Flight Delays-keeping travelers on time

Milestone 3.2

By Orpheas, Marco and Adam

The Problem we are trying to solve

- Through this project we aim to solve a problem faced by thousands per day – FLIGHT DELAYS
- This problem has affected all three of us and most definitely you to, therefore we would like to find a possible solution to aid in this issue
- Flight delays may seem minor at first but if we look into a problem as such, the severity can be clearly seen
- Flight delays lead to:

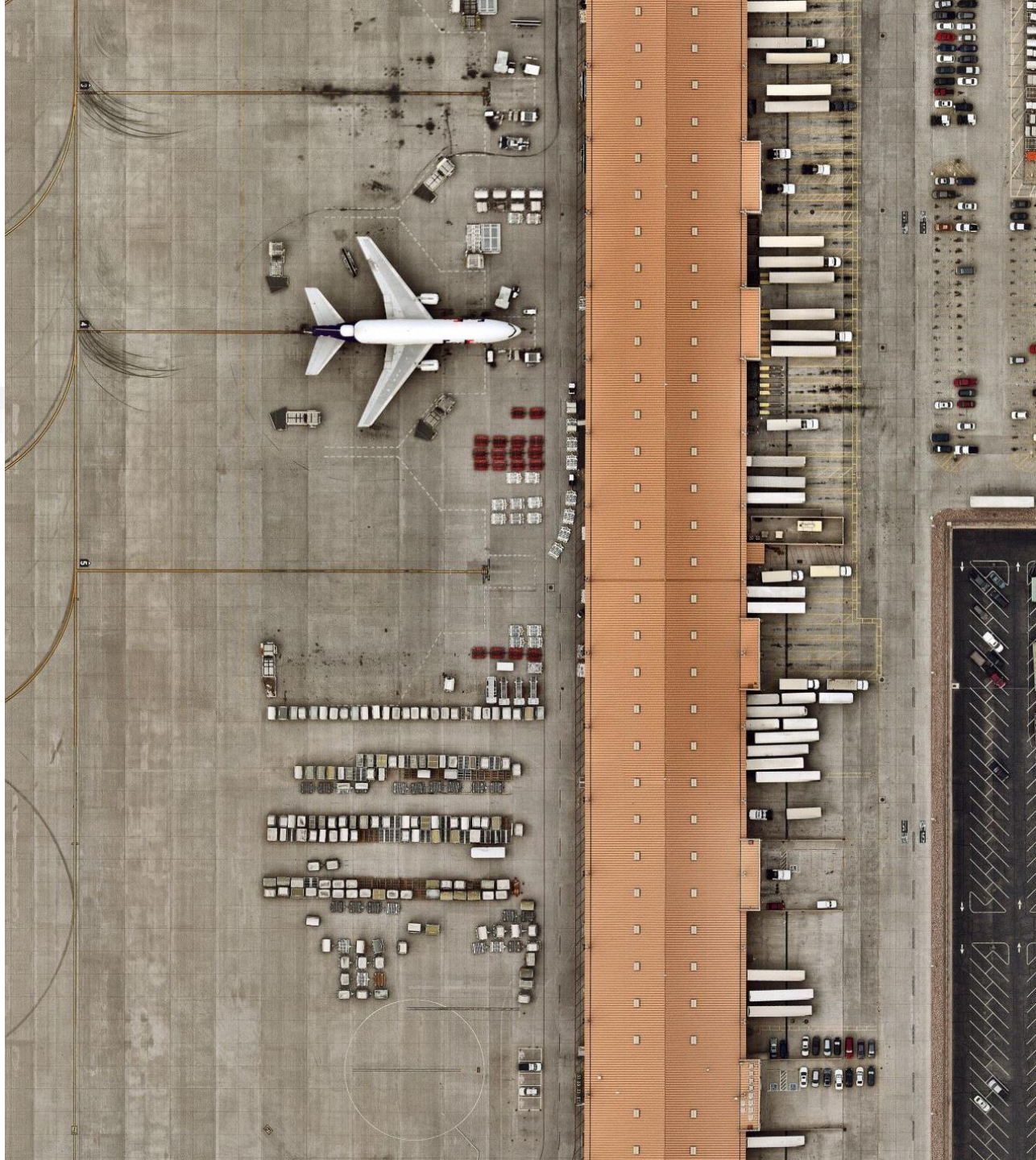
1. Passenger frustration – affect schedules (connecting flights, plans etc.) and causes inconvenience
2. Economic impact – cost to airlines, passengers and businesses
3. Uncertainty in travels plans - ruins a schedule, makes it difficult to plan

The Problem we are trying to solve

- We hope to find any correlations through the data from the causes which will be able to give us answers and solutions to help aid in the minimization of global delays in airports and the cost that comes with it.
- Our aim is to analyze historical flight data to identify patterns and predict the likelihood of a flight being delayed as well as the reasons for this delay.

