

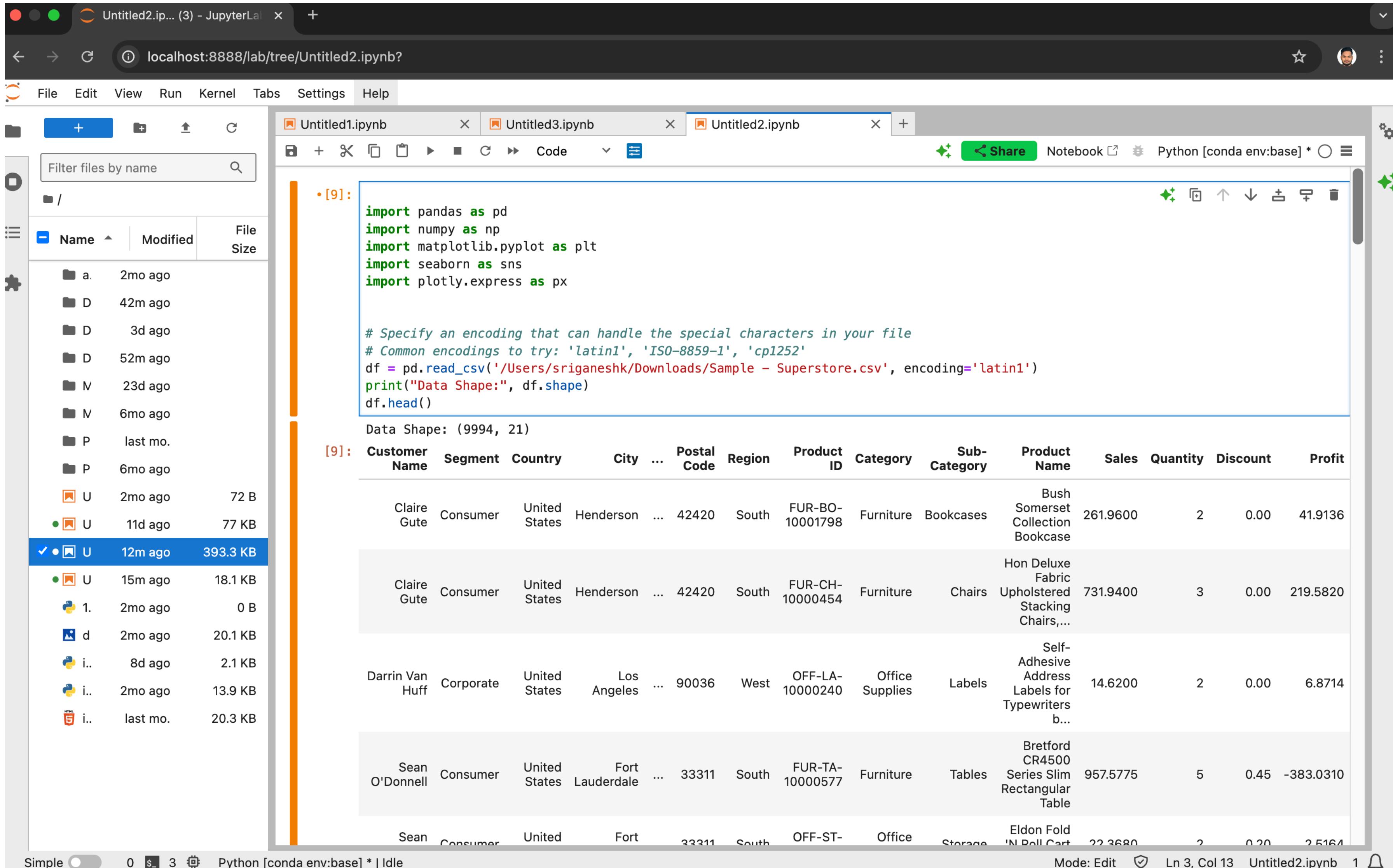
# **Walmart**

## **Data Analysis using**

## **Pandas, Numpy, Matplotlib, Seaborn.**

**S G V S Thribhuvan Kambhamettu**

# Importing Python Libraries.



The screenshot shows a Jupyter Notebook interface with three tabs: Untitled1.ipynb, Untitled3.ipynb, and Untitled2.ipynb. The Untitled2.ipynb tab is active, displaying the following code:

```
• [9]:  
import pandas as pd  
import numpy as np  
import matplotlib.pyplot as plt  
import seaborn as sns  
import plotly.express as px  
  
# Specify an encoding that can handle the special characters in your file  
# Common encodings to try: 'latin1', 'ISO-8859-1', 'cp1252'  
df = pd.read_csv('/Users/srganeshk/Downloads/Sample - Superstore.csv', encoding='latin1')  
print("Data Shape:", df.shape)  
df.head()  
  
Data Shape: (9994, 21)  
[9]:
```

