Introduction

A bird strike is strictly defined as a collision between a bird and an aircraft which is in flight or on a take-off or landing roll. The term is often expanded to cover other wildlife strikes - with bats or ground animals. Bird Strike is common and can be a significant threat to aircraft safety. For smaller aircraft, significant damage may be caused to the aircraft structure and all aircraft, especially jet-engine ones, are vulnerable to the loss of thrust which can follow the ingestion of birds into engine air intakes. This has resulted in several fatal accidents. Bird strikes may occur during any phase of flight, but are most likely during the take-off, initial climb, approach and landing phases due to the greater numbers of birds in flight at lower levels. To have a closer look the following document visually depicts the data collected on Bird Strikes by FAA between 2000-2011.

Import Libraries

In [1]:
```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns
%matplotlib inline
```

In [2]:
```
!pip install tidyverse
!pip install patchwork
!pip install tidytext
!pip install lattice
!pip install maps
!pip install mapdata
```

```
Requirement already satisfied: tidyverse in c:\users\payal\anaconda3\lib\site-packa
ges (0.1.0)
Requirement already satisfied: plotnine<0.11.0,>=0.10.1 in c:\users\payal\anaconda
3\lib\site-packages (from tidyverse) (0.10.1)
Requirement already satisfied: siuba<0.4.0,>=0.3.0 in c:\users\payal\anaconda3\lib\
site-packages (from tidyverse) (0.3.0)
Requirement already satisfied: matplotlib>=3.5.0 in c:\users\payal\anaconda3\lib\si
te-packages (from plotnine<0.11.0,>=0.10.1->tidyverse) (3.7.1)
Requirement already satisfied: mizani>=0.8.1 in c:\users\payal\anaconda3\lib\site-p
ackages (from plotnine<0.11.0,>=0.10.1->tidyverse) (0.10.0)
Requirement already satisfied: numpy>=1.19.0 in c:\users\payal\anaconda3\lib\site-p
ackages (from plotnine<0.11.0,>=0.10.1->tidyverse) (1.24.3)
Requirement already satisfied: pandas>=1.3.5 in c:\users\payal\anaconda3\lib\site-p
ackages (from plotnine<0.11.0,>=0.10.1->tidyverse) (1.5.3)
Requirement already satisfied: patsy>=0.5.1 in c:\users\payal\anaconda3\lib\site-pa
ckages (from plotnine<0.11.0,>=0.10.1->tidyverse) (0.5.3)
Requirement already satisfied: scipy>=1.5.0 in c:\users\payal\anaconda3\lib\site-pa
ckages (from plotnine<0.11.0,>=0.10.1->tidyverse) (1.10.1)
Requirement already satisfied: statsmodels>=0.13.2 in c:\users\payal\anaconda3\lib\
site-packages (from plotnine<0.11.0,>=0.10.1->tidyverse) (0.13.5)
Requirement already satisfied: SQLAlchemy>=1.2.19 in c:\users\payal\anaconda3\lib\s
ite-packages (from siuba<0.4.0,>=0.3.0->tidyverse) (1.4.39)
Requirement already satisfied: PyYAML>=3.0.0 in c:\users\payal\anaconda3\lib\site-p
ackages (from siuba<0.4.0,>=0.3.0->tidyverse) (6.0)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\payal\anaconda3\lib\sit
e-packages (from matplotlib>=3.5.0->plotnine<0.11.0,>=0.10.1->tidyverse) (1.0.5)
Requirement already satisfied: cycler>=0.10 in c:\users\payal\anaconda3\lib\site-pa
ckages (from matplotlib>=3.5.0->plotnine<0.11.0,>=0.10.1->tidyverse) (0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\payal\anaconda3\lib\si
te-packages (from matplotlib>=3.5.0->plotnine<0.11.0,>=0.10.1->tidyverse) (4.25.0)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\payal\anaconda3\lib\si
te-packages (from matplotlib>=3.5.0->plotnine<0.11.0,>=0.10.1->tidyverse) (1.4.4)
Requirement already satisfied: packaging>=20.0 in c:\users\payal\anaconda3\lib\site
-packages (from matplotlib>=3.5.0->plotnine<0.11.0,>=0.10.1->tidyverse) (23.0)
Requirement already satisfied: pillow>=6.2.0 in c:\users\payal\anaconda3\lib\site-p
ackages (from matplotlib>=3.5.0->plotnine<0.11.0,>=0.10.1->tidyverse) (9.4.0)
Requirement already satisfied: pyparsing>=2.3.1 in c:\users\payal\anaconda3\lib\sit
e-packages (from matplotlib>=3.5.0->plotnine<0.11.0,>=0.10.1->tidyverse) (3.0.9)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\payal\anaconda3\li
b\site-packages (from matplotlib>=3.5.0->plotnine<0.11.0,>=0.10.1->tidyverse) (2.8.
2)
Requirement already satisfied: tzdata in c:\users\payal\anaconda3\lib\site-packages
(from mizani>=0.8.1->plotnine<0.11.0,>=0.10.1->tidyverse) (2023.4)
Requirement already satisfied: pytz>=2020.1 in c:\users\payal\anaconda3\lib\site-pa
ckages (from pandas>=1.3.5->plotnine<0.11.0,>=0.10.1->tidyverse) (2022.7)
Requirement already satisfied: six in c:\users\payal\anaconda3\lib\site-packages (f
rom patsy>=0.5.1->plotnine<0.11.0,>=0.10.1->tidyverse) (1.16.0)
Requirement already satisfied: greenlet!=0.4.17 in c:\users\payal\anaconda3\lib\sit
e-packages (from SQLAlchemy>=1.2.19->siuba<0.4.0,>=0.3.0->tidyverse) (3.0.3)
Requirement already satisfied: patchwork in c:\users\payal\anaconda3\lib\site-packa
ges (1.0.1)
Requirement already satisfied: fabric<3.0,>=2.0 in c:\users\payal\anaconda3\lib\sit
e-packages (from patchwork) (2.7.1)
Requirement already satisfied: invoke<2.0,>=1.3 in c:\users\payal\anaconda3\lib\sit
e-packages (from fabric<3.0,>=2.0->patchwork) (1.7.3)
Requirement already satisfied: paramiko>=2.4 in c:\users\payal\anaconda3\lib\site-p
ackages (from fabric<3.0,>=2.0->patchwork) (2.8.1)
Requirement already satisfied: pathlib2 in c:\users\payal\anaconda3\lib\site-packag
es (from fabric<3.0,>=2.0->patchwork) (2.3.7.post1)
```

```
Requirement already satisfied: bcrypt>=3.1.3 in c:\users\payal\anaconda3\lib\site-p
ackages (from paramiko>=2.4->fabric<3.0,>=2.0->patchwork) (3.2.0)
Requirement already satisfied: cryptography>=2.5 in c:\users\payal\anaconda3\lib\si
te-packages (from paramiko>=2.4->fabric<3.0,>=2.0->patchwork) (39.0.1)
Requirement already satisfied: pynacl>=1.0.1 in c:\users\payal\anaconda3\lib\site-p
ackages (from paramiko>=2.4->fabric<3.0,>=2.0->patchwork) (1.5.0)
Requirement already satisfied: six in c:\users\payal\anaconda3\lib\site-packages (f
rom pathlib2->fabric<3.0,>=2.0->patchwork) (1.16.0)
Requirement already satisfied: cffi>=1.1 in c:\users\payal\anaconda3\lib\site-packa
ges (from bcrypt>=3.1.3->paramiko>=2.4->fabric<3.0,>=2.0->patchwork) (1.16.0)
Requirement already satisfied: pycparser in c:\users\payal\anaconda3\lib\site-packa
ges (from cffi>=1.1->bcrypt>=3.1.3->paramiko>=2.4->fabric<3.0,>=2.0->patchwork) (2.
21)
Requirement already satisfied: tidytext in c:\users\payal\anaconda3\lib\site-packag
es (0.0.1)
Requirement already satisfied: siuba in c:\users\payal\anaconda3\lib\site-packages
(from tidytext) (0.3.0)
Requirement already satisfied: pandas>=0.24.0 in c:\users\payal\anaconda3\lib\site-
packages (from siuba->tidytext) (1.5.3)
Requirement already satisfied: numpy>=1.12.0 in c:\users\payal\anaconda3\lib\site-p
ackages (from siuba->tidytext) (1.24.3)
Requirement already satisfied: SQLAlchemy>=1.2.19 in c:\users\payal\anaconda3\lib\s
ite-packages (from siuba->tidytext) (1.4.39)
Requirement already satisfied: PyYAML>=3.0.0 in c:\users\payal\anaconda3\lib\site-p
ackages (from siuba->tidytext) (6.0)
Requirement already satisfied: python-dateutil>=2.8.1 in c:\users\payal\anaconda3\l
ib\site-packages (from pandas>=0.24.0->siuba->tidytext) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in c:\users\payal\anaconda3\lib\site-pa
ckages (from pandas>=0.24.0->siuba->tidytext) (2022.7)
Requirement already satisfied: greenlet!=0.4.17 in c:\users\payal\anaconda3\lib\sit
e-packages (from SQLAlchemy>=1.2.19->siuba->tidytext) (3.0.3)
Requirement already satisfied: six>=1.5 in c:\users\payal\anaconda3\lib\site-packag
es (from python-dateutil>=2.8.1->pandas>=0.24.0->siuba->tidytext) (1.16.0)
Requirement already satisfied: lattice in c:\users\payal\anaconda3\lib\site-package
s (0.1.4)
Requirement already satisfied: Jinja2 in c:\users\payal\anaconda3\lib\site-packages
(from lattice) (3.1.2)
Requirement already satisfied: cbor2 in c:\users\payal\anaconda3\lib\site-packages
(from lattice) (5.5.1)
Requirement already satisfied: jsonschema in c:\users\payal\anaconda3\lib\site-pack
ages (from lattice) (4.17.3)
Requirement already satisfied: pygit2<2.0.0,>=1.10.0 in c:\users\payal\anaconda3\li
b\site-packages (from lattice) (1.13.3)
Requirement already satisfied: pyyaml in c:\users\payal\anaconda3\lib\site-packages
(from lattice) (6.0)
Requirement already satisfied: stringcase==1.2.0 in c:\users\payal\anaconda3\lib\si
te-packages (from lattice) (1.2.0)
Requirement already satisfied: cffi>=1.16.0 in c:\users\payal\anaconda3\lib\site-pa
ckages (from pygit2<2.0.0,>=1.10.0->lattice) (1.16.0)
Requirement already satisfied: MarkupSafe>=2.0 in c:\users\payal\anaconda3\lib\site
-packages (from Jinja2->lattice) (2.1.1)
Requirement already satisfied: attrs>=17.4.0 in c:\users\payal\anaconda3\lib\site-p
ackages (from jsonschema->lattice) (22.1.0)
Requirement already satisfied: pyrsistent!=0.17.0,!=0.17.1,!=0.17.2,>=0.14.0 in c:\
users\payal\anaconda3\lib\site-packages (from jsonschema->lattice) (0.18.0)
Requirement already satisfied: pycparser in c:\users\payal\anaconda3\lib\site-packa
ges (from cffi>=1.16.0->pygit2<2.0.0,>=1.10.0->lattice) (2.21)
Requirement already satisfied: maps in c:\users\payal\anaconda3\lib\site-packages
(5.1.1)
```

```
Requirement already satisfied: mapdata in c:\users\payal\anaconda3\lib\site-package
s (2.16.5)
```

In [3]:
```
import tidyverse
import patchwork
import tidytext
import lattice
import maps
import mapdata
```

In [4]:
```
Bird=pd.read_excel('C:/Users/payal/Desktop/Payal/Internkaksha Project/Data Visualis
```

In [5]:
```
Bird.head(3)
```

Out[5]:

	Record ID	Aircraft: Type	Airport: Name	Altitude bin	Aircraft: Make/Model	Wildlife: Number struck	Wildlife: Number Struck Actual	Effect: Impact to flight	FlightDate
0	202152	Airplane	LAGUARDIA NY	> 1000 ft	B-737-400	Over 100	859	Engine Shut Down	2000-11-23
1	208159	Airplane	DALLAS/FORT WORTH INTL ARPT	< 1000 ft	MD-80	Over 100	424	None	2001-07-25
2	207601	Airplane	LAKEFRONT AIRPORT	< 1000 ft	C-500	Over 100	261	None	2001-09-14

3 rows × 26 columns

In [6]:
```
Bird.shape
```

Out[6]:
```
(25558, 26)
```

In [7]:
```
Bird.describe()
```

Out[7]:

	Record ID	Wildlife: Number Struck Actual	Cost: Total $	Feet above ground	Number of people injured
count	25558.000000	25558.000000	2.555800e+04	25429.000000	25558.000000
mean	253916.085609	2.691525	5.567354e+03	799.028432	0.001056
std	38510.453382	12.793975	1.219713e+05	1740.079843	0.050420
min	1195.000000	1.000000	0.000000e+00	0.000000	0.000000
25%	225783.750000	1.000000	0.000000e+00	0.000000	0.000000
50%	248749.000000	1.000000	0.000000e+00	50.000000	0.000000
75%	269168.750000	1.000000	0.000000e+00	700.000000	0.000000
max	321909.000000	942.000000	1.239775e+07	18000.000000	6.000000

In [8]:
```
Bird.isnull().sum()
```

Out[8]:
```
Record ID                                0
Aircraft: Type                         129
Airport: Name                          129
Altitude bin                           129
Aircraft: Make/Model                     0
Wildlife: Number struck                129
Wildlife: Number Struck Actual           0
Effect: Impact to flight               129
FlightDate                             129
Effect: Indicated Damage                 0
Aircraft: Number of engines?           267
Aircraft: Airline/Operator             129
Origin State                           449
When: Phase of flight                  129
Conditions: Precipitation                0
Remains of wildlife collected?           0
Remains of wildlife sent to Smithsonian  0
Remarks                               4771
Wildlife: Size                         129
Conditions: Sky                          0
Wildlife: Species                        0
Pilot warned of birds or wildlife?     129
Cost: Total $                            0
Feet above ground                      129
Number of people injured                 0
Is Aircraft Large?                     129
dtype: int64
```

In [9]:
```
Bird['Remarks'].unique()
```

Out[9]:
```
array(['FLT 753. PILOT REPTD A HUNDRED BIRDS ON UNKN TYPE. #1 ENG WAS SHUT DOWN AND
DIVERTED TO EWR. SLIGHT VIBRATION. A/C WAS OUT OF SVC FOR REPAIRS TO COWLING, FAN D
UCT ACCOUSTIC PANEL. INGESTION. DENTED FAN BLADE #26 IN #1 ENG. HEAVY BLOOD STAINS
ON L WINGTIP',
       '102 CARCASSES FOUND. 1 LDG LIGHT ON NOSE GEAR WAS DAMAGED AND REPLACED.',
       'FLEW UNDER A VERY LARGE FLOCK OF BIRDS OVER APCH END OF RWY. NO DMG. JUST A
LOT OF BIRD DROPPINGS ON WINDSCREEN.',
       ...,
       'STRUCK BIRD ON RT FRONT DURING T/O. BIRD REPTD AS BROWN/WHITE. TWY.',
       'PILOTS REPORT STRIKING UNKNOWN BIRD ON RWY 21L BTWN TWY F & J. NO REMAINS F
OUND ON RWY OR ON A/C. NO DMG TO A/C.',
       'HIT CENTER OF RADOME, CAVING IN ABOUT 12". RADOME WAS REPLACED. CARCASS FOU
ND IN SAFETY ARA ON RT SIDE OF RWY 22 AT INTXN OF RWY 18/36.'],
      dtype=object)
```

In [10]:
```
Bird['Origin State'].unique()
```

Out[10]:
```
array(['New York', 'Texas', 'Louisiana', 'Washington', 'Virginia', nan,
       'Delaware', 'DC', 'Georgia', 'Florida', 'California', 'Illinois',
       'Connecticut', 'Missouri', 'Rhode Island', 'Hawaii', 'Arizona',
       'Tennessee', 'South Carolina', 'South Dakota', 'New Jersey',
       'Colorado', 'Minnesota', 'Alabama', 'Ohio', 'Wisconsin',
       'Michigan', 'Massachusetts', 'Alaska', 'North Carolina',
       'Kentucky', 'Indiana', 'Oregon', 'Pennsylvania', 'New Hampshire',
       'Arkansas', 'Nevada', 'Mississippi', 'Maryland', 'Maine', 'Quebec',
       'Idaho', 'British Columbia', 'Utah', 'Nebraska', 'Iowa',
       'New Mexico', 'West Virginia', 'Oklahoma', 'North Dakota',
       'Vermont', 'Wyoming', 'Kansas', 'Prince Edward Island', 'Montana',
       'Puerto Rico', 'Ontario', 'Virgin Islands',
       'Newfoundland and Labrador', 'Alberta', 'Saskatchewan'],
      dtype=object)
```

In [11]:
```
Bird['When: Phase of flight'].unique()
```

Out[11]:
```
array(['Climb', 'Landing Roll', 'Approach', 'Take-off run', 'Descent',
       nan, 'Taxi', 'Parked'], dtype=object)
```

In [12]:
```
Bird['Aircraft: Type'].unique()
```

Out[12]:
```
array(['Airplane', nan], dtype=object)
```

In [13]:
```
Bird.columns
```

Out[13]:
```
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')
```

In [14]:
```
Bird.size
```

Out[14]:
```
664508
```

In [15]:
```
Bird.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 25558 entries, 0 to 25557
Data columns (total 26 columns):
 #   Column                                  Non-Null Count  Dtype
---  ------                                  --------------  -----
 0   Record ID                               25558 non-null  int64
 1   Aircraft: Type                          25429 non-null  object
 2   Airport: Name                           25429 non-null  object
 3   Altitude bin                            25429 non-null  object
 4   Aircraft: Make/Model                    25558 non-null  object
 5   Wildlife: Number struck                 25429 non-null  object
 6   Wildlife: Number Struck Actual          25558 non-null  int64
 7   Effect: Impact to flight                25429 non-null  object
 8   FlightDate                              25429 non-null  datetime64[ns]
 9   Effect: Indicated Damage                25558 non-null  object
 10  Aircraft: Number of engines?            25291 non-null  object
 11  Aircraft: Airline/Operator              25429 non-null  object
 12  Origin State                            25109 non-null  object
 13  When: Phase of flight                   25429 non-null  object
 14  Conditions: Precipitation               25558 non-null  object
 15  Remains of wildlife collected?          25558 non-null  bool
 16  Remains of wildlife sent to Smithsonian 25558 non-null  bool
 17  Remarks                                 20787 non-null  object
 18  Wildlife: Size                          25429 non-null  object
 19  Conditions: Sky                         25558 non-null  object
 20  Wildlife: Species                       25558 non-null  object
 21  Pilot warned of birds or wildlife?      25429 non-null  object
 22  Cost: Total $                           25558 non-null  int64
 23  Feet above ground                       25429 non-null  float64
 24  Number of people injured                25558 non-null  int64
 25  Is Aircraft Large?                      25429 non-null  object
dtypes: bool(2), datetime64[ns](1), float64(1), int64(4), object(18)
memory usage: 4.7+ MB
```

In [16]:
```
Bird.dropna(inplace=True)
```

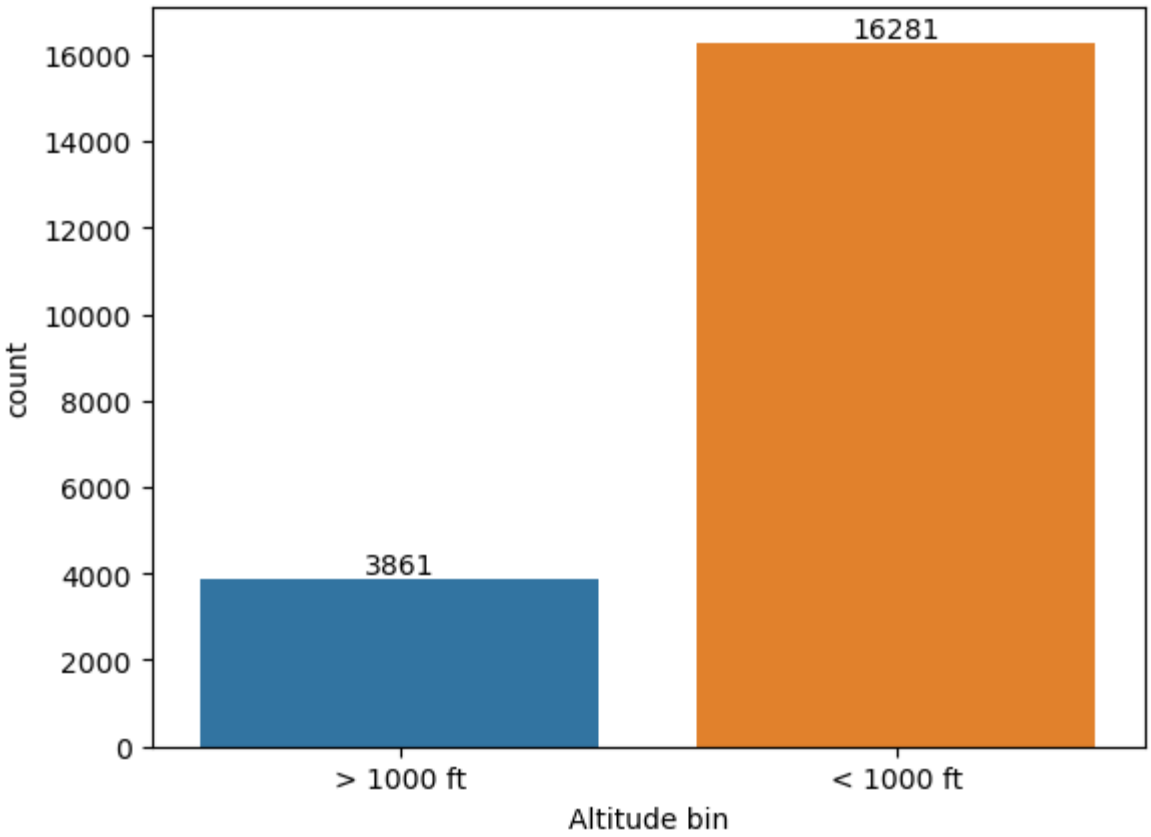
In [17]:
```
Bird.isnull().sum()
```

Out[17]:
```
Record ID                                  0
Aircraft: Type                             0
Airport: Name                              0
Altitude bin                               0
Aircraft: Make/Model                       0
Wildlife: Number struck                    0
Wildlife: Number Struck Actual             0
Effect: Impact to flight                   0
FlightDate                                 0
Effect: Indicated Damage                   0
Aircraft: Number of engines?               0
Aircraft: Airline/Operator                 0
Origin State                               0
When: Phase of flight                      0
Conditions: Precipitation                  0
Remains of wildlife collected?             0
Remains of wildlife sent to Smithsonian    0
Remarks                                    0
Wildlife: Size                             0
Conditions: Sky                            0
Wildlife: Species                          0
Pilot warned of birds or wildlife?         0
Cost: Total $                              0
Feet above ground                          0
Number of people injured                   0
Is Aircraft Large?                         0
dtype: int64
```

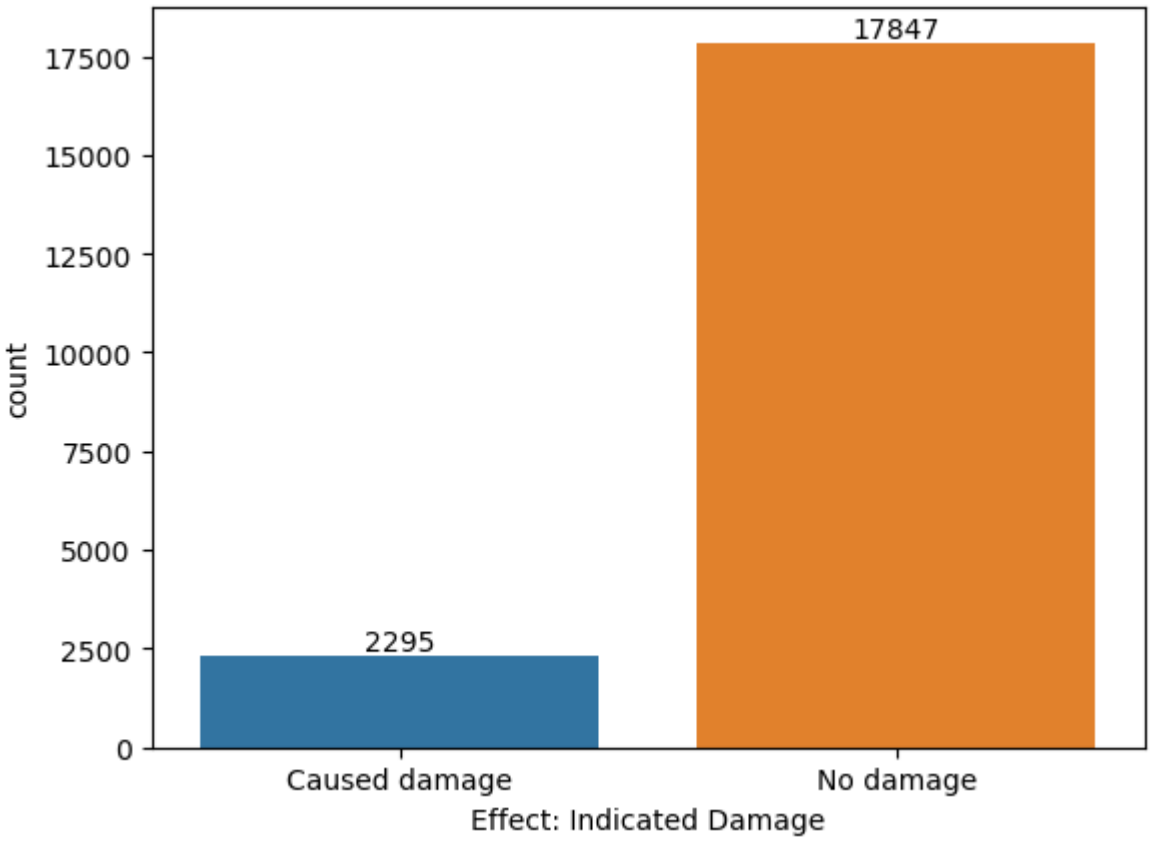
In [18]:
```
Bird.shape
```

Out[18]:
```
(20142, 26)
```

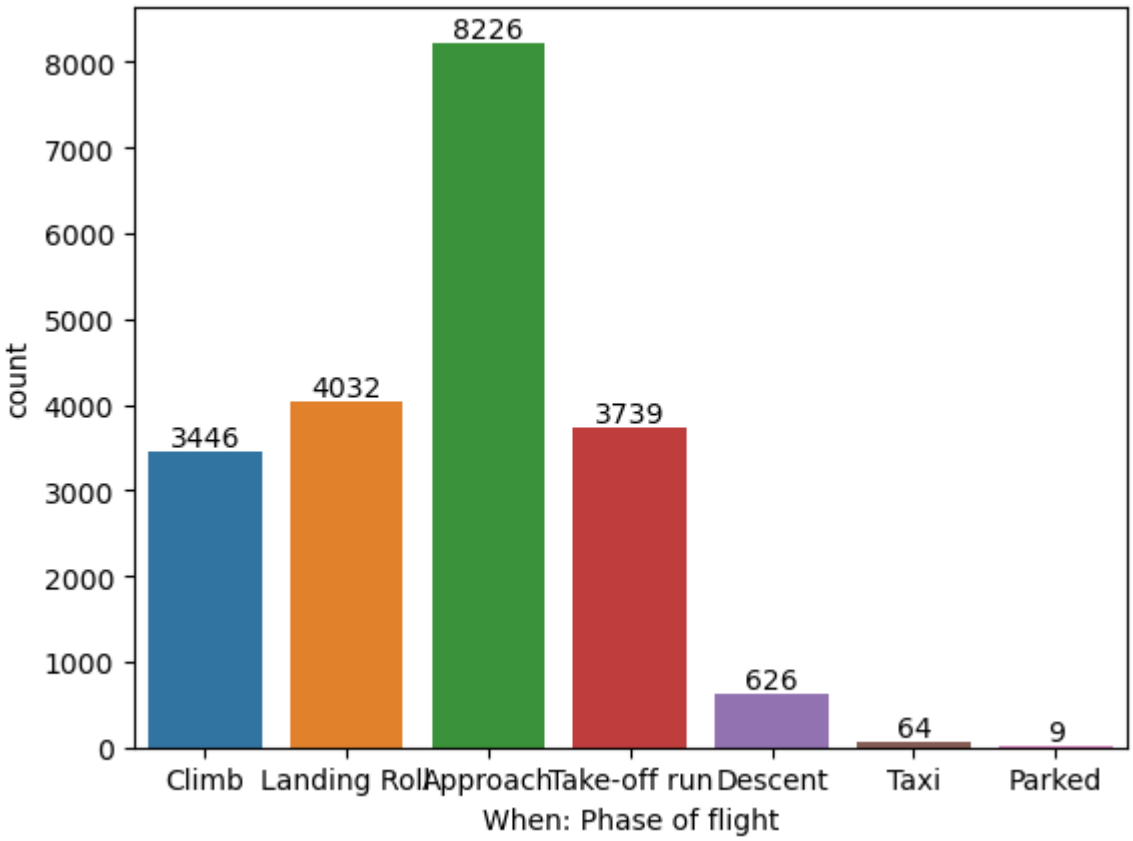
In [19]:
```
Altitude=sns.countplot(x='Altitude bin',data=Bird)
for a in Altitude.containers:
    Altitude.bar_label(a)
```



In [20]:
```
Damage=sns.countplot(x='Effect: Indicated Damage',data=Bird)
for a in Damage.containers:
    Damage.bar_label(a)
```



In [21]:
```
Flight_phase=sns.countplot(x='When: Phase of flight',data=Bird)
for a in Flight_phase.containers:
    Flight_phase.bar_label(a)
```



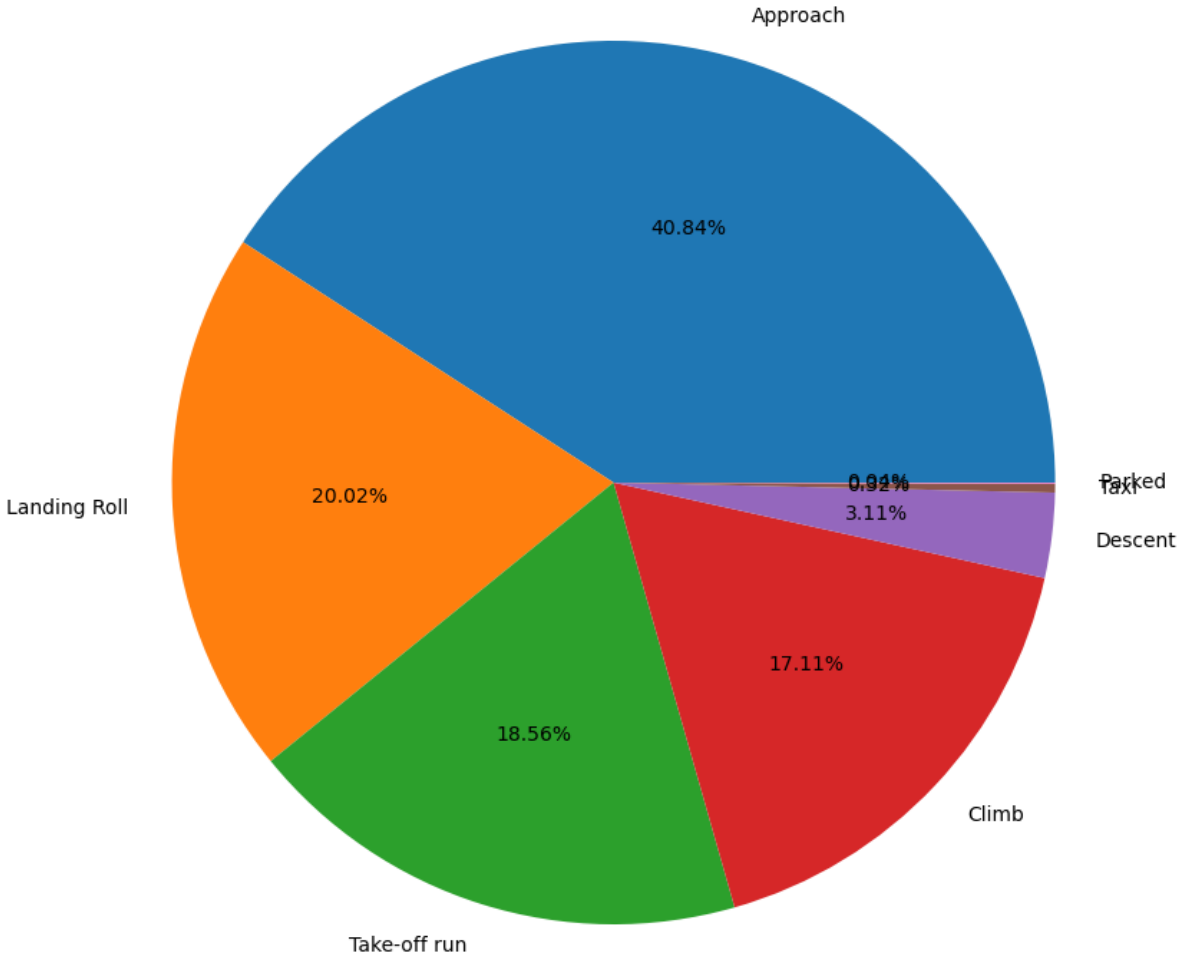
In [22]:
```
data= Bird['When: Phase of flight'].value_counts()
data.index
```

Out[22]:
```
Index(['Approach', 'Landing Roll', 'Take-off run', 'Climb', 'Descent', 'Taxi',
       'Parked'],
      dtype='object')
```

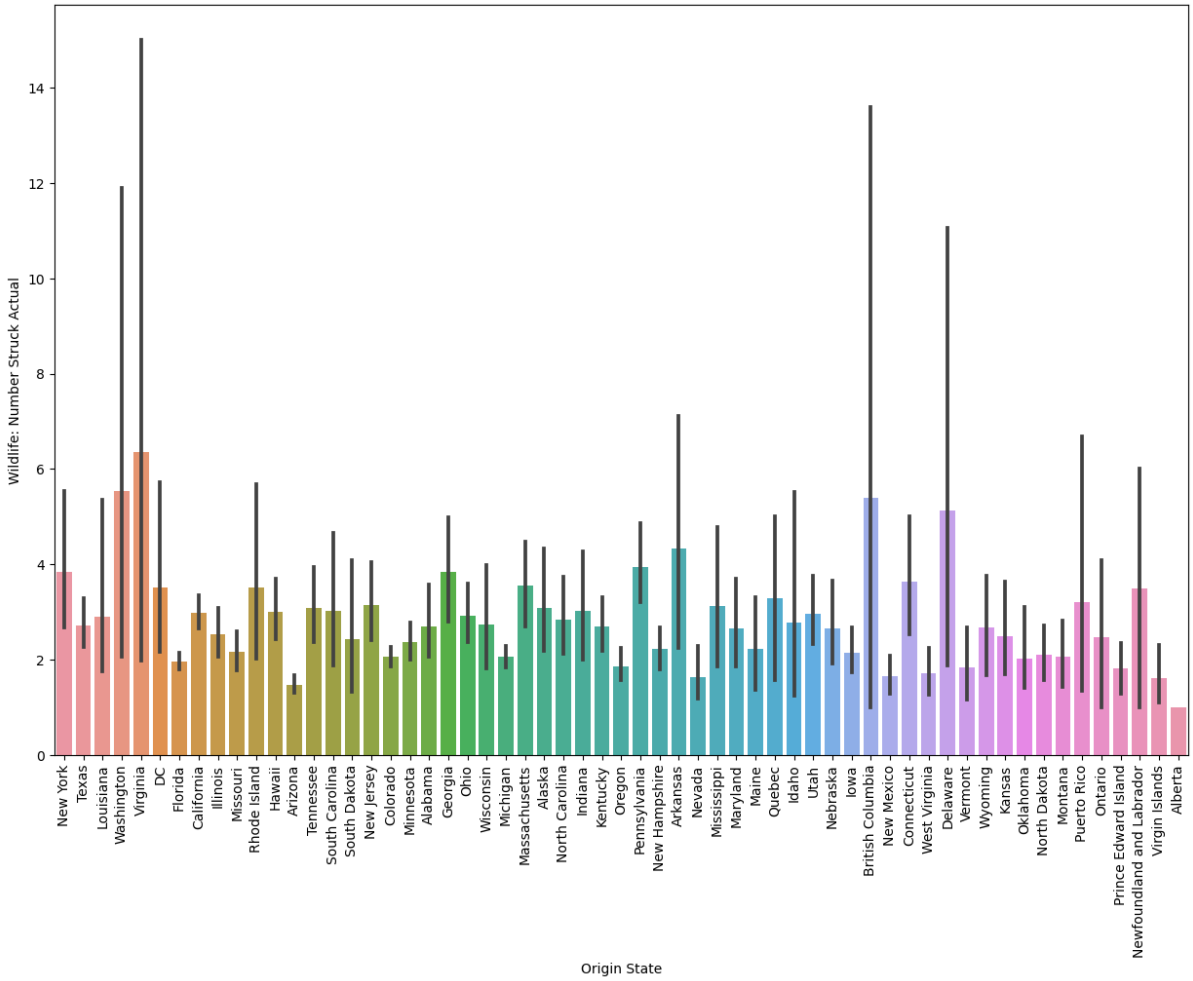
In [23]:
```
data.values
```

Out[23]:
```
array([8226, 4032, 3739, 3446,  626,   64,    9], dtype=int64)
```

In [24]:
```
plt.figure(figsize=(10,10))
plt.pie(data.values, labels= data.index,autopct="%0.2f%%")
plt.show()
```



In [25]:
```
plt.figure(figsize=(15,10))
sns.barplot(data=Bird,x="Origin State",y="Wildlife: Number Struck Actual")
plt.xticks(rotation=90)
plt.show()
```



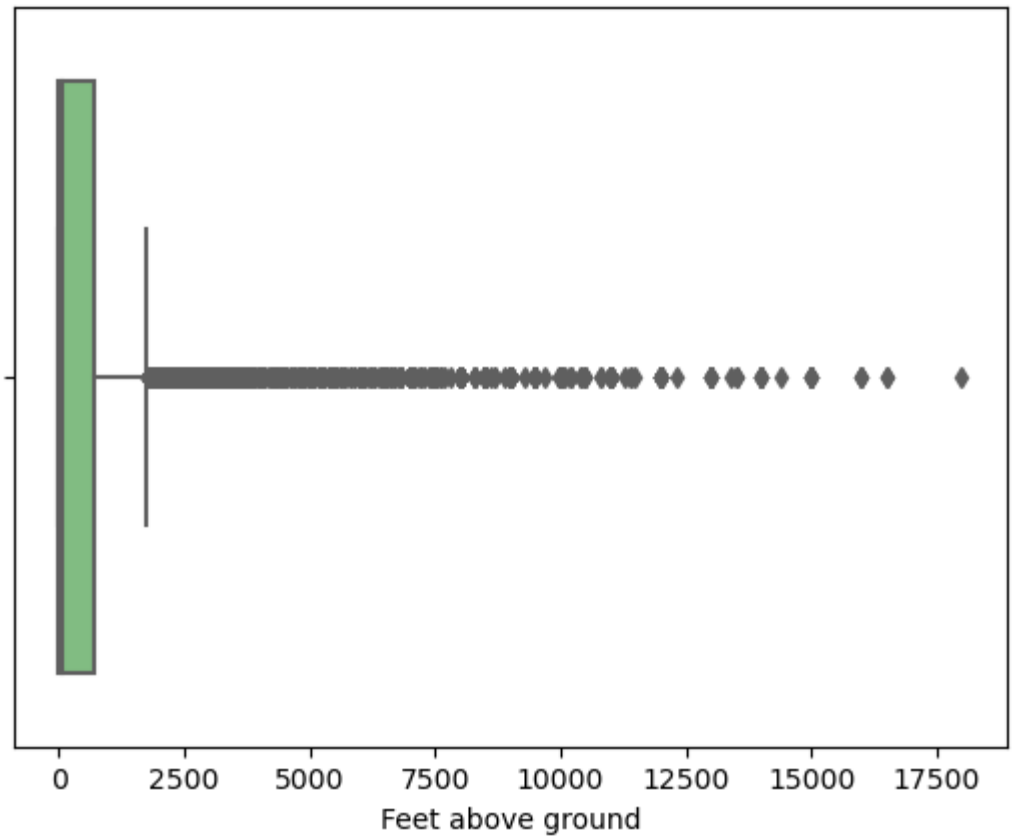
In [26]:
```
plt.figure(figsize=(12,10))
sns.heatmap(Bird.corr(),cmap='plasma',annot=True)
```

```
C:\Users\payal\AppData\Local\Temp\ipykernel_24680\2625734219.py:2: FutureWarning: T
he default value of numeric_only in DataFrame.corr is deprecated. In a future versi
on, it will default to False. Select only valid columns or specify the value of num
eric_only to silence this warning.
  sns.heatmap(Bird.corr(),cmap='plasma',annot=True)
```

Out[26]:
```
<Axes: >
```

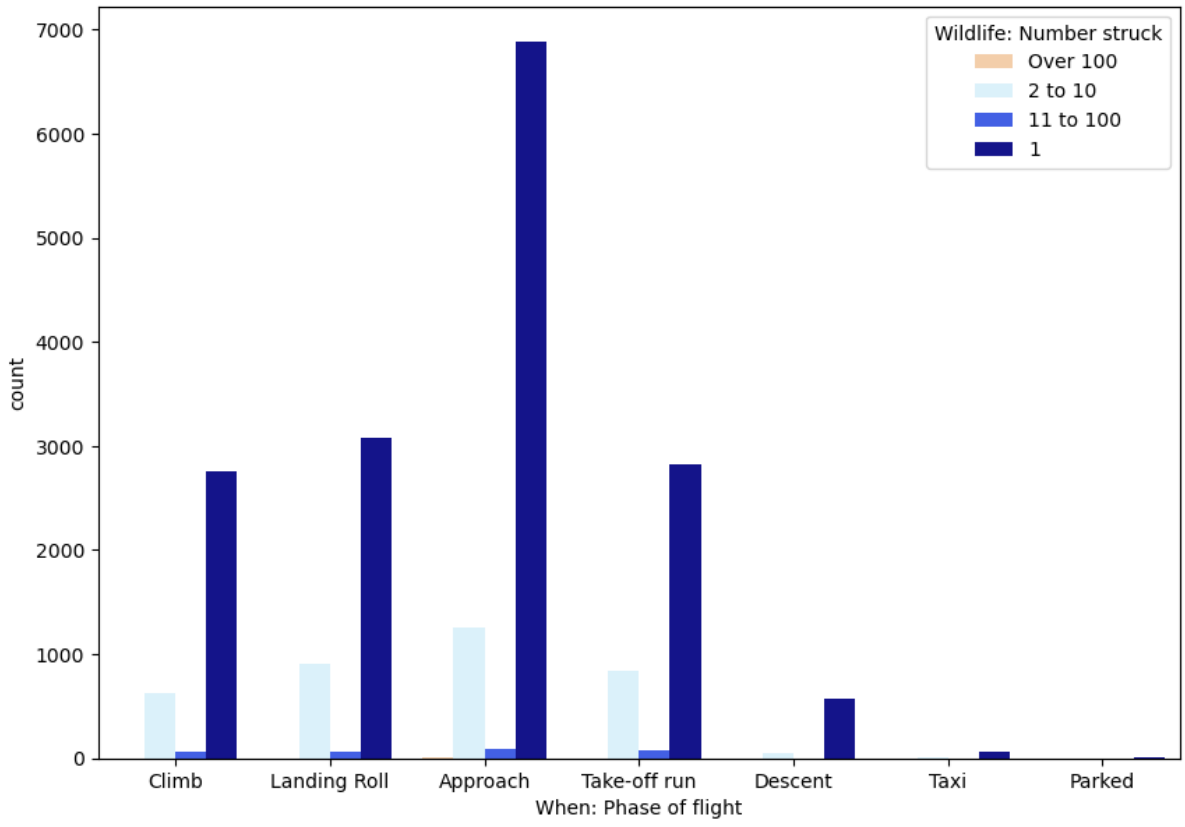
In [27]:
```
#boxplot on mean of feet above ground
sns.boxplot(x=Bird['Feet above ground'],palette='YlGn')
```

Out[27]:
```
<Axes: xlabel='Feet above ground'>
```



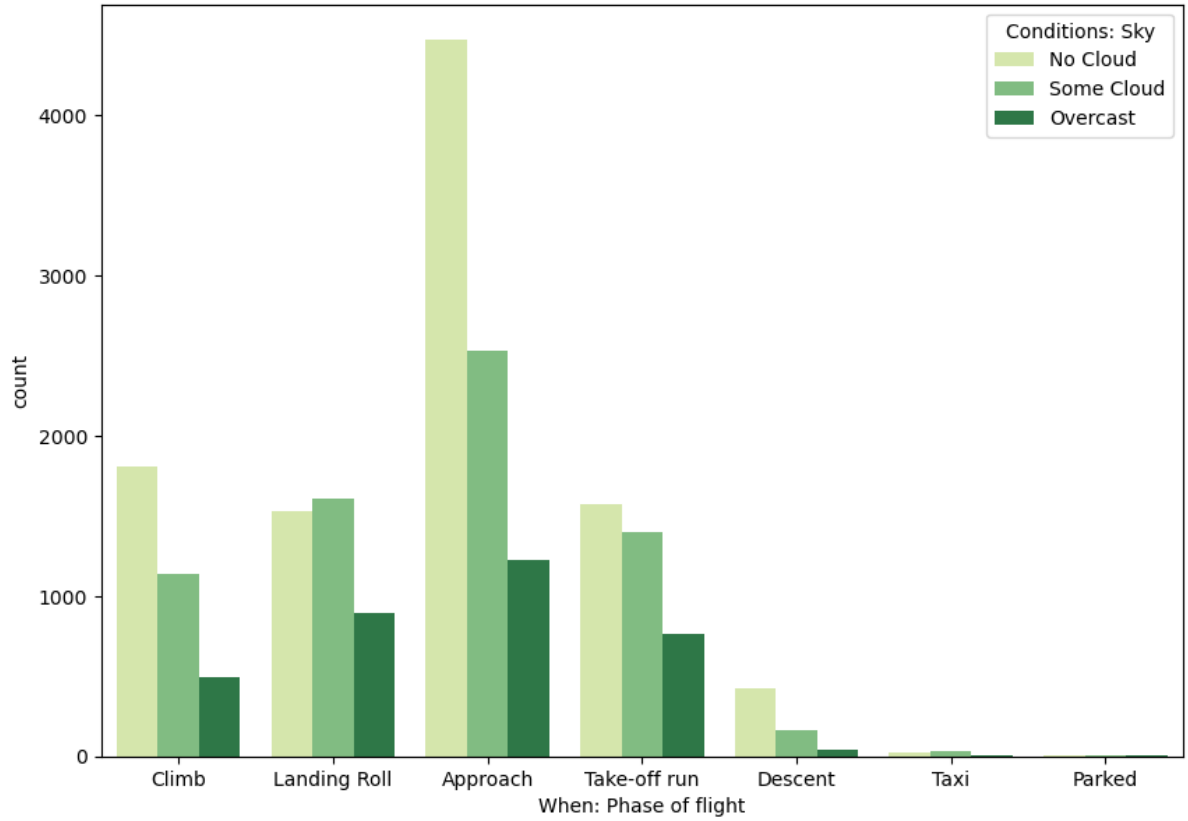
In [28]:
```
plt.figure(figsize=(10,7))
sns.countplot(x='When: Phase of flight',hue ='Wildlife: Number struck',data=Bird, p
```

Out[28]:
```
<Axes: xlabel='When: Phase of flight', ylabel='count'>
```



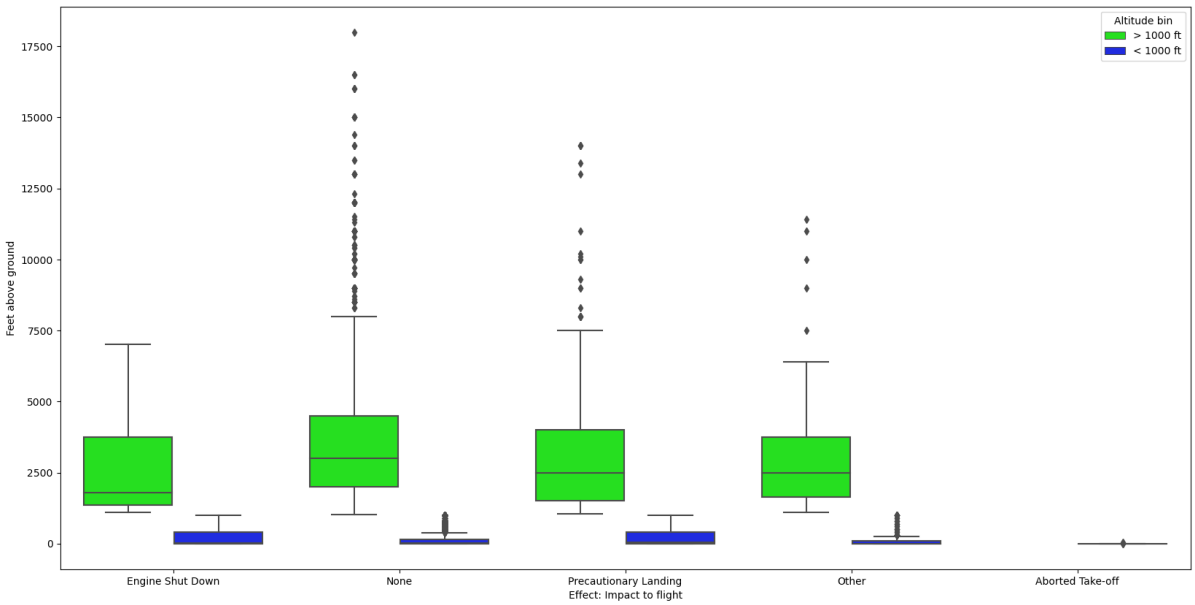
In [29]:
```
plt.figure(figsize=(10,7))
sns.countplot(x='When: Phase of flight',hue ='Conditions: Sky',data=Bird, palette='
```

Out[29]:
```
<Axes: xlabel='When: Phase of flight', ylabel='count'>
```



In [30]:
```
plt.figure(figsize=(20,10))
sns.boxplot(x='Effect: Impact to flight', y='Feet above ground', hue ='Altitude bin
```

Out[30]:
```
<Axes: xlabel='Effect: Impact to flight', ylabel='Feet above ground'>
```



Add year column

In [31]:
```
Bird['Year']=pd.DatetimeIndex(Bird.FlightDate).year
```

In [32]:
```
Bird.head(5)
```

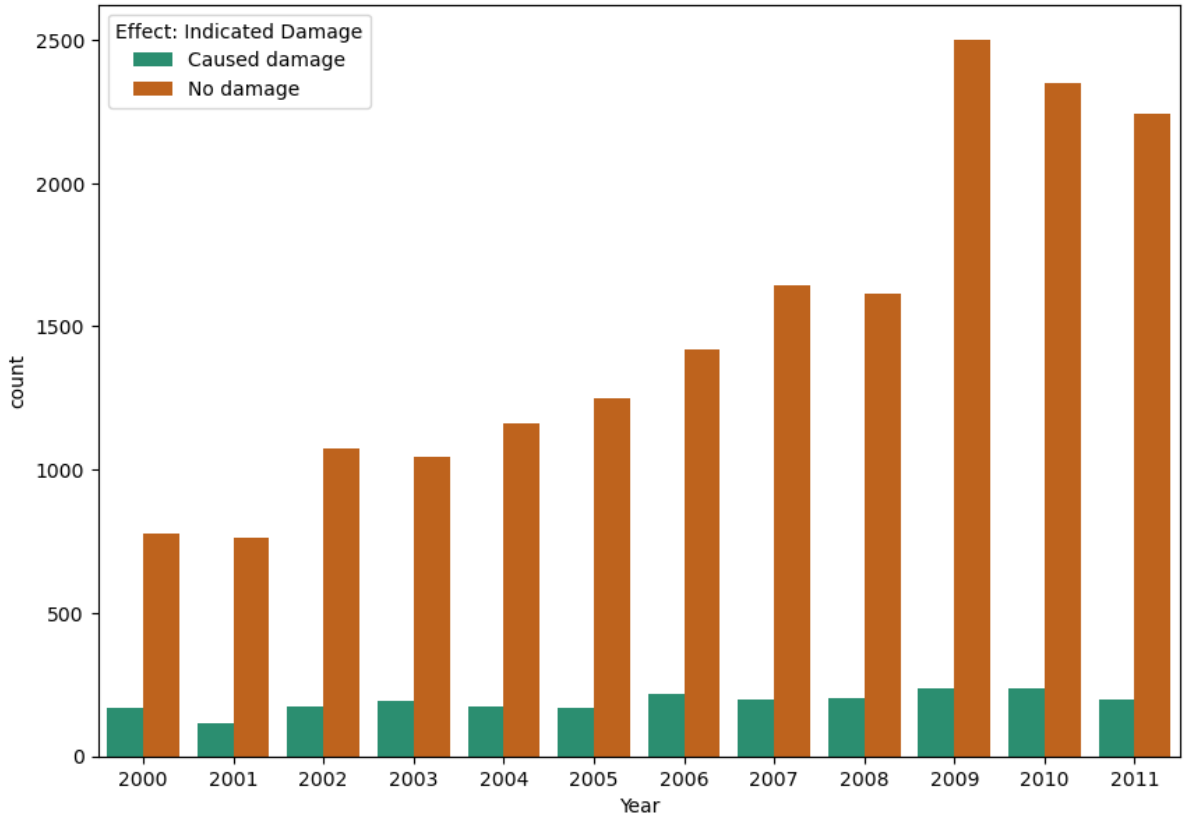
Out[32]:

	Record ID	Aircraft: Type	Airport: Name	Altitude bin	Aircraft: Make/Model	Wildlife: Number struck	Wildlife: Number Struck Actual	Effect: Impact to flight	Flight
0	202152	Airplane	LAGUARDIA NY	> 1000 ft	B-737-400	Over 100	859	Engine Shut Down	2000-
1	208159	Airplane	DALLAS/FORT WORTH INTL ARPT	< 1000 ft	MD-80	Over 100	424	None	2001-(
2	207601	Airplane	LAKEFRONT AIRPORT	< 1000 ft	C-500	Over 100	261	None	2001-(
3	215953	Airplane	SEATTLE-TACOMA INTL	< 1000 ft	B-737-400	Over 100	806	Precautionary Landing	2002-(
4	219878	Airplane	NORFOLK INTL	< 1000 ft	CL-RJ100/200	Over 100	942	None	2003-(

5 rows × 27 columns

In [33]:
```
plt.figure(figsize=(10,7))
sns.countplot(x ='Year',hue='Effect: Indicated Damage',data=Bird, palette='Dark2')
```

Out[33]:
```
<Axes: xlabel='Year', ylabel='count'>
```


Effect: Indicated Damage
Caused damage
No damage
count
Year
0
500
1000
1500
2000
2500
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011

In [ ]:

In [ ]:

In [ ]:

In [ ]: