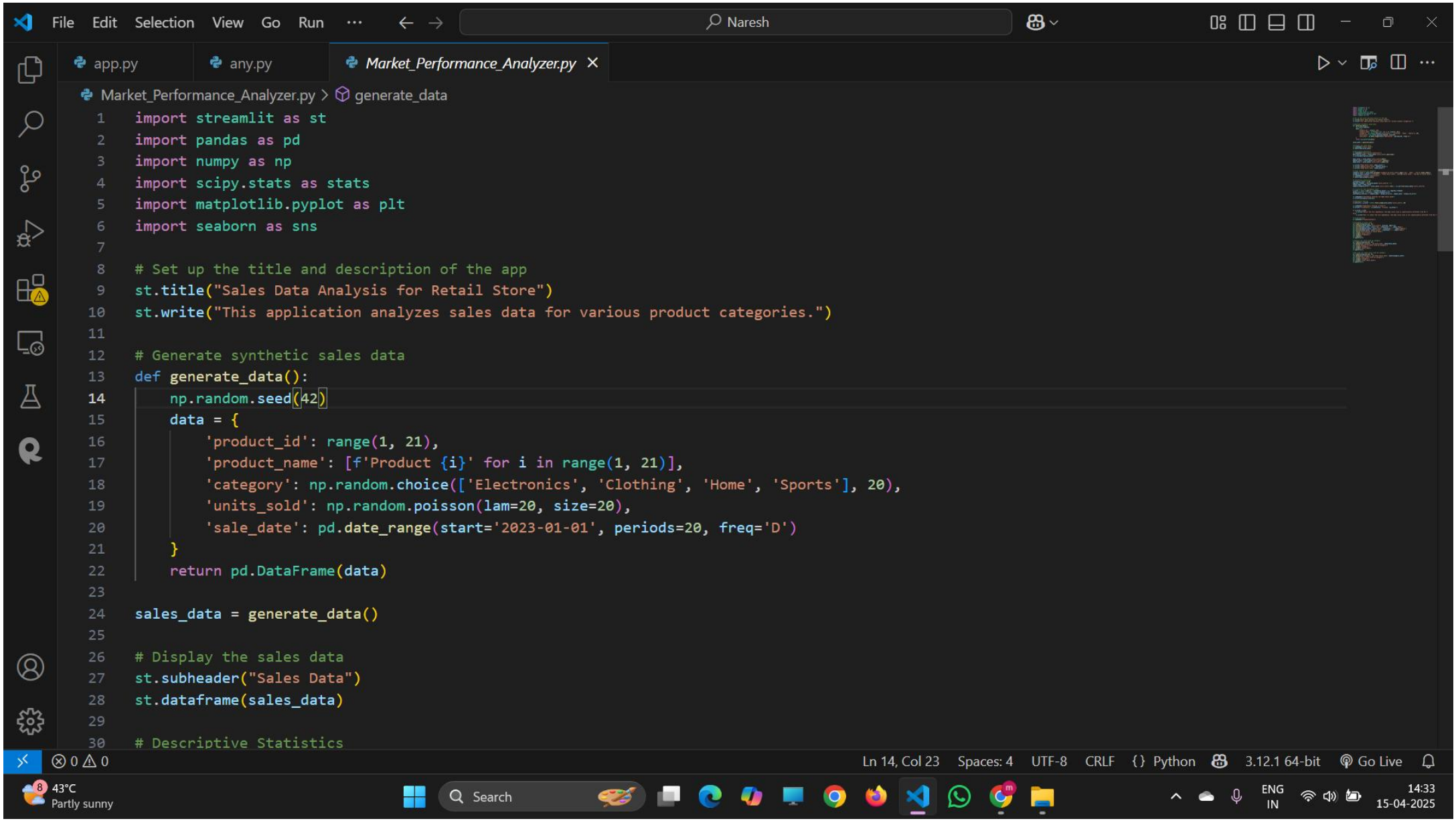
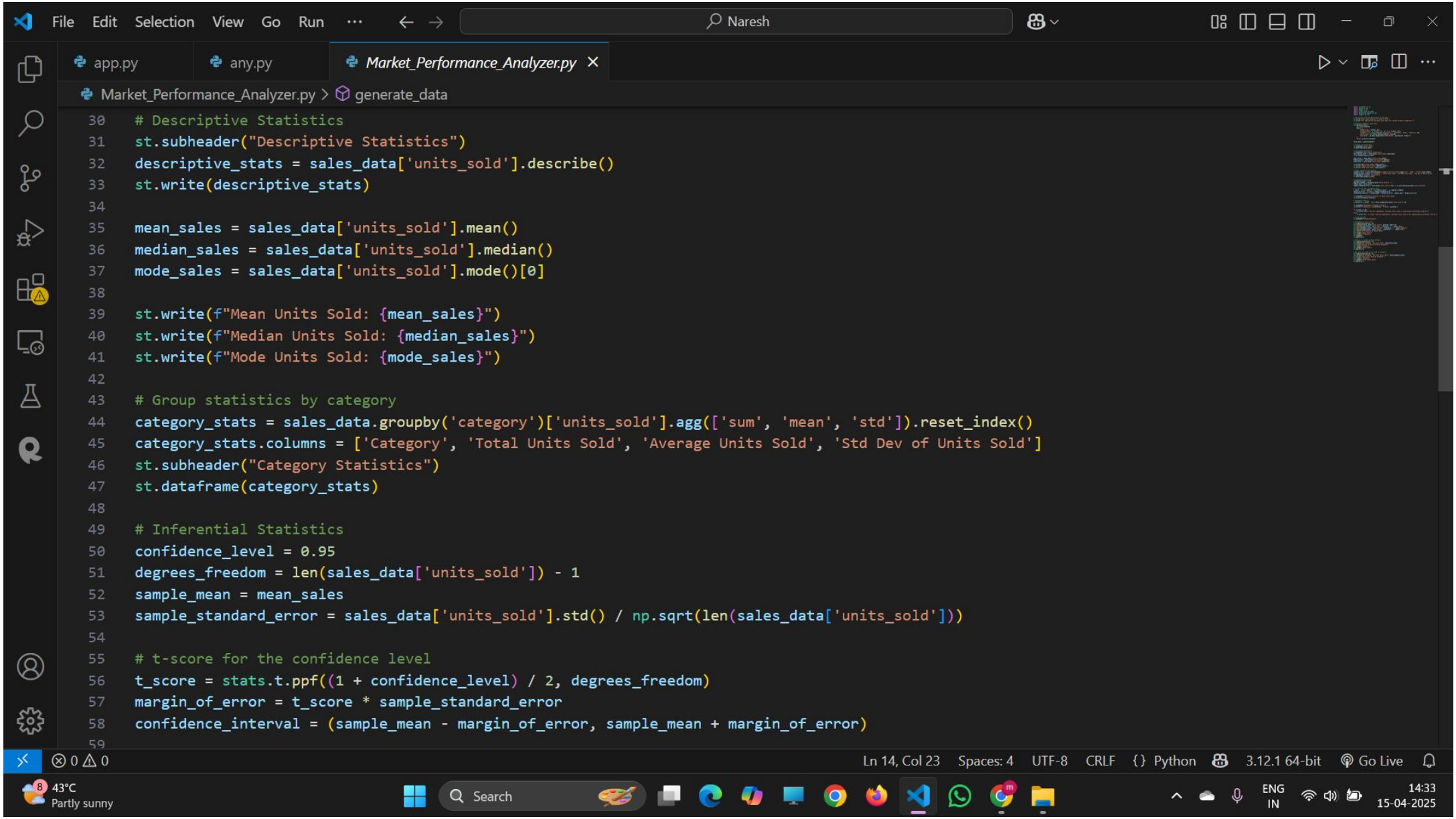