The notebook also displays a preview of the first few rows of the dataset:

| Customer Name   | Segment   | Country       | City            | Postal Code | Region | Product ID      | Category        | Sub-Category | Product Name                                      | Sales    | Quantity | Discount | Profit    |
|-----------------|-----------|---------------|-----------------|-------------|--------|-----------------|-----------------|--------------|---|----------|----------|----------|-----------|
| Claire Gute     | Consumer  | United States | Henderson       | 42420       | South  | FUR-BO-10001798 | Furniture       | Bookcases    | Bush Somerset Collection Bookcase                 | 261.9600 | 2        | 0.00     | 41.9136   |
| Claire Gute     | Consumer  | United States | Henderson       | 42420       | South  | FUR-CH-10000454 | Furniture       | Chairs       | Hon Deluxe Fabric Upholstered Stacking Chairs,... | 731.9400 | 3        | 0.00     | 219.5820  |
| Darrin Van Huff | Corporate | United States | Los Angeles     | 90036       | West   | OFF-LA-10000240 | Office Supplies | Labels       | Self-Adhesive Address Labels for Typewriters b... | 14.6200  | 2        | 0.00     | 6.8714    |
| Sean O'Donnell  | Consumer  | United States | Fort Lauderdale | 33311       | South  | FUR-TA-10000577 | Furniture       | Tables       | Bretford CR4500 Series Slim Rectangular Table     | 957.5775 | 5        | 0.45     | -383.0310 |
| Sean            | Consumer  | United        | Fort            | 33311       | South  | OFF-ST-         | Office          | Storage      | Eldon Fold 'N Roll Cart                           | 22.3680  | 2        | 0.20     | 2.5164    |

We used Jupyter notebook to do this project.

We Imported several libraries in Python to read the data and visualise it.

Dataset: <https://www.kaggle.com/datasets/vivek468/superstore-dataset-final>

# To Remove Duplicates and to check missing values.

The screenshot shows a Jupyter Notebook interface with three tabs open: Untitled1.ipynb, Untitled3.ipynb, and Untitled2.ipynb. The Untitled2.ipynb tab is active and contains the following Python code:

```
# Remove duplicates
df.drop_duplicates(inplace=True)

# Convert date columns
df['Order Date'] = pd.to_datetime(df['Order Date'])
df['Ship Date'] = pd.to_datetime(df['Ship Date'])

# Create additional time features
df['Year'] = df['Order Date'].dt.year
df['Month'] = df['Order Date'].dt.month_name()
df['Month_Year'] = df['Order Date'].dt.to_period('M')

# Check for missing values
print(df.isnull().sum())
```

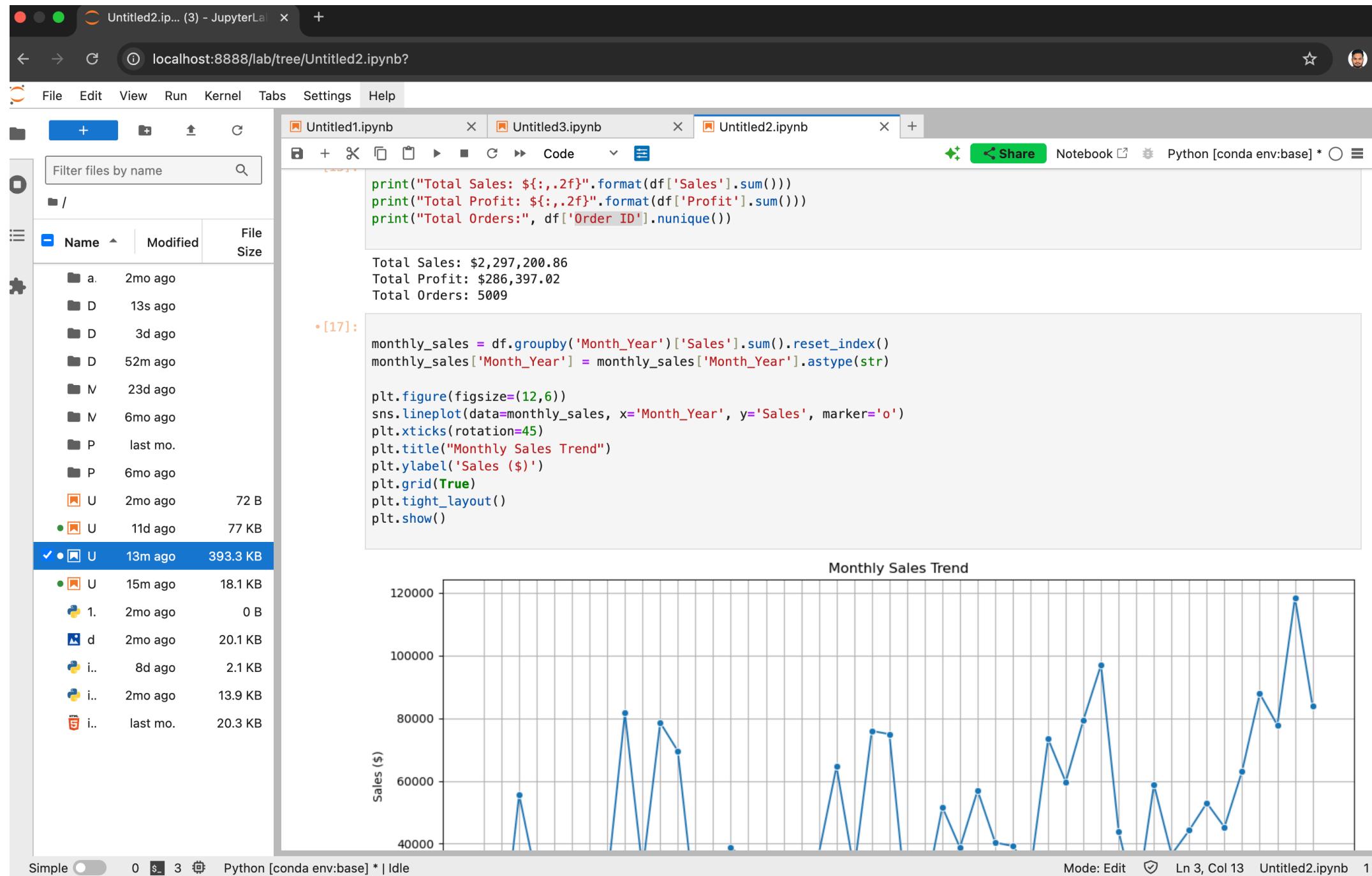
Below the code, the output of the `print(df.isnull().sum())` command is displayed as a table:

|               | Row ID | 0 |
|---------------|--------|---|
| Order ID      | 0      |   |
| Order Date    | 0      |   |
| Ship Date     | 0      |   |
| Ship Mode     | 0      |   |
| Customer ID   | 0      |   |
| Customer Name | 0      |   |
| Segment       | 0      |   |
| Country       | 0      |   |
| City          | 0      |   |
| State         | 0      |   |
| Postal Code   | 0      |   |
| Region        | 0      |   |
| Product ID    | 0      |   |
| Category      | 0      |   |
| Sub-Category  | 0      |   |
| Product Name  | 0      |   |
| Sales         | 0      |   |
| Quantity      | 0      |   |
| Discount      | 0      |   |
| Profit        | 0      |   |
| Year          | 0      |   |
| Month         | 0      |   |
| Month_Year    | 0      |   |

The sidebar on the left shows a file tree with various files and folders. The file `Untitled2.ipynb` is selected.

# Displaying Total sales/ Total Profits/ Total Orders.

## Displaying Monthly Sales Trends.



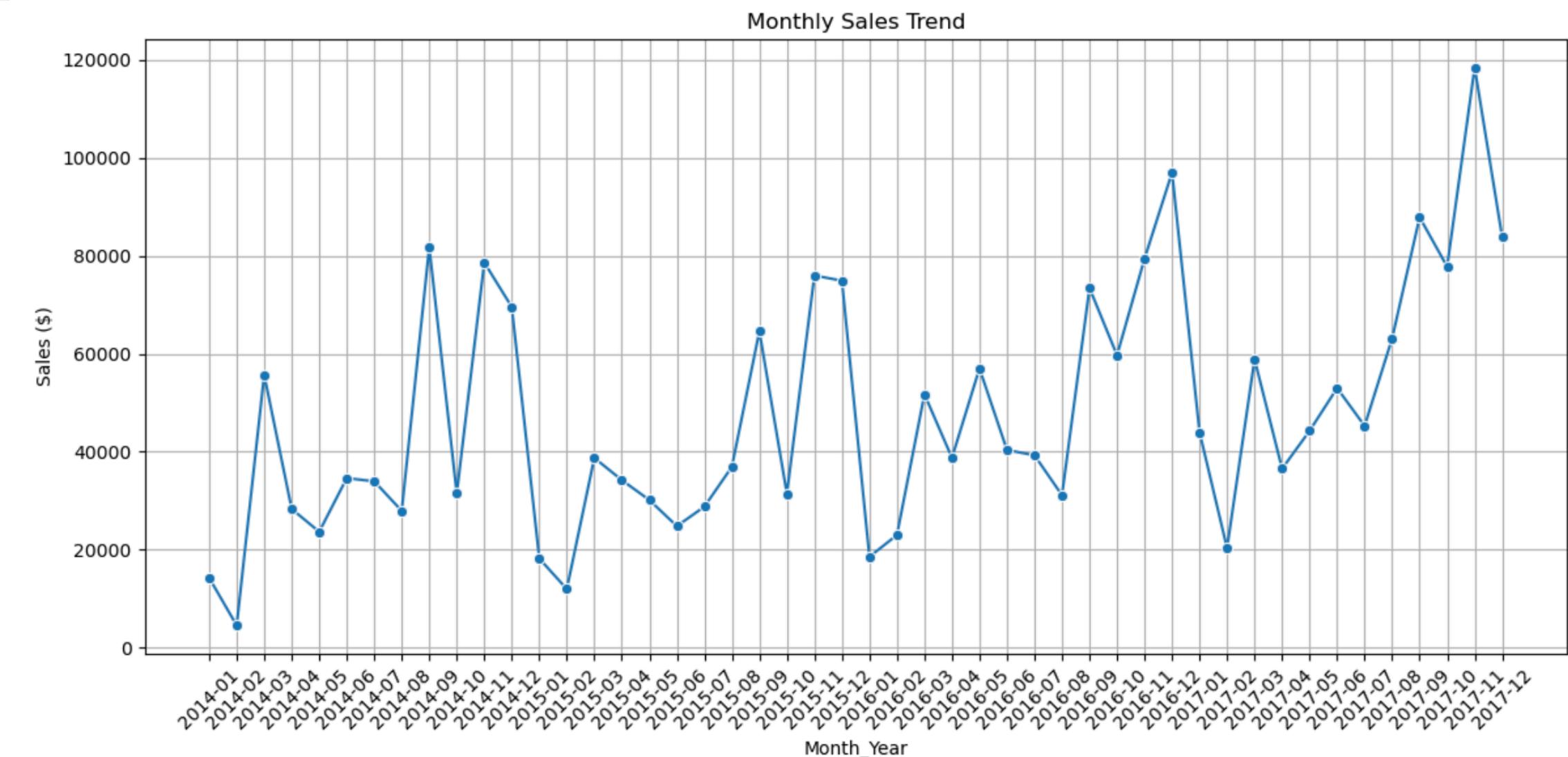
The screenshot shows a Jupyter Notebook interface with three tabs: Untitled1.ipynb, Untitled3.ipynb, and Untitled2.ipynb. The Untitled2.ipynb tab is active, displaying the following Python code:

```
print("Total Sales: ${:,2f}".format(df['Sales'].sum()))
print("Total Profit: ${:,2f}".format(df['Profit'].sum()))
print("Total Orders:", df['Order ID'].nunique())
```

