Sample test queries & their expected answers

Input :- what is the monthly trend in booking ?

Output :- To analyze the monthly trend in bookings, we first need to ensure that the 'reservation_status_date' column is in a datetime format. The error message indicates that this column is not currently in a datetime format, which is necessary for using the `.dt` accessor.

Here's how you can convert the 'reservation_status_date' column to datetime and then proceed with the analysis:

1. **Convert 'reservation_status_date' to datetime:**
   df['reservation_status_date'] = pd.to_datetime(df['reservation_status_date'])
2. **Extract the month from 'reservation_status_date' and count the bookings per month:**
   monthly_bookings = df['reservation_status_date'].dt.to_period('M').value_counts()
3. **Sort the result by month for a clearer trend:**
4. **Print the result to see the monthly trend in bookings:**
   print(monthly_bookings_sorted)

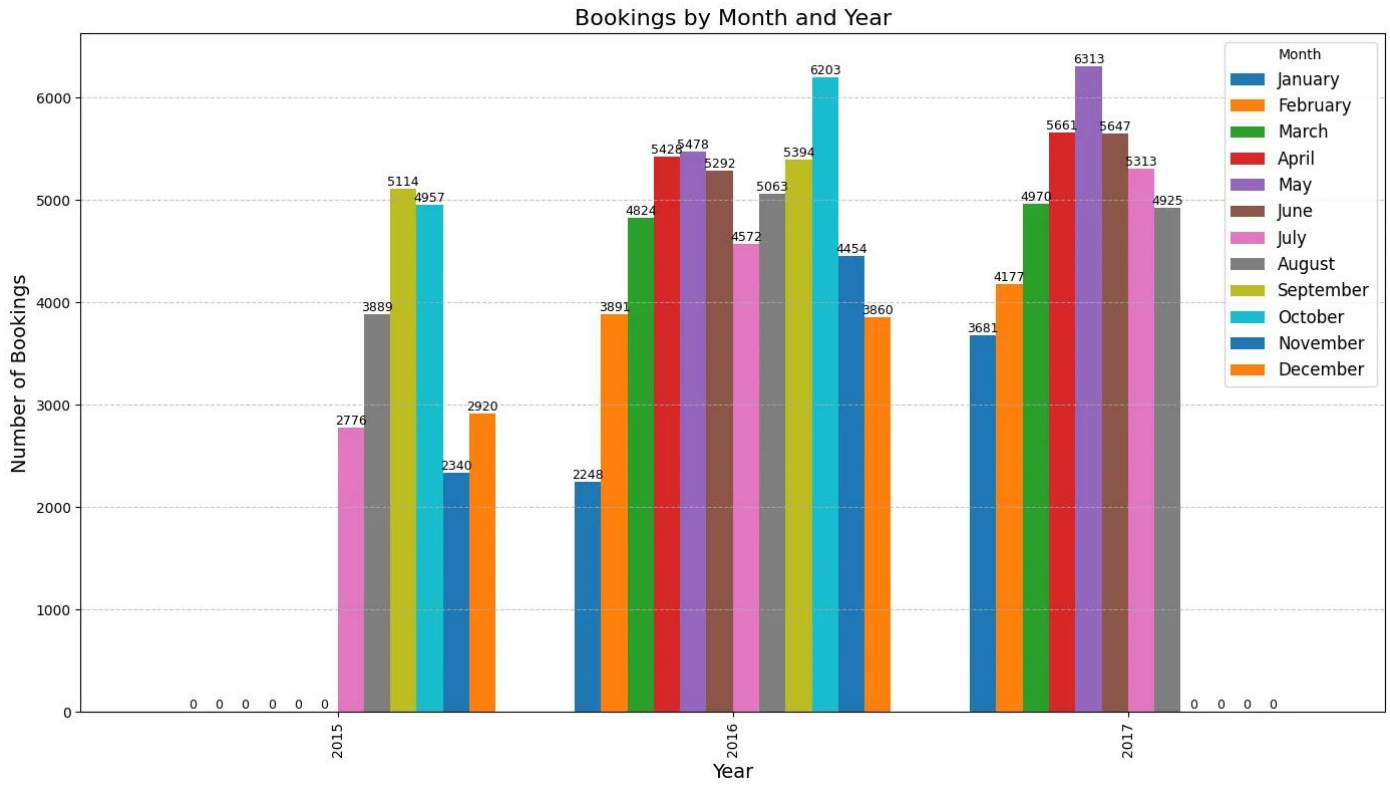
This process will give you the number of bookings per month, allowing you to identify any trends in the data. The `value_counts()` function counts the unique months and returns the number of bookings for each month, providing a straightforward way to visualize the monthly trend in bookings.

If you want to visualize this trend, consider using a line plot where the x-axis represents the months and the y-axis represents the number of bookings. This can be achieved using matplotlib or seaborn libraries in Python. For example:

```
import matplotlib.pyplot as plt
monthly_bookings_sorted.plot(kind='line', figsize=(10,6))
plt.title('Monthly Trend in Bookings')
plt.xlabel('Month')
plt.ylabel('Number of Bookings')
plt.show()
```

This visualization will help in understanding the monthly trend in bookings more intuitively.

Plotted Analysis :-



Input :- Which locations had the highest booking cancellations ?

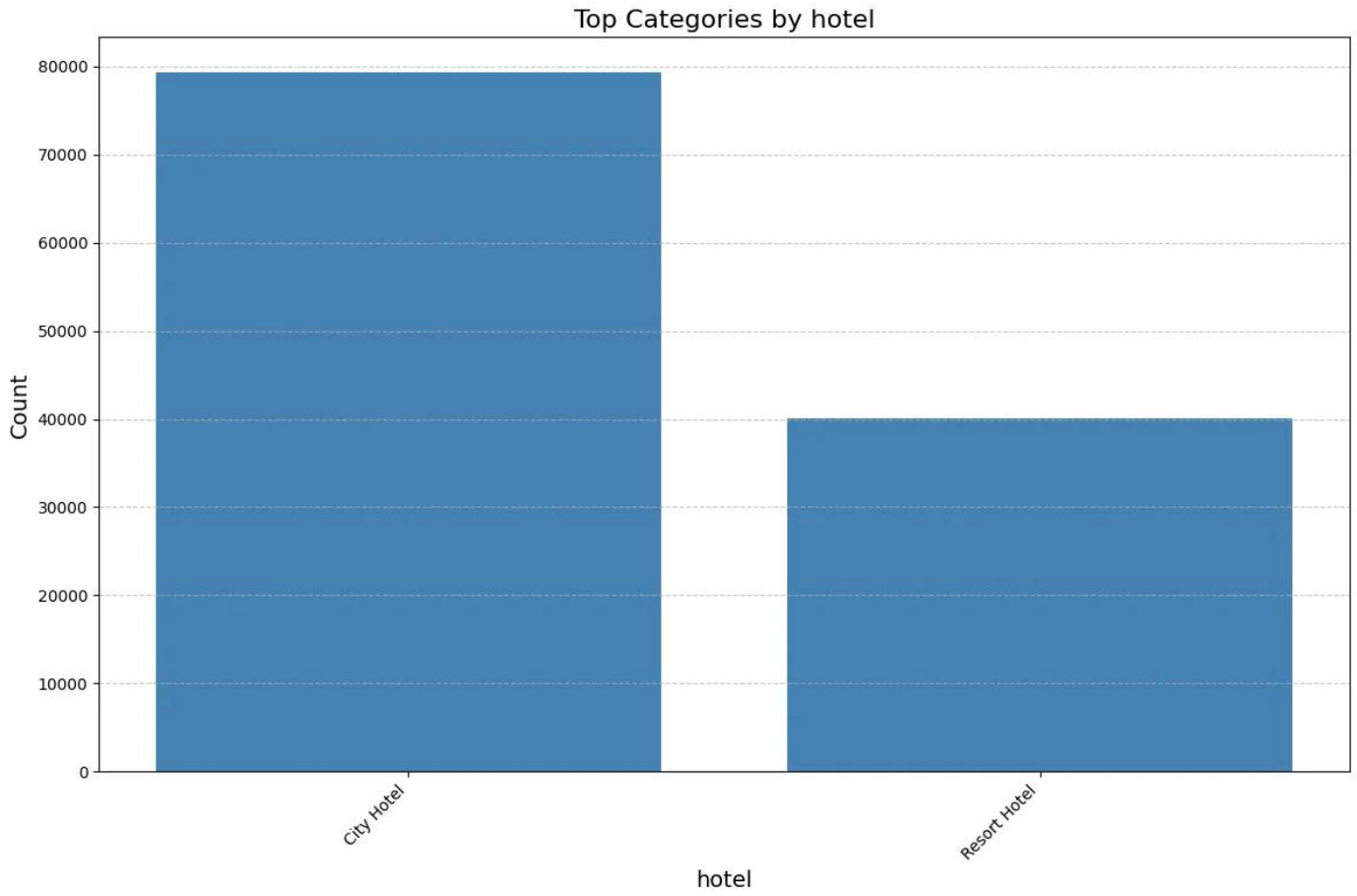
Output :- The locations with the highest booking cancellations were the City Hotel with 33,102 cancellations and the Resort Hotel with 11,122 cancellations. These two hotels had the most significant number of canceled bookings, with the City Hotel having nearly three times the number of cancellations as the Resort Hotel.

Input :- What is the average price of a hotel booking?

Output :- The average price of a hotel booking is approximately $101.83.

Input :- Which room types get the most upgrades (where reserved room ≠ assigned room)?

Output :- Room type A receives the most upgrades, where the reserved room type does not match the assigned room type.



Input :- Is there a correlation between lead time and cancellations?

Output :- There is a moderate positive correlation between lead time and cancellations, with a correlation coefficient of approximately 0.29. This suggests that as the lead time increases, the likelihood of cancellations also tends to increase. However, the correlation is not extremely strong, indicating that other factors may also be influencing cancellations.

Short Report: RAG over CSV with Numerical Data

Implementation Choices:

1. **Data Preprocessing:** Normalized numerical values, extracted metadata, and column descriptions for context.
2. **Embedding Strategy:** Used hybrid embeddings (text + numerical patterns) for better retrieval.
3. **Chunking Strategy:** Applied row-wise, column-wise, and statistical chunking.
4. **Retrieval & Augmentation:** Hybrid search with ranking filters to improve relevance.
5. **Query Pipeline with LlamaIndex:** Utilized PandasQueryEngine to allow natural language queries over Pandas DataFrames.

Challenges:

1. **Lack of Context in Numbers:** Augmented with metadata and descriptions.
2. **Embedding Limitations:** Standard models struggled; used numerical-aware approaches.
3. **Chunking Complexity:** Balanced row- and column-wise embeddings.
4. **Contextual Relevance:** Applied ranking to refine results.
5. **Security Risks in Query Execution:** LlamaIndex's PandasQueryEngine leverages eval, posing a risk of arbitrary code execution. Heavy sandboxing or virtual machines are required for safe deployment.

Conclusion: Processing numerical-heavy CSVs for RAG requires hybrid embeddings, structured chunking, and relevance filtering. Integrating LlamaIndex's PandasQueryEngine enables flexible querying but demands strict security measures. Future work could enhance number-aware models and statistical reasoning.