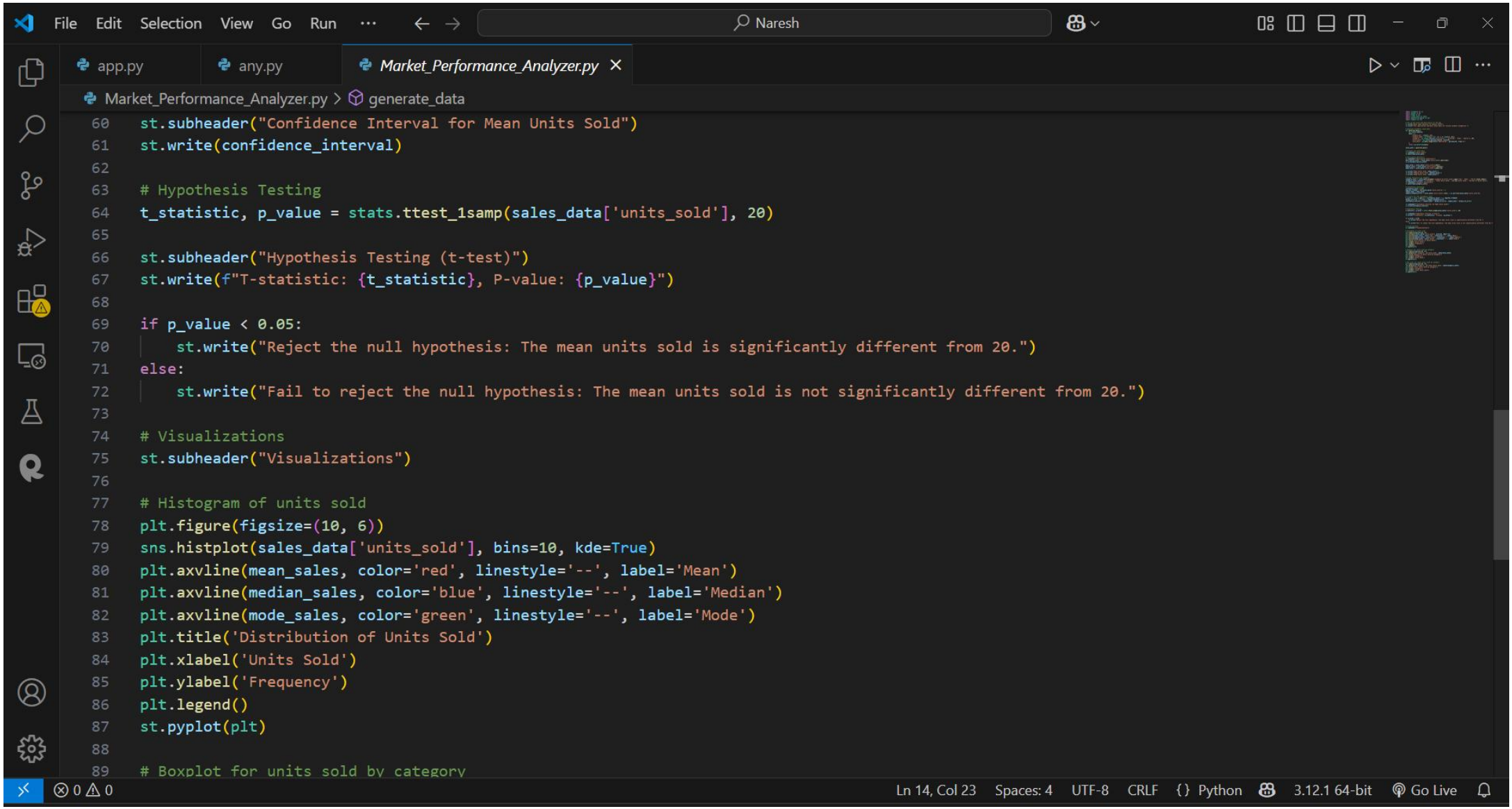
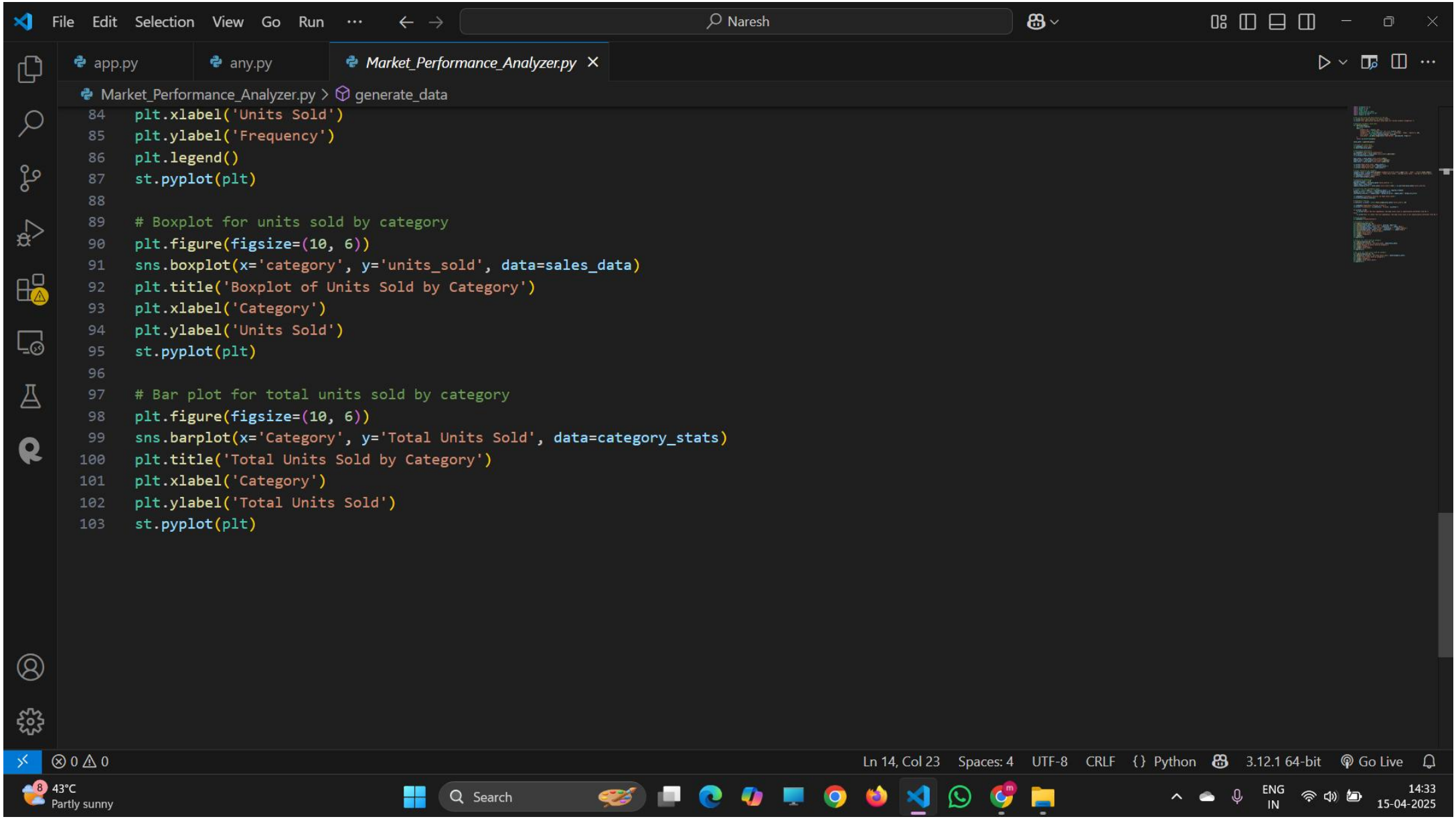

