconda distribution, which simplifies the management of Python environments and packages.

**1.3. Data Accessibility:** The data required for this project includes past sales records during Diwali, which can typically be obtained from retail stores or e-commerce platforms. Assuming access to this data is secured, the technical feasibility is high as the necessary tools and computational power are readily available.

## Economic Feasibility

**2.1. Cost of Tools and Technologies:** The project leverages open-source tools such as Python, Jupyter Notebook, and various Python libraries, which are all free to use. This significantly reduces the economic burden of software procurement. Additionally, any additional software (e.g., version control tools like Git) is also available for free.

**2.2. Hardware Costs:** Most modern computers meet the recommended hardware specifications. If an upgrade is necessary, the cost is relatively low compared to the potential benefits. An investment in upgrading RAM or storage would be minimal and provide long-term benefits for other data analysis tasks as well.

**2.3. Human Resources:** The project requires individuals with skills in data analysis, Python programming, and data visualization. Hiring or training employees to acquire these skills represents an investment, but the return on investment (ROI) is high due to the actionable insights that can drive increased sales and improved business strategies.

## 3. Operational Feasibility

**3.1. Skill Set Availability:** The required skill set includes proficiency in Python, data analysis, and visualization techniques. These skills are widely taught in data science and analytics courses, and there is a large talent pool of data analysts and scientists who can carry out this project. Additionally, numerous online resources and tutorials are available to support learning and implementation.

**3.2. Implementation Timeline:** The project can be divided into distinct phases: data collection, data preprocessing, exploratory data analysis, detailed analysis, and reporting. Each phase can be managed within a reasonable timeframe, allowing the project to be completed in a few months, depending on the dataset's size and complexity.

**3.3. Integration with Business Operations:** The insights gained from this analysis can be seamlessly integrated into the business operations of retail stores. Recommendations based on the analysis, such as inventory management, targeted marketing, and promotional strategies, can be implemented with minimal disruption to existing processes.

## 4. Legal and Ethical Feasibility

**4.1. Data Privacy:** Ensuring the privacy and security of customer data is paramount. The project must comply with data protection regulations such as GDPR or India's Personal Data Protection Bill. Anonymizing data and securing storage systems can mitigate privacy risks.

**4.2. Ethical Considerations:** The analysis should be conducted ethically, ensuring transparency and fairness in how data is handled and reported. The insights should be used to benefit both the business and its customers, enhancing the shopping experience without exploiting consumer behavior.

Conclusion

The Diwali Sales Analysis project is highly feasible from technical, economic, operational, and legal perspectives. The required hardware and software are accessible and affordable, the necessary skills are widely available, and the project can provide significant economic benefits through improved business strategies. Ensuring compliance with legal standards and maintaining ethical practices will further support the successful implementation of the project.

# System Design

## 1. Overview

The Diwali Sales Analysis project aims to analyze sales data to uncover trends, patterns, and insights that can help retailers optimize their strategies during the Diwali festive season. The system design encompasses data collection, processing, analysis, and visualization components, integrating various tools and technologies to achieve the project objectives.

## 2. System Architecture

The system architecture consists of several interconnected components that work together to process and analyze the sales data. The primary components include:

* Data Collection
* Data Storage
* Data Processing and Analysis
* Data Visualization
* Reporting

## 3. Components and Workflow

**3.1. Data Collection**

Sources of Data:

* Retail stores' sales databases
* E-commerce platforms' sales records
* Customer demographics and transaction history
* Data Collection Methods:
* Database exports (CSV, Excel)
* APIs for real-time data retrieval
* Web scraping (if necessary)

**3.2. Data Storage**

Data Storage Solutions:

* Local storage: Storing datasets on a local machine for analysis
* Cloud storage: Using cloud services (e.g., AWS S3, Google Cloud Storage) for scalability and accessibility
* Data Storage Format:
* CSV files
* Excel sheets
* SQL databases for structured storage

**3.3. Data Processing and Analysis**

Data Processing Steps:

* Data Cleaning: Removing duplicates, handling missing values, and standardizing formats
* Data Transformation: Converting data types, creating new features, and normalizing data
* Exploratory Data Analysis (EDA): Initial analysis to understand data distributions, detect anomalies, and

identify trends

**3.4. Data Visualization**

**Visualization Techniques:**

* Line Charts: To visualize sales trends over time
* Bar Graphs: To compare sales volumes across product categories
* Pie Charts: To show the distribution of sales among customer demographics
* Heatmaps: To visualize correlations between variables

**Tools:**

**Matplotlib and Seaborn:** For creating a variety of static and interactive plots

**Plotly:** For more interactive and web-based visualizations (optional)

**3.5. Reporting**

Report Generation:

Compile analysis results and visualizations into a comprehensive report

Use Jupyter Notebook to combine code, analysis, and narrative in a single document

Export reports in formats like HTML, PDF, or Jupyter Notebook (.ipynb) for sharing

## Presentation Tools:

**PowerPoint or Google Slides:** For creating presentations to communicate findings

**Dash by Plotly (optional):** For creating interactive web-based dashboards

**4. Data Flow Diagram (DFD)**

**Level 0: Context Diagram**

Input: Raw sales data from various sources

Process: Data collection, storage, processing, analysis, and visualization

Output: Actionable insights and reports

**Level 1: Detailed DFD**

Data Collection: Import data from sources -> Store raw data in storage solution

Data Processing: Clean and transform data -> Perform EDA

Data Analysis: Conduct detailed analysis -> Generate visualizations

Reporting: Compile results -> Create and share reports

**5. System Requirements**

**Hardware**:

* Processor: Intel Core i5/Ryzen 5 or better
* Memory: Minimum 8 GB RAM (16 GB recommended)
* Storage: Minimum 256 GB SSD (512 GB recommended)
* Internet Connection: Stable connection for data retrieval and software updates

**Software:**

* Operating System: Windows 10/11, macOS, or Linux
* Python: Version 3.7 or higher
* IDE: Jupyter Notebook (via Anaconda distribution)

**Libraries:** NumPy, Pandas, Matplotlib, Seaborn, scikit-learn (optional)

**Version Control: Git (optional), GitHub**

**6. Security and Privacy**

**Data Privacy:**

* Ensure compliance with data protection regulations (e.g., GDPR)
* Anonymize customer data to protect privacy

**Data Security**:

* Use encryption for data storage and transmission
* Implement access controls to restrict data access to authorized personnel

Conclusion

The system design for the Diwali Sales Analysis project outlines a comprehensive and structured approach to collecting, processing, analyzing, and visualizing sales data. By leveraging robust tools and techniques, this design ensures that valuable insights are derived to help retailers optimize their strategies during the Diwali festive season. The integration of hardware and software components is planned to achieve efficient and effective data analysis, providing a solid foundation for informed decision-making.

# Tools and Technologies:

**Python**: The primary language for data processing and analysis

**Jupyter** Notebook: An interactive environment for coding, visualizing data, and documenting the analysis

## Libraries:

**NumPy**: For numerical operations and handling arrays

**Pandas**: For data manipulation, cleaning, and transformation

**Matplotlib**: For creating static visualizations

**Seaborn**: For advanced statistical visualizations

**scikit**-**learn**: For potential machine learning models (optional)

# MODULE DESCRIPTION

## Module Description

The Diwali Sales Analysis project is divided into several modules, each responsible for specific tasks in the data analysis pipeline. These modules work together to collect, process, analyze, and visualize sales data, ultimately providing actionable insights to retailers. Below is a description of each module:

**1. Data Collection Module**

Description: This module is responsible for collecting sales data from various sources, including retail stores' databases, e-commerce platforms, and customer demographics. It employs methods such as database exports, APIs, and web scraping to gather relevant data.

**2. Data Preprocessing Module**

Description: The Data Preprocessing Module cleans and transforms the raw sales data to ensure accuracy and consistency. Tasks include removing duplicates, handling missing values, standardizing formats, and creating new features for analysis.

**3. Exploratory Data Analysis (EDA) Module**

Description: EDA is conducted to gain initial insights into the data. This module explores data distributions, detects anomalies, identifies trends, and visualizes relationships between variables using techniques such as statistical summaries, histograms, and scatter plots.

**4. Detailed Analysis Module**

Description: The Detailed Analysis Module delves deeper into the data to extract meaningful insights. It conducts advanced statistical analyses, time series analysis, category-wise analysis, and customer demographics analysis to identify patterns and trends in sales data.

**5. Data Visualization Module**

Description: This module creates visual representations of the analysis results to facilitate understanding and communication. It utilizes techniques such as line charts, bar graphs, pie charts, and heatmaps to visualize sales trends, category-wise sales, customer demographics, and correlations between variables.

**6. Reporting Module**

Description: The Reporting Module compiles the analysis results and visualizations into a comprehensive report. It integrates code, analysis, and narrative text using tools such as Jupyter Notebook, and exports reports in formats like HTML, PDF, or Jupyter Notebook (.ipynb) for sharing with stakeholders.

**7. Integration and Deployment Module**

Description: The Integration and Deployment Module integrates the individual modules into a cohesive system and deploys it for use. It ensures seamless communication between modules and provides documentation for system usage and maintenance.

**8. Security and Compliance Module**

Description: This module addresses security and compliance considerations, ensuring that data privacy regulations are adhered to. It implements encryption, access controls, and anonymization techniques to protect sensitive data and maintain compliance with legal standards.

Each module plays a crucial role in the overall data analysis process, contributing to the generation of actionable insights that can inform strategic decision-making for retailers during the Diwali festive season.

# CODING

**{**

**"cells": [**

**{**

**"cell\_type": "code",**

**"execution\_count": 4,**

**"id": "aa2b25c4-5f5b-4af3-a415-5f767ed27aa2",**

**"metadata": {},**

**"outputs": [**

**{**

**"name": "stdout",**

**"output\_type": "stream",**

**"text": [**

**"Requirement already satisfied: matplotlib in c:\\users\\frz\\appdata\\local\\programs\\python\\python312\\lib\\site-packages (3.9.0)\n",**

**"Requirement already satisfied: contourpy>=1.0.1 in c:\\users\\frz\\appdata\\local\\programs\\python\\python312\\lib\\site-packages (from matplotlib) (1.2.1)\n",**

**"Requirement already satisfied: cycler>=0.10 in c:\\users\\frz\\appdata\\local\\programs\\python\\python312\\lib\\site-packages (from matplotlib) (0.12.1)\n",**

**"Requirement already satisfied: fonttools>=4.22.0 in c:\\users\\frz\\appdata\\local\\programs\\python\\python312\\lib\\site-packages (from matplotlib) (4.51.0)\n",**

**"Requirement already satisfied: kiwisolver>=1.3.1 in c:\\users\\frz\\appdata\\local\\programs\\python\\python312\\lib\\site-packages (from matplotlib) (1.4.5)\n",**

**"Requirement already satisfied: numpy>=1.23 in c:\\users\\frz\\appdata\\local\\programs\\python\\python312\\lib\\site-packages (from matplotlib) (1.26.4)\n",**

**"Requirement already satisfied: packaging>=20.0 in c:\\users\\frz\\appdata\\local\\programs\\python\\python312\\lib\\site-packages (from matplotlib) (24.0)\n",**

**"Requirement already satisfied: pillow>=8 in c:\\users\\frz\\appdata\\local\\programs\\python\\python312\\lib\\site-packages (from matplotlib) (10.3.0)\n",**

**"Requirement already satisfied: pyparsing>=2.3.1 in c:\\users\\frz\\appdata\\local\\programs\\python\\python312\\lib\\site-packages (from matplotlib) (3.1.2)\n",**

**"Requirement already satisfied: python-dateutil>=2.7 in c:\\users\\frz\\appdata\\local\\programs\\python\\python312\\lib\\site-packages (from matplotlib) (2.9.0.post0)\n",**

**"Requirement already satisfied: six>=1.5 in c:\\users\\frz\\appdata\\local\\programs\\python\\python312\\lib\\site-packages (from python-dateutil>=2.7->matplotlib) (1.16.0)\n"**

**]**

**}**

**],**

**"source": [**

**"!pip install matplotlib"**

**]**

**},**

**{**

**"cell\_type": "code",**

**"execution\_count": 1,**

**"id": "0979edbb",**

**"metadata": {},**

**"outputs": [],**

**"source": [**

**"# import python libraries\n",**

**"\n",**

**"import numpy as np \n",**

**"import pandas as pd \n",**

**"import matplotlib.pyplot as plt # visualizing data\n",**

**"%matplotlib inline\n",**

**"import seaborn as sns"**

**]**

**},**

**{**

**"cell\_type": "code",**

**"execution\_count": 2,**

**"id": "7fbfb436",**

**"metadata": {},**

**"outputs": [],**

**"source": [**

**"# import csv file\n",**

**"df = pd.read\_csv('Diwali Sales Data.csv', encoding= 'unicode\_escape')"**

**]**

**},**

**{**

**"cell\_type": "code",**

**"execution\_count": 3,**

**"id": "b8db4a5b",**

**"metadata": {},**

**"outputs": [**

**{**

**"data": {**

**"text/plain": [**

**"(11251, 15)"**

**]**

**},**

**"execution\_count": 3,**

**"metadata": {},**

**"output\_type": "execute\_result"**

**}**

**],**

**"source": [**

**"df.shape"**

**]**

**},**

**{**

**"cell\_type": "code",**

**"execution\_count": 4,**

**"id": "662c471c",**

**"metadata": {},**

**"outputs": [**

**{**

**"data": {**

**"text/html": [**

**"<div>\n",**

**"<style scoped>\n",**

**" .dataframe tbody tr th:only-of-type {\n",**

**" vertical-align: middle;\n",**

**" }\n",**

**"\n",**

**" .dataframe tbody tr th {\n",**

**" vertical-align: top;\n",**

**" }\n",**

**"\n",**

**" .dataframe thead th {\n",**

**" text-align: right;\n",**

**" }\n",**

**"</style>\n",**

**"<table border=\"1\" class=\"dataframe\">\n",**

**" <thead>\n",**

**" <tr style=\"text-align: right;\">\n",**

**" <th></th>\n",**

**" <th>User\_ID</th>\n",**

**" <th>Cust\_name</th>\n",**

**" <th>Product\_ID</th>\n",**

**" <th>Gender</th>\n",**

**" <th>Age Group</th>\n",**

**" <th>Age</th>\n",**

**" <th>Marital\_Status</th>\n",**

**" <th>State</th>\n",**

**" <th>Zone</th>\n",**

**" <th>Occupation</th>\n",**

**" <th>Product\_Category</th>\n",**

**" <th>Orders</th>\n",**

**" <th>Amount</th>\n",**

**" <th>Status</th>\n",**

**" <th>unnamed1</th>\n",**

**" </tr>\n",**

**" </thead>\n",**

**" <tbody>\n",**

**" <tr>\n",**

**" <th>0</th>\n",**

**" <td>1002903</td>\n",**

**" <td>Sanskriti</td>\n",**

**" <td>P00125942</td>\n",**

**" <td>F</td>\n",**

**" <td>26-35</td>\n",**

**" <td>28</td>\n",**

**" <td>0</td>\n",**

**" <td>Maharashtra</td>\n",**

**" <td>Western</td>\n",**

**" <td>Healthcare</td>\n",**

**" <td>Auto</td>\n",**

**" <td>1</td>\n",**

**" <td>23952.0</td>\n",**

**" <td>NaN</td>\n",**

**" <td>NaN</td>\n",**

**" </tr>\n",**

**" <tr>\n",**

**" <th>1</th>\n",**

**" <td>1000732</td>\n",**

**" <td>Kartik</td>\n",**

**" <td>P00110942</td>\n",**

**" <td>F</td>\n",**

**" <td>26-35</td>\n",**

**" <td>35</td>\n",**

**" <td>1</td>\n",**

**" <td>Andhra Pradesh</td>\n",**

**" <td>Southern</td>\n",**

**" <td>Govt</td>\n",**

**" <td>Auto</td>\n",**

**" <td>3</td>\n",**

**" <td>23934.0</td>\n",**

**" <td>NaN</td>\n",**

**" <td>NaN</td>\n",**

**" </tr>\n",**

**" <tr>\n",**

**" <th>2</th>\n",**

**" <td>1001990</td>\n",**

**" <td>Bindu</td>\n",**

**" <td>P00118542</td>\n",**

**" <td>F</td>\n",**

**" <td>26-35</td>\n",**

**" <td>35</td>\n",**

**" <td>1</td>\n",**

**" <td>Uttar Pradesh</td>\n",**

**" <td>Central</td>\n",**

**" <td>Automobile</td>\n",**

**" <td>Auto</td>\n",**

**" <td>3</td>\n",**

**" <td>23924.0</td>\n",**

**" <td>NaN</td>\n",**

**" <td>NaN</td>\n",**

**" </tr>\n",**

**" <tr>\n",**

**" <th>3</th>\n",**

**" <td>1001425</td>\n",**

**" <td>Sudevi</td>\n",**

**" <td>P00237842</td>\n",**

**" <td>M</td>\n",**

**" <td>0-17</td>\n",**

**" <td>16</td>\n",**

**" <td>0</td>\n",**

**" <td>Karnataka</td>\n",**

**" <td>Southern</td>\n",**

**" <td>Construction</td>\n",**

**" <td>Auto</td>\n",**

**" <td>2</td>\n",**

**" <td>23912.0</td>\n",**

**" <td>NaN</td>\n",**

**" <td>NaN</td>\n",**

**" </tr>\n",**

**" <tr>\n",**

**" <th>4</th>\n",**

**" <td>1000588</td>\n",**

**" <td>Joni</td>\n",**

**" <td>P00057942</td>\n",**

**" <td>M</td>\n",**

**" <td>26-35</td>\n",**

**" <td>28</td>\n",**

**" <td>1</td>\n",**

**" <td>Gujarat</td>\n",**

**" <td>Western</td>\n",**

**" <td>Food Processing</td>\n",**

**" <td>Auto</td>\n",**

**" <td>2</td>\n",**

**" <td>23877.0</td>\n",**

**" <td>NaN</td>\n",**

**" <td>NaN</td>\n",**

**" </tr>\n",**

**" </tbody>\n",**

**"</table>\n",**

**"</div>"**

**],**

**"text/plain": [**

**" User\_ID Cust\_name Product\_ID Gender Age Group Age Marital\_Status \\\n",**

**"0 1002903 Sanskriti P00125942 F 26-35 28 0 \n",**

**"1 1000732 Kartik P00110942 F 26-35 35 1 \n",**

**"2 1001990 Bindu P00118542 F 26-35 35 1 \n",**

**"3 1001425 Sudevi P00237842 M 0-17 16 0 \n",**

**"4 1000588 Joni P00057942 M 26-35 28 1 \n",**

**"\n",**

**" State Zone Occupation Product\_Category Orders \\\n",**

**"0 Maharashtra Western Healthcare Auto 1 \n",**

**"1 Andhra Pradesh Southern Govt Auto 3 \n",**

**"2 Uttar Pradesh Central Automobile Auto 3 \n",**

**"3 Karnataka Southern Construction Auto 2 \n",**

**"4 Gujarat Western Food Processing Auto 2 \n",**

**"\n",**

**" Amount Status unnamed1 \n",**

**"0 23952.0 NaN NaN \n",**

**"1 23934.0 NaN NaN \n",**

**"2 23924.0 NaN NaN \n",**

**"3 23912.0 NaN NaN \n",**

**"4 23877.0 NaN NaN "**

**]**

**},**

**"execution\_count": 4,**

**"metadata": {},**

**"output\_type": "execute\_result"**

**}**

**],**

**"source": [**

**"df.head()"**

**]**

**},**

**{**

**"cell\_type": "code",**

**"execution\_count": 5,**

**"id": "0814c45b",**

**"metadata": {},**

**"outputs": [**

**{**

**"name": "stdout",**

**"output\_type": "stream",**

**"text": [**

**"<class 'pandas.core.frame.DataFrame'>\n",**

**"RangeIndex: 11251 entries, 0 to 11250\n",**

**"Data columns (total 15 columns):\n",**

**" # Column Non-Null Count Dtype \n",**

**"--- ------ -------------- ----- \n",**

**" 0 User\_ID 11251 non-null int64 \n",**

**" 1 Cust\_name 11251 non-null object \n",**

**" 2 Product\_ID 11251 non-null object \n",**

**" 3 Gender 11251 non-null object \n",**

**" 4 Age Group 11251 non-null object \n",**

**" 5 Age 11251 non-null int64 \n",**

**" 6 Marital\_Status 11251 non-null int64 \n",**

**" 7 State 11251 non-null object \n",**

**" 8 Zone 11251 non-null object \n",**

**" 9 Occupation 11251 non-null object \n",**

**" 10 Product\_Category 11251 non-null object \n",**

**" 11 Orders 11251 non-null int64 \n",**

**" 12 Amount 11239 non-null float64\n",**

**" 13 Status 0 non-null float64\n",**

**" 14 unnamed1 0 non-null float64\n",**

**"dtypes: float64(3), int64(4), object(8)\n",**

**"memory usage: 1.3+ MB\n"**

**]**

**}**

**],**

**"source": [**

**"df.info()"**

**]**

**},**

**{**

**"cell\_type": "code",**

**"execution\_count": 6,**

**"id": "29afa95b",**

**"metadata": {},**

**"outputs": [],**

**"source": [**

**"#drop unrelated/blank columns\n",**

**"df.drop(['Status', 'unnamed1'], axis=1, inplace=True)"**

**]**

**},**

**{**

**"cell\_type": "code",**

**"execution\_count": 7,**

**"id": "5cee58eb",**

**"metadata": {},**

**"outputs": [**

**{**

**"data": {**

**"text/plain": [**

**"User\_ID 0\n",**

**"Cust\_name 0\n",**

**"Product\_ID 0\n",**

**"Gender 0\n",**

**"Age Group 0\n",**

**"Age 0\n",**

**"Marital\_Status 0\n",**

**"State 0\n",**

**"Zone 0\n",**

**"Occupation 0\n",**

**"Product\_Category 0\n",**

**"Orders 0\n",**

**"Amount 12\n",**

**"dtype: int64"**

**]**

**},**

**"execution\_count": 7,**

**"metadata": {},**

**"output\_type": "execute\_result"**

**}**

**],**

**"source": [**

**"#check for null values\n",**

**"pd.isnull(df).sum()"**

**]**

**},**

**{**

**"cell\_type": "code",**

**"execution\_count": 8,**

**"id": "65de74e3",**

**"metadata": {},**

**"outputs": [],**

**"source": [**

**"# drop null values\n",**

**"df.dropna(inplace=True)"**

**]**

**},**

**{**

**"cell\_type": "code",**

**"execution\_count": 9,**

**"id": "14e4ff48",**

**"metadata": {},**

**"outputs": [],**

**"source": [**

**"# change data type\n",**

**"df['Amount'] = df['Amount'].astype('int')"**

**]**

**},**

**{**

**"cell\_type": "code",**

**"execution\_count": 10,**

**"id": "aa6752dc",**

**"metadata": {},**

**"outputs": [**

**{**

**"data": {**

**"text/plain": [**

**"dtype('int32')"**

**]**

**},**

**"execution\_count": 10,**

**"metadata": {},**

**"output\_type": "execute\_result"**

**}**

**],**

**"source": [**

**"df['Amount'].dtypes"**

**]**

**},**

**{**

**"cell\_type": "code",**

**"execution\_count": 11,**

**"id": "5f723a99",**

**"metadata": {},**

**"outputs": [**

**{**

**"data": {**

**"text/plain": [**

**"Index(['User\_ID', 'Cust\_name', 'Product\_ID', 'Gender', 'Age Group', 'Age',\n",**

**" 'Marital\_Status', 'State', 'Zone', 'Occupation', 'Product\_Category',\n",**

**" 'Orders', 'Amount'],\n",**

**" dtype='object')"**

**]**

**},**

**"execution\_count": 11,**

**"metadata": {},**

**"output\_type": "execute\_result"**

**}**

**],**

**"source": [**

**"df.columns"**

**]**

**},**

**{**

**"cell\_type": "code",**

**"execution\_count": 12,**

**"id": "a85bb3c3",**

**"metadata": {},**

**"outputs": [**

**{**

**"data": {**

**"text/html": [**

**"<div>\n",**

**"<style scoped>\n",**

**" .dataframe tbody tr th:only-of-type {\n",**

**" vertical-align: middle;\n",**

**" }\n",**

**"\n",**

**" .dataframe tbody tr th {\n",**

**" vertical-align: top;\n",**

**" }\n",**

**"\n",**

**" .dataframe thead th {\n",**

**" text-align: right;\n",**

**" }\n",**

**"</style>\n",**

**"<table border=\"1\" class=\"dataframe\">\n",**

**" <thead>\n",**

**" <tr style=\"text-align: right;\">\n",**

**" <th></th>\n",**

**" <th>User\_ID</th>\n",**

**" <th>Cust\_name</th>\n",**

**" <th>Product\_ID</th>\n",**

**" <th>Gender</th>\n",**

**" <th>Age Group</th>\n",**

**" <th>Age</th>\n",**

**" <th>Shaadi</th>\n",**

**" <th>State</th>\n",**

**" <th>Zone</th>\n",**

**" <th>Occupation</th>\n",**

**" <th>Product\_Category</th>\n",**

**" <th>Orders</th>\n",**

**" <th>Amount</th>\n",**

**" </tr>\n",**

**" </thead>\n",**

**" <tbody>\n",**

**" <tr>\n",**

**" <th>0</th>\n",**

**" <td>1002903</td>\n",**

**" <td>Sanskriti</td>\n",**

**" <td>P00125942</td>\n",**

**" <td>F</td>\n",**

**" <td>26-35</td>\n",**

**" <td>28</td>\n",**

**" <td>0</td>\n",**

**" <td>Maharashtra</td>\n",**

**" <td>Western</td>\n",**

**" <td>Healthcare</td>\n",**

**" <td>Auto</td>\n",**

**" <td>1</td>\n",**

**" <td>23952</td>\n",**

**" </tr>\n",**

**" <tr>\n",**

**" <th>1</th>\n",**

**" <td>1000732</td>\n",**

**" <td>Kartik</td>\n",**

**" <td>P00110942</td>\n",**

**" <td>F</td>\n",**

**" <td>26-35</td>\n",**

**" <td>35</td>\n",**

**" <td>1</td>\n",**

**" <td>Andhra Pradesh</td>\n",**

**" <td>Southern</td>\n",**

**" <td>Govt</td>\n",**

**" <td>Auto</td>\n",**

**" <td>3</td>\n",**

**" <td>23934</td>\n",**

**" </tr>\n",**

**" <tr>\n",**

**" <th>2</th>\n",**

**" <td>1001990</td>\n",**

**" <td>Bindu</td>\n",**

**" <td>P00118542</td>\n",**

**" <td>F</td>\n",**

**" <td>26-35</td>\n",**

**" <td>35</td>\n",**

**" <td>1</td>\n",**

**" <td>Uttar Pradesh</td>\n",**

**" <td>Central</td>\n",**

**" <td>Automobile</td>\n",**

**" <td>Auto</td>\n",**

**" <td>3</td>\n",**

**" <td>23924</td>\n",**

**" </tr>\n",**

**" <tr>\n",**

**" <th>3</th>\n",**

**" <td>1001425</td>\n",**

**" <td>Sudevi</td>\n",**

**" <td>P00237842</td>\n",**

**" <td>M</td>\n",**

**" <td>0-17</td>\n",**

**" <td>16</td>\n",**

**" <td>0</td>\n",**

**" <td>Karnataka</td>\n",**

**" <td>Southern</td>\n",**

**" <td>Construction</td>\n",**

**" <td>Auto</td>\n",**

**" <td>2</td>\n",**

**" <td>23912</td>\n",**

**" </tr>\n",**

**" <tr>\n",**

**" <th>4</th>\n",**

**" <td>1000588</td>\n",**

**" <td>Joni</td>\n",**

**" <td>P00057942</td>\n",**

**" <td>M</td>\n",**

**" <td>26-35</td>\n",**

**" <td>28</td>\n",**

**" <td>1</td>\n",**

**" <td>Gujarat</td>\n",**

**" <td>Western</td>\n",**

**" <td>Food Processing</td>\n",**

**" <td>Auto</td>\n",**

**" <td>2</td>\n",**

**" <td>23877</td>\n",**

**" </tr>\n",**

**" <tr>\n",**

**" <th>...</th>\n",**

**" <td>...</td>\n",**

**" <td>...</td>\n",**

**" <td>...</td>\n",**

**" <td>...</td>\n",**

**" <td>...</td>\n",**

**" <td>...</td>\n",**

**" <td>...</td>\n",**

**" <td>...</td>\n",**

**" <td>...</td>\n",**

**" <td>...</td>\n",**

**" <td>...</td>\n",**

**" <td>...</td>\n",**

**" <td>...