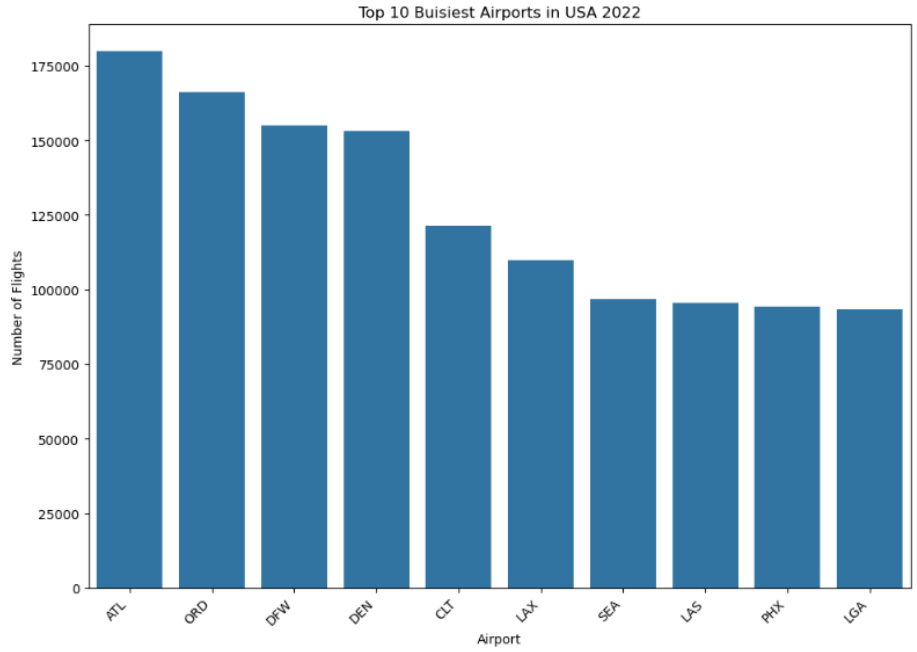
Our Data – Key Plots- What We Learnt

- After importing, reading the data and checking for any missing values and duplicates we created bar plots of the busiest and least busy airports through their airport codes (“Origin”) and the number of flights that they had in the specific year of the data.
- We made this plot to get a visualization of the airports with the highest activity which will come in handy later