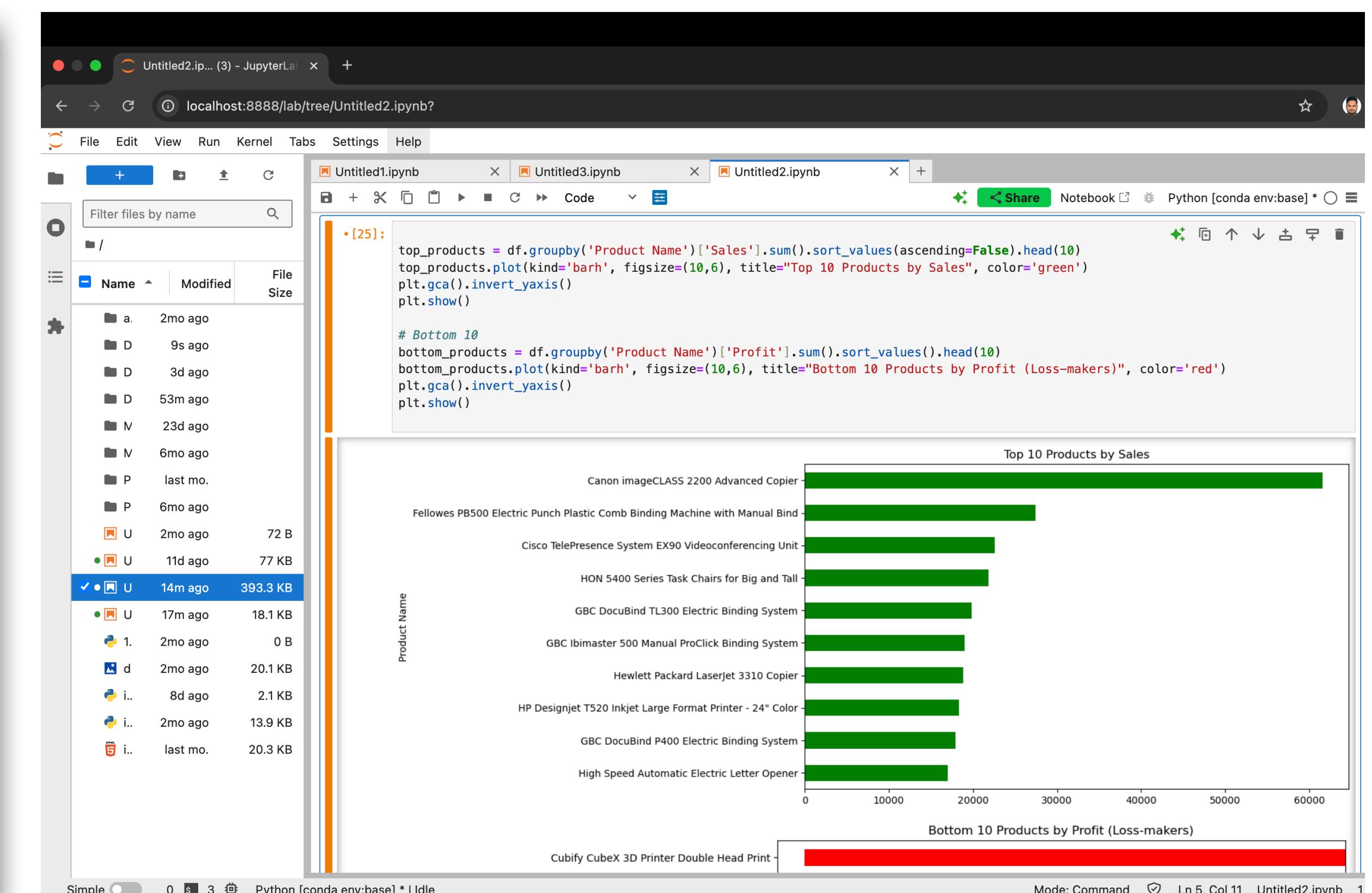