File Edit Selection View Go Run
Naresh
app.py
any.py
Market_Performance_Analyzer.py
Market_Performance_Analyzer.py > generate_data
import streamlit as st
import pandas as pd
import numpy as np
import scipy.stats as stats
import matplotlib.pyplot as plt
import seaborn as sns
# Set up the title and description of the app
st.title("Sales Data Analysis for Retail Store")
st.write("This application analyzes sales data for various product categories.")
# Generate synthetic sales data
def generate_data():
np.random.seed(42)
data = {
'product_id': range(1, 21),
'product_name': [f'Product {i}' for i in range(1, 21)],
'category': np.random.choice(['Electronics', 'Clothing', 'Home', 'Sports'], 20),
'units_sold': np.random.poisson(lam=20, size=20),
'sale_date': pd.date_range(start='2023-01-01', periods=20, freq='D')
}
return pd.DataFrame(data)
sales_data = generate_data()
# Display the sales data
st.subheader("Sales Data")
st.dataframe(sales_data)
# Descriptive Statistics
Ln 14, Col 23 Spaces: 4 UTF-8 CRLF Python 3.12.1 64-bit Go Live
43°C Partly sunny
Search
ENG IN
14:33 15-04-2025


File Edit Selection View Go Run ···
Naresh
app.py
any.py
Market_Performance_Analyzer.py
Market_Performance_Analyzer.py > generate_data
# Descriptive Statistics
st.subheader("Descriptive Statistics")
descriptive_stats = sales_data['units_sold'].describe()
st.write(descriptive_stats)
mean_sales = sales_data['units_sold'].mean()
median_sales = sales_data['units_sold'].median()
mode_sales = sales_data['units_sold'].mode()[0]
st.write(f"Mean Units Sold: {mean_sales}")
st.write(f"Median Units Sold: {median_sales}")
st.write(f"Mode Units Sold: {mode_sales}")
# Group statistics by category
category_stats = sales_data.groupby('category')['units_sold'].agg(['sum', 'mean', 'std']).reset_index()
category_stats.columns = ['Category', 'Total Units Sold', 'Average Units Sold', 'Std Dev of Units Sold']
st.subheader("Category Statistics")
st.dataframe(category_stats)
# Inferential Statistics
confidence_level = 0.95
degrees_freedom = len(sales_data['units_sold']) - 1
sample_mean = mean_sales
sample_standard_error = sales_data['units_sold'].std() / np.sqrt(len(sales_data['units_sold']))
# t-score for the confidence level
t_score = stats.t.ppf((1 + confidence_level) / 2, degrees_freedom)
margin_of_error = t_score * sample_standard_error
confidence_interval = (sample_mean - margin_of_error, sample_mean + margin_of_error)
Ln 14, Col 23 Spaces: 4 UTF-8 CRLF { } Python 3.12.1 64-bit Go Live
43°C Partly sunny
Search
ENG IN
14:33 15-04-2025


