In [1]:

```python
# Import necessary libraries
import pandas as pd

# Load the dataset
file_path = 'Bird Strikes.csv'
bird_strikes_df = pd.read_csv(file_path)

# Display the first few rows of the dataset to inspect its structure
bird_strikes_df.head(), bird_strikes_df.columns
```

```
C:\Users\vivek\anaconda3\lib\site-packages\pandas\core\computation\expressions.py:21: UserWarning: Pandas requires version '2.8.4' or newer of 'numexpr' (version '2.7.3' currently installed).
  from pandas.core.computation.check import NUMEXPR_INSTALLED
C:\Users\vivek\anaconda3\lib\site-packages\pandas\core\arrays\masked.py:60: UserWarning: Pandas requires version '1.3.6' or newer of 'bottleneck' (version '1.3.2' currently installed).
  from pandas.core import (
```

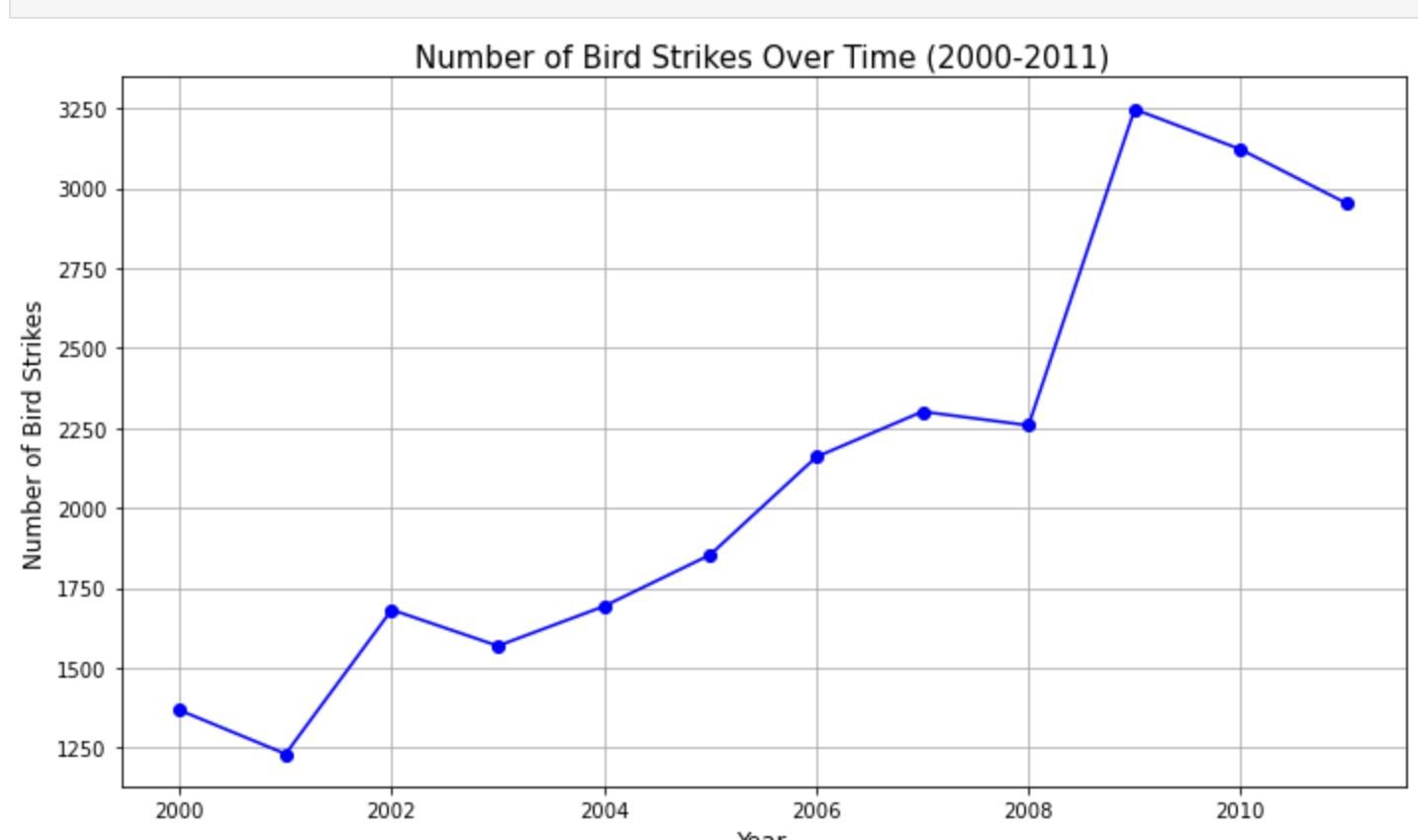
Out[1]:

```
(   Record ID Aircraft: Type                Airport: Name Altitude bin  \
 0     202152       Airplane                 LAGUARDIA NY    > 1000 ft
 1     208159       Airplane  DALLAS/FORT WORTH INTL ARPT    < 1000 ft
 2     207601       Airplane            LAKEFRONT AIRPORT    < 1000 ft
 3     215953       Airplane          SEATTLE-TACOMA INTL    < 1000 ft
 4     219878       Airplane                 NORFOLK INTL    < 1000 ft

   Aircraft: Make/Model Wildlife: Number struck  \
 0            B-737-400                Over 100
 1                MD-80                Over 100
 2                C-500                Over 100
 3            B-737-400                Over 100
 4         CL-RJ100/200                Over 100

    Wildlife: Number Struck Actual Effect: Impact to flight     FlightDate  \
 0                             859         Engine Shut Down  11/23/00 0:00
 1                             424                      NaN   7/25/01 0:00
 2                             261                      NaN   9/14/01 0:00
 3                             806    Precautionary Landing    9/5/02 0:00
 4                             942                      NaN   6/23/03 0:00

   Effect: Indicated Damage  ... Remains of wildlife sent to Smithsonian  \
 0            Caused damage  ...                                   False
 1            Caused damage  ...                                   False
 2                No damage  ...                                   False
 3                No damage  ...                                   False
 4                No damage  ...                                   False

                                          Remarks Wildlife: Size  \
 0  FLT 753. PILOT REPTD A HUNDRED BIRDS ON UNKN T...         Medium
 1  102 CARCASSES FOUND. 1 LDG LIGHT ON NOSE GEAR ...          Small
 2  FLEW UNDER A VERY LARGE FLOCK OF BIRDS OVER AP...          Small
 3  NOTAM WARNING. 26 BIRDS HIT THE A/C, FORCING A...          Small
 4                                    NO DMG REPTD.          Small

   Conditions: Sky      Wildlife: Species  Pilot warned of birds or wildlife?  \
 0        No Cloud  Unknown bird - medium                                   N
 1      Some Cloud            Rock pigeon                                   Y
 2        No Cloud      European starling                                   N
 3      Some Cloud      European starling                                   Y
 4        No Cloud      European starling                                   N

    Cost: Total $ Feet above ground Number of people injured Is Aircraft Large?
 0         30,736             1,500                        0                Yes
 1              0                 0                        0                 No
 2              0                50                        0                 No
 3              0                50                        0                Yes
 4              0                50                        0                 No

 [5 rows x 26 columns],
 Index(['Record ID', 'Aircraft: Type', 'Airport: Name', 'Altitude bin',
        'Aircraft: Make/Model', 'Wildlife: Number struck',
        'Wildlife: Number Struck Actual', 'Effect: Impact to flight',
        'FlightDate', 'Effect: Indicated Damage',
        'Aircraft: Number of engines?', 'Aircraft: Airline/Operator',
        'Origin State', 'When: Phase of flight', 'Conditions: Precipitation',
        'Remains of wildlife collected?',
        'Remains of wildlife sent to Smithsonian', 'Remarks', 'Wildlife: Size',
        'Conditions: Sky', 'Wildlife: Species',
        'Pilot warned of birds or wildlife?', 'Cost: Total $',
        'Feet above ground', 'Number of people injured', 'Is Aircraft Large?'],
       dtype='object'))
```