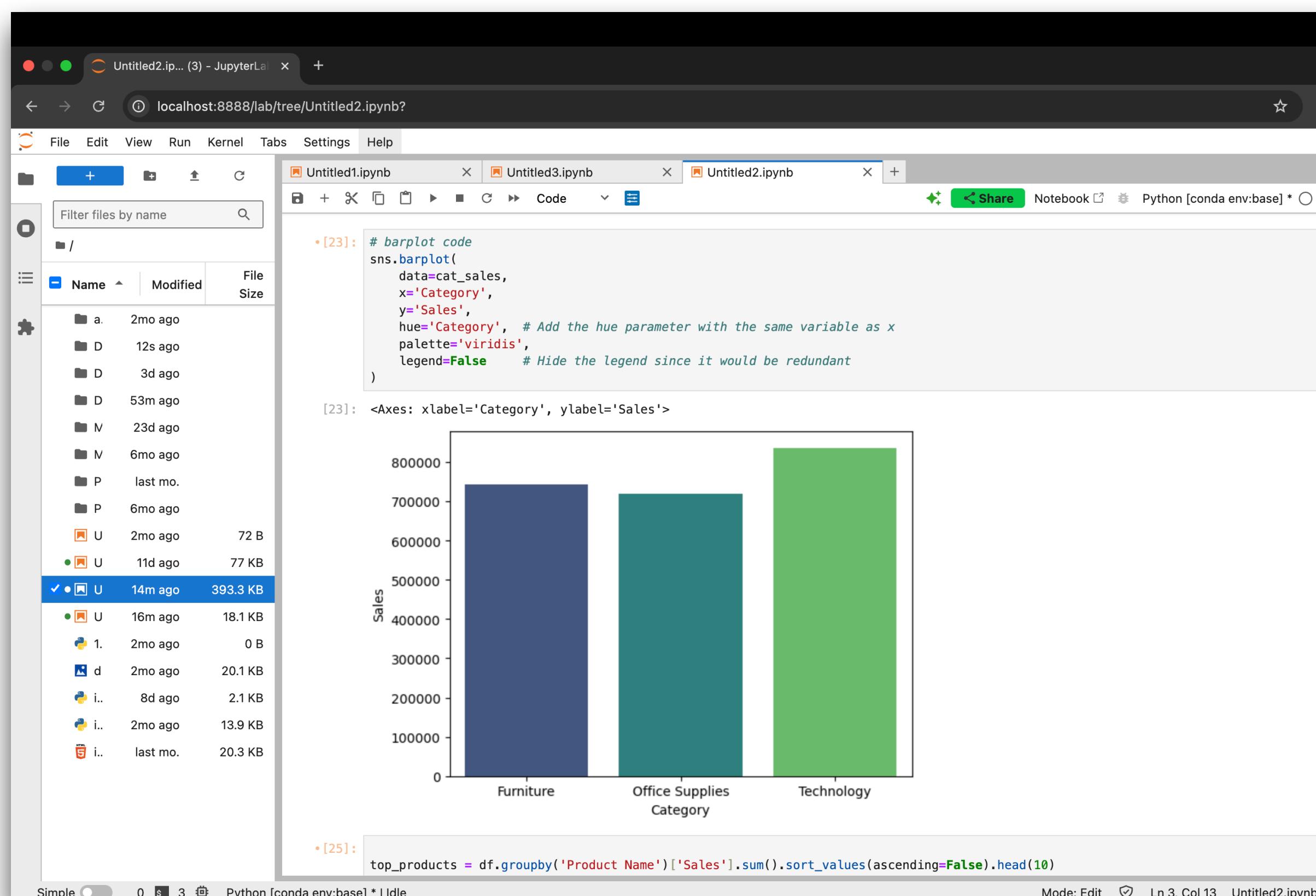
The output of the code is:

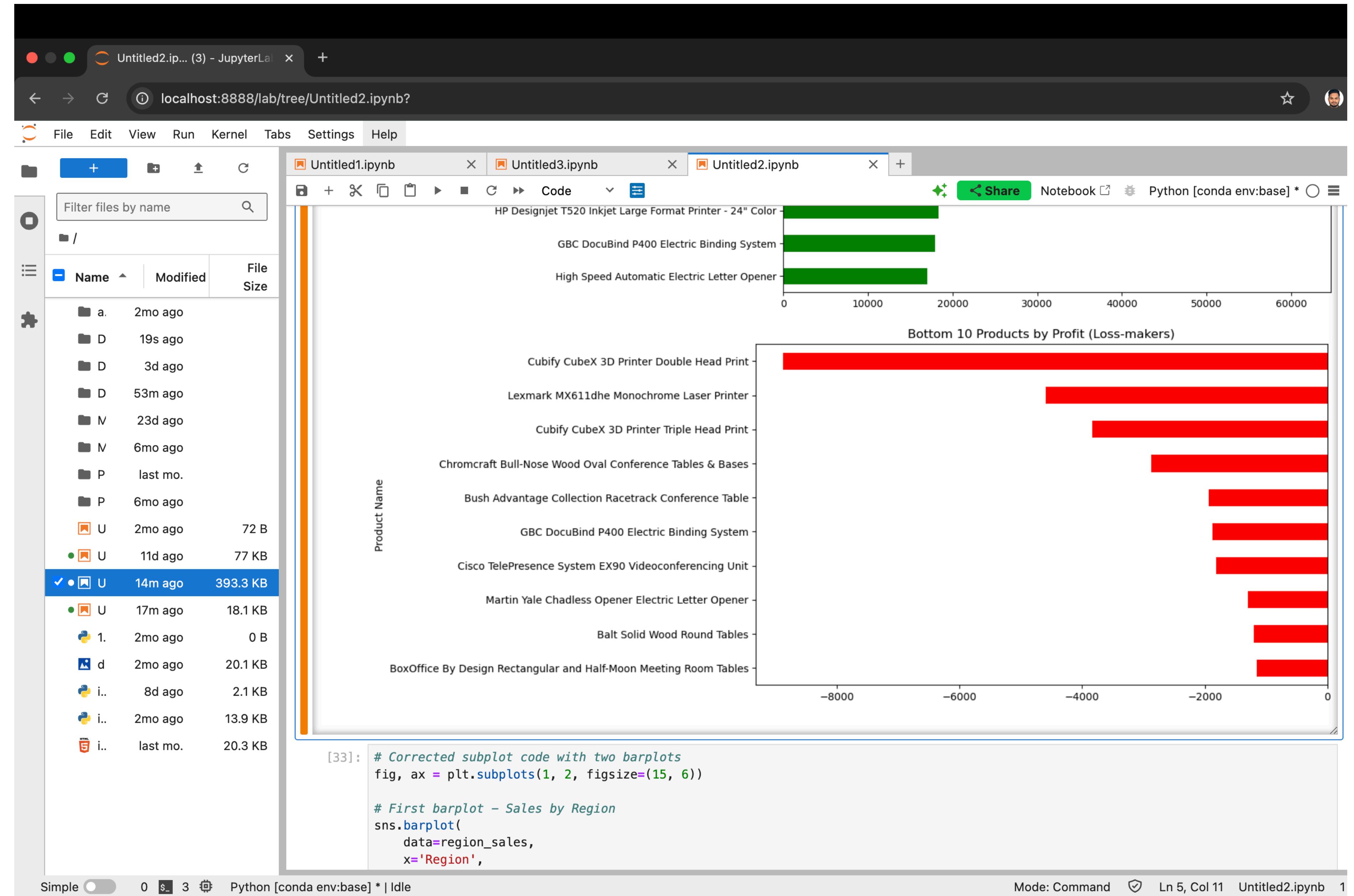
```
Total Sales: $2,297,200.86
Total Profit: $286,397.02
Total Orders: 5009
```

Below the code cell is a line plot titled "Monthly Sales Trend". The plot shows monthly sales in dollars over time. The y-axis is labeled "Sales (\$)" and ranges from 40,000 to 120,000. The x-axis is labeled "Month\_Year" and shows dates from 2014-01 to 2017-11. The plot displays a highly volatile trend with several sharp peaks and troughs, indicating significant seasonal fluctuations in sales.