File Edit Selection View Go Run ···
Naresh
app.py
any.py
Market_Performance_Analyzer.py
Market_Performance_Analyzer.py > generate_data
st.subheader("Confidence Interval for Mean Units Sold")
st.write(confidence_interval)
# Hypothesis Testing
t_statistic, p_value = stats.ttest_1samp(sales_data['units_sold'], 20)
st.subheader("Hypothesis Testing (t-test)")
st.write(f"T-statistic: {t_statistic}, P-value: {p_value}")
if p_value < 0.05:
st.write("Reject the null hypothesis: The mean units sold is significantly different from 20.")
else:
st.write("Fail to reject the null hypothesis: The mean units sold is not significantly different from 20.")
# Visualizations
st.subheader("Visualizations")
# Histogram of units sold
plt.figure(figsize=(10, 6))
sns.histplot(sales_data['units_sold'], bins=10, kde=True)
plt.axvline(mean_sales, color='red', linestyle='--', label='Mean')
plt.axvline(median_sales, color='blue', linestyle='--', label='Median')
plt.axvline(mode_sales, color='green', linestyle='--', label='Mode')
plt.title('Distribution of Units Sold')
plt.xlabel('Units Sold')
plt.ylabel('Frequency')
plt.legend()
st.pyplot(plt)
# Boxplot for units sold by category
0 0
Ln 14, Col 23 Spaces: 4 UTF-8 CRLF {} Python 3.12.1 64-bit Go Live


File Edit Selection View Go Run ···
Naresh
app.py
any.py
Market_Performance_Analyzer.py
Market_Performance_Analyzer.py > generate_data
84 plt.xlabel('Units Sold')
85 plt.ylabel('Frequency')
86 plt.legend()
87 st.pyplot(plt)
88
89 # Boxplot for units sold by category
90 plt.figure(figsize=(10, 6))
91 sns.boxplot(x='category', y='units_sold', data=sales_data)
92 plt.title('Boxplot of Units Sold by Category')
93 plt.xlabel('Category')
94 plt.ylabel('Units Sold')
95 st.pyplot(plt)
96
97 # Bar plot for total units sold by category
98 plt.figure(figsize=(10, 6))
99 sns.barplot(x='Category', y='Total Units Sold', data=category_stats)
100 plt.title('Total Units Sold by Category')
101 plt.xlabel('Category')
102 plt.ylabel('Total Units Sold')
103 st.pyplot(plt)
⊗ 0 ⚠ 0
Ln 14, Col 23 Spaces: 4 UTF-8 CRLF { } Python 3.12.1 64-bit Go Live
43°C
Partly sunny
Search
ENG
IN
14:33
15-04-2025

Sales Data Analysis for Retail Store

This application analyzes sales data for various product categories.

Sales Data

	product_id	product_name	category	units_sold	sale_date
0	1	Product 1	Home	25	2023-01-01 00:00:00
1	2	Product 2	Sports	15	2023-01-02 00:00:00
2	3	Product 3	Electronics	17	2023-01-03 00:00:00
3	4	Product 4	Home	19	2023-01-04 00:00:00
4	5	Product 5	Home	21	2023-01-05 00:00:00
5	6	Product 6	Sports	17	2023-01-06 00:00:00
6	7	Product 7	Electronics	19	2023-01-07 00:00:00
7	8	Product 8	Electronics	16	2023-01-08 00:00:00
8	9	Product 9	Home	21	2023-01-09 00:00:00
9	10	Product 10	Clothing	21	2023-01-10 00:00:00

Descriptive Statistics

	units_sold
count	20
mean	18.8
std	3.3023
min	13
25%	17
50%	18.5
75%	21
max	25

Mean Units Sold: 18.8

Median Units Sold: 18.5

Mode Units Sold: 17

Category Statistics

	Category	Total Units Sold	Average Units Sold	Std Dev of Units Sold
0	Clothing	21	21	None

Category Statistics

	Category	Total Units Sold	Average Units Sold	Std Dev of Units Sold
0	Clothing	21	21	None
1	Electronics	73	18.25	2.2174
2	Home	181	20.1111	3.7231
3	Sports	101	16.8333	2.7142

Confidence Interval for Mean Units Sold

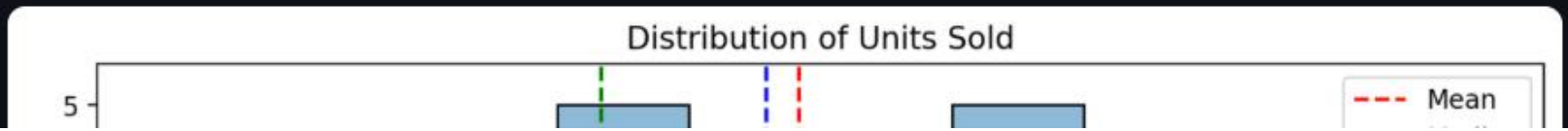
```
(np.float64(17.254470507823573), np.float64(20.34552949217643))
```