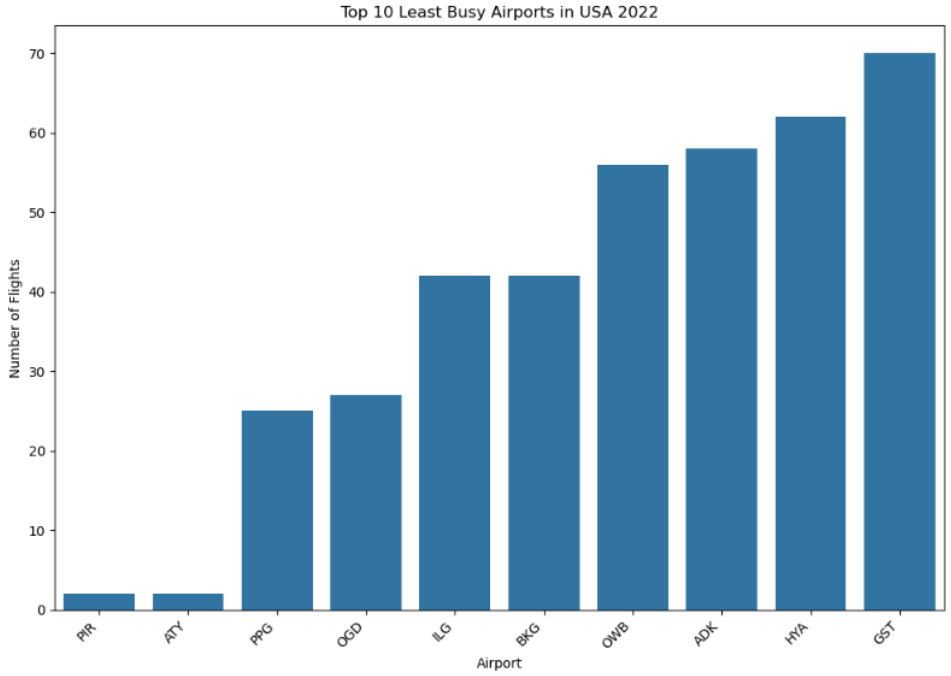


```
Name: count, dtype: int64
```

[28]:
```
# Bar plot to visualise top 10 buisiest airports
plt.figure(figsize=(10,7))
sns.barplot(x=buisiest_airports.index, y=buisiest_airports.values)
plt.title('Top 10 Buisiest Airports in USA 2022')
plt.xlabel('Airport')
plt.ylabel('Number of Flights')
plt.tight_layout()
plt.xticks(rotation=45, ha='right')
plt.show()
```



[30]:
```
# Bar plot to visualise top 10 least busy airports
plt.figure(figsize=(10,7))
sns.barplot(x=least_busy_airports.index, y=least_busy_airports.values)
plt.title('Top 10 Least Busy Airports in USA 2022')
plt.xlabel('Airport')
plt.ylabel('Number of Flights')
plt.tight_layout()
plt.xticks(rotation=45, ha='right')
plt.show()
```

