Project Documentation

- **Project title:** Restaurant Sales and Feedback Analysis
- **Name:** Daroor Pavan Kalyan
- **Date:** 22/12/2024

Introduction

In the competitive restaurant industry, understanding sales trends and customer feedback is crucial for improving operations and customer satisfaction. This project aims to analyse restaurant data to uncover actionable insights.

Project Objective

The aim of this project is to analyse restaurant sales and customer feedback to derive actionable insights, improve customer satisfaction, and identify potential growth opportunities for restaurant business.

Data Source

1) Sales Data: Information about restaurant orders, total sales amount, items sold, and customer preferences.
    - **File:** cleaned_sales_data.csv
    - **Key Columns:** Restaurant Name, Order Date, Total Amount, Items Sold, City
2) Feedback Data: Customer reviews and ratings for various restaurants.
    - **File:** fixed_feedback_data.csv
    - **Key Columns:** Restaurant Name, Rating, Reviews Count, City, Date.

Tools And Technologies Used

- **Python:** Data Cleaning, Data Processing and visualization.
- **Libraries:** Pandas, Plotly, Streamlit, Psutil.
- **Streamlit:** Dashboard Creation for interactive analysis.

Data Cleaning

- **Objective:** Ensure data quality and consistency by addressing missing values, formatting issues, and spelling errors.
- **Step Taken:**

1) **Missing Values:** Filled missing numerical values with their column mean
2) **Inconsistent Date Formats:** Standardized the Order Date column into single date format (YYYY-MM-DD) to ensure consistency and allow accurate time-series analysis.
3) **Spelling Mistakes in City Names:** Standardized common spelling errors in city names to maintain the consistency.
4) **Duplicate Records:** Removed duplicate rows to ensure data integrity

- **Outcome:** Cleaned and validated datasets ready for analysis

Sales Data Dashboard

Objective: The goal is to provide insights into restaurant sales trends, performance by restaurant and city, and customer preferences based on sales data. The dashboard allows users to filter the data by selecting a specific restaurant.

Key Insights

These insights are derived from the interactive dashboard, which provides detailed visualizations and analysis. The main findings are:

Sales Data

- **Sales by Restaurant:** Restaurant with higher sales volumes attract more customers and likely have better marketing strategies or prime locations.
- **Items Sold Over Time:** Sales show seasonal spikes, especially during weekends and holidays.
- **Sales by City:** Urban areas contribute significantly to total sales.
- **Average Items Sold per Restaurant:** Restaurant with higher average items sold are likely to have a larger menu.