</td>\n",**

**" </tr>\n",**

**" <tr>\n",**

**" <th>11246</th>\n",**

**" <td>1000695</td>\n",**

**" <td>Manning</td>\n",**

**" <td>P00296942</td>\n",**

**" <td>M</td>\n",**

**" <td>18-25</td>\n",**

**" <td>19</td>\n",**

**" <td>1</td>\n",**

**" <td>Maharashtra</td>\n",**

**" <td>Western</td>\n",**

**" <td>Chemical</td>\n",**

**" <td>Office</td>\n",**

**" <td>4</td>\n",**

**" <td>370</td>\n",**

**" </tr>\n",**

**" <tr>\n",**

**" <th>11247</th>\n",**

**" <td>1004089</td>\n",**

**" <td>Reichenbach</td>\n",**

**" <td>P00171342</td>\n",**

**" <td>M</td>\n",**

**" <td>26-35</td>\n",**

**" <td>33</td>\n",**

**" <td>0</td>\n",**

**" <td>Haryana</td>\n",**

**" <td>Northern</td>\n",**

**" <td>Healthcare</td>\n",**

**" <td>Veterinary</td>\n",**

**" <td>3</td>\n",**

**" <td>367</td>\n",**

**" </tr>\n",**

**" <tr>\n",**

**" <th>11248</th>\n",**

**" <td>1001209</td>\n",**

**" <td>Oshin</td>\n",**

**" <td>P00201342</td>\n",**

**" <td>F</td>\n",**

**" <td>36-45</td>\n",**

**" <td>40</td>\n",**

**" <td>0</td>\n",**

**" <td>Madhya Pradesh</td>\n",**

**" <td>Central</td>\n",**

**" <td>Textile</td>\n",**

**" <td>Office</td>\n",**

**" <td>4</td>\n",**

**" <td>213</td>\n",**

**" </tr>\n",**

**" <tr>\n",**

**" <th>11249</th>\n",**

**" <td>1004023</td>\n",**

**" <td>Noonan</td>\n",**

**" <td>P00059442</td>\n",**

**" <td>M</td>\n",**

**" <td>36-45</td>\n",**

**" <td>37</td>\n",**

**" <td>0</td>\n",**

**" <td>Karnataka</td>\n",**

**" <td>Southern</td>\n",**

**" <td>Agriculture</td>\n",**

**" <td>Office</td>\n",**

**" <td>3</td>\n",**

**" <td>206</td>\n",**

**" </tr>\n",**

**" <tr>\n",**

**" <th>11250</th>\n",**

**" <td>1002744</td>\n",**

**" <td>Brumley</td>\n",**

**" <td>P00281742</td>\n",**

**" <td>F</td>\n",**

**" <td>18-25</td>\n",**

**" <td>19</td>\n",**

**" <td>0</td>\n",**

**" <td>Maharashtra</td>\n",**

**" <td>Western</td>\n",**

**" <td>Healthcare</td>\n",**

**" <td>Office</td>\n",**

**" <td>3</td>\n",**

**" <td>188</td>\n",**

**" </tr>\n",**

**" </tbody>\n",**

**"</table>\n",**

**"<p>11239 rows × 13 columns</p>\n",**

**"</div>"**

**],**

**"text/plain": [**

**" User\_ID Cust\_name Product\_ID Gender Age Group Age Shaadi \\\n",**

**"0 1002903 Sanskriti P00125942 F 26-35 28 0 \n",**

**"1 1000732 Kartik P00110942 F 26-35 35 1 \n",**

**"2 1001990 Bindu P00118542 F 26-35 35 1 \n",**

**"3 1001425 Sudevi P00237842 M 0-17 16 0 \n",**

**"4 1000588 Joni P00057942 M 26-35 28 1 \n",**

**"... ... ... ... ... ... ... ... \n",**

**"11246 1000695 Manning P00296942 M 18-25 19 1 \n",**

**"11247 1004089 Reichenbach P00171342 M 26-35 33 0 \n",**

**"11248 1001209 Oshin P00201342 F 36-45 40 0 \n",**

**"11249 1004023 Noonan P00059442 M 36-45 37 0 \n",**

**"11250 1002744 Brumley P00281742 F 18-25 19 0 \n",**

**"\n",**

**" State Zone Occupation Product\_Category Orders \\\n",**

**"0 Maharashtra Western Healthcare Auto 1 \n",**

**"1 Andhra Pradesh Southern Govt Auto 3 \n",**

**"2 Uttar Pradesh Central Automobile Auto 3 \n",**

**"3 Karnataka Southern Construction Auto 2 \n",**

**"4 Gujarat Western Food Processing Auto 2 \n",**

**"... ... ... ... ... ... \n",**

**"11246 Maharashtra Western Chemical Office 4 \n",**

**"11247 Haryana Northern Healthcare Veterinary 3 \n",**

**"11248 Madhya Pradesh Central Textile Office 4 \n",**

**"11249 Karnataka Southern Agriculture Office 3 \n",**

**"11250 Maharashtra Western Healthcare Office 3 \n",**

**"\n",**

**" Amount \n",**

**"0 23952 \n",**

**"1 23934 \n",**

**"2 23924 \n",**

**"3 23912 \n",**

**"4 23877 \n",**

**"... ... \n",**

**"11246 370 \n",**

**"11247 367 \n",**

**"11248 213 \n",**

**"11249 206 \n",**

**"11250 188 \n",**

**"\n",**

**"[11239 rows x 13 columns]"**

**]**

**},**

**"execution\_count": 12,**

**"metadata": {},**

**"output\_type": "execute\_result"**

**}**

**],**

**"source": [**

**"#rename column\n",**

**"df.rename(columns= {'Marital\_Status':'Shaadi'})"**

**]**

**},**

**{**

**"cell\_type": "code",**

**"execution\_count": 13,**

**"id": "a2484863",**

**"metadata": {},**

**"outputs": [**

**{**

**"data": {**

**"text/html": [**

**"<div>\n",**

**"<style scoped>\n",**

**" .dataframe tbody tr th:only-of-type {\n",**

**" vertical-align: middle;\n",**

**" }\n",**

**"\n",**

**" .dataframe tbody tr th {\n",**

**" vertical-align: top;\n",**

**" }\n",**

**"\n",**

**" .dataframe thead th {\n",**

**" text-align: right;\n",**

**" }\n",**

**"</style>\n",**

**"<table border=\"1\" class=\"dataframe\">\n",**

**" <thead>\n",**

**" <tr style=\"text-align: right;\">\n",**

**" <th></th>\n",**

**" <th>User\_ID</th>\n",**

**" <th>Age</th>\n",**

**" <th>Marital\_Status</th>\n",**

**" <th>Orders</th>\n",**

**" <th>Amount</th>\n",**

**" </tr>\n",**

**" </thead>\n",**

**" <tbody>\n",**

**" <tr>\n",**

**" <th>count</th>\n",**

**" <td>1.123900e+04</td>\n",**

**" <td>11239.000000</td>\n",**

**" <td>11239.000000</td>\n",**

**" <td>11239.000000</td>\n",**

**" <td>11239.000000</td>\n",**

**" </tr>\n",**

**" <tr>\n",**

**" <th>mean</th>\n",**

**" <td>1.003004e+06</td>\n",**

**" <td>35.410357</td>\n",**

**" <td>0.420055</td>\n",**

**" <td>2.489634</td>\n",**

**" <td>9453.610553</td>\n",**

**" </tr>\n",**

**" <tr>\n",**

**" <th>std</th>\n",**

**" <td>1.716039e+03</td>\n",**

**" <td>12.753866</td>\n",**

**" <td>0.493589</td>\n",**

**" <td>1.114967</td>\n",**

**" <td>5222.355168</td>\n",**

**" </tr>\n",**

**" <tr>\n",**

**" <th>min</th>\n",**

**" <td>1.000001e+06</td>\n",**

**" <td>12.000000</td>\n",**

**" <td>0.000000</td>\n",**

**" <td>1.000000</td>\n",**

**" <td>188.000000</td>\n",**

**" </tr>\n",**

**" <tr>\n",**

**" <th>25%</th>\n",**

**" <td>1.001492e+06</td>\n",**

**" <td>27.000000</td>\n",**

**" <td>0.000000</td>\n",**

**" <td>2.000000</td>\n",**

**" <td>5443.000000</td>\n",**

**" </tr>\n",**

**" <tr>\n",**

**" <th>50%</th>\n",**

**" <td>1.003064e+06</td>\n",**

**" <td>33.000000</td>\n",**

**" <td>0.000000</td>\n",**

**" <td>2.000000</td>\n",**

**" <td>8109.000000</td>\n",**

**" </tr>\n",**

**" <tr>\n",**

**" <th>75%</th>\n",**

**" <td>1.004426e+06</td>\n",**

**" <td>43.000000</td>\n",**

**" <td>1.000000</td>\n",**

**" <td>3.000000</td>\n",**

**" <td>12675.000000</td>\n",**

**" </tr>\n",**

**" <tr>\n",**

**" <th>max</th>\n",**

**" <td>1.006040e+06</td>\n",**

**" <td>92.000000</td>\n",**

**" <td>1.000000</td>\n",**

**" <td>4.000000</td>\n",**

**" <td>23952.000000</td>\n",**

**" </tr>\n",**

**" </tbody>\n",**

**"</table>\n",**

**"</div>"**

**],**

**"text/plain": [**

**" User\_ID Age Marital\_Status Orders Amount\n",**

**"count 1.123900e+04 11239.000000 11239.000000 11239.000000 11239.000000\n",**

**"mean 1.003004e+06 35.410357 0.420055 2.489634 9453.610553\n",**

**"std 1.716039e+03 12.753866 0.493589 1.114967 5222.355168\n",**

**"min 1.000001e+06 12.000000 0.000000 1.000000 188.000000\n",**

**"25% 1.001492e+06 27.000000 0.000000 2.000000 5443.000000\n",**

**"50% 1.003064e+06 33.000000 0.000000 2.000000 8109.000000\n",**

**"75% 1.004426e+06 43.000000 1.000000 3.000000 12675.000000\n",**

**"max 1.006040e+06 92.000000 1.000000 4.000000 23952.000000"**

**]**

**},**

**"execution\_count": 13,**

**"metadata": {},**

**"output\_type": "execute\_result"**

**}**

**],**

**"source": [**

**"# describe() method returns description of the data in the DataFrame (i.e. count, mean, std, etc)\n",**

**"df.describe()"**

**]**

**},**

**{**

**"cell\_type": "code",**

**"execution\_count": 14,**

**"id": "2c380030",**

**"metadata": {},**

**"outputs": [**

**{**

**"data": {**

**"text/html": [**

**"<div>\n",**

**"<style scoped>\n",**

**" .dataframe tbody tr th:only-of-type {\n",**

**" vertical-align: middle;\n",**

**" }\n",**

**"\n",**

**" .dataframe tbody tr th {\n",**

**" vertical-align: top;\n",**

**" }\n",**

**"\n",**

**" .dataframe thead th {\n",**

**" text-align: right;\n",**

**" }\n",**

**"</style>\n",**

**"<table border=\"1\" class=\"dataframe\">\n",**

**" <thead>\n",**

**" <tr style=\"text-align: right;\">\n",**

**" <th></th>\n",**

**" <th>Age</th>\n",**

**" <th>Orders</th>\n",**

**" <th>Amount</th>\n",**

**" </tr>\n",**

**" </thead>\n",**

**" <tbody>\n",**

**" <tr>\n",**

**" <th>count</th>\n",**

**" <td>11239.000000</td>\n",**

**" <td>11239.000000</td>\n",**

**" <td>11239.000000</td>\n",**

**" </tr>\n",**

**" <tr>\n",**

**" <th>mean</th>\n",**

**" <td>35.410357</td>\n",**

**" <td>2.489634</td>\n",**

**" <td>9453.610553</td>\n",**

**" </tr>\n",**

**" <tr>\n",**

**" <th>std</th>\n",**

**" <td>12.753866</td>\n",**

**" <td>1.114967</td>\n",**

**" <td>5222.355168</td>\n",**

**" </tr>\n",**

**" <tr>\n",**

**" <th>min</th>\n",**

**" <td>12.000000</td>\n",**

**" <td>1.000000</td>\n",**

**" <td>188.000000</td>\n",**

**" </tr>\n",**

**" <tr>\n",**

**" <th>25%</th>\n",**

**" <td>27.000000</td>\n",**

**" <td>2.000000</td>\n",**

**" <td>5443.000000</td>\n",**

**" </tr>\n",**

**" <tr>\n",**

**" <th>50%</th>\n",**

**" <td>33.000000</td>\n",**

**" <td>2.000000</td>\n",**

**" <td>8109.000000</td>\n",**

**" </tr>\n",**

**" <tr>\n",**

**" <th>75%</th>\n",**

**" <td>43.000000</td>\n",**

**" <td>3.000000</td>\n",**

**" <td>12675.000000</td>\n",**

**" </tr>\n",**

**" <tr>\n",**

**" <th>max</th>\n",**

**" <td>92.000000</td>\n",**

**" <td>4.000000</td>\n",**

**" <td>23952.000000</td>\n",**

**" </tr>\n",**

**" </tbody>\n",**

**"</table>\n",**

**"</div>"**

**],**

**"text/plain": [**

**" Age Orders Amount\n",**

**"count 11239.000000 11239.000000 11239.000000\n",**

**"mean 35.410357 2.489634 9453.610553\n",**

**"std 12.753866 1.114967 5222.355168\n",**

**"min 12.000000 1.000000 188.000000\n",**

**"25% 27.000000 2.000000 5443.000000\n",**

**"50% 33.000000 2.000000 8109.000000\n",**

**"75% 43.000000 3.000000 12675.000000\n",**

**"max 92.000000 4.000000 23952.000000"**

**]**

**},**

**"execution\_count": 14,**

**"metadata": {},**

**"output\_type": "execute\_result"**

**}**

**],**

**"source": [**

**"# use describe() for specific columns\n",**

**"df[['Age', 'Orders', 'Amount']].describe()"**

**]**

**},**

**{**

**"cell\_type": "markdown",**

**"id": "cc649558",**

**"metadata": {},**

**"source": [**

**"# Exploratory Data Analysis"**

**]**

**},**

**{**

**"cell\_type": "markdown",**

**"id": "504b419c",**

**"metadata": {},**

**"source": [**

**"### Gender"**

**]**

**},**

**{**

**"cell\_type": "code",**

**"execution\_count": 15,**

**"id": "84c7918d",**

**"metadata": {},**

**"outputs": [**

**{**

**"data": {**

**"image/png": "",**

**"text/plain": [**

**"<Figure size 640x480 with 1 Axes>"**

**]**

**},**

**"metadata": {},**

**"output\_type": "display\_data"**

**}**

**],**

**"source": [**

**"# plotting a bar chart for Gender and it's count\n",**

**"\n",**

**"ax = sns.countplot(x = 'Gender',data = df)\n",**

**"\n",**

**"for bars in ax.containers:\n",**

**" ax.bar\_label(bars)"**

**]**

**},**

**{**

**"cell\_type": "code",**

**"execution\_count": 16,**

**"id": "c5fd4566",**

**"metadata": {},**

**"outputs": [**

**{**

**"data": {**

**"text/plain": [**

**"<Axes: xlabel='Gender', ylabel='Amount'>"**

**]**

**},**

**"execution\_count": 16,**

**"metadata": {},**

**"output\_type": "execute\_result"**

**},**

**{**

**"data": {**

**"image/png": "",**

**"text/plain": [**

**"<Figure size 640x480 with 1 Axes>"**

**]**

**},**

**"metadata": {},**

**"output\_type": "display\_data"**

**}**

**],**

**"source": [**

**"# plotting a bar chart for gender vs total amount\n",**

**"\n",**

**"sales\_gen = df.groupby(['Gender'], as\_index=False)['Amount'].sum().sort\_values(by='Amount', ascending=False)\n",**

**"\n",**

**"sns.barplot(x = 'Gender',y= 'Amount' ,data = sales\_gen)"**

**]**

**},**

**{**

**"cell\_type": "markdown",**

**"id": "1e108cec",**

**"metadata": {},**

**"source": [**

**"\*From above graphs we can see that most of the buyers are females and even the purchasing power of females are greater than men\*"**

**]**

**},**

**{**

**"cell\_type": "markdown",**

**"id": "dcc927f9",**

**"metadata": {},**

**"source": [**

**"### Age"**

**]**

**},**

**{**

**"cell\_type": "code",**

**"execution\_count": 17,**

**"id": "88563dec",**

**"metadata": {},**

**"outputs": [**

**{**

**"data": {**

**"image/png": "",**

**"text/plain": [**

**"<Figure size 640x480 with 1 Axes>"**

**]**

**},**

**"metadata": {},**

**"output\_type": "display\_data"**

**}**

**],**

**"source": [**

**"ax = sns.countplot(data = df, x = 'Age Group', hue = 'Gender')\n",**

**"\n",**

**"for bars in ax.containers:\n",**

**" ax.bar\_label(bars)"**

**]**

**},**

**{**

**"cell\_type": "code",**

**"execution\_count": 18,**

**"id": "bfbe3937",**

**"metadata": {},**

**"outputs": [**

**{**

**"data": {**

**"text/plain": [**

**"<Axes: xlabel='Age Group', ylabel='Amount'>"**

**]**

**},**

**"execution\_count": 18,**

**"metadata": {},**

**"output\_type": "execute\_result"**

**},**

**{**

**"data": {**

**"image/png": "",**

**"text/plain": [**

**"<Figure size 640x480 with 1 Axes>"**

**]**

**},**

**"metadata": {},**

**"output\_type": "display\_data"**

**}**

**],**

**"source": [**

**"# Total Amount vs Age Group\n",**

**"sales\_age = df.groupby(['Age Group'], as\_index=False)['Amount'].sum().sort\_values(by='Amount', ascending=False)\n",**