Case studies

In [4]:

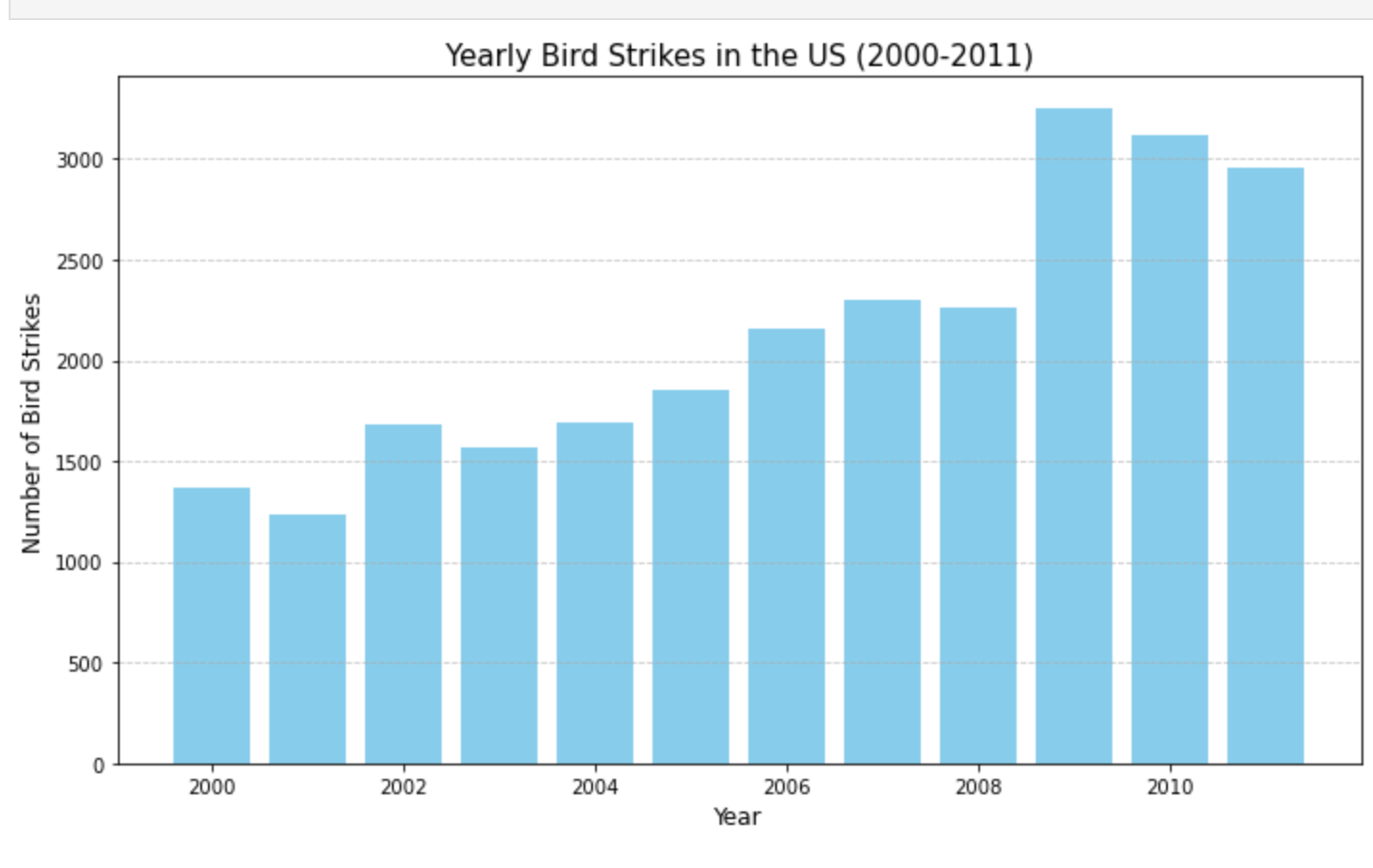
```python
import matplotlib.pyplot as plt
import numpy as np

# Check for any NaN or invalid values in 'Year' and 'Number of Bird Strikes'
bird_strikes_per_year.dropna(subset=['Year', 'Number of Bird Strikes'], inplace=True)

# Convert to integer and numeric types, if necessary
bird_strikes_per_year['Year'] = pd.to_numeric(bird_strikes_per_year['Year'], errors='coerce').astype(int)
bird_strikes_per_year['Number of Bird Strikes'] = pd.to_numeric(bird_strikes_per_year['Number of Bird Strikes'], errors='coerce')

# Ensure the columns are numpy arrays to avoid multi-dimensional indexing issues
years = bird_strikes_per_year['Year'].to_numpy()
strikes = bird_strikes_per_year['Number of Bird Strikes'].to_numpy()

# Plot the data
plt.figure(figsize=(10, 6))
plt.plot(years, strikes, marker='o', color='b')
plt.title('Number of Bird Strikes Over Time (2000-2011)', fontsize=15)
plt.xlabel('Year', fontsize=12)
plt.ylabel('Number of Bird Strikes', fontsize=12)
plt.grid(True)
plt.tight_layout()
plt.show()
```



1. Visuals Depicting the Number of Bird Strikes

In [5]:

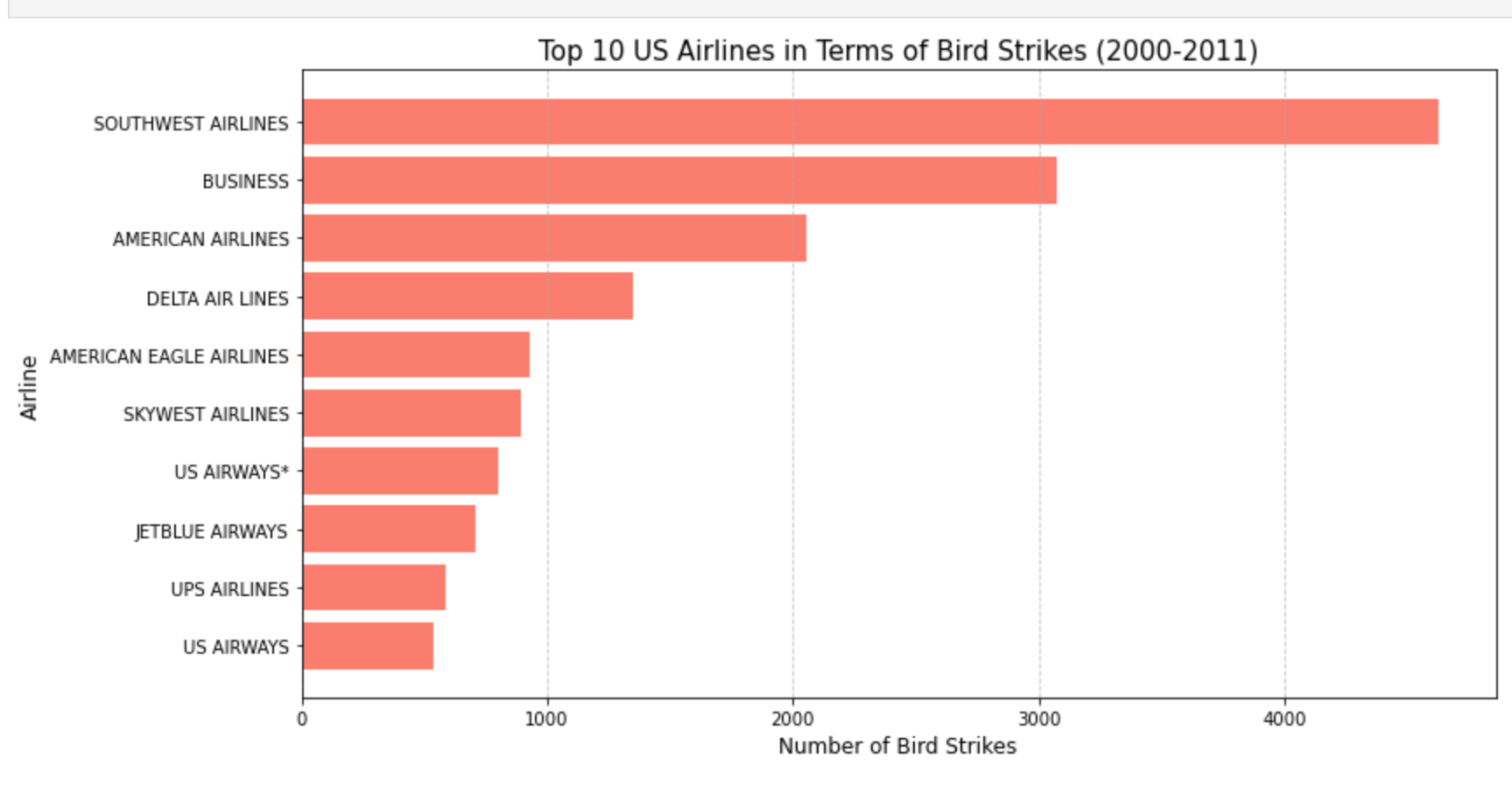
```python
plt.figure(figsize=(10, 6))
plt.bar(bird_strikes_per_year['Year'], bird_strikes_per_year['Number of Bird Strikes'], color='skyblue')
plt.title('Yearly Bird Strikes in the US (2000-2011)', fontsize=15)
plt.xlabel('Year', fontsize=12)
plt.ylabel('Number of Bird Strikes', fontsize=12)
plt.grid(axis='y', linestyle='--', alpha=0.7)
plt.tight_layout()
plt.show()
```



2. Yearly Analysis & Bird Strikes in the US

In [6]:

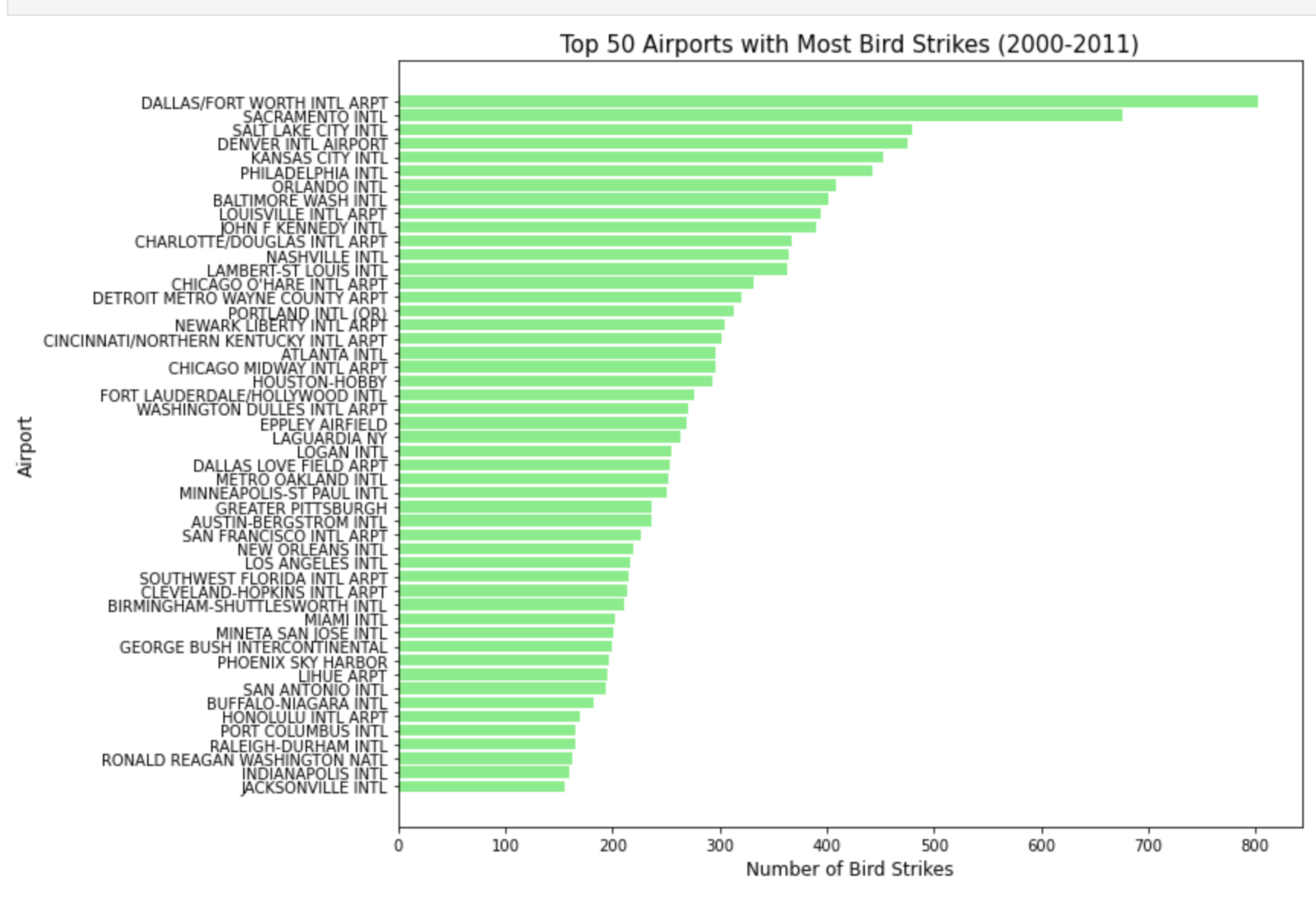
```python
# Group by the 'Aircraft: Airline/Operator' column to find the top 10 airlines with the most bird strikes
top_airlines = bird_strikes_df.groupby('Aircraft: Airline/Operator').size().reset_index(name='Number of Bird Strikes')

# Sort the airlines by the number of bird strikes in descending order and take the top 10
top_10_airlines = top_airlines.sort_values(by='Number of Bird Strikes', ascending=False).head(10)

# Plot the top 10 airlines
plt.figure(figsize=(12, 6))
plt.barh(top_10_airlines['Aircraft: Airline/Operator'], top_10_airlines['Number of Bird Strikes'], color='salmon')
plt.title('Top 10 US Airlines in Terms of Bird Strikes (2000-2011)', fontsize=15)
plt.xlabel('Number of Bird Strikes', fontsize=12)
plt.ylabel('Airline', fontsize=12)
plt.gca().invert_yaxis()  # Invert y-axis to have the top airline at the top
plt.grid(axis='x', linestyle='--', alpha=0.7)
plt.tight_layout()
plt.show()
```



3: Top 10 US Airlines in Terms of Bird Strikes

In [7]:

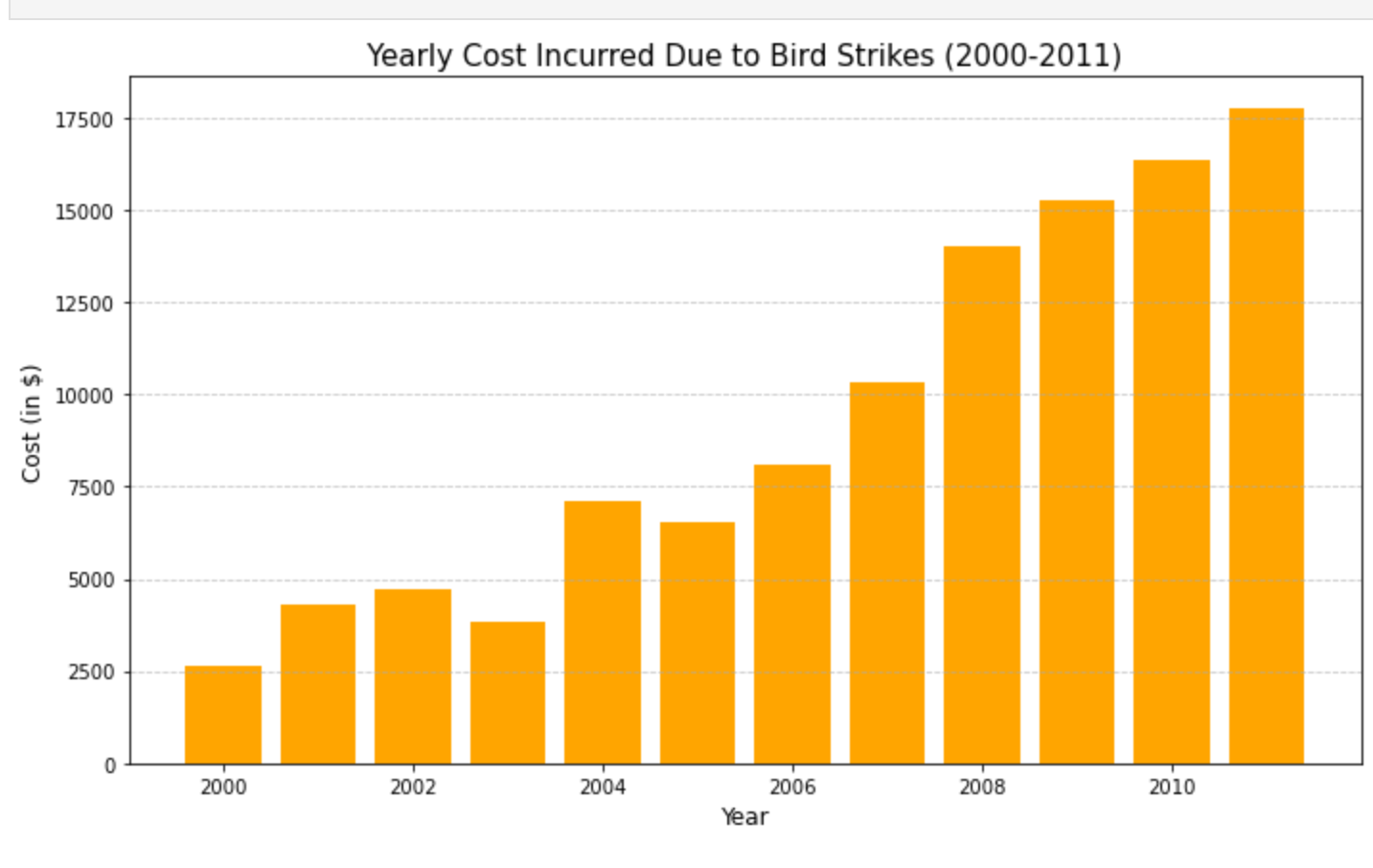
```python
top_airports = bird_strikes_df.groupby('Airport: Name').size().reset_index(name='Number of Bird Strikes')
top_50_airports = top_airports.sort_values(by='Number of Bird Strikes', ascending=False).head(50)

plt.figure(figsize=(12, 8))
plt.barh(top_50_airports['Airport: Name'], top_50_airports['Number of Bird Strikes'], color='lightgreen')
plt.title('Top 50 Airports with Most Bird Strikes (2000-2011)', fontsize=15)
plt.xlabel('Number of Bird Strikes', fontsize=12)
plt.ylabel('Airport', fontsize=12)
plt.gca().invert_yaxis()
plt.tight_layout()
plt.show()
```



4. Airports with Most Incidents of Bird Strikes – Top 50

In [8]:

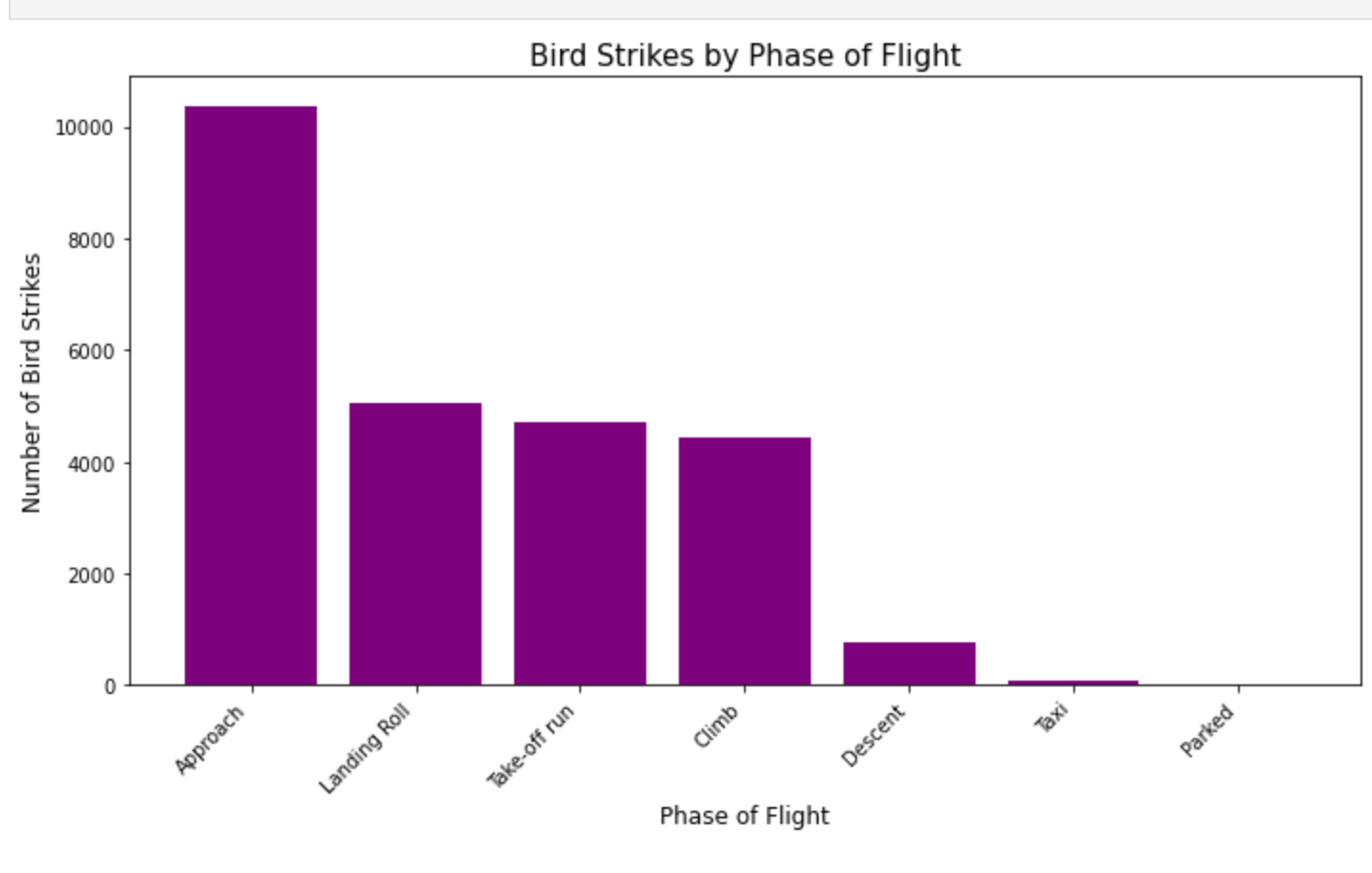
```python
bird_strikes_df['Cost: Total $'] = pd.to_numeric(bird_strikes_df['Cost: Total $'], errors='coerce')
yearly_cost = bird_strikes_df.groupby('Year')['Cost: Total $'].sum().reset_index()

plt.figure(figsize=(10, 6))
plt.bar(yearly_cost['Year'], yearly_cost['Cost: Total $'], color='orange')
plt.title('Yearly Cost Incurred Due to Bird Strikes (2000-2011)', fontsize=15)
plt.xlabel('Year', fontsize=12)
plt.ylabel('Cost (in $)', fontsize=12)
plt.grid(axis='y', linestyle='--', alpha=0.7)
plt.tight_layout()
plt.show()
```



5. Yearly Cost Incurred Due to Bird Strikes

In [30]:

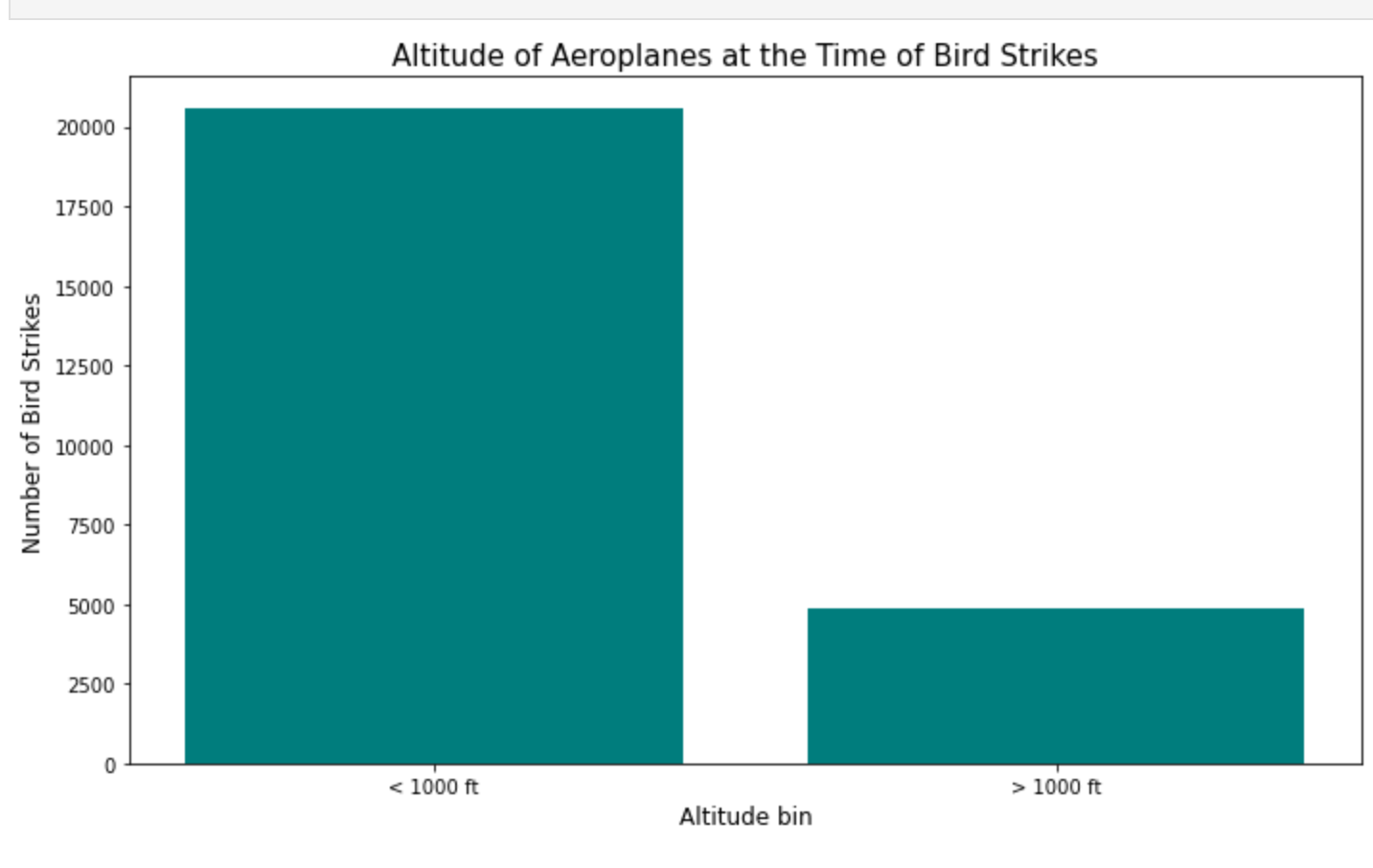
```python
import pandas as pd
import matplotlib.pyplot as plt

# Load the dataset
bird_strikes_data = pd.read_csv('Bird Strikes.csv')



# Perform the analysis with the correct column name
phase_of_flight_counts = bird_strikes_data['When: Phase of flight'].value_counts().reset_index()

# Renaming the columns for clarity
phase_of_flight_counts.columns = ['Flight Phase', 'Number of Bird Strikes']

# Plotting the bar chart
plt.figure(figsize=(10, 6))
plt.bar(phase_of_flight_counts['Flight Phase'], phase_of_flight_counts['Number of Bird Strikes'], color='purple')
plt.title('Bird Strikes by Phase of Flight', fontsize=15)
plt.xlabel('Phase of Flight', fontsize=12)
plt.ylabel('Number of Bird Strikes', fontsize=12)
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
plt.show()
```