Hypothesis Testing (t-test)

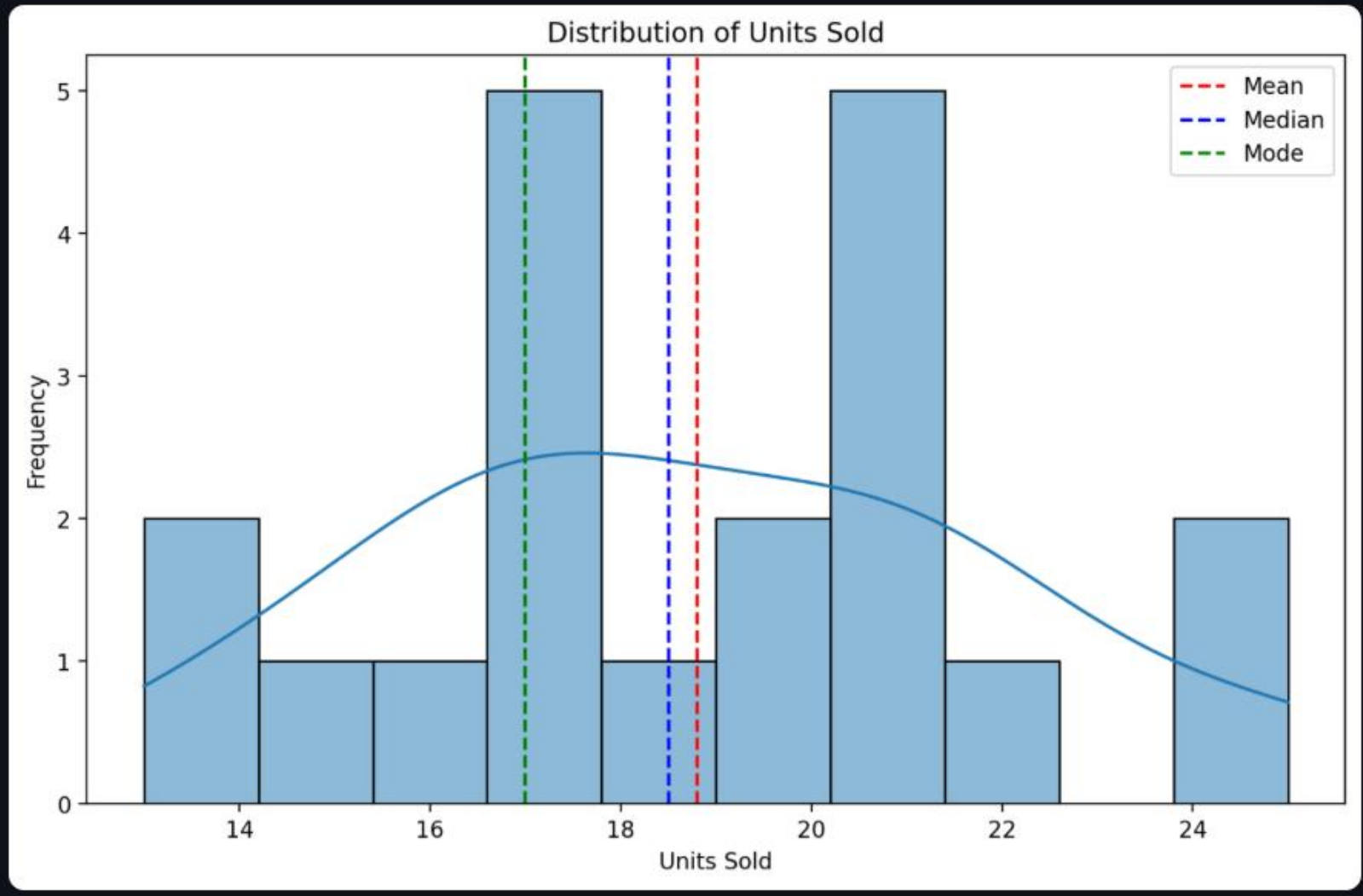
T-statistic: -1.6250928099424466, P-value: 0.12061572226781002

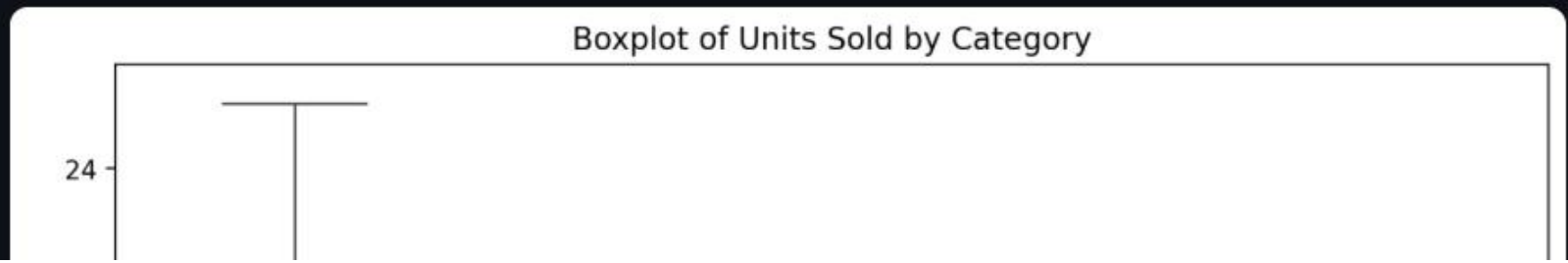
Fail to reject the null hypothesis: The mean units sold is not significantly different from 20.

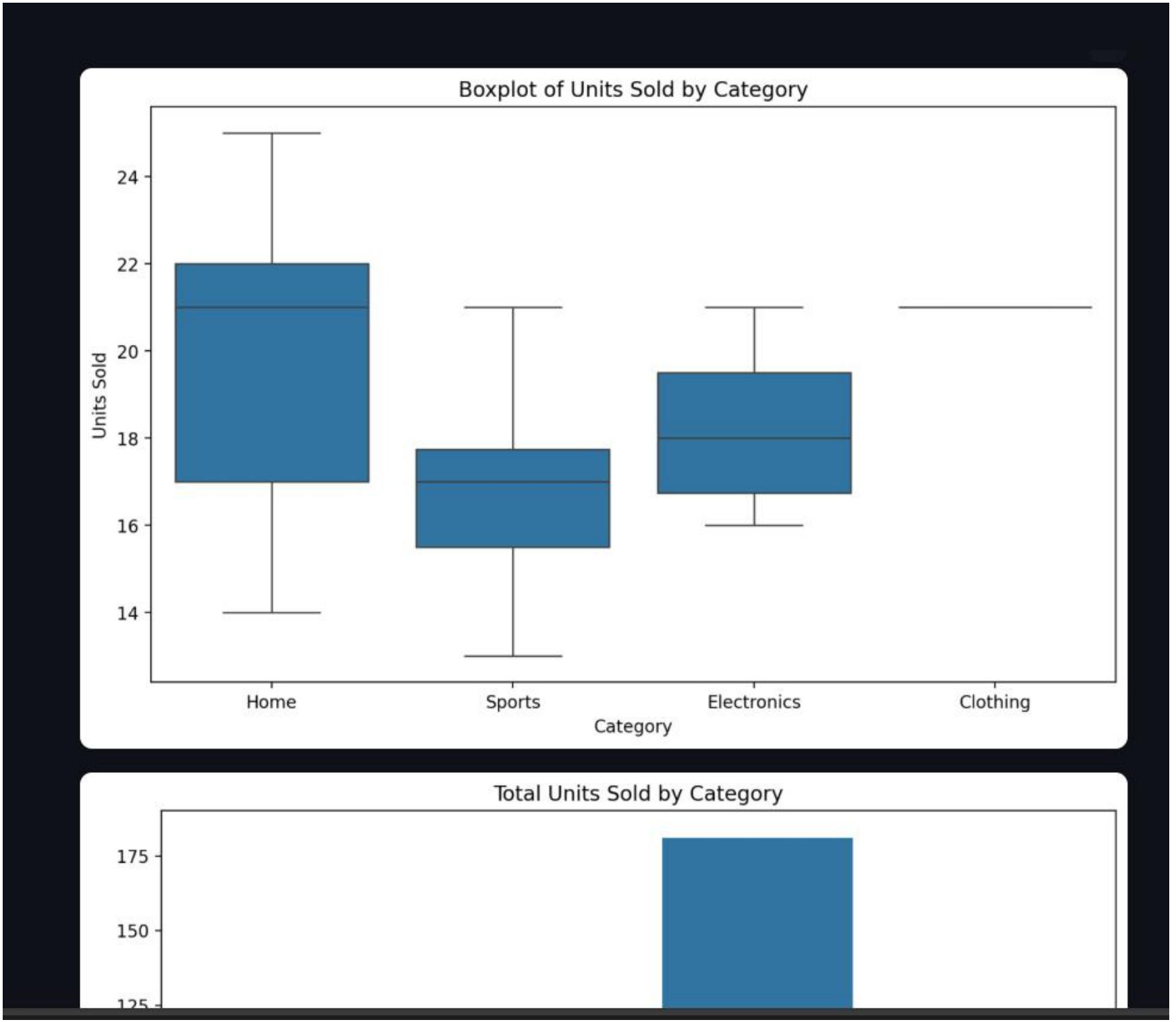
Visualizations

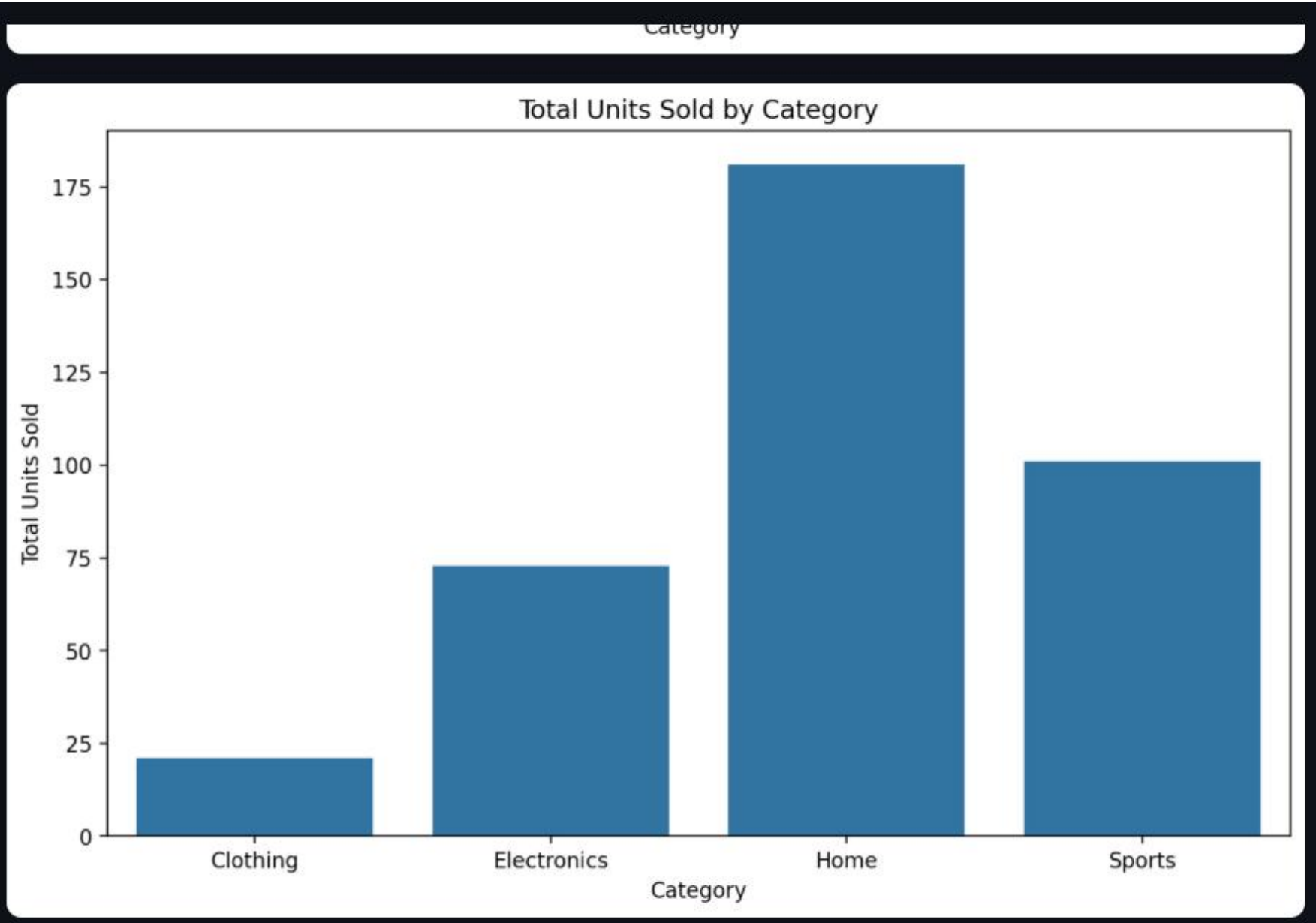


Visualizations






Boxplot of Units Sold by Category
Units Sold
24
22
20
18
16
14
Home
Sports
Electronics
Clothing
Category
Total Units Sold by Category
175
150
125


Category
Total Units Sold by Category
175
150
125
100
75
50
25
0
Total Units Sold
Clothing
Electronics
Home
Sports
Category