Feedback Data Dashboard

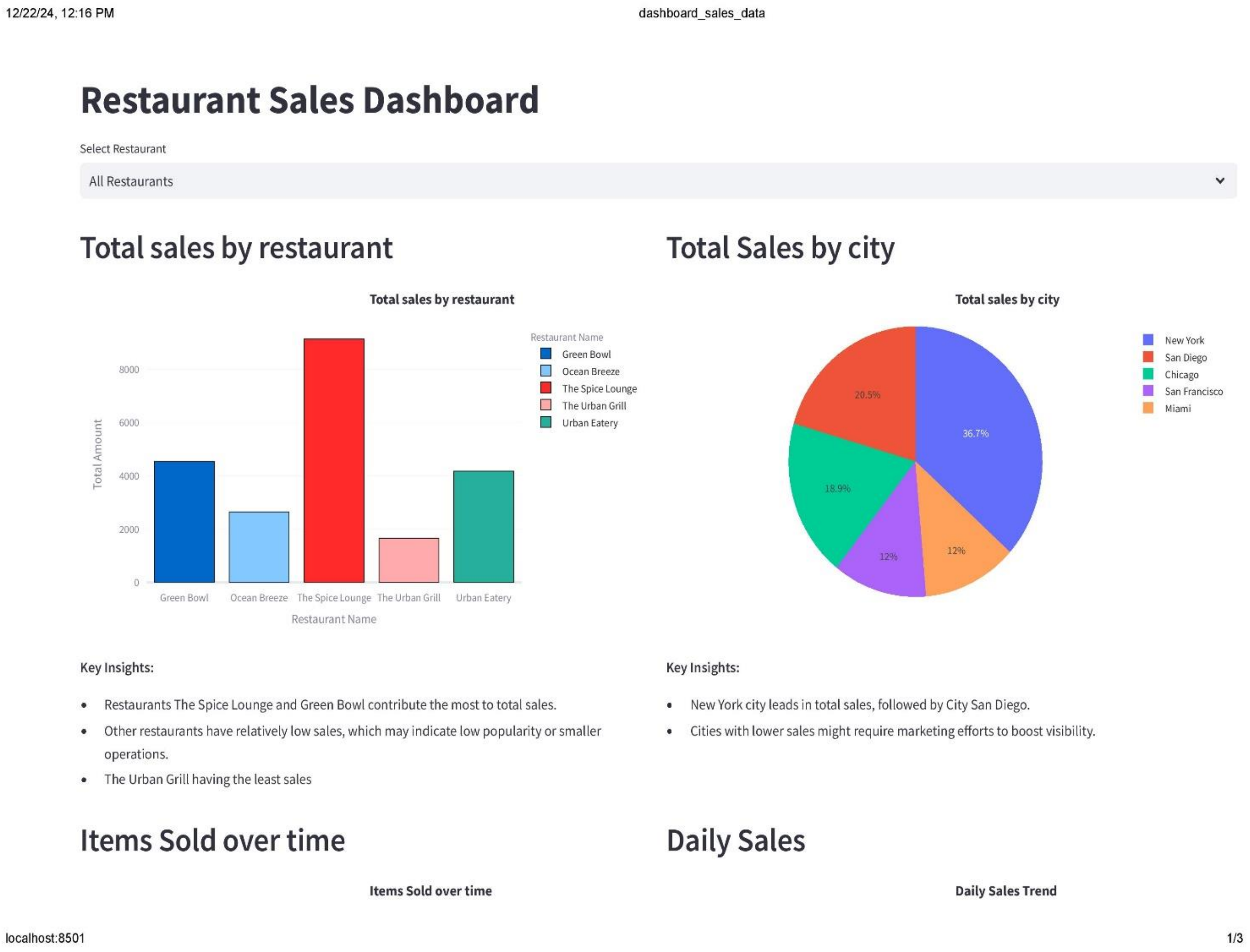
Objective: The purpose of this dashboard is to analyse customer satisfaction based on restaurant ratings and reviews. The dashboard allows users to filter the data by selecting a specific restaurant.

Feedback Data

- **Average Ratings by Restaurant:** Restaurants with higher ratings demonstrate better customer service and food quality.

- **Rating Distribution:** Some restaurants consistently receive lower ratings, indicating areas for improvement in service or menu quality.
- **Reviews Per Restaurant:** Restaurants with a higher number of reviews may have stronger customer engagement or social media presence.
- **Ratings Over Time:** Positive trends in ratings over time suggest improvements in service or food quality.
- **Average Rating Per City:** Certain cities have higher average ratings, possibly due to cultural or regional preferences.

Sales Dashboard



12/22/24, 12:16 PM dashboard_sales_data

Restaurant Sales Dashboard

Select Restaurant

All Restaurants

Total sales by restaurant

Key Insights:

- Restaurants The Spice Lounge and Green Bowl contribute the most to total sales.
- Other restaurants have relatively low sales, which may indicate low popularity or smaller operations.
- The Urban Grill having the least sales

Total Sales by city

Key Insights:

- New York city leads in total sales, followed by City San Diego.
- Cities with lower sales might require marketing efforts to boost visibility.

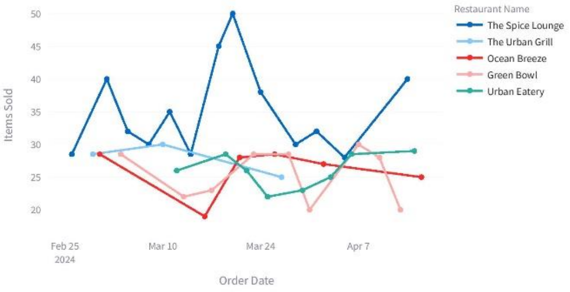
Items Sold over time

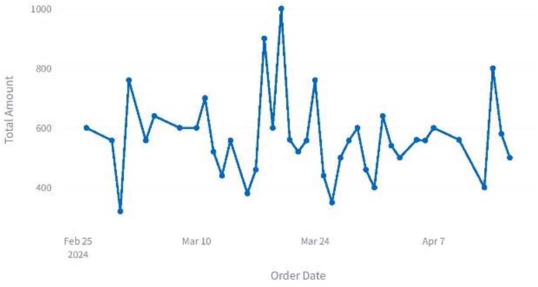
Items Sold over time

Daily Sales

Daily Sales Trend

localhost:8501 1/3





Key Insights:

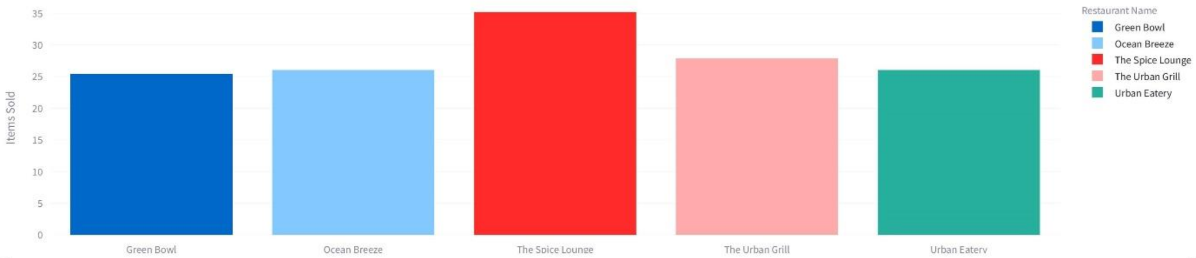
- The Spice Lounge Restaurant shows good sales for items both on weekdays and weekends, likely because it is located in a busy area.
- Items sold show seasonal spikes, particularly during weekends and holidays.
- Consistent sales trends indicate stable demand for popular restaurants.

Key Insights:

- Sales are highest during festival seasons and weekends.
- A noticeable dip in sales during weekdays may indicate a need for weekday promotions.

Average Items Sold Per Restaurant

Average Items Sold Per Restaurant



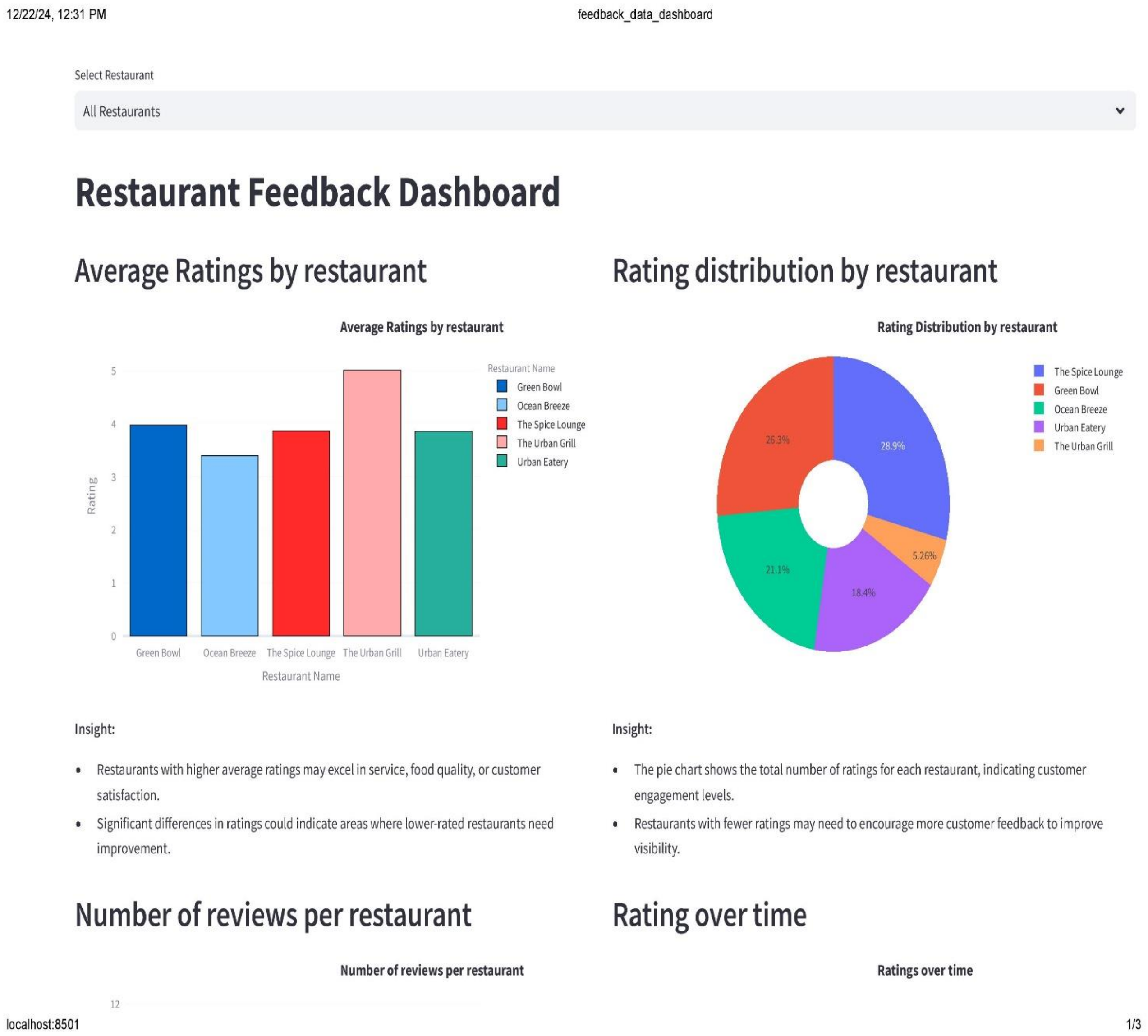
Key Insights:

- Restaurants with higher average items sold are likely to have a larger menu or higher customer loyalty.
- Lower average sales could indicate a need for menu refinement or improved marketing efforts to attract more customers.

Conclusion:

- Some restaurants like The Spice Lounge,Urban Eatery are doing better than others because of factors like location, menu, or loyal customers.
- Restaurants with lower sales may want to improve their menu, try promotions, or focus on their location.
- It's a good idea for restaurants to look at their sales trends and focus on what customers like the most.

Feedback Data Dashboard



12/22/24, 12:31 PM feedback_data_dashboard

Select Restaurant

All Restaurants

Restaurant Feedback Dashboard

Average Ratings by restaurant

Insight:

- Restaurants with higher average ratings may excel in service, food quality, or customer satisfaction.
- Significant differences in ratings could indicate areas where lower-rated restaurants need improvement.

Rating distribution by restaurant

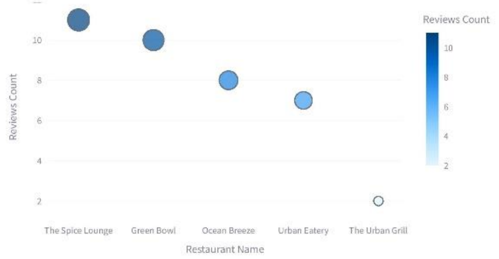
Insight:

- The pie chart shows the total number of ratings for each restaurant, indicating customer engagement levels.
- Restaurants with fewer ratings may need to encourage more customer feedback to improve visibility.

Number of reviews per restaurant

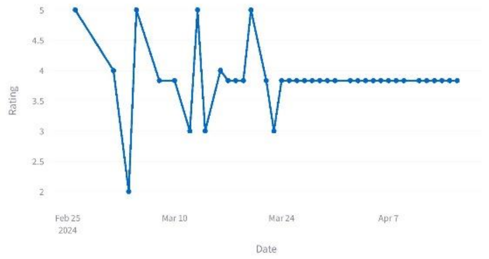
Rating over time

localhost:8501 1/3



Insight:

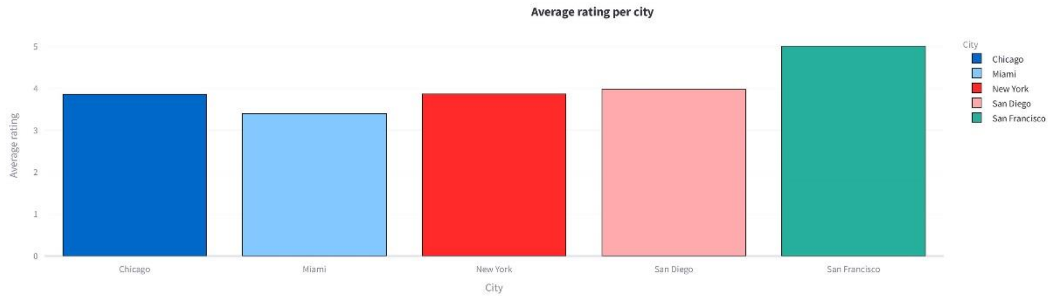
- Restaurants with more reviews are likely more popular or visible to customers.
- Disparities in review counts suggest opportunities for marketing and outreach for less-reviewed restaurants.



Insight:

- Ratings over time reveal trends such as seasonal customer satisfaction or potential improvements in service.
- Sudden drops in ratings may indicate service issues or negative customer experiences that need to be addressed.

Average rating per city



Insight:

- Cities with higher average ratings might represent regions with better restaurant standards or higher customer satisfaction.
- Cities with lower ratings may highlight the need for service improvements or new customer engagement strategies.

Final Conclusion

Overall Insights:

- Restaurants with higher average ratings and reviews demonstrate strong customer satisfaction and consistent quality. These establishments should continue their successful practices while exploring opportunities for expansion.
- Cities with lower ratings or fewer reviews represent untapped opportunities for targeted improvements in service quality and marketing strategies.
- Seasonal variations and trends in ratings over time highlight the importance of maintaining consistent service during high-demand periods, such as weekends and holidays.
- Encouraging customers to provide feedback, especially for underrepresented restaurants, can help improve visibility and drive growth.

Actionable Steps:

- Focus on improving the service and visibility of lower-rated restaurants.
- Invest in targeted campaigns for cities with lower average ratings to boost customer engagement.
- Monitor seasonal trends and proactively prepare for high-demand periods to ensure customer satisfaction.
- Develop advertisements highlighting each restaurant's unique offerings, and include a variety of international cuisines to attract a diverse customer base.

Conclusion

This project highlights key sales and feedback tends that can help restaurant business optimize their performance. By leveraging these insights, restaurants can enhance customer satisfaction, improve operational efficiency, and drive growth. The dashboards provide an interactive platform for monitoring performance and making data-driven decisions

Project Code

Data Cleaning

Sales Data cleaning

```
import pandas as pd

try:
    data=pd.read_csv("C:\\Projects\\DataAnalysisProjects\\Restaurant_Sales_And_Feedback_Analysis\\Unclean_data\\Sales_data.csv")
    print("Data loaded successfully.")
except Exception as e:
    print("Error in data loading:",e)


try:
    city_corrections={
        "Chicgo":"Chicago",
        "Chigago":"Chicago",
        "San Diefo":"San Diego",
        "San Diege":"San Diego",
        "Maimi":"Miami",
        "New Yrk":"New York",
        "NYC":"New York"
    }
    if 'City' in data.columns:
        data["City"]=data["City"].replace(city_corrections)
        print("City names changed successfully.")
    else:
        print("City column is not found.")
except Exception as e:
    print("Error standardising the city name:",e)

try:
    if 'Items Sold' in data.columns:
        mean_items_sold=data['Items Sold'].mean()
        data['Items Sold']=data['Items Sold'].fillna(mean_items_sold)
        print("Missing values in 'Items Sold' column handled successfully.")
    if 'Total Amount' in data.columns:
        mean_total_amount=data['Total Amount'].mean()
        data['Total Amount']=data['Total Amount'].fillna(mean_total_amount)
        print("Missing values in 'Total Amount' handled successfully.")
except Exception as e:
    print("Error in handling the missing values:",e)

try:
    if 'Order Date' in data.columns:
        data['Order Date']=pd.to_datetime(data['Order Date'],errors='coerce')
        data=data.dropna(subset=['Order Date'])
        print("Order Date normalised and invalid rows removed successfully.")
    else:
        print("Order Date column not found.")
except Exception as e:
    print("Error in normalising Order Date:",e)
```

```
try:
    data=data.drop_duplicates()
    print("Duplicates removed successfully.")
except Exception as e:
    print("Error in removing duplicates:",e)

try:
    cleaned_data_path="C:\\Projects\\DataAnalysisProjects\\Restaurant_Sales_And_Feedback_Analysis\\Clean_data\\cleaned_sales_data.csv"
    data.to_csv(cleaned_data_path,index=False)
    print("Cleaned data saved successfully.")
except Exception as e:
    print("Error in loading data:",e)
```