**"\n",**

**"sns.barplot(x = 'Age Group',y= 'Amount' ,data = sales\_age)"**

**]**

**},**

**{**

**"cell\_type": "markdown",**

**"id": "abaa6c07",**

**"metadata": {},**

**"source": [**

**"\*From above graphs we can see that most of the buyers are of age group between 26-35 yrs female\*"**

**]**

**},**

**{**

**"cell\_type": "markdown",**

**"id": "83a49098",**

**"metadata": {},**

**"source": [**

**"### State"**

**]**

**},**

**{**

**"cell\_type": "code",**

**"execution\_count": 19,**

**"id": "dfac4f2c",**

**"metadata": {},**

**"outputs": [**

**{**

**"data": {**

**"text/plain": [**

**"<Axes: xlabel='State', ylabel='Orders'>"**

**]**

**},**

**"execution\_count": 19,**

**"metadata": {},**

**"output\_type": "execute\_result"**

**},**

**{**

**"data": {**

**"image/png": "",**

**"text/plain": [**

**"<Figure size 1500x500 with 1 Axes>"**

**]**

**},**

**"metadata": {},**

**"output\_type": "display\_data"**

**}**

**],**

**"source": [**

**"# total number of orders from top 10 states\n",**

**"\n",**

**"sales\_state = df.groupby(['State'], as\_index=False)['Orders'].sum().sort\_values(by='Orders', ascending=False).head(10)\n",**

**"\n",**

**"sns.set(rc={'figure.figsize':(15,5)})\n",**

**"sns.barplot(data = sales\_state, x = 'State',y= 'Orders')"**

**]**

**},**

**{**

**"cell\_type": "code",**

**"execution\_count": 20,**

**"id": "a66617bc",**

**"metadata": {},**

**"outputs": [**

**{**

**"data": {**

**"text/plain": [**

**"<Axes: xlabel='State', ylabel='Amount'>"**

**]**

**},**

**"execution\_count": 20,**

**"metadata": {},**

**"output\_type": "execute\_result"**

**},**

**{**

**"data": {**

**"image/png": "",**

**"text/plain": [**

**"<Figure size 1500x500 with 1 Axes>"**

**]**

**},**

**"metadata": {},**

**"output\_type": "display\_data"**

**}**

**],**

**"source": [**

**"# total amount/sales from top 10 states\n",**

**"\n",**

**"sales\_state = df.groupby(['State'], as\_index=False)['Amount'].sum().sort\_values(by='Amount', ascending=False).head(10)\n",**

**"\n",**

**"sns.set(rc={'figure.figsize':(15,5)})\n",**

**"sns.barplot(data = sales\_state, x = 'State',y= 'Amount')"**

**]**

**},**

**{**

**"cell\_type": "markdown",**

**"id": "39d46e9e",**

**"metadata": {},**

**"source": [**

**"\*From above graphs we can see that most of the orders & total sales/amount are from Uttar Pradesh, Maharashtra and Karnataka respectively\*\n"**

**]**

**},**

**{**

**"cell\_type": "markdown",**

**"id": "89f865a5",**

**"metadata": {},**

**"source": [**

**"### Marital Status"**

**]**

**},**

**{**

**"cell\_type": "code",**

**"execution\_count": 21,**

**"id": "77e6635e",**

**"metadata": {},**

**"outputs": [**

**{**

**"data": {**

**"image/png": "",**

**"text/plain": [**

**"<Figure size 1500x500 with 1 Axes>"**

**]**

**},**

**"metadata": {},**

**"output\_type": "display\_data"**

**}**

**],**

**"source": [**

**"ax = sns.countplot(data = df, x = 'Marital\_Status')\n",**

**"\n",**

**"sns.set(rc={'figure.figsize':(7,5)})\n",**

**"for bars in ax.containers:\n",**

**" ax.bar\_label(bars)"**

**]**

**},**

**{**

**"cell\_type": "code",**

**"execution\_count": 22,**

**"id": "2d8c7b27",**

**"metadata": {},**

**"outputs": [**

**{**

**"data": {**

**"text/plain": [**

**"<Axes: xlabel='Marital\_Status', ylabel='Amount'>"**

**]**

**},**

**"execution\_count": 22,**

**"metadata": {},**

**"output\_type": "execute\_result"**

**},**

**{**

**"data": {**

**"image/png": "",**

**"text/plain": [**

**"<Figure size 600x500 with 1 Axes>"**

**]**

**},**

**"metadata": {},**

**"output\_type": "display\_data"**

**}**

**],**

**"source": [**

**"sales\_state = df.groupby(['Marital\_Status', 'Gender'], as\_index=False)['Amount'].sum().sort\_values(by='Amount', ascending=False)\n",**

**"\n",**

**"sns.set(rc={'figure.figsize':(6,5)})\n",**

**"sns.barplot(data = sales\_state, x = 'Marital\_Status',y= 'Amount', hue='Gender')"**

**]**

**},**

**{**

**"cell\_type": "markdown",**

**"id": "37afbb83",**

**"metadata": {},**

**"source": [**

**"\*From above graphs we can see that most of the buyers are married (women) and they have high purchasing power\*"**

**]**

**},**

**{**

**"cell\_type": "markdown",**

**"id": "230cfe66",**

**"metadata": {},**

**"source": [**

**"### Occupation"**

**]**

**},**

**{**

**"cell\_type": "code",**

**"execution\_count": 23,**

**"id": "4deb41f4",**

**"metadata": {},**

**"outputs": [**

**{**

**"data": {**

**"image/png": "",**

**"text/plain": [**

**"<Figure size 2000x500 with 1 Axes>"**

**]**

**},**

**"metadata": {},**

**"output\_type": "display\_data"**

**}**

**],**

**"source": [**

**"sns.set(rc={'figure.figsize':(20,5)})\n",**

**"ax = sns.countplot(data = df, x = 'Occupation')\n",**

**"\n",**

**"for bars in ax.containers:\n",**

**" ax.bar\_label(bars)"**

**]**

**},**

**{**

**"cell\_type": "code",**

**"execution\_count": 24,**

**"id": "4eb620f6",**

**"metadata": {},**

**"outputs": [**

**{**

**"data": {**

**"text/plain": [**

**"<Axes: xlabel='Occupation', ylabel='Amount'>"**

**]**

**},**

**"execution\_count": 24,**

**"metadata": {},**

**"output\_type": "execute\_result"**

**},**

**{**

**"data": {**

**"image/png": "",**

**"text/plain": [**

**"<Figure size 2000x500 with 1 Axes>"**

**]**

**},**

**"metadata": {},**

**"output\_type": "display\_data"**

**}**

**],**

**"source": [**

**"sales\_state = df.groupby(['Occupation'], as\_index=False)['Amount'].sum().sort\_values(by='Amount', ascending=False)\n",**

**"\n",**

**"sns.set(rc={'figure.figsize':(20,5)})\n",**

**"sns.barplot(data = sales\_state, x = 'Occupation',y= 'Amount')"**

**]**

**},**

**{**

**"cell\_type": "markdown",**

**"id": "26c22532",**

**"metadata": {},**

**"source": [**

**"\*From above graphs we can see that most of the buyers are working in IT, Healthcare and Aviation sector\*"**

**]**

**},**

**{**

**"cell\_type": "markdown",**

**"id": "40518579",**

**"metadata": {},**

**"source": [**

**"### Product Category"**

**]**

**},**

**{**

**"cell\_type": "code",**

**"execution\_count": 25,**

**"id": "3e7e4d39",**

**"metadata": {},**

**"outputs": [**

**{**

**"data": {**

**"image/png": "",**

**"text/plain": [**

**"<Figure size 2000x500 with 1 Axes>"**

**]**

**},**

**"metadata": {},**

**"output\_type": "display\_data"**

**}**

**],**

**"source": [**

**"sns.set(rc={'figure.figsize':(20,5)})\n",**

**"ax = sns.countplot(data = df, x = 'Product\_Category')\n",**

**"\n",**

**"for bars in ax.containers:\n",**

**" ax.bar\_label(bars)"**

**]**

**},**

**{**

**"cell\_type": "code",**

**"execution\_count": 26,**

**"id": "99f5c6f3",**

**"metadata": {},**

**"outputs": [**

**{**

**"data": {**

**"text/plain": [**

**"<Axes: xlabel='Product\_Category', ylabel='Amount'>"**

**]**

**},**

**"execution\_count": 26,**

**"metadata": {},**

**"output\_type": "execute\_result"**

**},**

**{**

**"data": {**

**"image/png": "",**

**"text/plain": [**

**"<Figure size 2000x500 with 1 Axes>"**

**]**

**},**

**"metadata": {},**

**"output\_type": "display\_data"**

**}**

**],**

**"source": [**

**"sales\_state = df.groupby(['Product\_Category'], as\_index=False)['Amount'].sum().sort\_values(by='Amount', ascending=False).head(10)\n",**

**"\n",**

**"sns.set(rc={'figure.figsize':(20,5)})\n",**

**"sns.barplot(data = sales\_state, x = 'Product\_Category',y= 'Amount')"**

**]**

**},**

**{**

**"cell\_type": "markdown",**

**"id": "5be54973",**

**"metadata": {},**

**"source": [**

**"\*From above graphs we can see that most of the sold products are from Food, Clothing and Electronics category\*"**

**]**

**},**

**{**

**"cell\_type": "code",**

**"execution\_count": 27,**

**"id": "f33d23c7",**

**"metadata": {},**

**"outputs": [**

**{**

**"data": {**

**"text/plain": [**

**"<Axes: xlabel='Product\_ID', ylabel='Orders'>"**

**]**

**},**

**"execution\_count": 27,**

**"metadata": {},**

**"output\_type": "execute\_result"**

**},**

**{**

**"data": {**

**"image/png": "",**

**"text/plain": [**

**"<Figure size 2000x500 with 1 Axes>"**

**]**

**},**

**"metadata": {},**

**"output\_type": "display\_data"**

**}**

**],**

**"source": [**

**"sales\_state = df.groupby(['Product\_ID'], as\_index=False)['Orders'].sum().sort\_values(by='Orders', ascending=False).head(10)\n",**

**"\n",**

**"sns.set(rc={'figure.figsize':(20,5)})\n",**

**"sns.barplot(data = sales\_state, x = 'Product\_ID',y= 'Orders')"**

**]**

**},**

**{**

**"cell\_type": "code",**

**"execution\_count": null,**

**"id": "17d2bb43",**

**"metadata": {**

**"scrolled": true**

**},**

**"outputs": [],**

**"source": [**

**"# top 10 most sold products (same thing as above)\n",**

**"fig1, ax1 = plt.subplots(figsize=(12,7))\n",**

**"df.groupby('Product\_ID')['Orders'].sum().nlargest(10).sort\_values(ascending=False).plot(kind='bar')"**

**]**

**},**

**{**

**"cell\_type": "markdown",**

**"id": "31f88f63",**

**"metadata": {},**

**"source": [**

**"## Conclusion:\n",**

**"\n",**

**"### "**

**]**

**},**

**{**

**"cell\_type": "markdown",**

**"id": "56e52bca",**

**"metadata": {},**

**"source": [**

**"\*Married women age group 26-35 yrs from UP, Maharastra and Karnataka working in IT, Healthcare and Aviation are more likely to buy products from Food, Clothing and Electronics category\*"**

**]**

**},**

**{**

**"cell\_type": "markdown",**

**"id": "1e91055b",**

**"metadata": {},**

**"source": [**

**" complete project on GitHub: https://github.com/Xeroworld/diwali-sales-analysis"**

**]**

**},**

**{**

**"cell\_type": "markdown",**

**"id": "d7bfbc58",**

**"metadata": {},**

**"source": [**

**"Thank you!\n",**

**"\n",**

**"Your's Sincerely\n",**

**"Mohd faraj"**

**]**

**}**

**],**

**"metadata": {**

**"kernelspec": {**

**"display\_name": "Python 3 (ipykernel)",**

**"language": "python",**

**"name": "python3"**

**},**

**"language\_info": {**

**"codemirror\_mode": {**

**"name": "ipython",**

**"version": 3**

**},**

**"file\_extension": ".py",**

**"mimetype": "text/x-python",**

**"name": "python",**

**"nbconvert\_exporter": "python",**

**"pygments\_lexer": "ipython3",**

**"version": "3.12.3"**

**}**

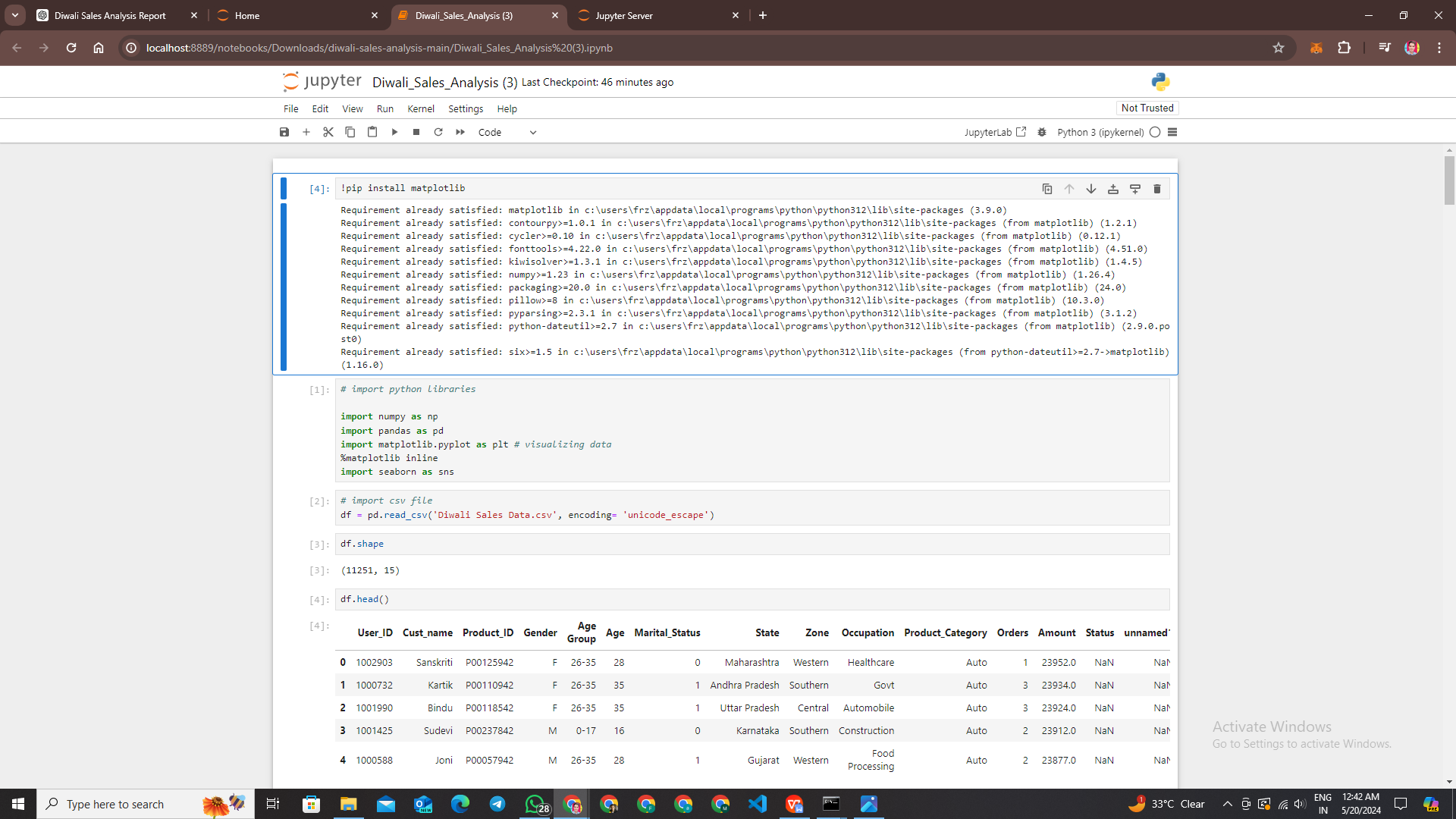
**},**

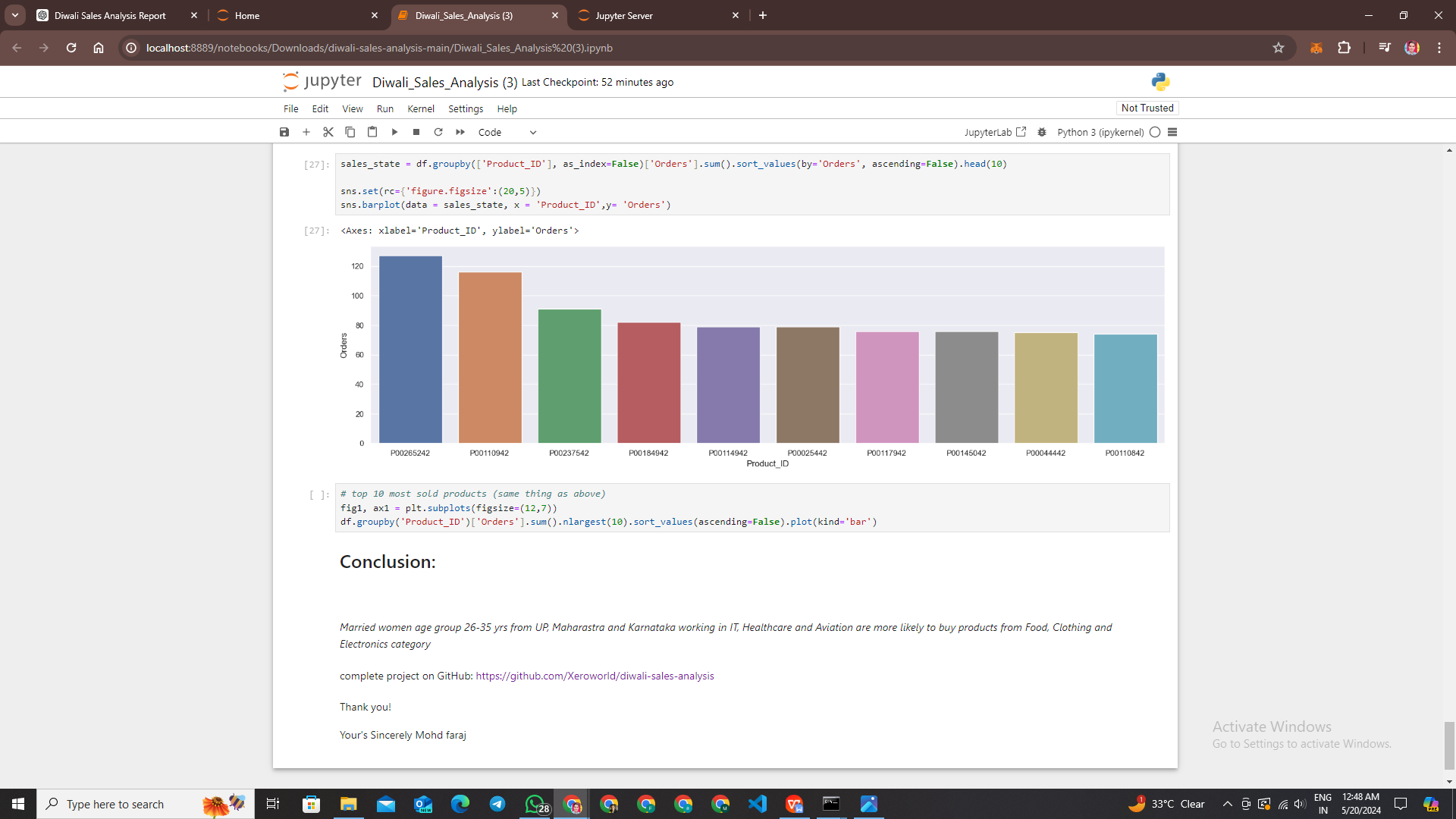
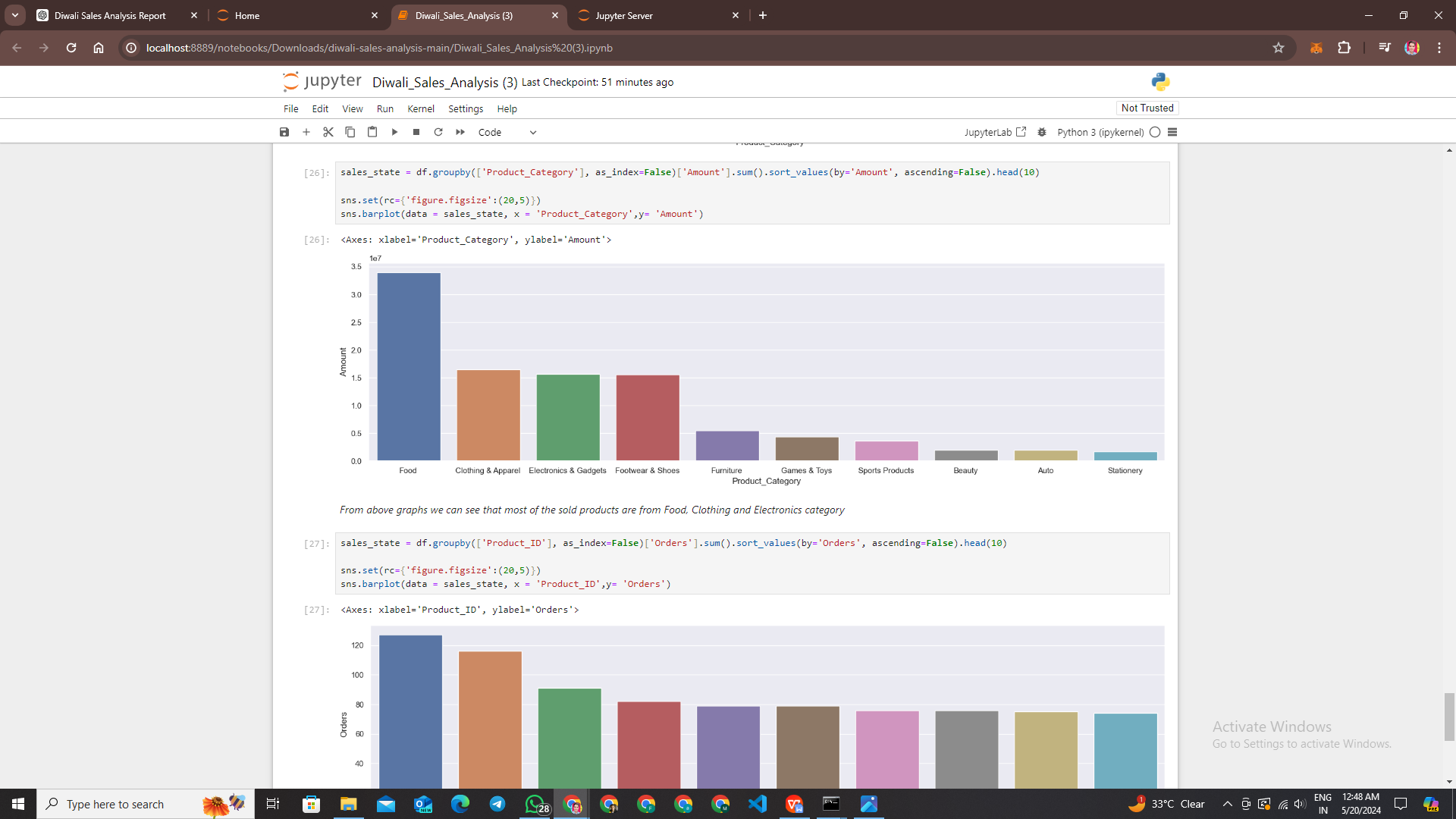
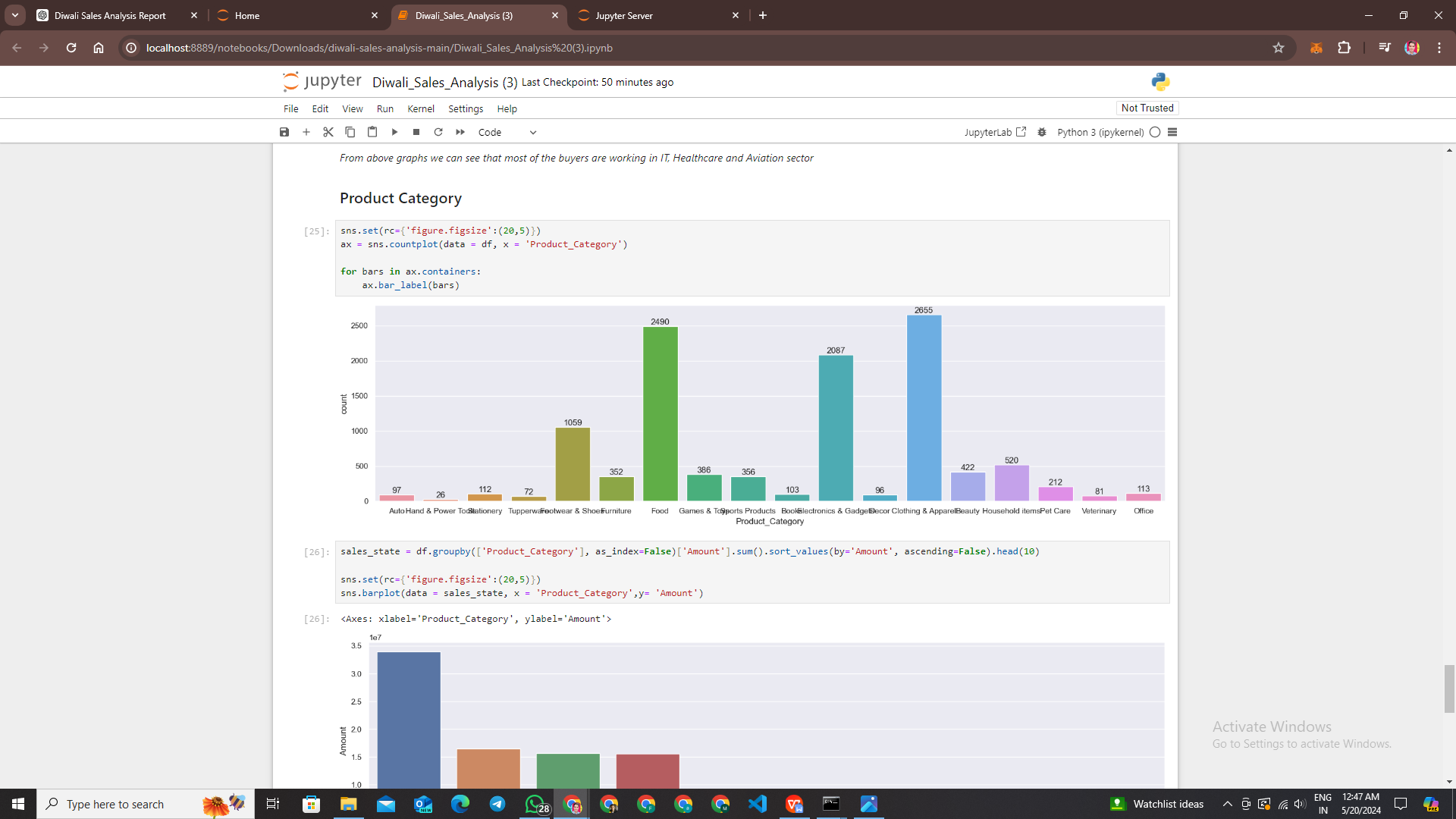
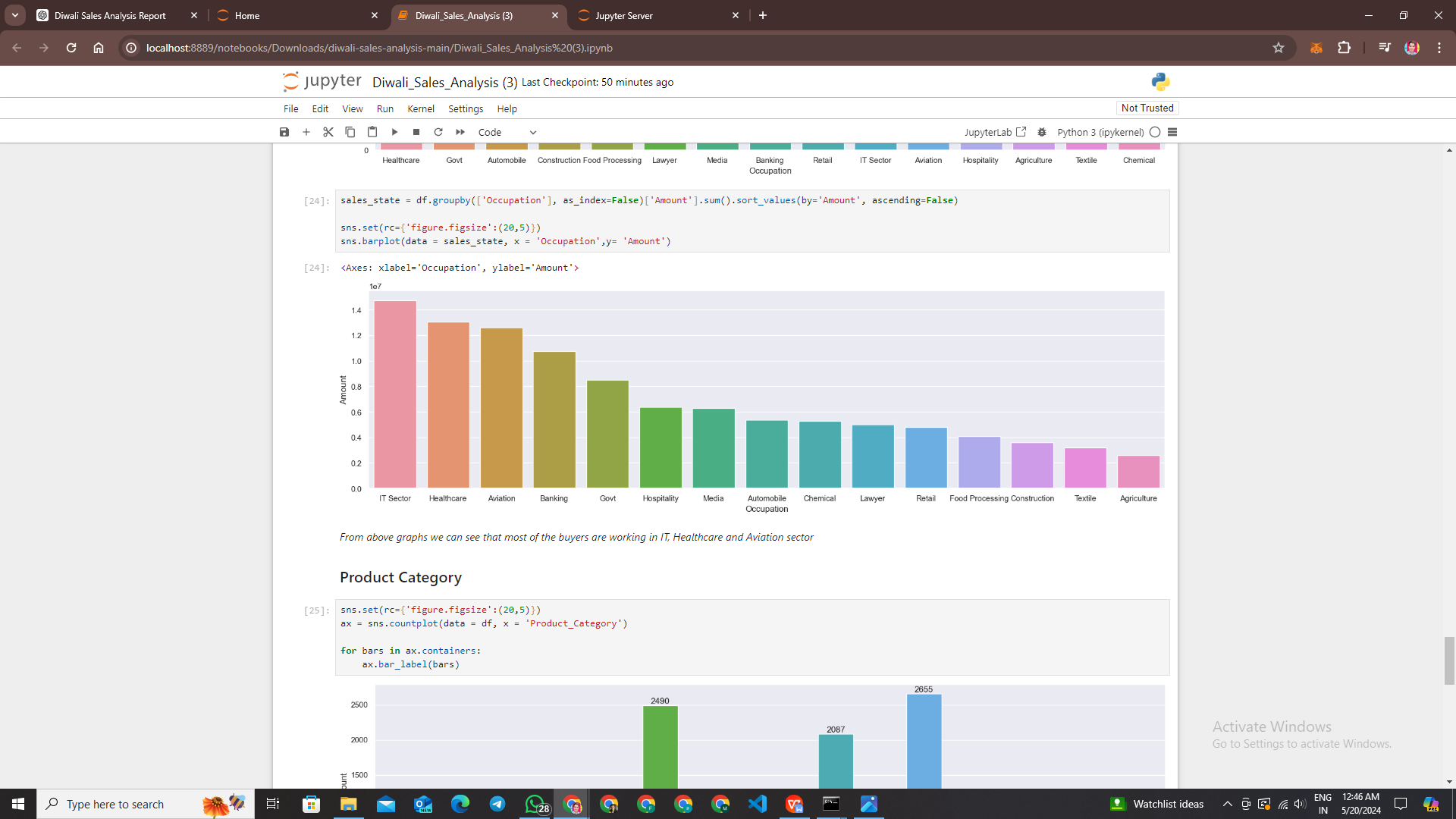
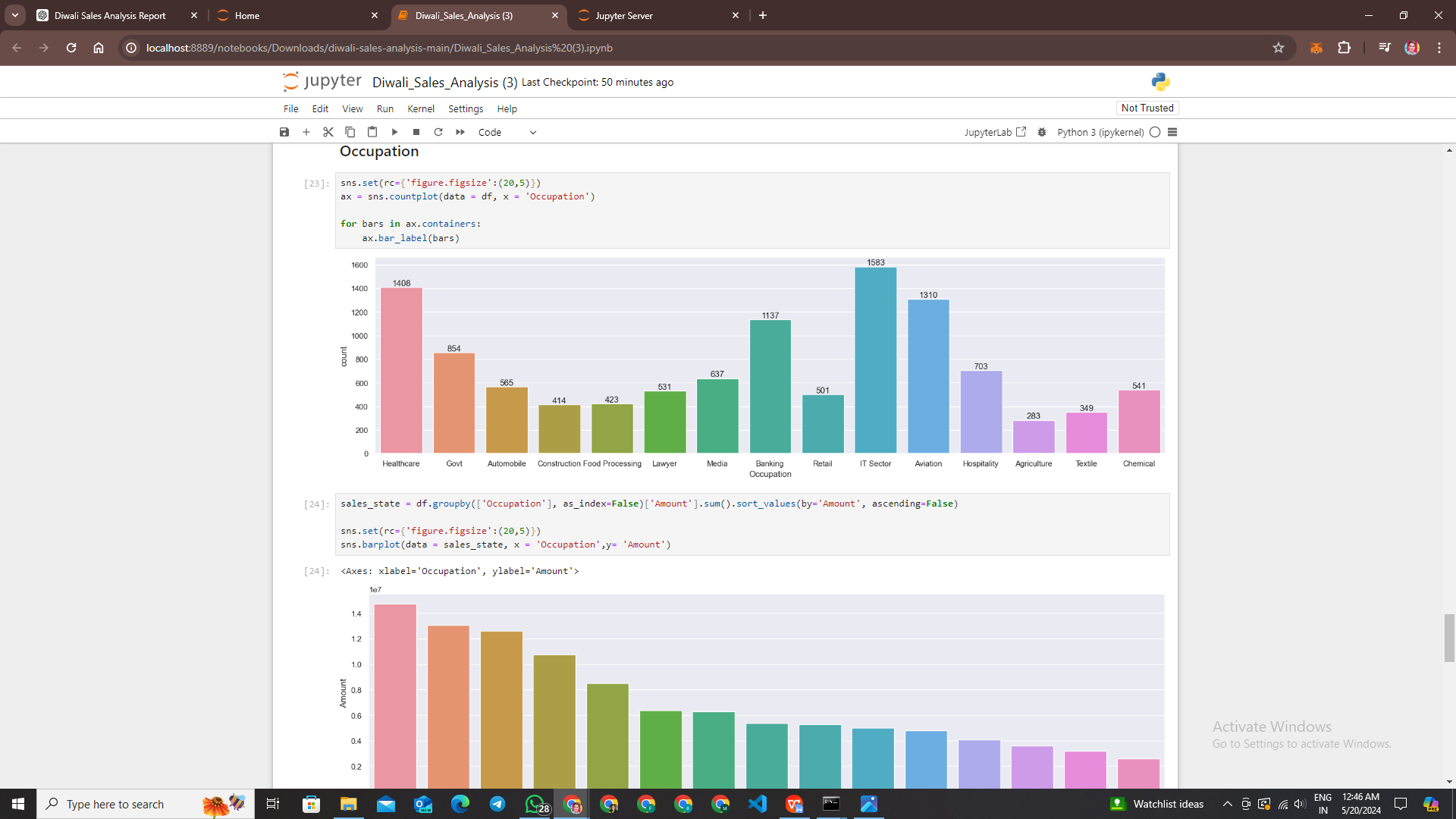
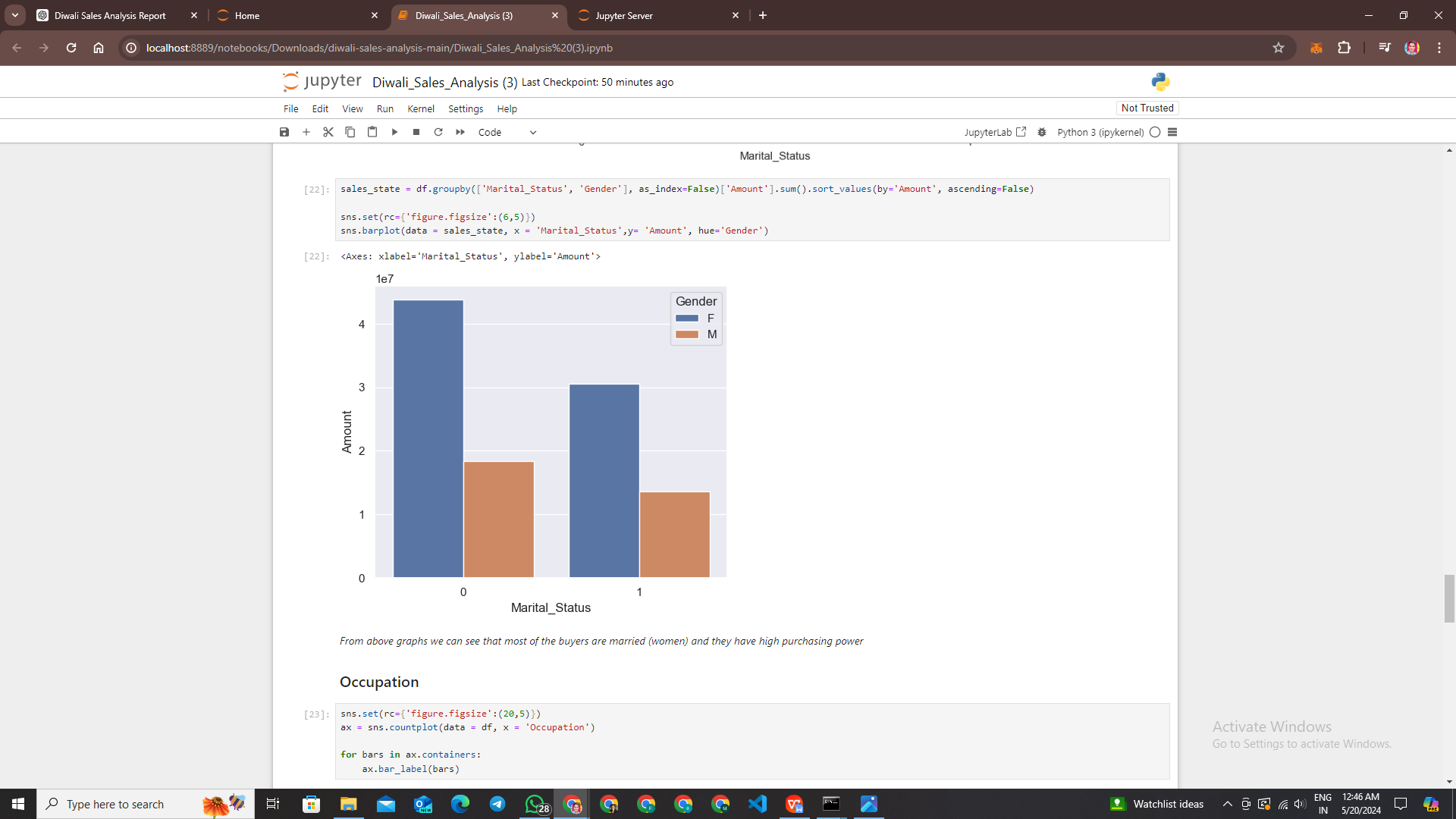
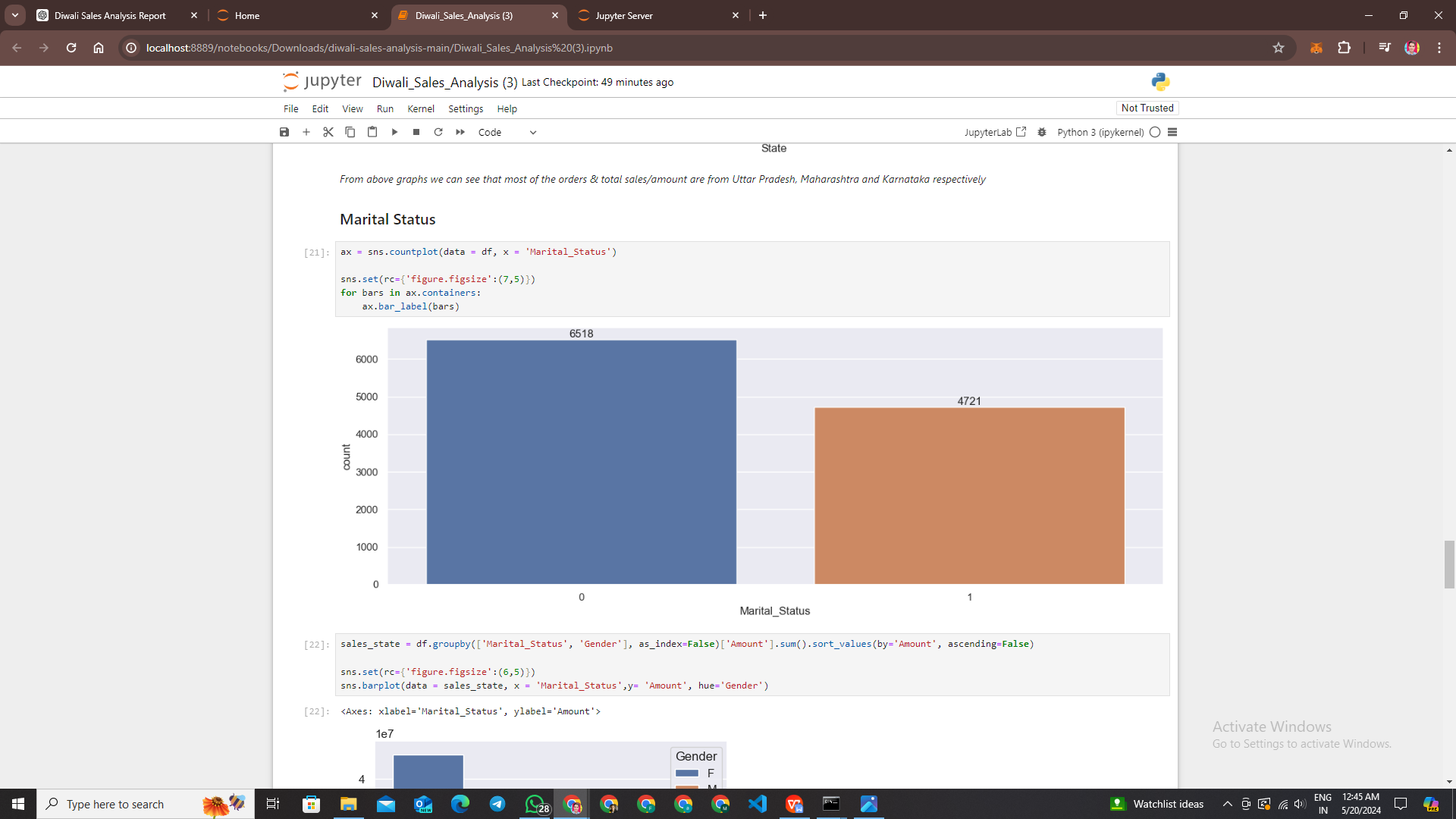
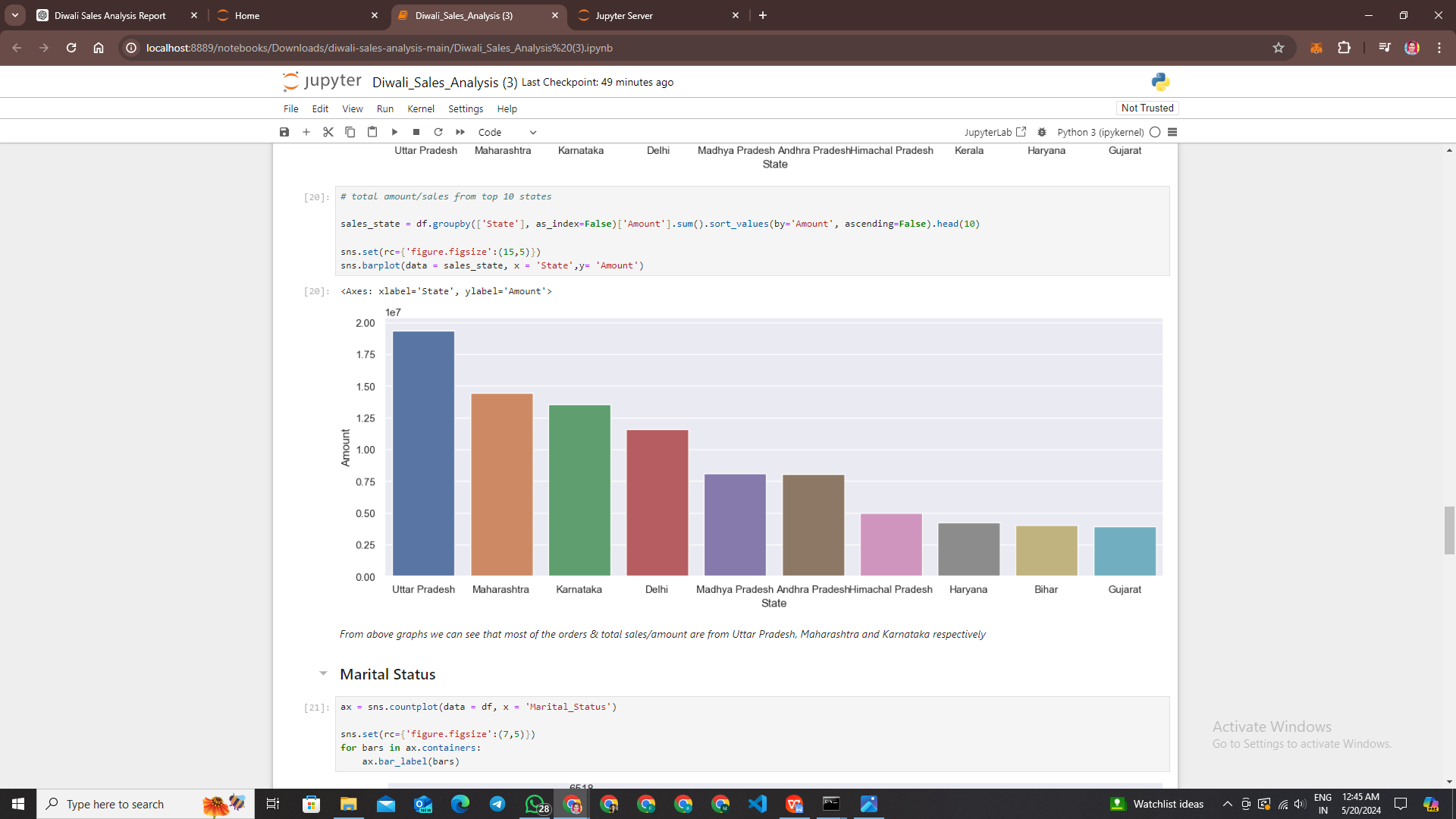
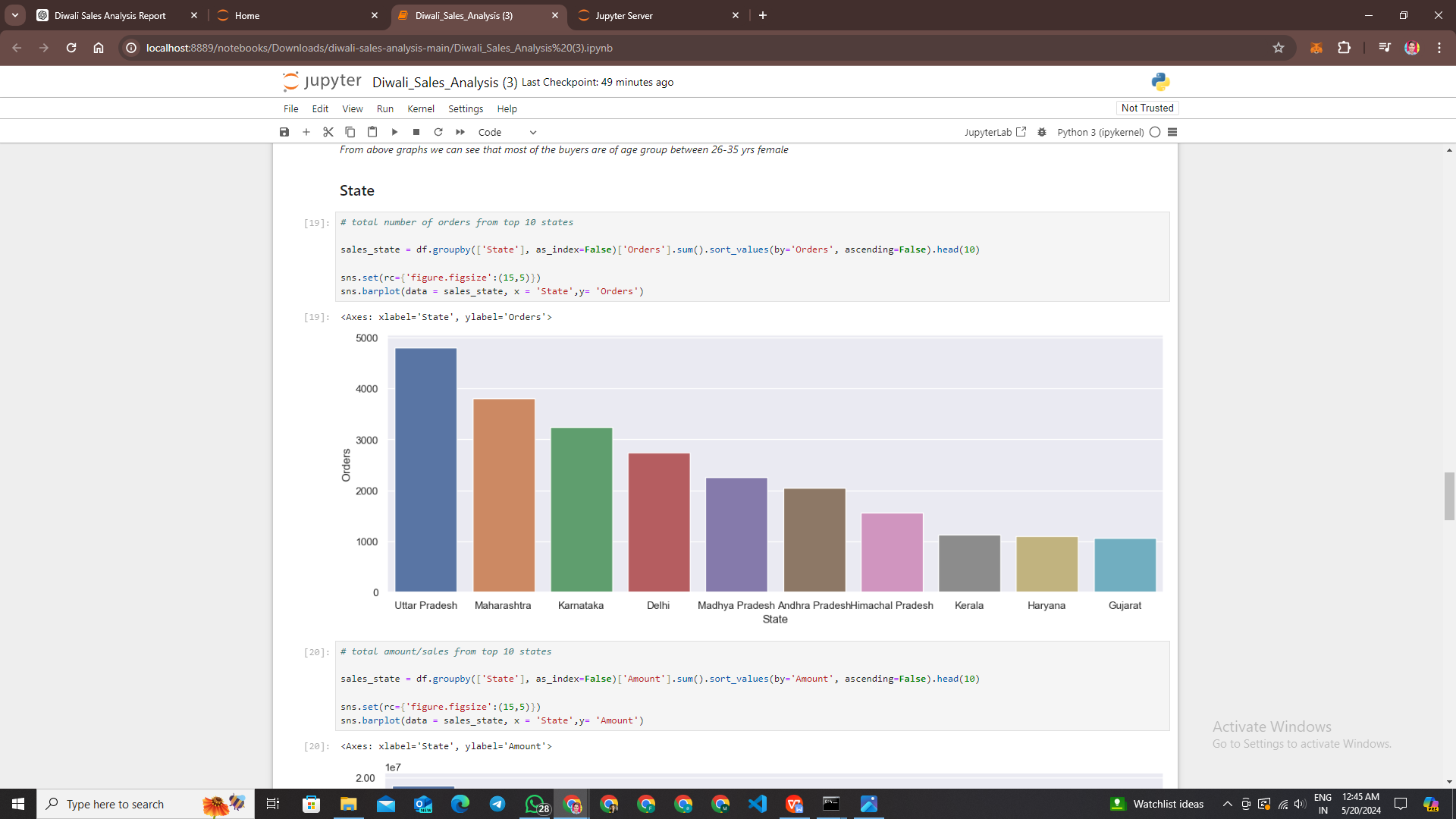
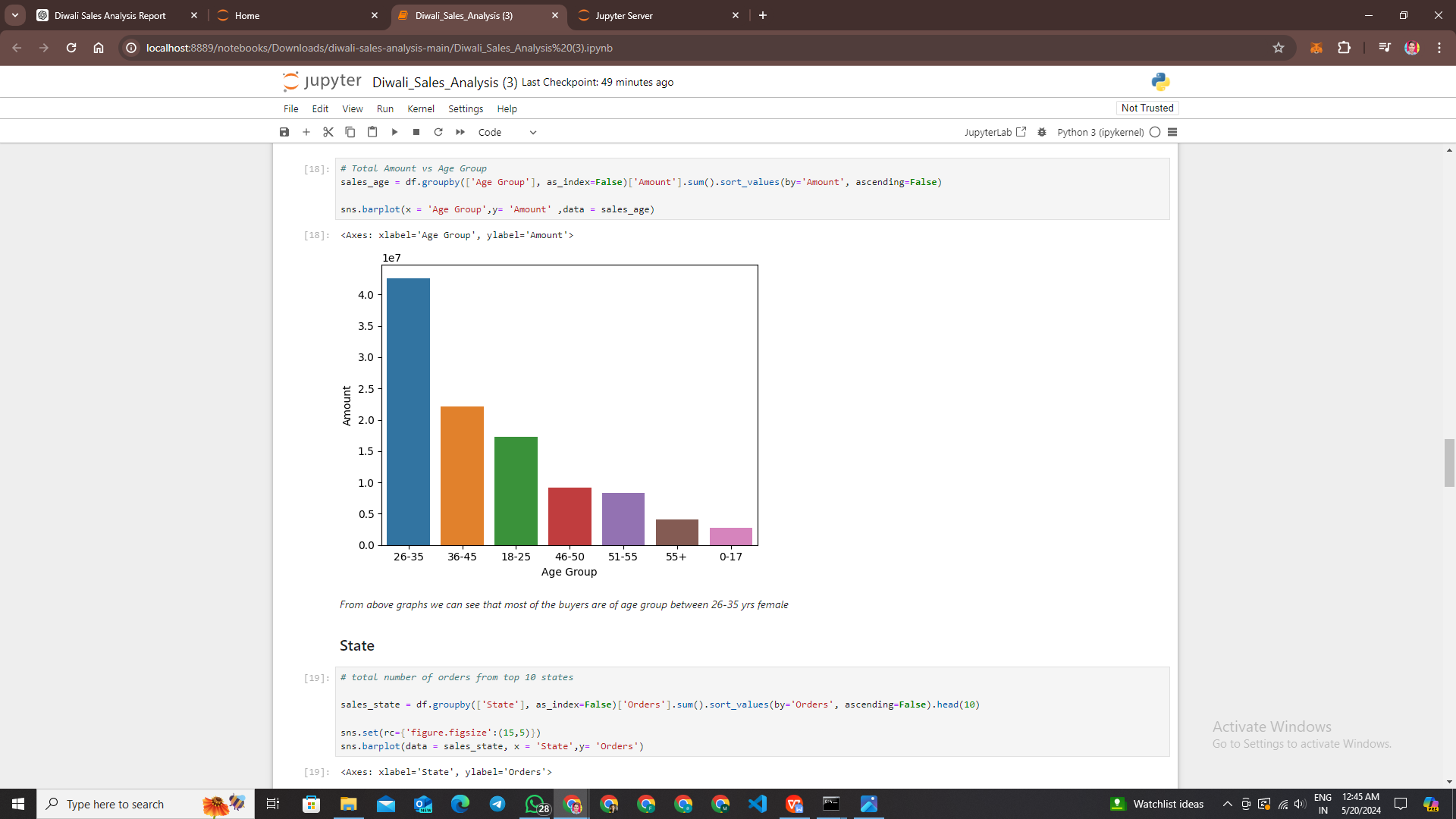
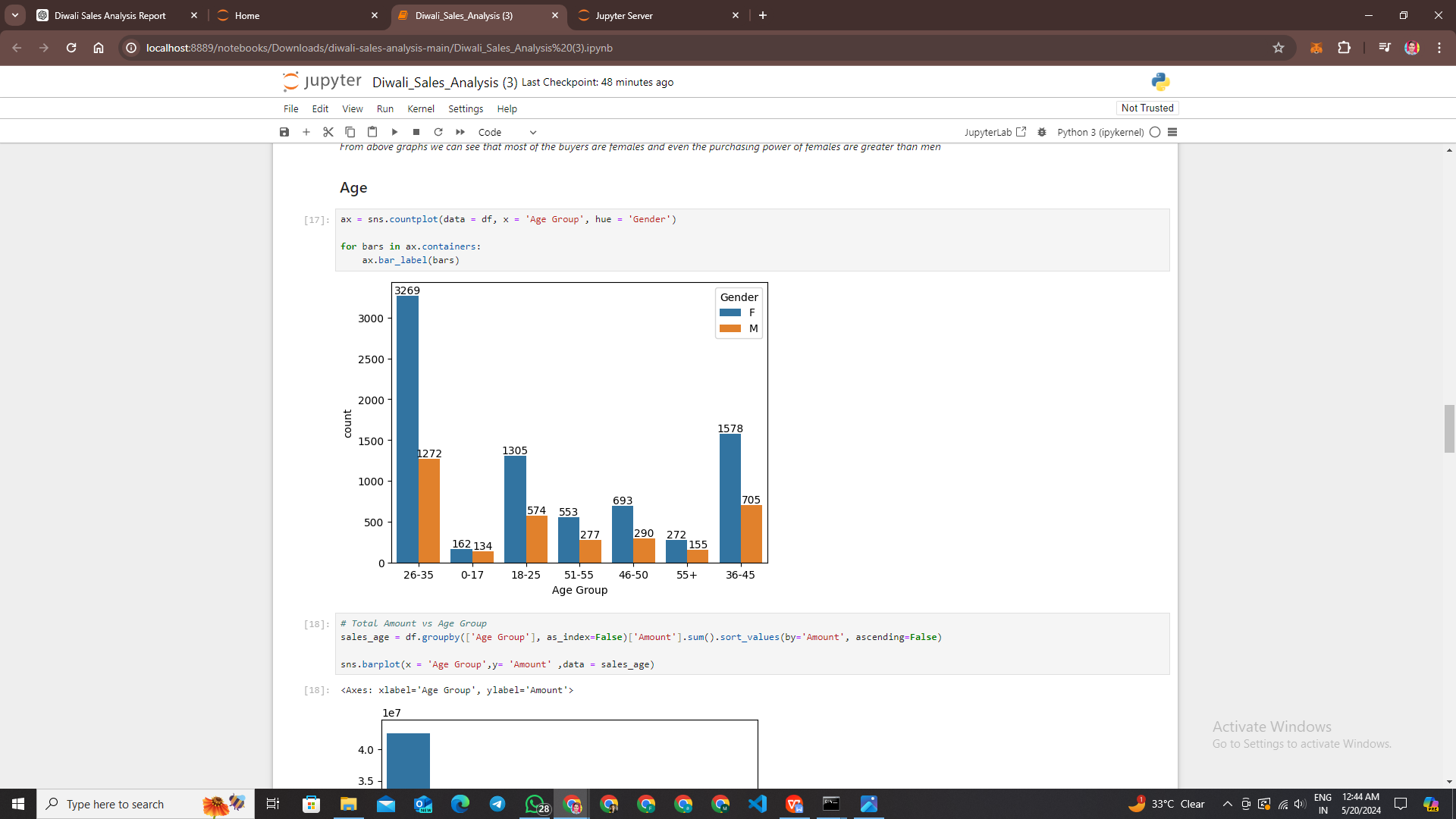
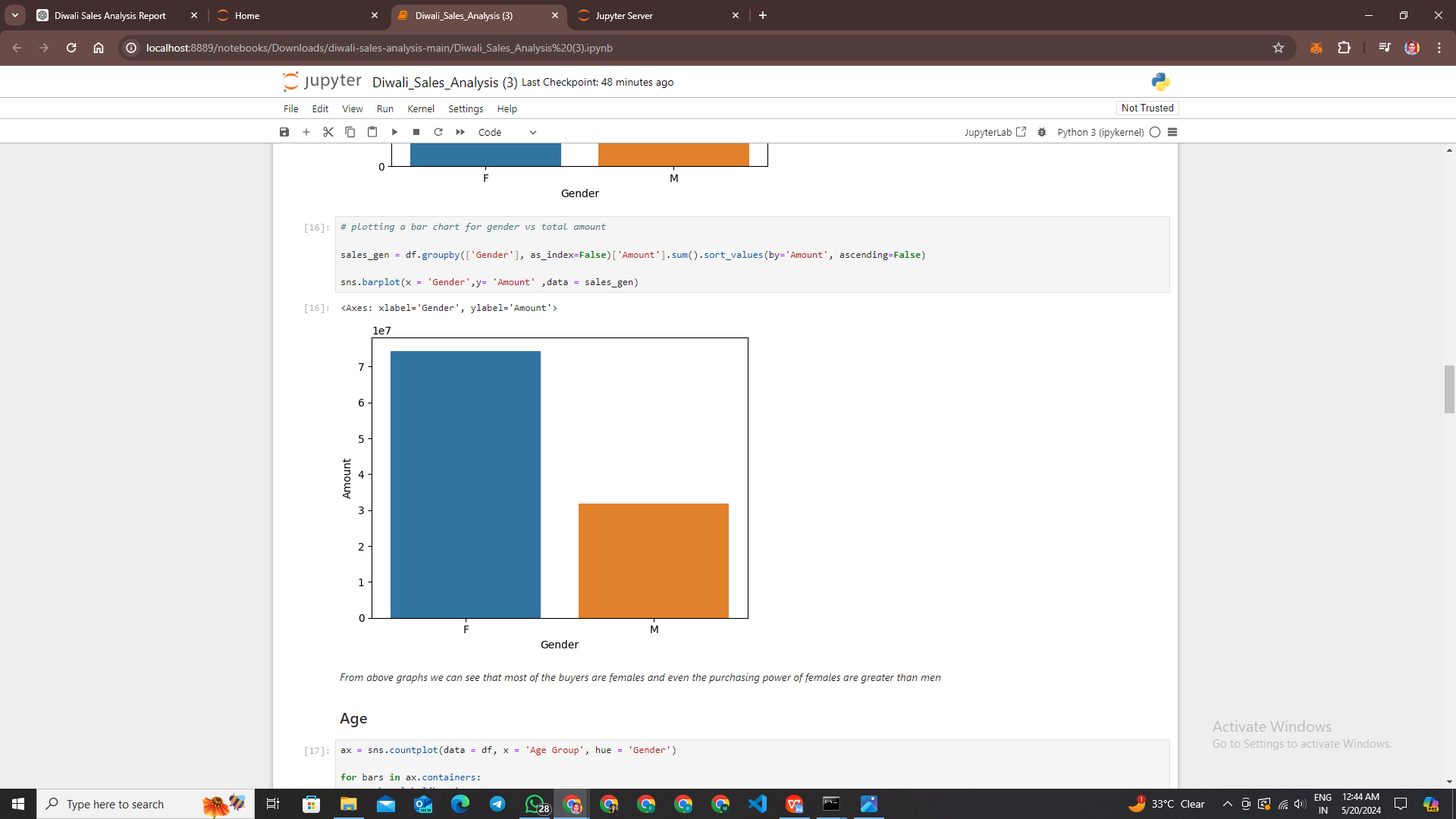
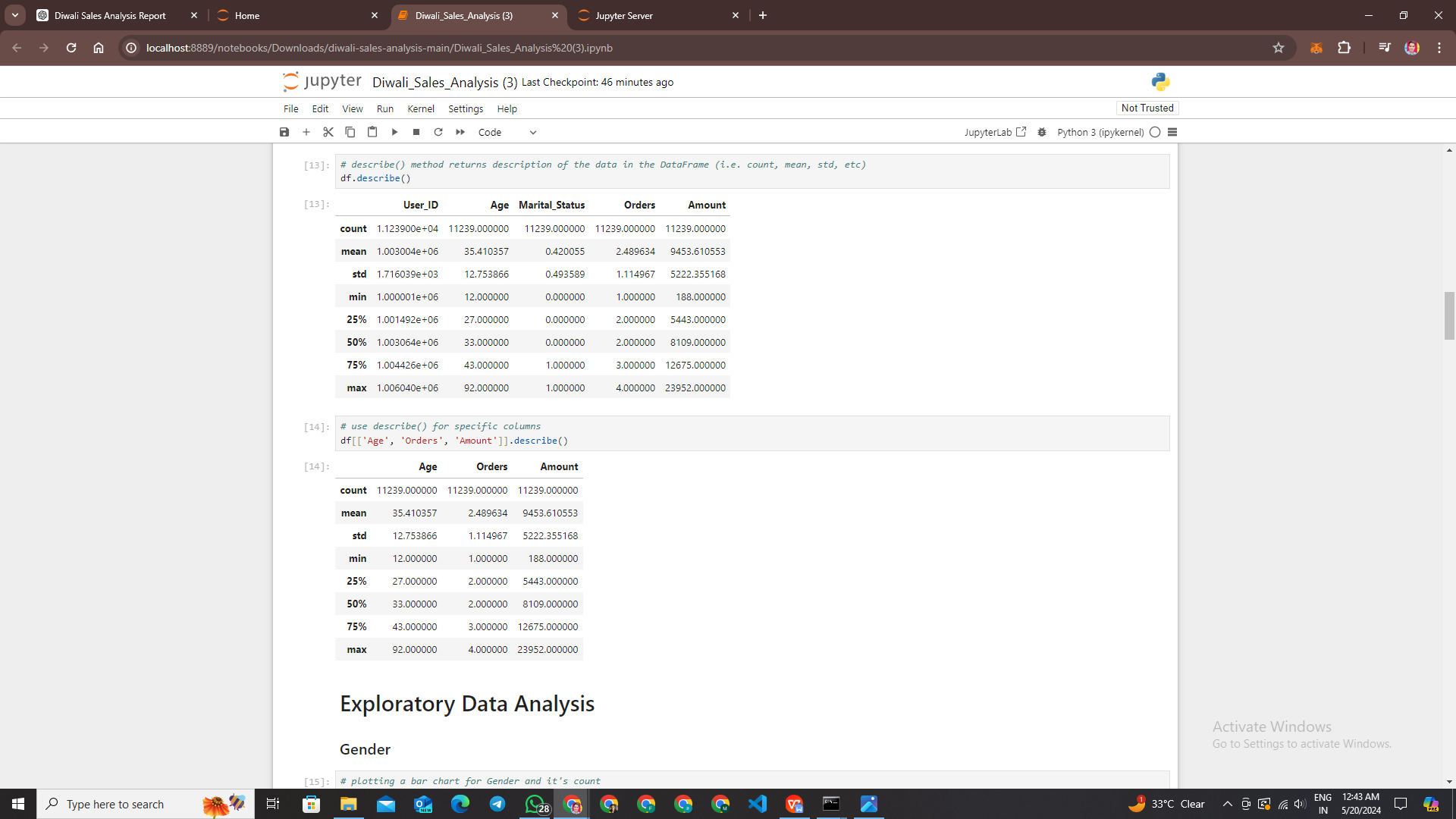
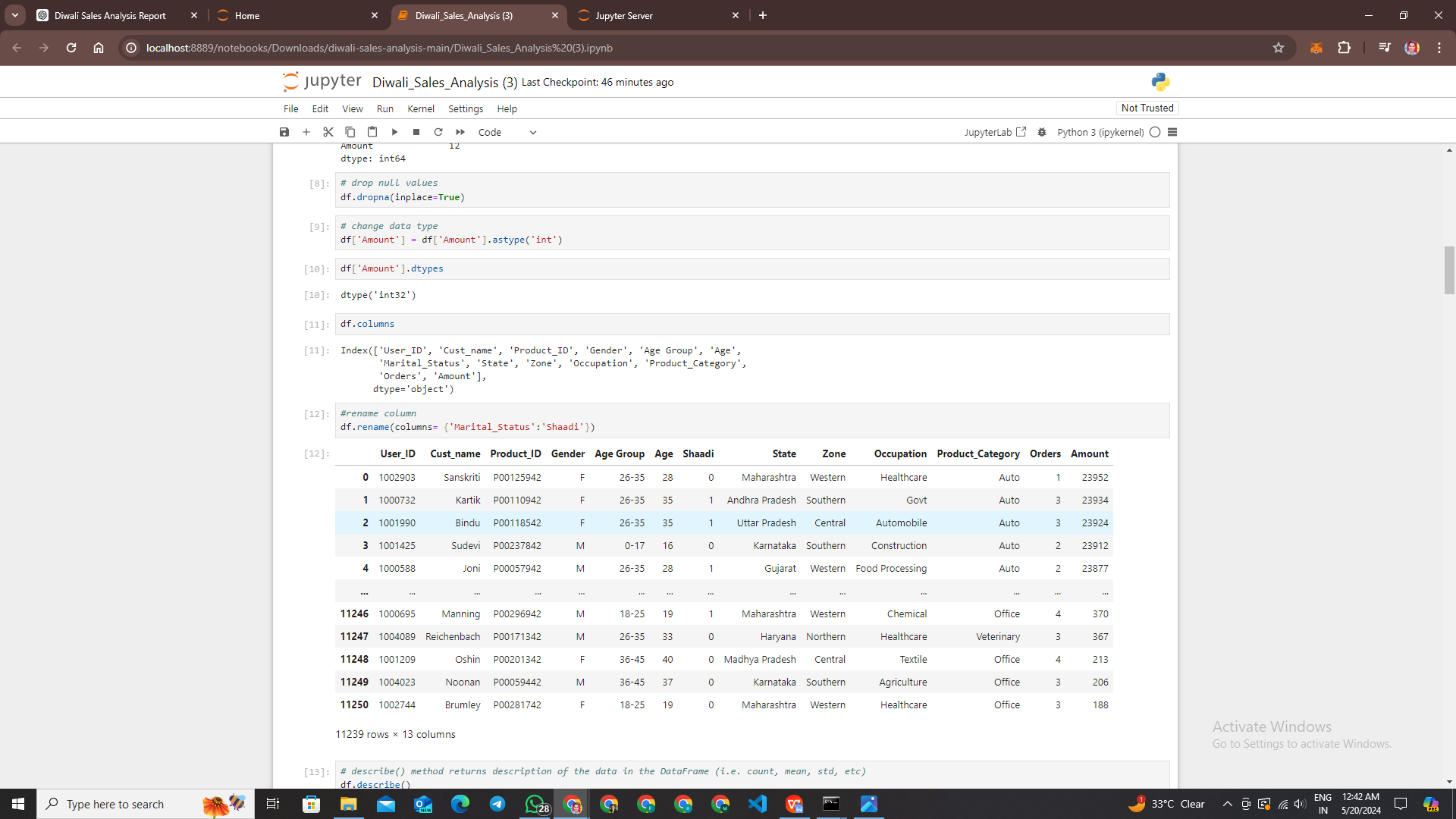
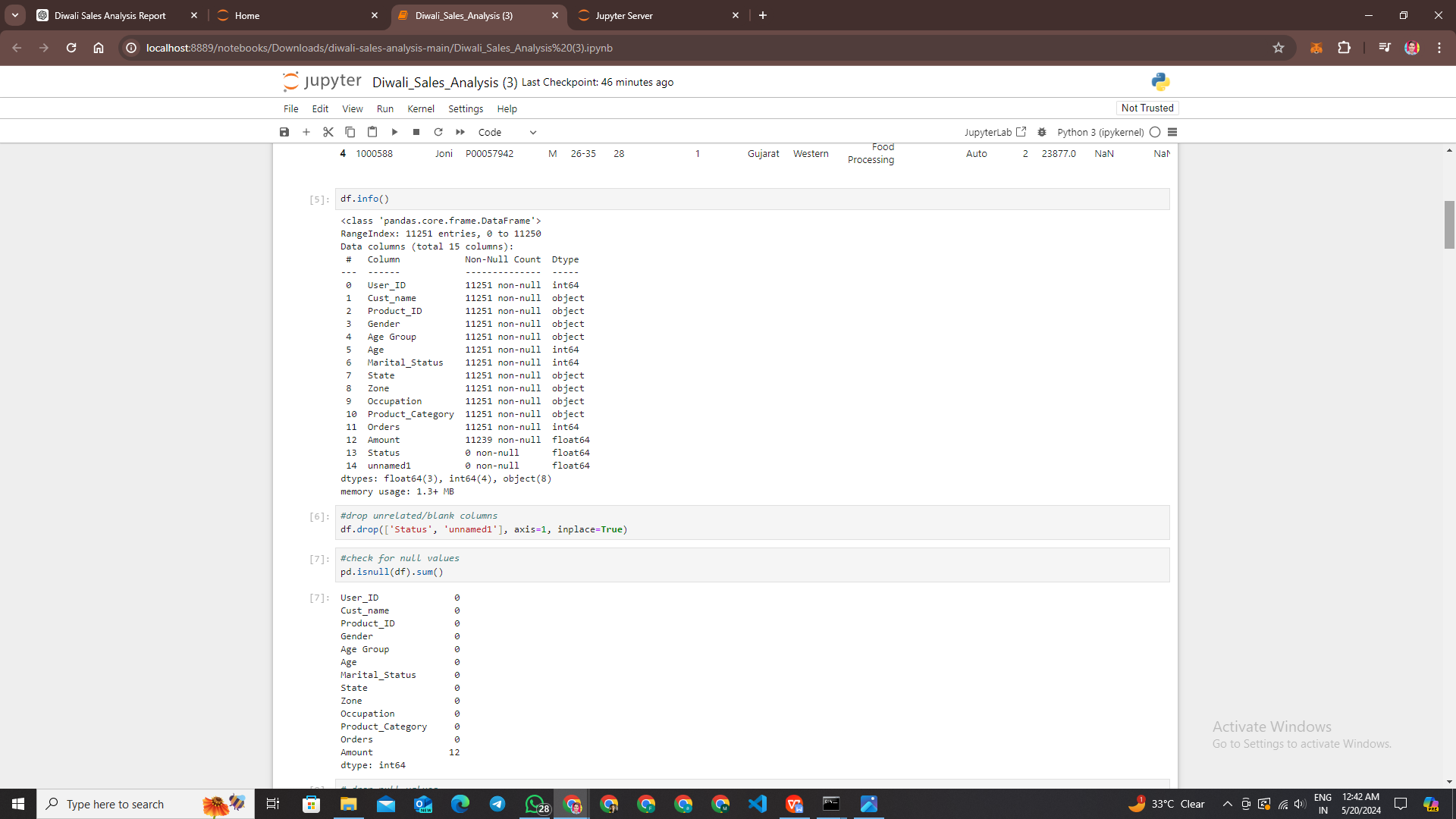
**"nbformat": 4,**

**"nbformat\_minor": 5**

**}**

# ScreenShots





# Future Scope of the Project

The Diwali Sales Analysis project lays the groundwork for further exploration and enhancement in several areas. Below are some potential avenues for future development and expansion:

**1. Predictive Modeling**

Implement machine learning algorithms to develop predictive models for forecasting Diwali sales trends.

Explore time series forecasting techniques such as ARIMA, SARIMA, or Prophet to predict future sales volumes.

Integrate external factors such as economic indicators, weather patterns, and social trends to improve prediction accuracy.

**2. Customer Segmentation**

Conduct more granular analysis of customer demographics to identify distinct segments based on purchasing behavior, preferences, and engagement levels.

Develop personalized marketing strategies and targeted promotions tailored to specific customer segments to enhance customer satisfaction and loyalty.

**3. Real-time Analysis**

Explore real-time data streaming technologies to analyze sales data as it becomes available, enabling retailers to make timely decisions and respond quickly to market changes during the Diwali season.

Implement interactive dashboards and visualizations that update dynamically to provide real-time insights into sales performance.

**4. Market Basket Analysis**

Conduct market basket analysis to identify product associations and patterns in customer purchasing behavior during Diwali.

Generate insights into cross-selling opportunities, product bundling strategies, and optimizing product placements to increase basket sizes and revenue.

**5. Geographic Expansion**

Expand the scope of analysis to include sales data from different regions and cities to understand regional variations in consumer behavior and preferences.

Tailor marketing strategies and product offerings based on regional insights to better cater to diverse customer needs and preferences.

**6. Integration with Retail Operations**

Integrate analysis findings with retail operations systems such as inventory management, supply chain logistics, and pricing strategies to optimize overall business performance.

Develop automated decision support systems that leverage analysis insights to streamline operations and improve efficiency.

**7. Enhanced Data Visualization**

Experiment with advanced visualization techniques such as interactive maps, network graphs, and 3D visualizations to present sales data in innovative and engaging ways.

Incorporate storytelling elements into visualizations to convey insights more effectively and make the analysis findings more compelling.

**8. Collaboration and Knowledge Sharing**

Foster collaboration with other retail industry stakeholders, academia, and research organizations to share insights, best practices, and emerging trends in Diwali sales analysis.

Contribute to open-source projects and participate in industry conferences and workshops to showcase project findings and foster knowledge exchange.

# Glossary

**1. Diwali:**

Also known as the Festival of Lights, Diwali is one of the most significant Hindu festivals celebrated with great enthusiasm across India and other parts of the world. It symbolizes the victory of light over darkness, good over evil, and knowledge over ignorance.

**2. Sales Analysis:**

The process of analyzing sales data to gain insights into sales trends, patterns, and performance. It involves examining sales volumes, revenue, customer behavior, and other factors to inform decision-making and optimize sales strategies.

**3. Data Preprocessing:**

The initial step in data analysis that involves cleaning, transforming, and preparing raw data for analysis. It includes tasks such as removing duplicates, handling missing values, standardizing formats, and creating new features.

**4. Exploratory Data Analysis (EDA):**

An approach to analyzing data sets to summarize their main characteristics, often with visual methods. EDA helps uncover patterns, detect anomalies, and formulate hypotheses, providing insights into the underlying structure of the data.

**5. Data Visualization:**

The graphical representation of data to facilitate understanding, analysis, and communication of insights. It encompasses various techniques such as charts, graphs, maps, and dashboards to visually present patterns, trends, and relationships in the data.

**6. Predictive Modeling:**

The process of using historical data to predict future outcomes. Predictive models leverage statistical algorithms and machine learning techniques to identify patterns and relationships in data and make predictions about future events or behaviors.

**7. Customer Segmentation:**

The division of customers into distinct groups based on shared characteristics such as demographics, purchasing behavior, and preferences. Customer segmentation enables targeted marketing efforts, personalized messaging, and tailored product offerings.

**8. Market Basket Analysis:**

A data mining technique used to identify relationships between products frequently purchased together by customers. Market basket analysis helps retailers understand consumer purchasing patterns and optimize product placement, cross-selling, and promotions.

**9. Time Series Analysis:**

A statistical technique used to analyze time-ordered data to identify patterns, trends, and seasonality. Time series analysis is commonly used in forecasting future values based on historical data and understanding the underlying patterns in time-varying phenomena.

**10. Machine Learning:**

A branch of artificial intelligence that enables computers to learn from data and make predictions or decisions without being explicitly programmed. Machine learning algorithms learn patterns and relationships in data to make predictions or take actions, often used in predictive modeling and pattern recognition tasks.

# BIBLIOGRAPHY

VanderPlas, Jake. "Python Data Science Handbook." O'Reilly Media, 2016.

This book provides comprehensive coverage of data science tools and techniques in Python, including NumPy, Pandas, Matplotlib, and Scikit-learn, which are essential for data analysis and machine learning.

McKinney, Wes. "Python for Data Analysis." O'Reilly Media, 2017.

Another valuable resource authored by Wes McKinney, the creator of Pandas library, focusing specifically on data manipulation and analysis using Python's Pandas library.

Seaborn Documentation. https://seaborn.pydata.org/

The official documentation for Seaborn library, offering detailed explanations, tutorials, and examples for creating attractive and informative statistical visualizations.

Matplotlib Documentation. [https://matplotlib.org/](https://matplotlib.org/" \t "https://chatgpt.com/c/_new)

The official documentation for Matplotlib library, providing comprehensive guides and examples for creating various types of plots and charts for data visualization.

GitHub Repository: Various repositories containing code snippets, examples, and tutorials related to data analysis, machine learning, and visualization techniques in Python.

YouTube Tutorials: Various YouTube channels offering tutorials and demonstrations on Python programming, data analysis, and machine learning techniques, which may have provided additional guidance and insights for your project.

Coursera Courses: Online courses on data analysis, machine learning, and related topics available on platforms like Coursera, which may have provided theoretical knowledge and practical skills applicable to your project.

Jupyter Notebook Documentation. [https://jupyter.org/](https://jupyter.org/" \t "https://chatgpt.com/c/_new)

The official documentation for Jupyter Notebook, providing guidance on using this interactive computing environment for writing code, visualizing data, and documenting analysis processes.

These sources likely contributed to my project by providing guidance, tutorials, documentation, and examples for using Python, Jupyter Notebook, data analysis libraries, and machine learning techniques.

Thank You for you time