Project Report

Title Page

- **Project Name:** Uber Rides Data Analysis using Python
- **Group Members:**

  1. Adwait Sabale: Team Lead

  2. Tejas Paigude: Data Analyst and Visualization Specialist

  3. Kshitija Bhagwat: Data Analyst

  4. Divesh Bari: Documentation Specialist
- **Guide/Supervisor:** Ms. Nikita Pawar
- **Date of Submission:** 24/01/2025

1. Objective

The project analyses Uber ride data to extract insights into ride patterns, such as peak hours, popular pickup/drop-off locations, and trends over time. The results aim to improve operational efficiency, customer experience, and strategic planning.

2. Sources

1. Tools and Libraries

- **Pandas**: Reference: McKinney, W. (2010). *Data Structures for Statistical Computing in Python*. Proceedings of the 9th Python in Science Conference. Retrieved from https://pandas.pydata.org
- **Matplotlib**: Reference: Hunter, J. D. (2007). *Matplotlib: A 2D Graphics Environment*. *Computing in Science & Engineering*. Retrieved from https://matplotlib.org
- **Seaborn**: Reference: Waskom, M. L. (2021). *Seaborn: Statistical Data Visualization*. Journal of Open Source Software, 6(60), 3021. Retrieved from https://seaborn.pydata.org

2. Dataset

- **Uber Rides Dataset**:
  - File: UberDataset_with_payment.csv
  - Source: Kaggle. Retrieved from https://www.kaggle.com.

3. Methods and Operations

- **Data Cleaning and Processing**:
  - Parsing date-time fields (pd.to_datetime) for START_DATE and END_DATE.
  - Dropping rows with missing critical fields (dropna).
- **Feature Engineering**:
  - Extracting time-based features such as day names, hours, and months from timestamps.
  - Calculating trip durations by subtracting START_DATE from END_DATE.
- **Visualization Techniques**:
  - Bar plots to analyze ride demand by hours and days.
  - Grouped bar charts to study payment preferences across pickup zones.
  - Line charts to observe ride volumes over time.

4. Analysis and Findings

- Identification of peak demand hours and days of the week.
- Insights into top pickup and drop-off locations.
- Evaluation of payment preferences in different zones.
- Zones with the longest average trip durations.

5. Software

- **Jupyter Notebook**: Used for executing Python code interactively and documenting the workflow.
- **Python 3.11**: Primary programming language for data analysis and visualization tasks.

---

3. Steps Taken

1. Importing Required Libraries

- The notebook begins by importing essential Python libraries:
  - **pandas**: For data manipulation and analysis.
  - **matplotlib.pyplot**: For creating static visualizations.
  - **seaborn**: For advanced statistical plotting.

2. Loading the Dataset

- The dataset, UberDataset_with_payment.csv, is loaded into a Pandas DataFrame using pd.read_csv.

3. Data Preprocessing

- The notebook applies the following pre-processing steps:
    - Converts START_DATE and END_DATE columns to datetime objects for easier manipulation.
    - Removes rows with missing values in key columns (START, STOP, START_DATE, END_DATE).

4. Feature Engineering

- Extracts new features from the START_DATE column:
    - **Day of the Week**: To identify trends by day.
    - **Hour**: To analyze peak usage times.
    - **Month**: For seasonal trends.

5. Visualization of Peak Hours

- A bar chart is plotted to identify hours with the highest demand for rides.
- Seaborn's countplot is used with the Hour column.

6. Rides by Day of the Week

- Analyzes ride distribution across days using another bar chart.
- Specifies the order of days to align with the calendar week.

7. Identifying Top Locations

- Computes the top 10 pickup and drop-off locations by counting occurrences in START and STOP columns.

8. Analyzing Trip Duration

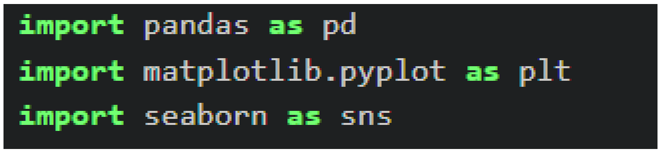
- Calculates trip duration in minutes by subtracting START_DATE from END_DATE.
- Aggregates trip duration by hour and day to find averages.

9. Payment Method Analysis

- If a payment column exists, explores the distribution of payment methods and preferences by zones.

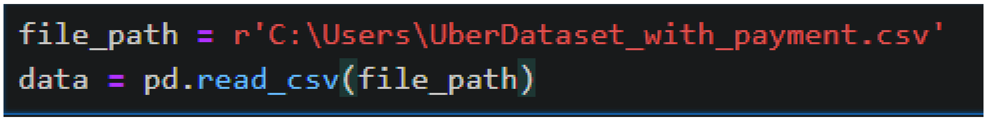
4. Code

1. Importing Required Libraries



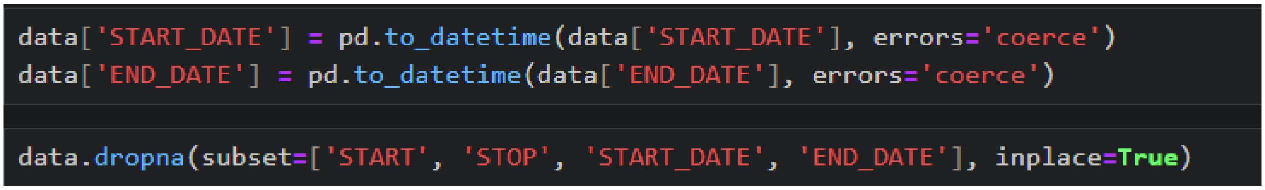
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

2. Loading the Dataset



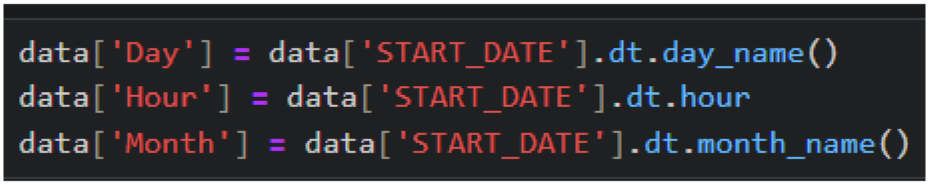
```
file_path = r'C:\Users\UberDataset_with_payment.csv'
data = pd.read_csv(file_path)
```

3. Data Pre-processing



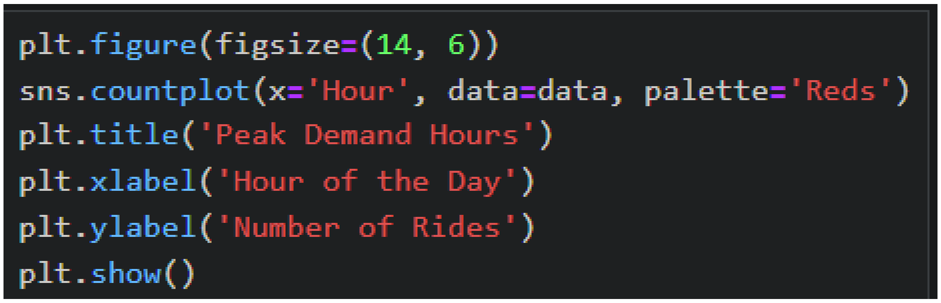
```
data['START_DATE'] = pd.to_datetime(data['START_DATE'], errors='coerce')
data['END_DATE'] = pd.to_datetime(data['END_DATE'], errors='coerce')

data.dropna(subset=['START', 'STOP', 'START_DATE', 'END_DATE'], inplace=True)
```

4. Feature Engineering

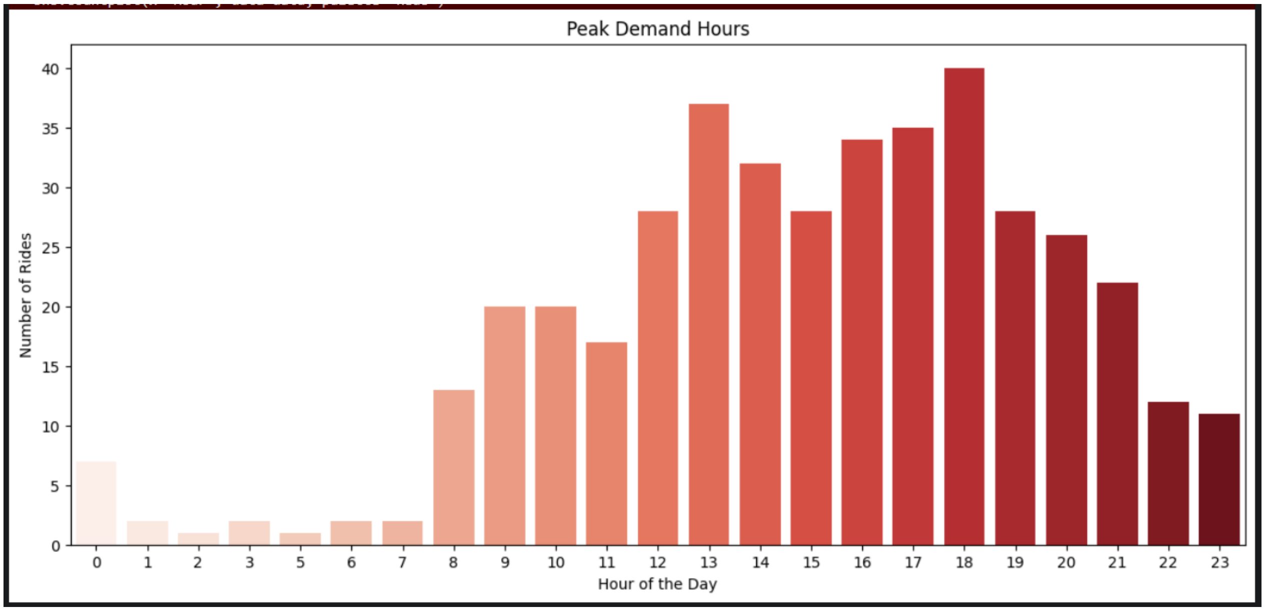


```
data['Day'] = data['START_DATE'].dt.day_name()
data['Hour'] = data['START_DATE'].dt.hour
data['Month'] = data['START_DATE'].dt.month_name()
```

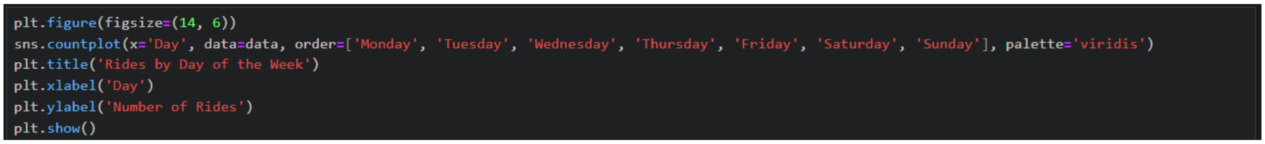
5. Visualization of Peak Hours



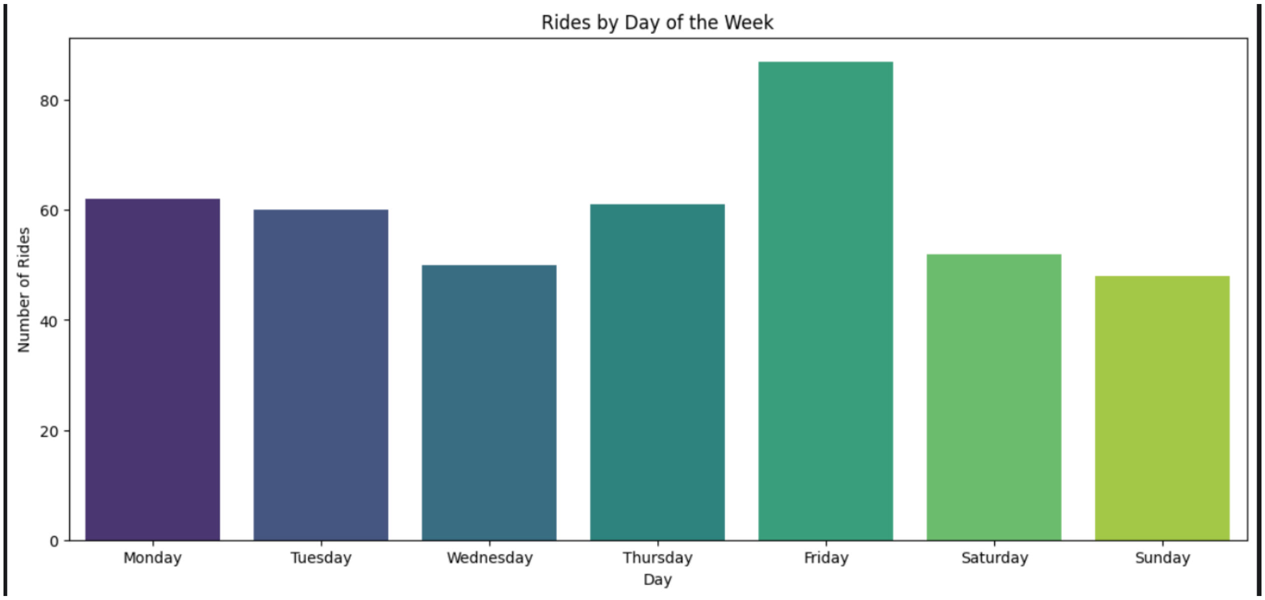
```
plt.figure(figsize=(14, 6))
sns.countplot(x='Hour', data=data, palette='Reds')
plt.title('Peak Demand Hours')
plt.xlabel('Hour of the Day')
plt.ylabel('Number of Rides')
plt.show()
```



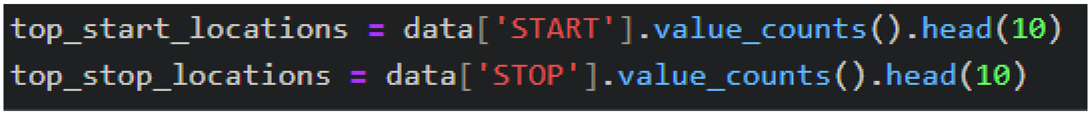
6. Rides by Day of the Week



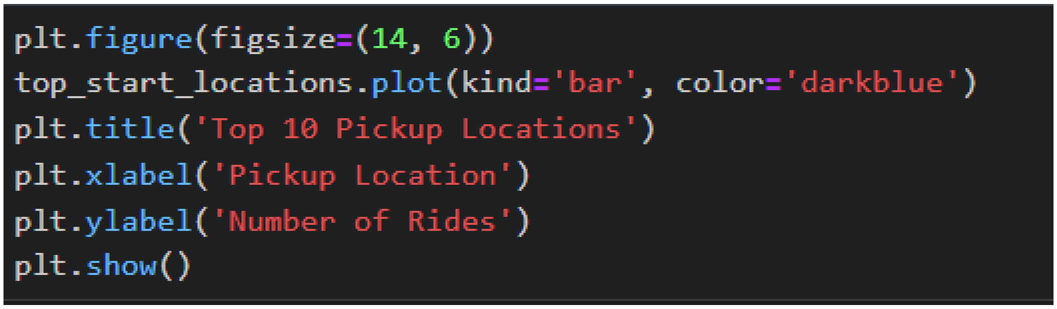
```
plt.figure(figsize=(14, 6))
sns.countplot(x='Day', data=data, order=['Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Saturday', 'Sunday'], palette='viridis')
plt.title('Rides by Day of the Week')
plt.xlabel('Day')
plt.ylabel('Number of Rides')
plt.show()
```



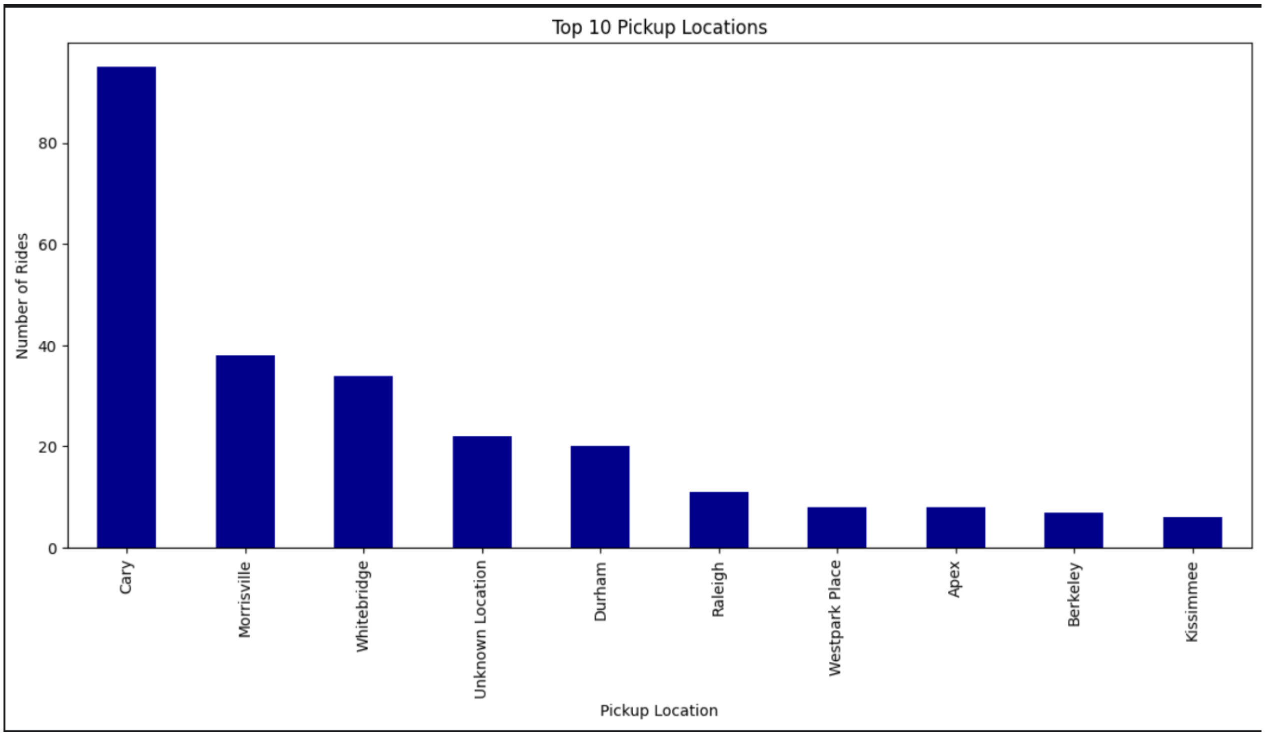
7. Identifying Top Locations

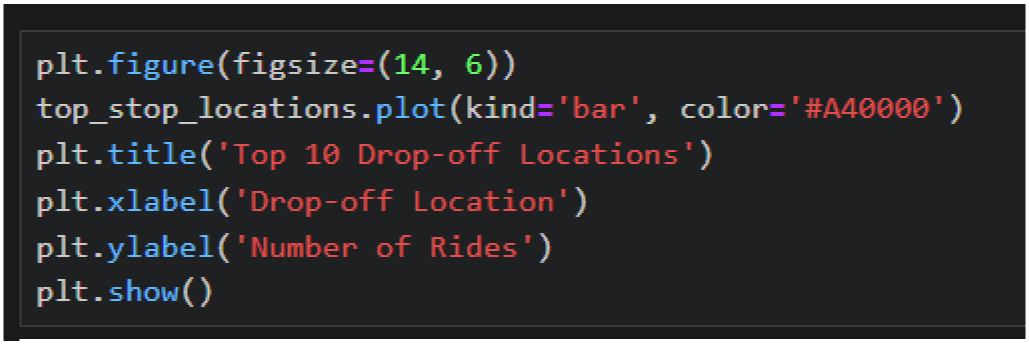


```
top_start_locations = data['START'].value_counts().head(10)
top_stop_locations = data['STOP'].value_counts().head(10)
```

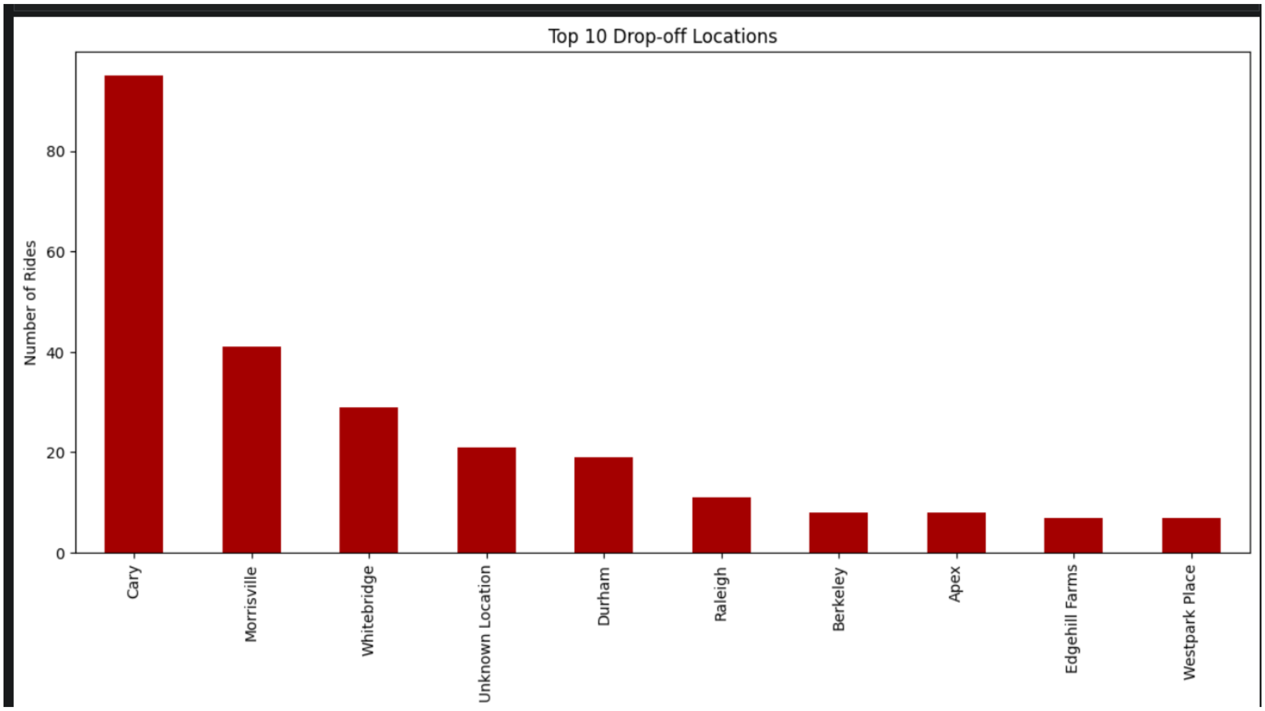


```
plt.figure(figsize=(14, 6))
top_start_locations.plot(kind='bar', color='darkblue')
plt.title('Top 10 Pickup Locations')
plt.xlabel('Pickup Location')
plt.ylabel('Number of Rides')
plt.show()
```

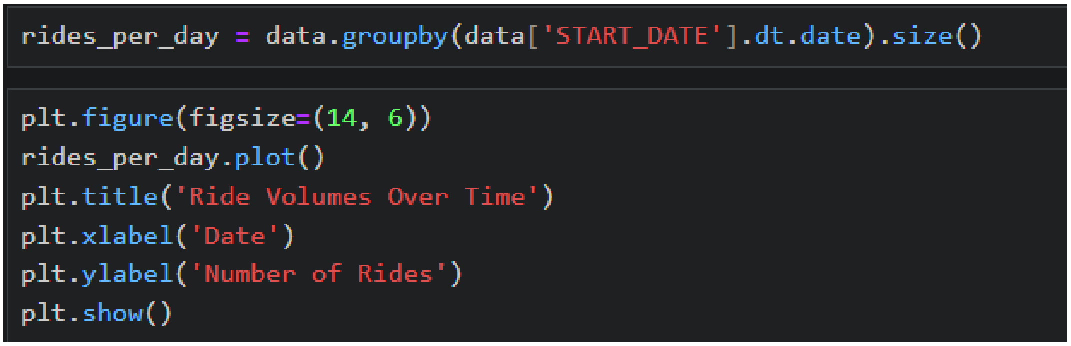




```
plt.figure(figsize=(14, 6))
top_stop_locations.plot(kind='bar', color='#A40000')
plt.title('Top 10 Drop-off Locations')
plt.xlabel('Drop-off Location')
plt.ylabel('Number of Rides')
plt.show()
```

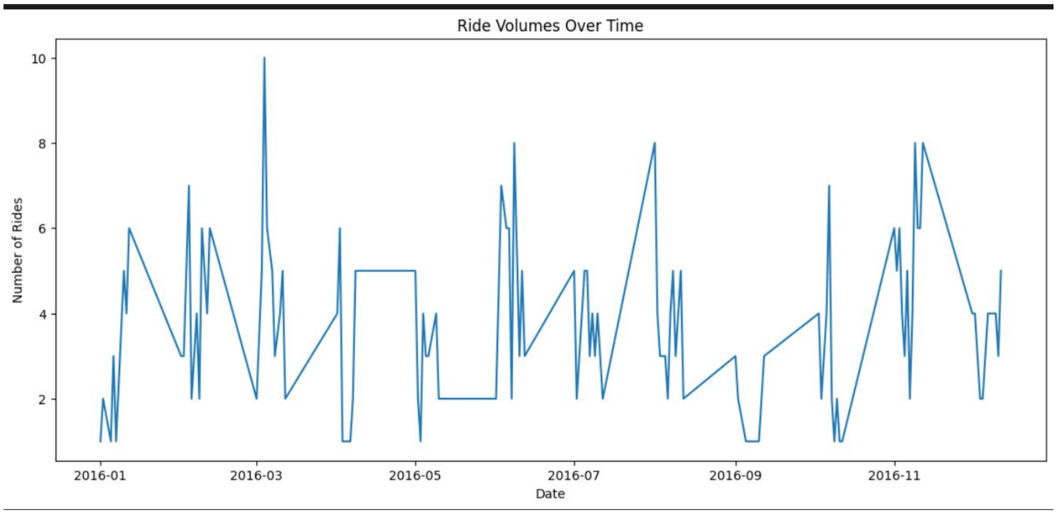


8. Rides per day

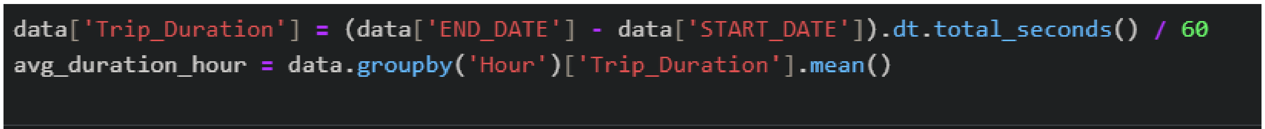


```
rides_per_day = data.groupby(data['START_DATE'].dt.date).size()
```

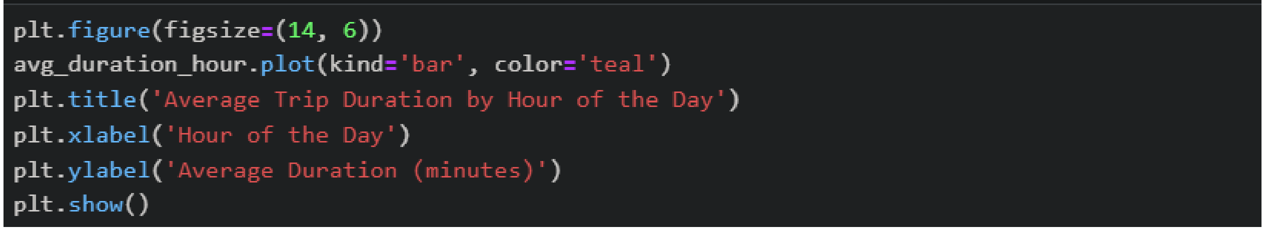
```
plt.figure(figsize=(14, 6))
rides_per_day.plot()
plt.title('Ride Volumes Over Time')
plt.xlabel('Date')
plt.ylabel('Number of Rides')
plt.show()
```



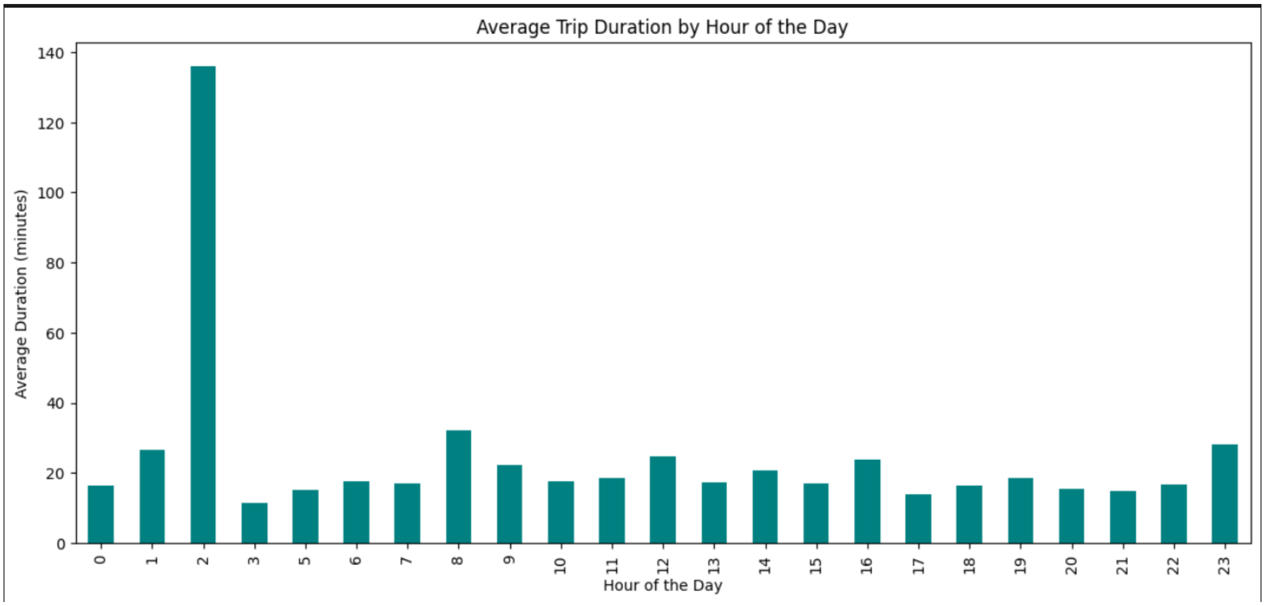
9. Analyzing Trip Duration

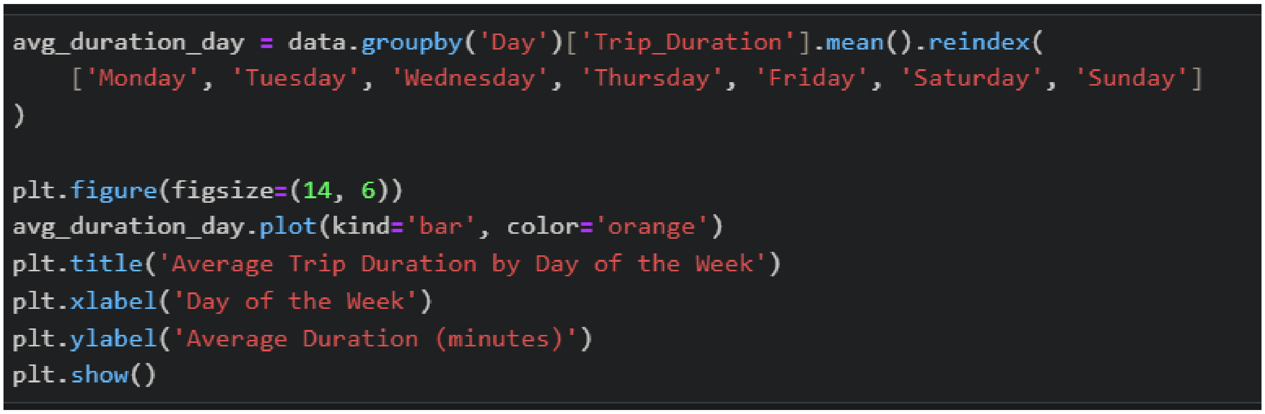


```
data['Trip_Duration'] = (data['END_DATE'] - data['START_DATE']).dt.total_seconds() / 60
avg_duration_hour = data.groupby('Hour')['Trip_Duration'].mean()
```

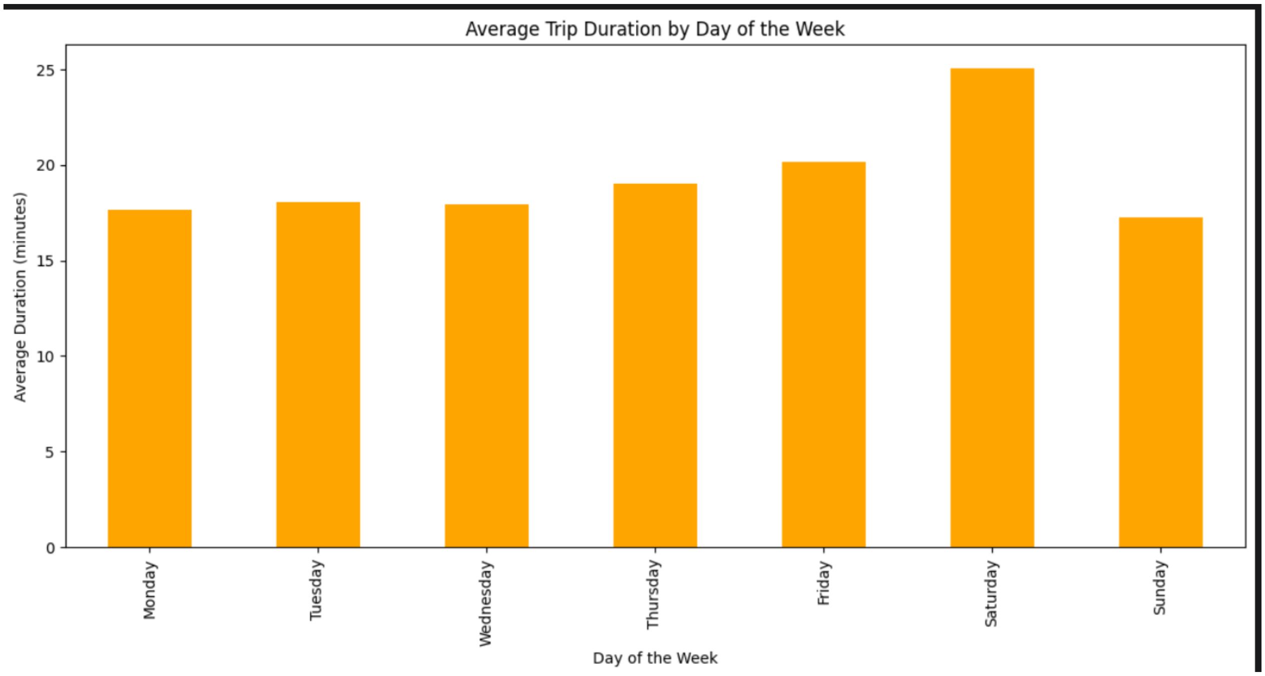


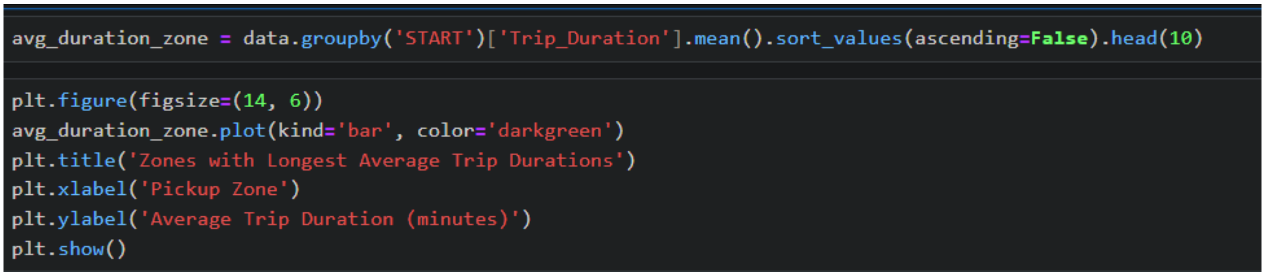
```
plt.figure(figsize=(14, 6))
avg_duration_hour.plot(kind='bar', color='teal')
plt.title('Average Trip Duration by Hour of the Day')
plt.xlabel('Hour of the Day')
plt.ylabel('Average Duration (minutes)')
plt.show()
```





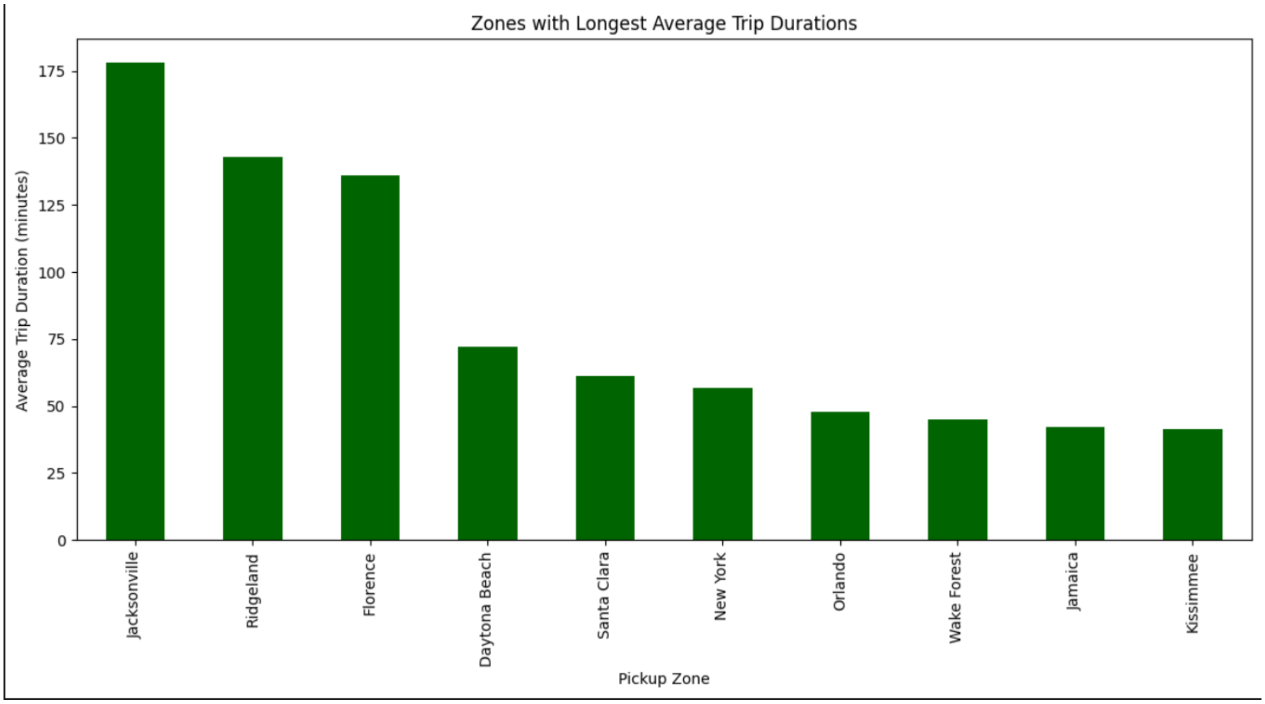
```
avg_duration_day = data.groupby('Day')['Trip_Duration'].mean().reindex(
    ['Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Saturday', 'Sunday']
)

plt.figure(figsize=(14, 6))
avg_duration_day.plot(kind='bar', color='orange')
plt.title('Average Trip Duration by Day of the Week')
plt.xlabel('Day of the Week')
plt.ylabel('Average Duration (minutes)')
plt.show()
```



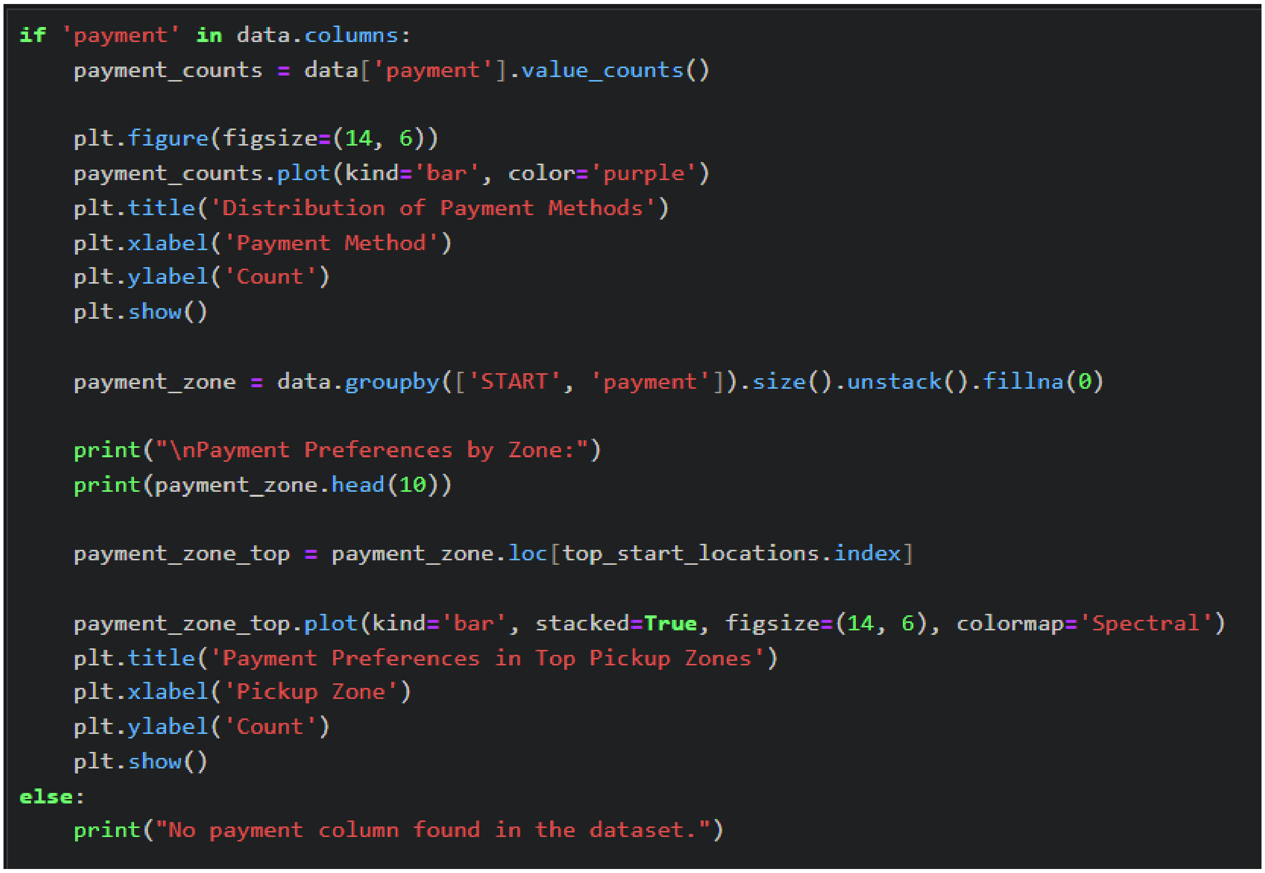


```
avg_duration_zone = data.groupby('START')['Trip_Duration'].mean().sort_values(ascending=False).head(10)
```

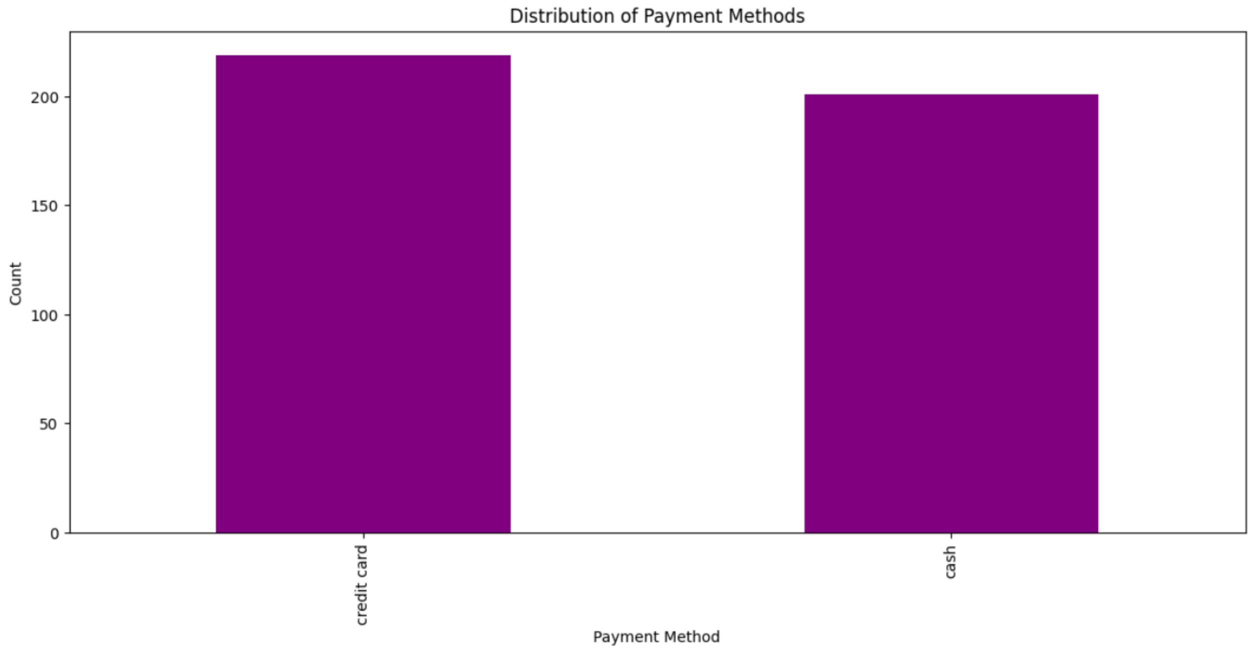
```
plt.figure(figsize=(14, 6))
avg_duration_zone.plot(kind='bar', color='darkgreen')
plt.title('Zones with Longest Average Trip Durations')
plt.xlabel('Pickup Zone')
plt.ylabel('Average Trip Duration (minutes)')
plt.show()
```

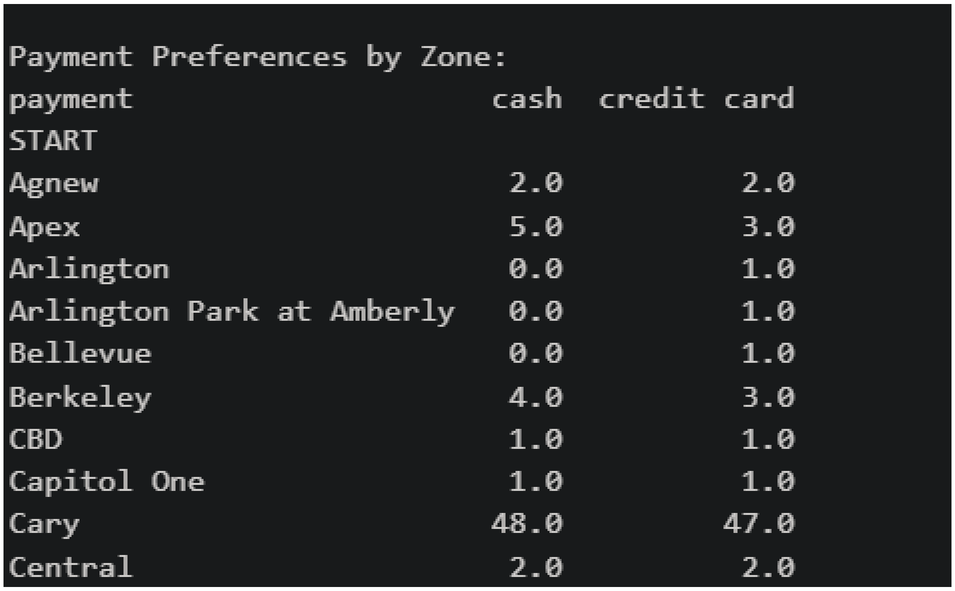


10. Payment Method Analysis

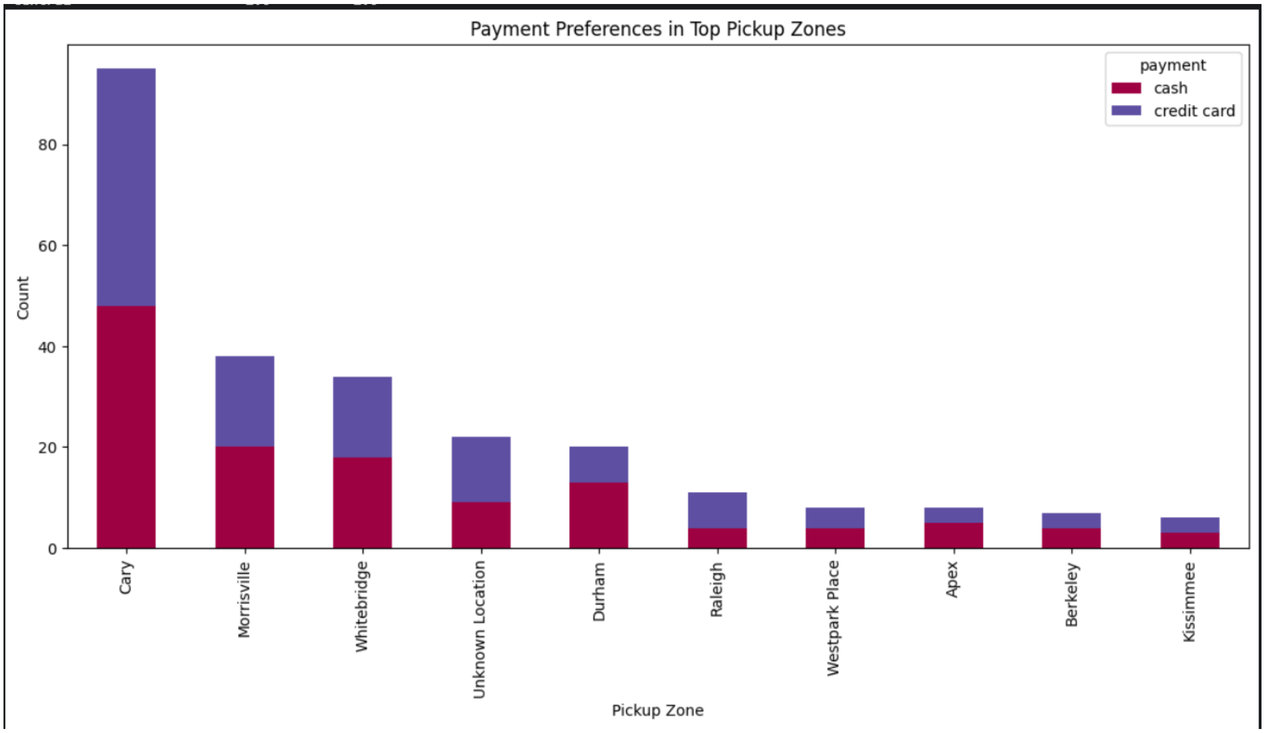


```python
if 'payment' in data.columns:
    payment_counts = data['payment'].value_counts()

    plt.figure(figsize=(14, 6))
    payment_counts.plot(kind='bar', color='purple')
    plt.title('Distribution of Payment Methods')
    plt.xlabel('Payment Method')
    plt.ylabel('Count')
    plt.show()

    payment_zone = data.groupby(['START', 'payment']).size().unstack().fillna(0)

    print("\nPayment Preferences by Zone:")
    print(payment_zone.head(10))

    payment_zone_top = payment_zone.loc[top_start_locations.index]

    payment_zone_top.plot(kind='bar', stacked=True, figsize=(14, 6), colormap='Spectral')
    plt.title('Payment Preferences in Top Pickup Zones')
    plt.xlabel('Pickup Zone')
    plt.ylabel('Count')
    plt.show()
else:
    print("No payment column found in the dataset.")
```





```
Payment Preferences by Zone:
payment                    cash  credit card
START
Agnew                       2.0          2.0
Apex                        5.0          3.0
Arlington                   0.0          1.0
Arlington Park at Amberly   0.0          1.0
Bellevue                    0.0          1.0
Berkeley                    4.0          3.0
CBD                         1.0          1.0
Capitol One                 1.0          1.0
Cary                       48.0         47.0
Central                     2.0          2.0
```



5. Results

Following the completion of the project, the results achieved were:

- **Peak Hours**: The busiest times for rides occur in the early mornings and late evenings, aligning with commute hours.
- **Popular Days**: Fridays and Saturdays show the highest number of rides, likely due to social and leisure activities.
- **Top Locations**: Specific zones are identified as the most frequent pickup and drop-off points, indicating key business or residential hubs.
- **Trip Duration**: Average trip durations are longest during off-peak hours, suggesting reduced traffic congestion.
- **Payment Methods**: Preferred payment methods vary across zones, reflecting regional user behavior and preferences.

6. Conclusion

The project successfully analyzed the Uber dataset to uncover critical insights about ride patterns, including peak demand times, popular locations, trip durations, and payment preferences. These findings provide valuable information for optimizing operations, enhancing user experiences, and strategizing resource allocation. By employing data preprocessing, feature engineering, and advanced visualization techniques, the project demonstrates the power of data-driven decision-making in the transportation sector.