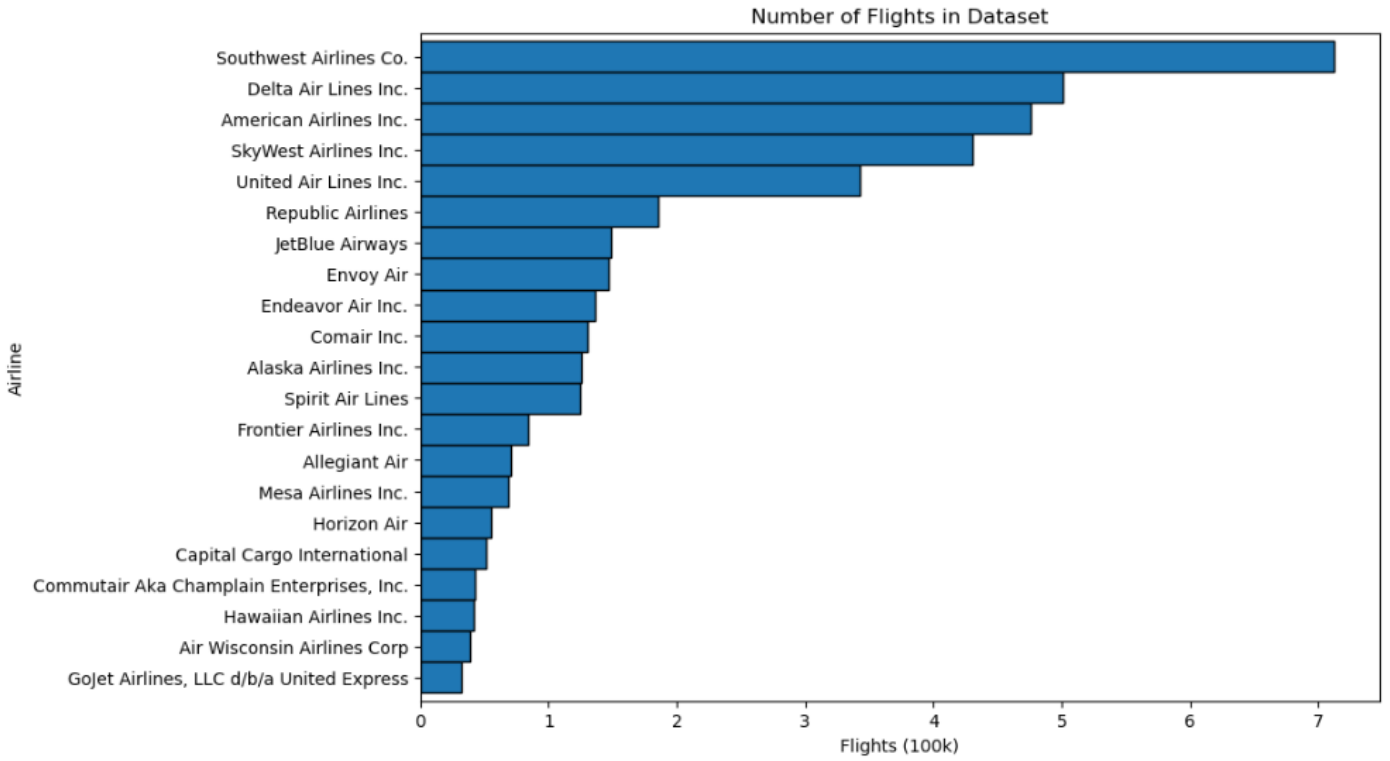


Our Data – Key Plots-What We Learnt

- We also made of plot with the Airline and the number of flights they had which we could use later to see if there is any correlation with Airline and other factors
- Southwest had to most flights

```
[47]: # which airline has the most flights
fig, ax = plt.subplots(figsize=(10, 7))
airlines_ordered = (df["Airline"].value_counts(ascending=True) / 100_000).plot(kind="barh", ax=ax, width=1, edgecolor="black")
ax.set_title("Number of Flights in Dataset")
ax.set_xlabel("Flights (100k)")

[47]: Text(0.5, 0, 'Flights (100k)')
```

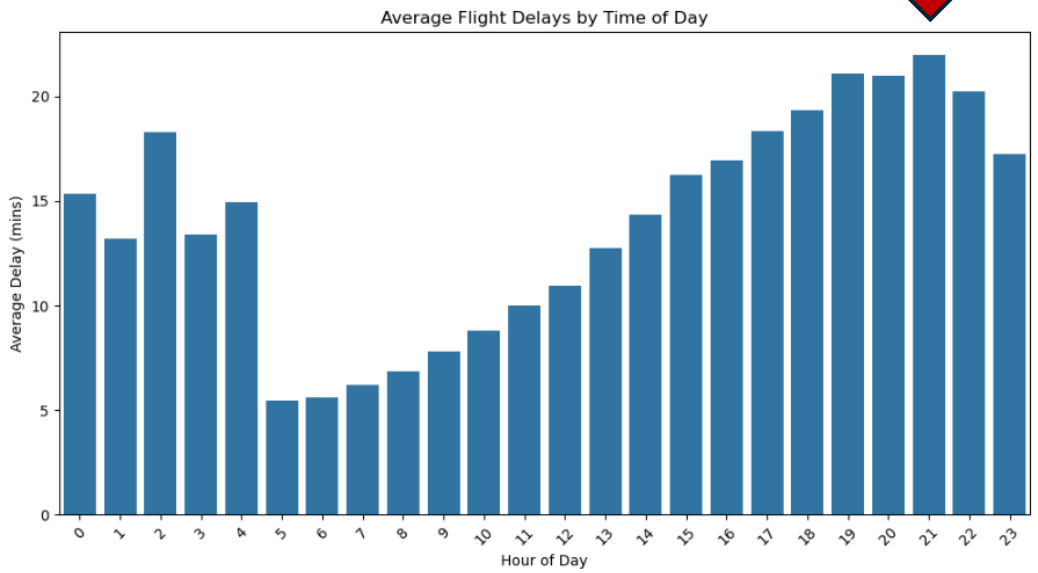


Our Data – Key Plots- What we Learnt

- We made a bar plot to visualize the distribution of Average Flight Delays by time of day and saw that the times with the longest average flight delays where between 7pm-10pm with the longest being at 9pm.
- Therefore, we concluded that at night is where the flight delays are longest.
- We also saw that from around 5am the average time of delays gradually starts rising until the night.

```
[31]: # Convert CRSDepTime to hours
      df['DepHour'] = df['CRSDepTime'] // 100

[32]: # Calculate avergage delay by hour
      average_delay_by_hour = df.groupby('DepHour')['DepDelay'].mean().reset_index()
      # Bar plot to visualize average flight delays by Time of Day
      plt.figure(figsize=(12, 6))
      sns.barplot(data=average_delay_by_hour, x='DepHour', y='DepDelay')
      plt.title('Average Flight Delays by Time of Day')
      plt.xlabel('Hour of Day')
      plt.ylabel('Average Delay (mins)')
      plt.xticks(rotation=45)
      plt.show()
```