Feedback data cleaning

```
import pandas as pd
from datetime import datetime

file_path="C:\\Projects\\DataAnalysisProjects\\Restaurant_Sales_And_Feedback_Analysis\\Unclean_data\\Feedback_analysis.csv"
try:
    data=pd.read_csv(file_path,header=0,on_bad_lines='skip')
    print("Data loaded successfully.")
except Exception as e:
    print("Error in loading data",e)
```

```
try:
    data.rename(columns=lambda x:x if "Unnamed" not in x else
"Date",inplace=True)
    print("Unnamed columns renamed successfully.")
except Exception as e:
    print("Error renaming Unnamed Columns: ",e)

try:
    data['Rating']=data['Rating'].fillna(data['Rating'].mean())
    print("Missing ratings filled successfully.")
except Exception as e:
    print("Erroro handling in missing ratings:",e)

try:
    data['Comments']=data['Comments'].fillna("No comments")
except Exception as e:
    print("Error handling missing comments:",e)

try:
    city_replacements={
    'NYC':'New York',
    'San Fran':'San Franciso',
    'Chi-Town':'Chicago'
    }
    data['City']=data['City'].replace(city_replacements)
    print("City names corrected successfully.")
except Exception as e:
    print("Error in correcting city name:",e)


try:
    data['Date']=pd.to_datetime(data['Date'],errors='coerce')
    print("Date Formats fixed successfully.")
except Exception as e:
    print("Error in fixing date format:",e)


try:
    intial_rows=len(data)
    data=data.dropna(subset=['Date'])
    removed_rows=intial_rows-len(data)
    print("Rows with missing dates removed successfully.",removed_rows," rows dropped.")
except Exception as e:
    print("Error removing rows with missing dates:",e)


fixed_file_path="C:\\Projects\\DataAnalysisProjects\\Restaurant_Sales_And_Feedback_Analysis\\Clean_data\\fixed_feedback_data.csv"
try:
    data.to_csv(fixed_file_path,index=False)
    print("Cleaned data successfully saved to",fixed_file_path)
except Exception as e:
    print("Error in saving cleaned data:",e)
```

Dashboard creation

Restaurant Sales Data

```
import pandas as pd
import plotly.express as px
import streamlit as st
import subprocess
import psutil

def is_streamlit_running():
    for proc in psutil.process_iter(attrs=['pid', 'name']):
        if 'streamlit' in proc.info['name'].lower():
            return True
    return False

if not is_streamlit_running():
    subprocess.run(['streamlit', 'run', 'C:\\Projects\\DataAnalysisProjects\\Restaurant_Sales_And_Feedback_Analysis\\Visualisations\\Sales_Data_Visualisations\\dashboard_sales_data.py'])
else:
    print("Streamlit app is already running!")

df=pd.read_csv("C:\\Projects\\DataAnalysisProjects\\Restaurant_Sales_And_Feedback_Analysis\\Clean_data\\cleaned_sales_data.csv")
df['Order Date']=pd.to_datetime(df['Order Date'])
st.set_page_config(layout="wide")

st.title("Restaurant Sales Dashboard")
```

```
restaurant_options=['All Restaurants']+df['Restaurant Name'].unique().tolist()
selected_restaurant=st.selectbox('Select Restaurant',restaurant_options)
if selected_restaurant!='All Restaurants':
    filtered_df=df[df['Restaurant Name']==selected_restaurant]
else:
    filtered_df=df

def create_figure(fig):
    fig.update_layout(
        height=400,
        width=700,
        title_x=0.5,
        margin=dict(t=50, l=50, r=50, b=50),
        showlegend=True,
        template="plotly_dark"
    )
    return fig

fig1=px.bar(filtered_df.groupby('Restaurant Name',as_index=False)['Total Amount'].sum(),x='Restaurant Name',y='Total Amount',title='Total sales by restaurant',color='Restaurant Name')
fig1=create_figure(fig1)

fig2=px.line(filtered_df,x='Order Date',y='Items Sold',color='Restaurant Name',title='Items Sold over time',markers=True)
fig2=create_figure(fig2)
```

```
fig3=px.pie(filtered_df,names='City',values='Total Amount',title='Total sales by city')
fig3=create_figure(fig3)

daily_sales=filtered_df.groupby("Order Date",as_index=False)["Total Amount"].sum()
fig4=px.line(daily_sales,x='Order Date',y='Total Amount',title='Daily Sales Trend',markers=True)
fig4=create_figure(fig4)

average_items=filtered_df.groupby('Restaurant Name',as_index=False)['Items Sold'].mean()
fig5=px.bar(average_items,x='Restaurant Name',y='Items Sold',color='Restaurant Name',title='Average Items Sold Per Restaurant')

col1, col2 = st.columns(2)
with col1:
    st.markdown("## Total sales by restaurant")
    st.plotly_chart(fig1, use_container_width=True)
    st.markdown("""
    **Key Insights:**
    - Restaurants The Spice Lounge and Green Bowl contribute the most to total sales.
    - Other restaurants have relatively low sales, which may indicate low popularity or smaller operations.
    - The Urban Grill having the least sales
    """)
```

```
with col2:
    st.markdown("## Total Sales by city")
    st.plotly_chart(fig3, use_container_width=True)
    st.markdown("""
    **Key Insights:**
    - New York city leads in total sales, followed by City San Diego.
    - Cities with lower sales might require marketing efforts to boost visibility.
    """)

col3, col4 = st.columns(2)
with col3:
    st.markdown("## Items Sold over time")
    st.plotly_chart(fig2, use_container_width=True)
    st.markdown("""
    **Key Insights:**
    - The Spice Lounge Restaurant shows good sales for items both on weekdays and weekends, likely because it is located in a busy area.
    - Items sold show seasonal spikes, particularly during weekends and holidays.
    - Consistent sales trends indicate stable demand for popular restaurants.
    """)


with col4:
    st.markdown("## Daily Sales")
    st.plotly_chart(fig4, use_container_width=True)
    st.markdown("""
    **Key Insights:**
    - Sales are highest during festival seasons and weekends.
    - A noticeable dip in sales during weekdays may indicate a need for weekday promotions.
    """)


st.markdown("## Average Items Sold Per Restaurant")
st.plotly_chart(fig5, use_container_width=True)
st.markdown("""
**Key Insights:**
- Restaurants with higher average items sold are likely to have a larger menu or higher customer loyalty.
- Lower average sales could indicate a need for menu refinement or improved marketing efforts to attract more customers.
""")
st.markdown("""
**Conclusion:**
- Some restaurants like The Spice Lounge,Urban Eatery are doing better than others because of factors like location, menu, or loyal customers.
- Restaurants with lower sales may want to improve their menu, try promotions, or focus on their location.
- It's a good idea for restaurants to look at their sales trends and focus on what customers like the most.
""")
```

Feedback Data

```
import streamlit as st
import plotly.express as px
import pandas as pd
import subprocess
import psutil
```

```
def is_streamlit_running():
    for proc in psutil.process_iter(attrs=['pid','name']):
        if 'streamlit' in proc.info['name'].lower():
            return True
    return False

if not is_streamlit_running():
    subprocess.run(['streamlit','run','C:\\Projects\\DataAnalysisProjects\\Restaurant_Sales_And_Feedback_Analysis\\Visualisations\\Feedback_data_visualisations\\feedback_data_dashboard.py'])
else:
    print("Streamlit app is already running!")

df=pd.read_csv("C:\\Projects\\DataAnalysisProjects\\Restaurant_Sales_And_Feedback_Analysis\\Clean_data\\fixed_feedback_data.csv")
df['Date']=pd.to_datetime(df['Date'])
st.set_page_config(layout='wide')

restaurant_options=['All Restaurants']+df['Restaurant Name'].unique().tolist()
select_restaurant=st.selectbox('Select Restaurant',restaurant_options)
if select_restaurant!='All Restaurants':
    filtered_df=df[df['Restaurant Name']==select_restaurant]
else:
    filtered_df=df

def create_figure(fig):
    fig.update_layout(
        height=400,
        width=700,
        title_x=0.5,
        margin=dict(t=50,l=50,r=50,b=50),
        showlegend=True,
        template='plotly_dark'
    )
    return fig

avg_rating=filtered_df.groupby('Restaurant Name')['Rating'].mean().reset_index()
reviews_count=filtered_df['Restaurant Name'].value_counts().reset_index()
reviews_count.columns=['Restaurant Name','Reviews Count']
ratings_over_time=filtered_df.groupby('Date')['Rating'].mean().reset_index()
avg_rating_city=filtered_df.groupby('City')['Rating'].mean().reset_index()
rating_distribution=filtered_df.groupby('Restaurant Name')['Rating'].value_counts().unstack().fillna(0)
rating_distribution['Total Ratings'] = rating_distribution.sum(axis=1)

fig1=create_figure(px.bar(avg_rating,x='Restaurant Name',y='Rating',title='Average Ratings by restaurant',color='Restaurant Name'))

fig2=create_figure(px.scatter(reviews_count,x='Restaurant Name',y='Reviews Count',title='Number of reviews per restaurant',color='Reviews Count',size='Reviews Count'))
```

```
fig3=create_figure(px.pie(rating_distribution,names=rating_distribution.index,values='Total Ratings',title='Rating Distribution by restaurant',hole=0.3))


fig4=create_figure(px.line(ratings_over_time,x='Date',y='Rating',title='Ratings over time',markers=True))


fig5=create_figure(px.bar(avg_rating_city,x='City',y='Rating',title='Average rating per city',color='City',labels={'Rating':'Average rating','City':'City'}))


st.title('Restaurant Feedback Dashboard')


col1,col2=st.columns(2)
with col1:
    st.markdown("## Average Ratings by restaurant")
    st.plotly_chart(fig1,use_container_width=True)
    st.markdown("""
    **Insight:**
    - Restaurants with higher average ratings may excel in service, food quality, or customer satisfaction.
    - Significant differences in ratings could indicate areas where lower-rated restaurants need improvement.
    """)
with col2:
    st.markdown("## Rating distribution by restaurant")
    st.plotly_chart(fig3,use_container_width=True)
    st.markdown("""
    **Insight:**
    - The pie chart shows the total number of ratings for each restaurant, indicating customer engagement levels.
    - Restaurants with fewer ratings may need to encourage more customer feedback to improve visibility.
    """)
st.markdown("## Average rating per city")
st.plotly_chart(fig5,use_container_width=True)
st.markdown("""
**Insight:**
- Cities with higher average ratings might represent regions with better restaurant standards or higher customer satisfaction.
- Cities with lower ratings may highlight the need for service improvements or new customer engagement strategies.
""")
with col1:
    st.markdown("## Number of reviews per restaurant")
    st.plotly_chart(fig2,use_container_width=True)
    st.markdown("""
    **Insight:**
    - Restaurants with more reviews are likely more popular or visible to customers.
    - Disparities in review counts suggest opportunities for marketing and outreach for less-reviewed restaurants.
    """)


with col2:
    st.markdown("## Rating over time")
    st.plotly_chart(fig4,use_container_width=True)
    st.markdown("""
    **Insight:**
    - Ratings over time reveal trends such as seasonal customer satisfaction or potential improvements in service.
    - Sudden drops in ratings may indicate service issues or negative customer experiences that need to be addressed.
    """)
st.markdown("## Final Conclusion")
st.markdown("""
**Overall Insights:**
- Restaurants with higher average ratings and reviews demonstrate strong customer satisfaction and consistent quality. These establishments should continue their successful practices while exploring opportunities for expansion.
- Cities with lower ratings or fewer reviews represent untapped opportunities for targeted improvements in service quality and marketing strategies.
- Seasonal variations and trends in ratings over time highlight the importance of maintaining consistent service during high-demand periods, such as weekends and holidays.
- Encouraging customers to provide feedback, especially for underrepresented restaurants, can help improve visibility and drive growth.

**Actionable Steps:**
- Focus on improving the service and visibility of lower-rated restaurants.
- Invest in targeted campaigns for cities with lower average ratings to boost customer engagement.
- Monitor seasonal trends and proactively prepare for high-demand periods to ensure customer satisfaction.
- Develop advertisements highlighting each restaurant's unique offerings, and include a variety of international cuisines to attract a diverse customer base.
""")
```