# Barplot/ Top 10 Products by sale top 10 / Top 10 Products by sale Bottom 10





# Subplot with 2 Barplots.

Untitled2.ipynb (3) - JupyterLab

localhost:8888/lab/tree/Untitled2.ipynb?

File Edit View Run Kernel Tabs Settings Help

Untitled1.ipynb Untitled3.ipynb Untitled2.ipynb

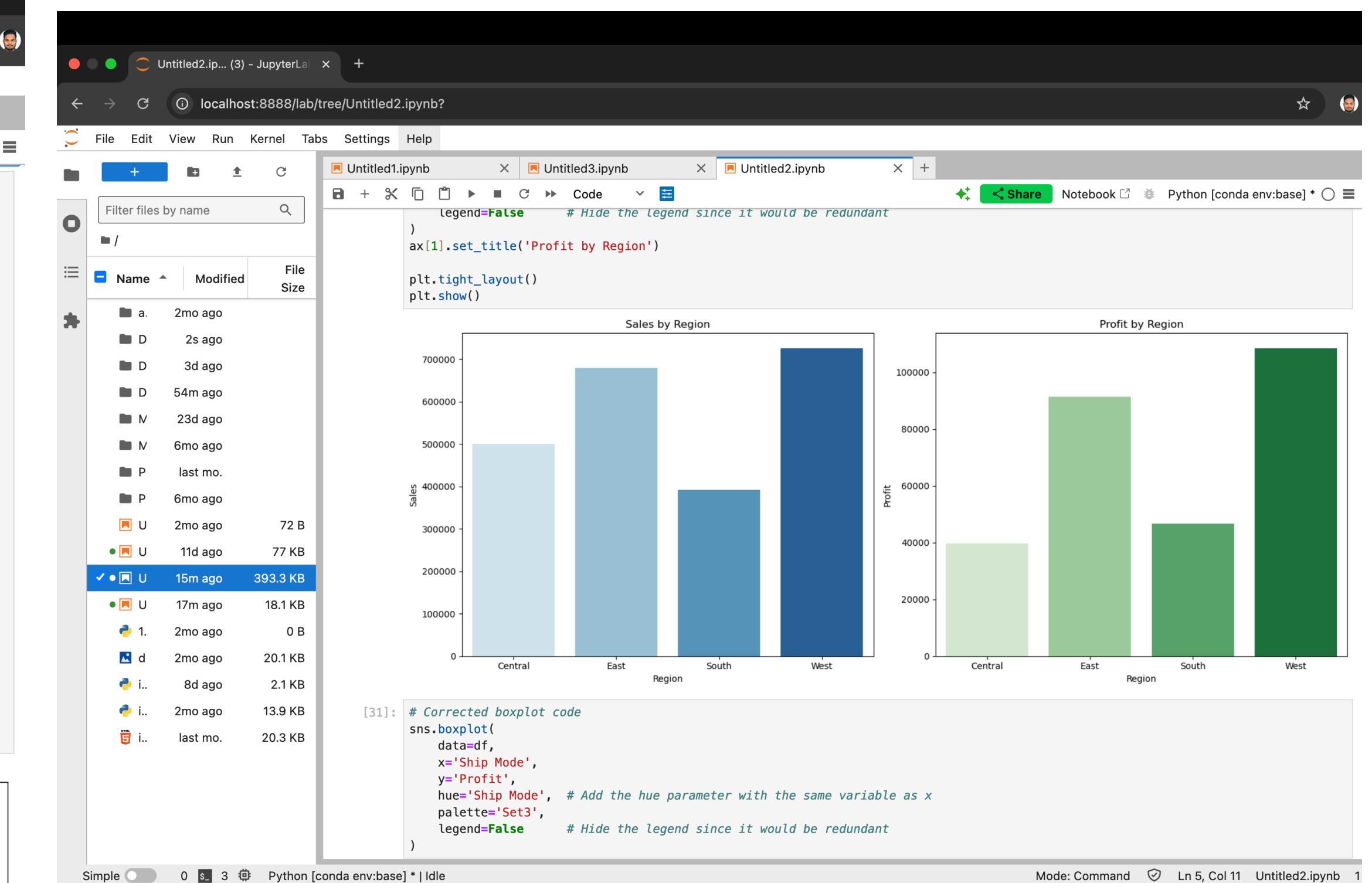
Share Notebook Python [conda env:base] \* |

Filter files by name

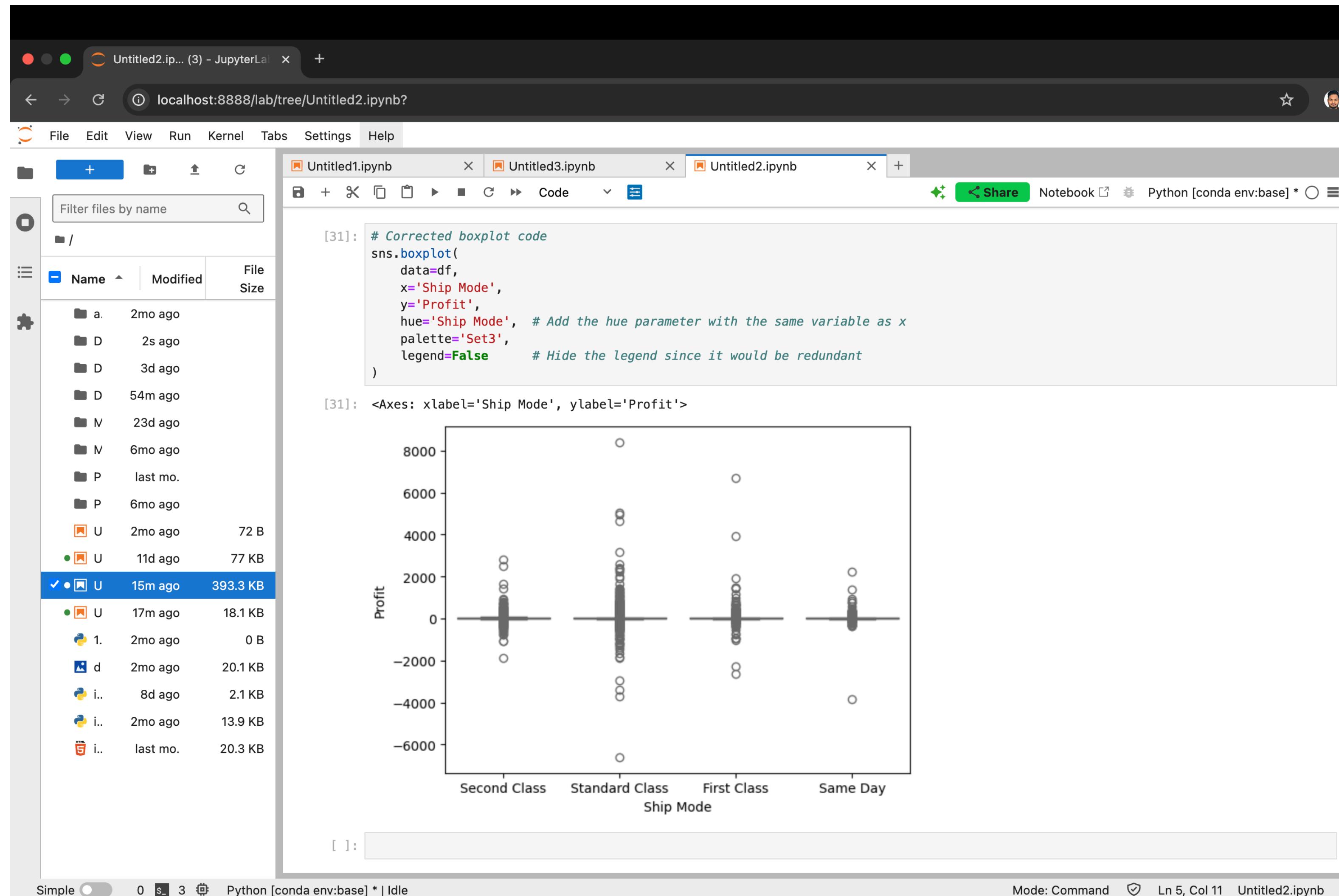
[33]: # Corrected subplot code with two barplots  
fig, ax = plt.subplots(1, 2, figsize=(15, 6))  
  
# First barplot - Sales by Region  
sns.barplot(  
 data=region\_sales,  
 x='Region',  
 y='Sales',  
 ax=ax[0],  
 hue='Region', # Add the hue parameter with the same variable as x  
 palette='Blues',  
 legend=False # Hide the legend since it would be redundant  
)  
ax[0].set\_title('Sales by Region')  
  
# Second barplot - Profit by Region  
sns.barplot(  
 data=region\_profit,  
 x='Region',  
 y='Profit',  
 ax=ax[1],  
 hue='Region', # Add the hue parameter with the same variable as x  
 palette='Greens',  
 legend=False # Hide the legend since it would be redundant  
)  
ax[1].set\_title('Profit by Region')  
  
plt.tight\_layout()  
plt.show()

Sales by Region

Profit by Region



# BarPlot.



# **Namratha Reddy Kaluvayee**

Linkedin: [www.linkedin.com/in/namratha-kaluvayee](https://www.linkedin.com/in/namratha-kaluvayee)

# **SGVS Thribhuvan Kambhamettu**

Linkedin: [www.linkedin.com/in/thribhuvankambhamettu](https://www.linkedin.com/in/thribhuvankambhamettu)