Our Data – Key Plots – What we Learnt

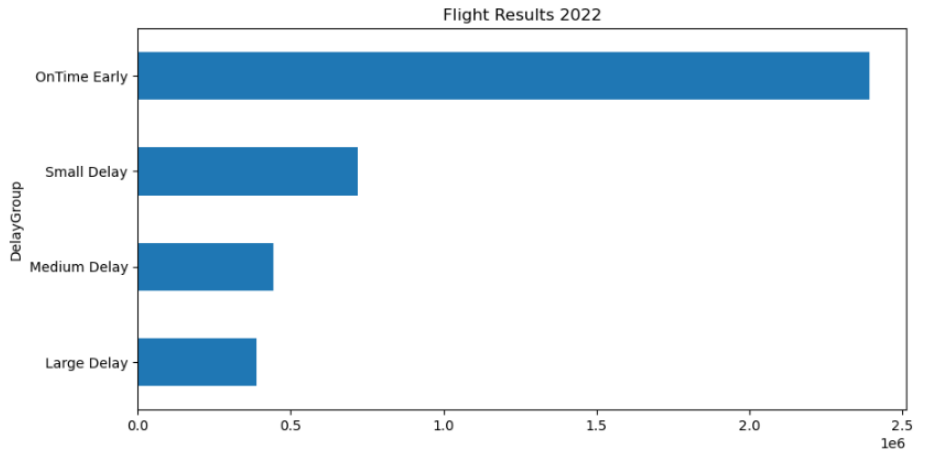
- We split the “DepDelayMinutes” column in 4 sub columns ranging with magnitude of time delay :

1. OnTime Early – No delay
2. Small Delay – Delay <= 15 mins
3. Medium Delay – 15mins< Delay <= 45mins
4. Large Delay – Delay > 45mins

- The plot showed that most flights have no delay but from the flights that have a delay “Small Delay” was the most common although with not a significant difference from the other two indicating that the probaility of each type of delay happening is similar – A great problem!

[70]:
```
# Grouping of delays
df["DelayGroup"] = None
df.loc[df["DepDelayMinutes"] == 0, "DelayGroup"] = "OnTime Early"
df.loc[(df["DepDelayMinutes"] > 0) & (df["DepDelayMinutes"] <= 15), "DelayGroup"] = "Small Delay"
df.loc[(df["DepDelayMinutes"] > 15) & (df["DepDelayMinutes"] <= 45), "DelayGroup"] = "Medium Delay"
df.loc[df["DepDelayMinutes"] > 45, "DelayGroup"] = "Large Delay"
df.loc[df["Cancelled"], "DelayGroup"] = "Cancelled"
```

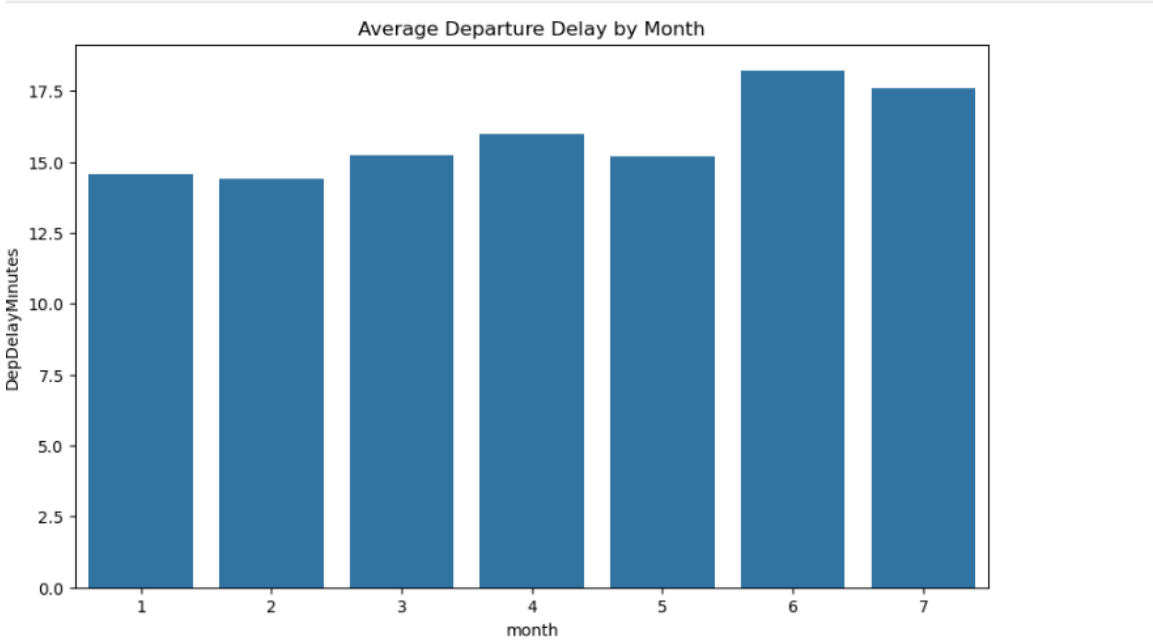
[72]:
```
# Plot to visualise type of delay
df["DelayGroup"].value_counts(ascending=True).plot(kind="barh", figsize=(10, 5), title="Flight Results 2022")
plt.show()
```



[38]:
```
# most flights are on time
# if there is a delay it is most likelt a small delay <= 15 mins
# however tendancy of medium or large delay are on similar frequencies
```

Our Data – Key Plots – What we Learnt

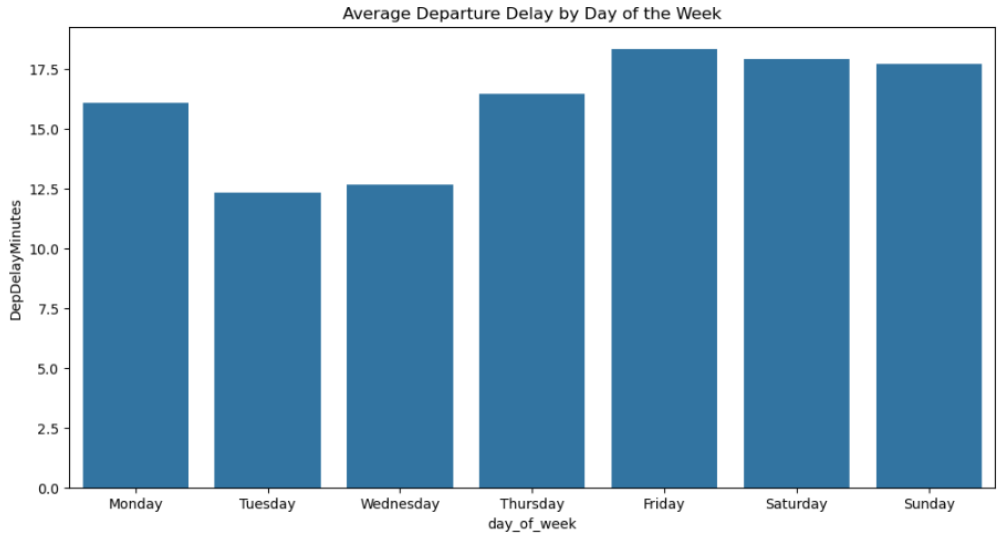
- Bar plot for average departure delay by Month
- This plot was unsuccessful as through the plot we realised that our data didn’t have flights on all the months of the year.
- However, it somewhat showed that the month doesn’t affect delays by a significant effect



Our Data – Key Plots – What we Learnt

- Since the months was unsuccessful, we split the data in days of the week.
- The bar plot of showed that the weekend and Fridays had the longest delays on average we seems logical.
- Friday had the longest delays and Tuesday the shortest.
- Delay time rises from Tuesday until it reaches Friday and then starts to decrease again.

```
[45]: # Plot to visualise delays by day of the week
plt.figure(figsize=(12, 6))
sns.barplot(x='day_of_week', y='DepDelayMinutes', data=df,
            order=['Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Saturday', 'Sunday'], errorbar=None)
plt.title('Average Departure Delay by Day of the Week')
plt.show()
```

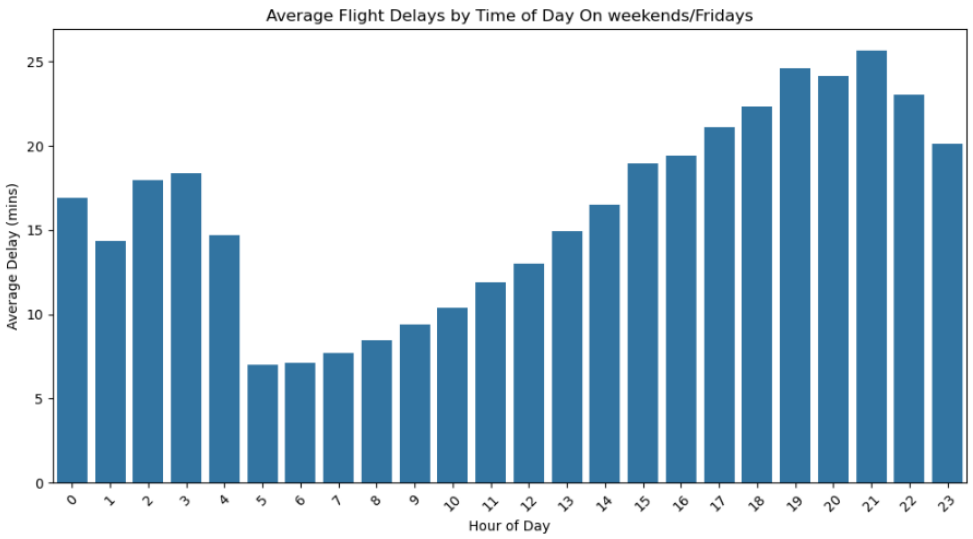


Our Data – Key Plots – What we Learnt

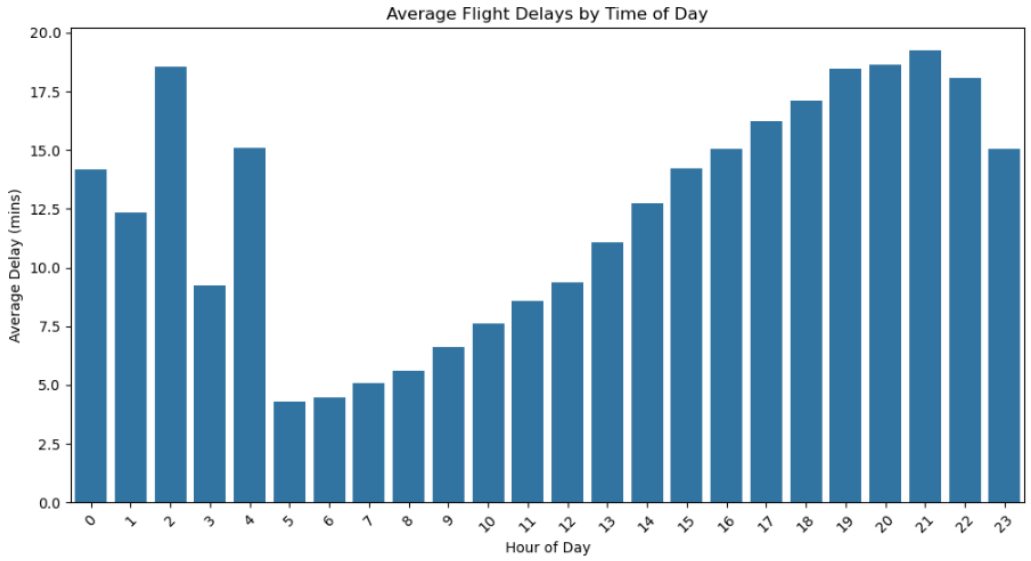
- Since weekends and Fridays had the greatest time delays we decided to sort our data into the three specified days to further examine the data and come to conclusions
- We split our data in “Weekend” – Fri, Sat, Sun and “Not Weekend”- All other days
- We then made a similar bar plot like before of time of day and average delay to compare and see any other differences

“Weekend” vs “Not Weekend”

Weekend



Not Weekend



Overall shape of two Bar charts looks the same however for weekend and Friday each time delay for the hour is of greater magnitude which was expected. Therefore, the day doesn’t affect the hour of delay that much