6. When Do Most Bird Strikes Occur?

In [24]:

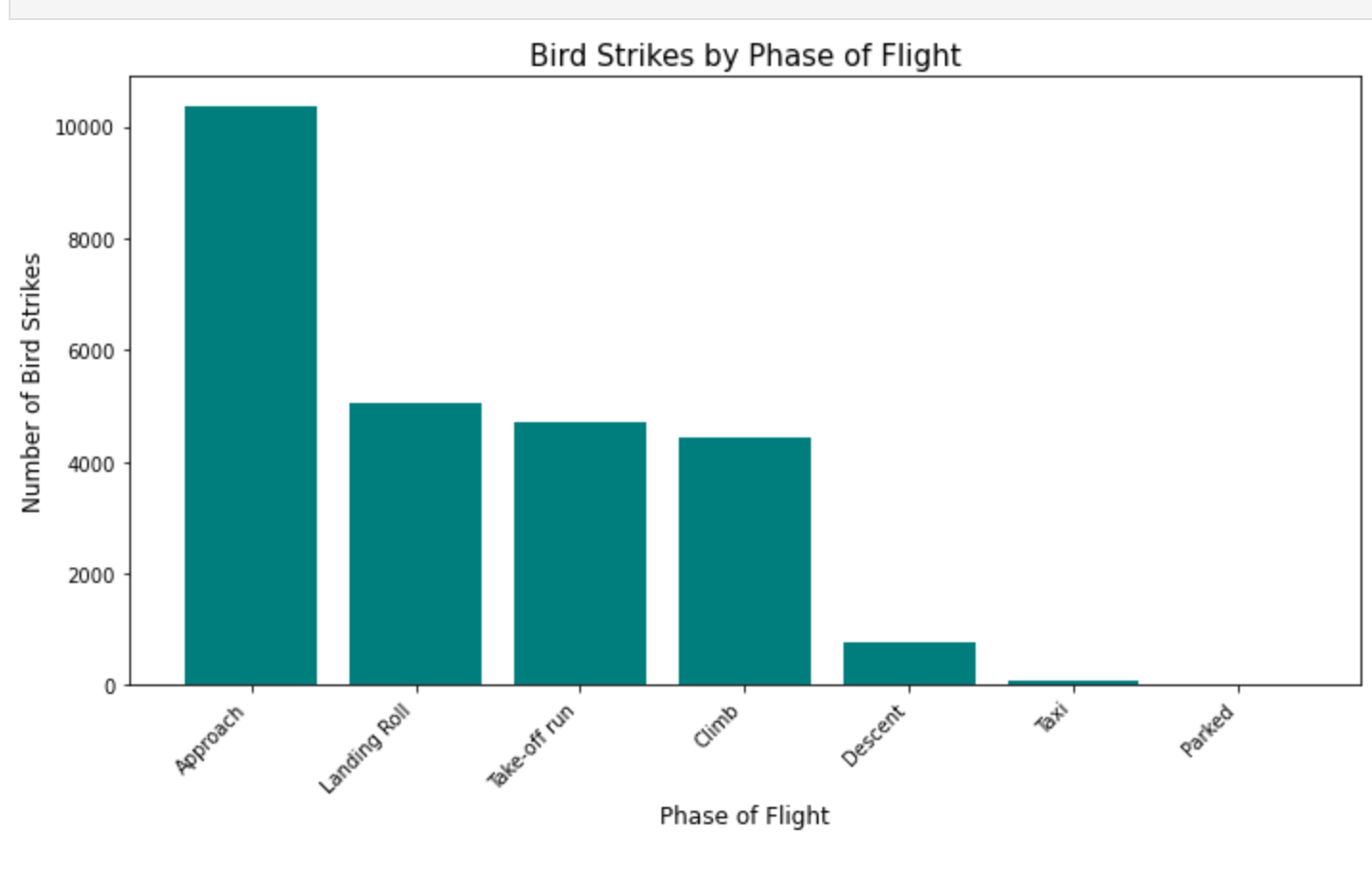
```python
altitude_counts = bird_strikes_data['Altitude bin'].value_counts().reset_index(name='Number of Bird Strikes')
altitude_counts.columns = ['Altitude bin', 'Number of Bird Strikes']

plt.figure(figsize=(10, 6))
plt.bar(altitude_counts['Altitude bin'], altitude_counts['Number of Bird Strikes'], color='teal')
plt.title('Altitude of Aeroplanes at the Time of Bird Strikes', fontsize=15)
plt.xlabel('Altitude bin', fontsize=12)
plt.ylabel('Number of Bird Strikes', fontsize=12)
plt.tight_layout()
plt.show()
```



7 Altitude of aeroplanes at the time of strike

In [25]:

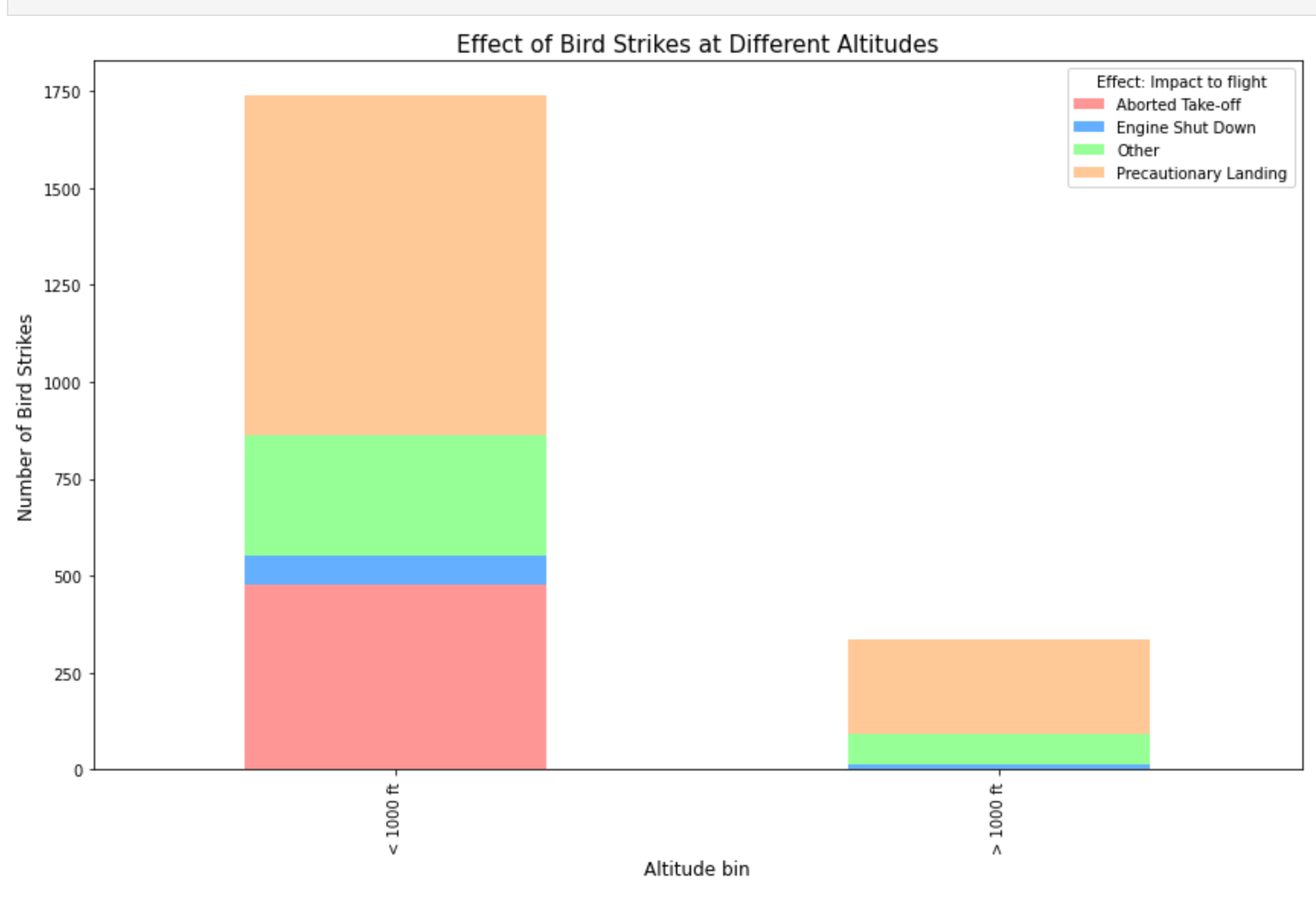
```python
import pandas as pd
import matplotlib.pyplot as plt

# Load the dataset
bird_strikes_data = pd.read_csv('Bird Strikes.csv')

# Count the occurrences of each phase of flight
phase_of_flight_counts = bird_strikes_data['When: Phase of flight'].value_counts().reset_index(name='Number of Bird Strikes')
phase_of_flight_counts.columns = ['Phase of Flight', 'Number of Bird Strikes']

# Plotting the bar chart
plt.figure(figsize=(10, 6))
plt.bar(phase_of_flight_counts['Phase of Flight'], phase_of_flight_counts['Number of Bird Strikes'], color='teal')
plt.title('Bird Strikes by Phase of Flight', fontsize=15)
plt.xlabel('Phase of Flight', fontsize=12)
plt.ylabel('Number of Bird Strikes', fontsize=12)
plt.xticks(rotation=45, ha='right')  # Rotate labels for better readability
plt.tight_layout()
plt.show()
```



8 Phase of flight at the time of the strike.

In [17]:

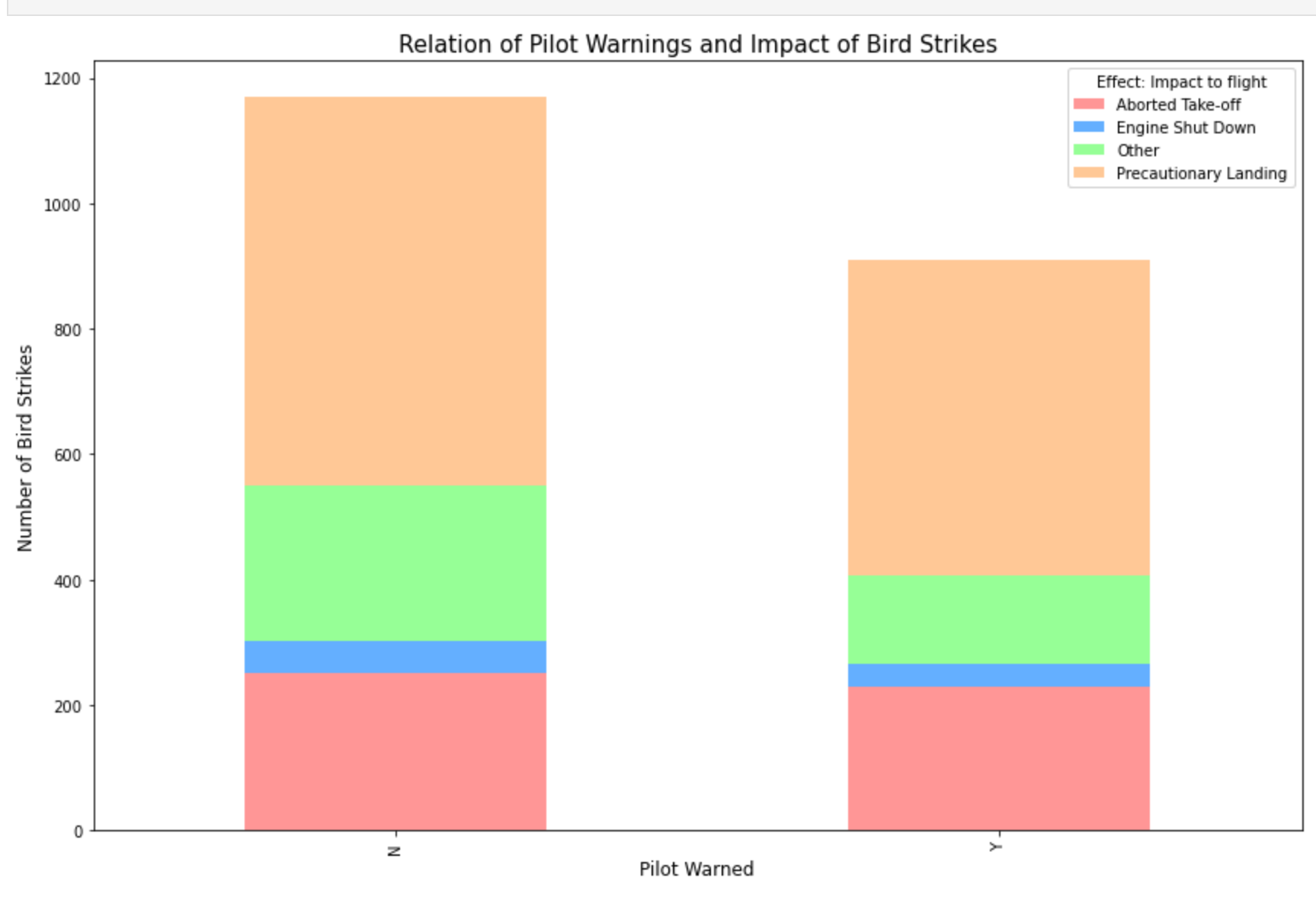
```python
altitude_effect = bird_strikes_df.groupby(['Altitude bin', 'Effect: Impact to flight']).size().unstack(fill_value=0)

altitude_effect.plot(kind='bar', stacked=True, figsize=(12, 8), color=['#ff9999','#66b3ff','#99ff99','#ffcc99'])
plt.title('Effect of Bird Strikes at Different Altitudes', fontsize=15)
plt.xlabel('Altitude bin', fontsize=12)
plt.ylabel('Number of Bird Strikes', fontsize=12)
plt.tight_layout()
plt.show()
```



9 Average Altitude of the aeroplanes in different phases at the time of strike

In [18]:

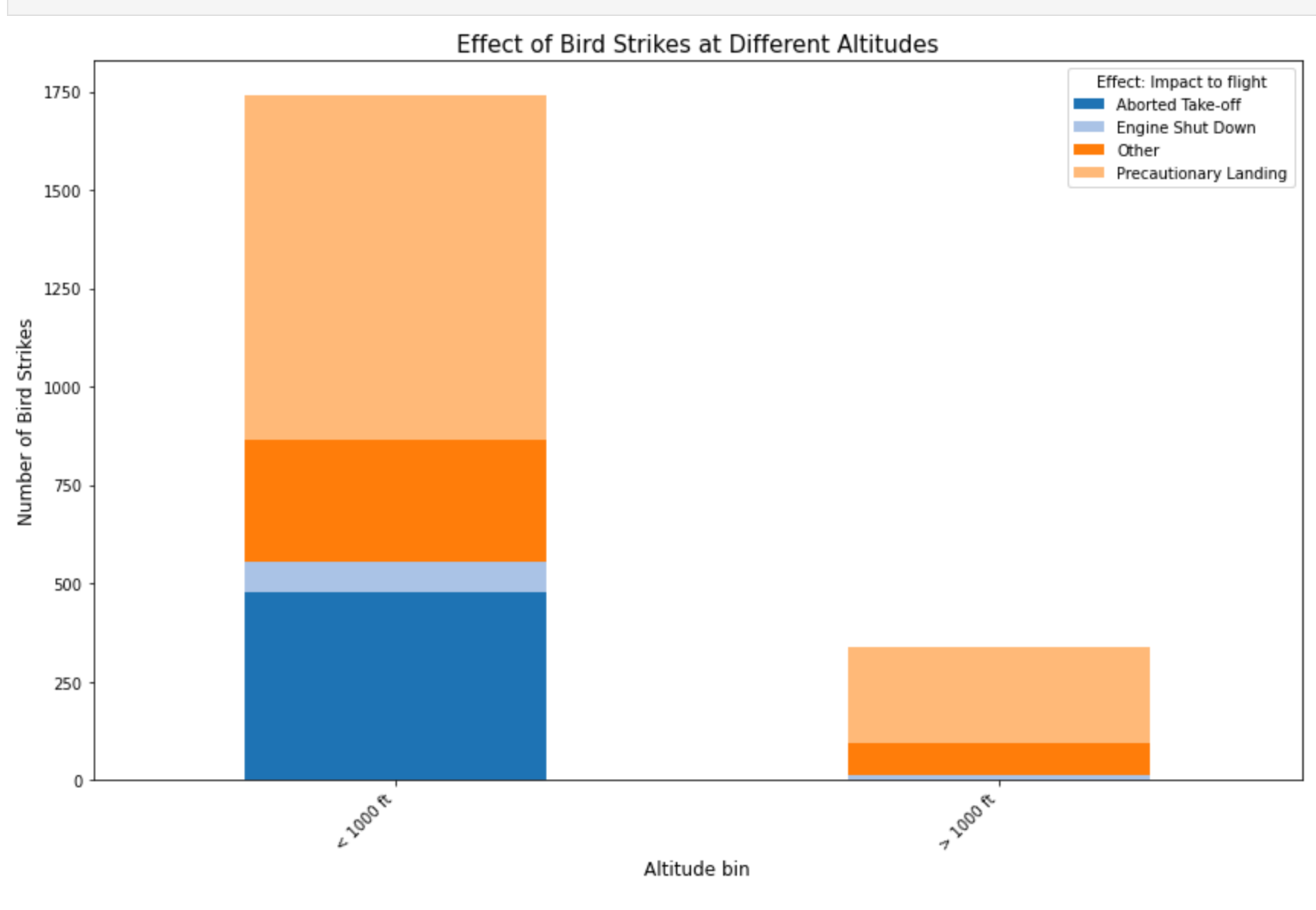
```python
warning_effect_relation = bird_strikes_df.groupby(['Pilot warned of birds or wildlife?', 'Effect: Impact to flight']).size().unstack(fill_value=0)

warning_effect_relation.plot(kind='bar', stacked=True, figsize=(12, 8), color=['#ff9999','#66b3ff','#99ff99','#ffcc99'])
plt.title('Relation of Pilot Warnings and Impact of Bird Strikes', fontsize=15)
plt.xlabel('Pilot Warned', fontsize=12)
plt.ylabel('Number of Bird Strikes', fontsize=12)
plt.tight_layout()
plt.show()
```



10 Effect of Bird Strikes & Impact on Flight

In [27]:

```python
import pandas as pd
import matplotlib.pyplot as plt

# Load the dataset
bird_strikes_data = pd.read_csv('Bird Strikes.csv')

# Group by 'Altitude bin' and 'Effect: Impact to flight' and count occurrences
effect_by_altitude = bird_strikes_data.groupby(['Altitude bin', 'Effect: Impact to flight']).size().unstack(fill_value=0)

# Plotting the stacked bar chart without using colormap
plt.figure(figsize=(12, 8))
effect_by_altitude.plot(kind='bar', stacked=True, ax=plt.gca(), color=plt.get_cmap('tab20').colors)
plt.title('Effect of Bird Strikes at Different Altitudes', fontsize=15)
plt.xlabel('Altitude bin', fontsize=12)
plt.ylabel('Number of Bird Strikes', fontsize=12)
plt.xticks(rotation=45, ha='right')  # Rotate labels for better readability
plt.tight_layout()
plt.show()
```



11 Effect of Strike at Different Altitude

In [28]:

```python
import pandas as pd
import matplotlib.pyplot as plt

# Load the dataset
bird_strikes_data = pd.read_csv('Bird Strikes.csv')

# Count occurrences of effects by whether pilots were warned or not
warning_effect_counts = bird_strikes_data.groupby(['Pilot warned of birds or wildlife?', 'Effect: Impact to flight']).size().unstack(fill_value=0)

# Plotting the stacked bar chart
plt.figure(figsize=(12, 8))
warning_effect_counts.plot(kind='bar', stacked=True, color=plt.get_cmap('tab20').colors)
plt.title('Effect of Bird Strikes with or without Prior Warning', fontsize=15)
plt.xlabel('Pilot Warned', fontsize=12)
plt.ylabel('Number of Bird Strikes', fontsize=12)
plt.xticks(rotation=0)  # Rotate labels for better readability
plt.tight_layout()
plt.show()
```

```
<Figure size 864x576 with 0 Axes>
```

12 Were Pilots Informed? & Prior Warning and Effect of Strike Relation

In [ ]: