

```
In [1]: # pandas is a software library written for the Python programming Language for data analysis
import pandas as pd
# NumPy is a library for the Python programming Language, adding support for large arrays and matrices,
import numpy as np
# Matplotlib is a plotting library for python and pyplot gives us a MatLab like plotting library
import matplotlib.pyplot as plt
# Seaborn is a Python data visualization library based on matplotlib. It provides a high-level interface
import seaborn as sns
```

```
In [2]: df=pd.read_csv("https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/CF-CO254302-2021-12/SpaceX/SpaceX.csv")

# If you were unable to complete the previous lab correctly you can uncomment and run the following code

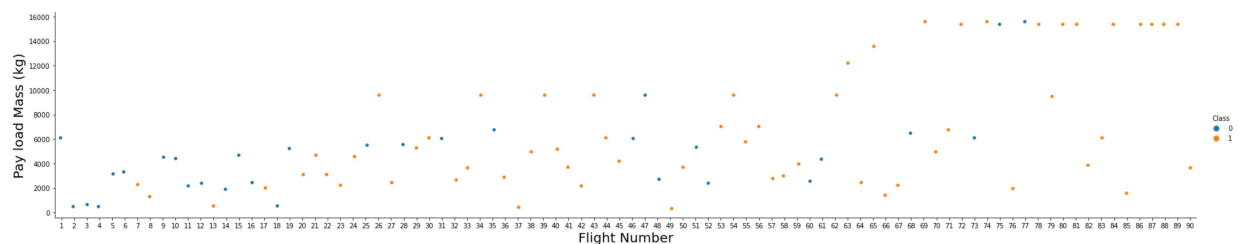
# df = pd.read_csv('https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/CF-CO254302-2021-12/SpaceX/SpaceX.csv')

df.head(5)
```

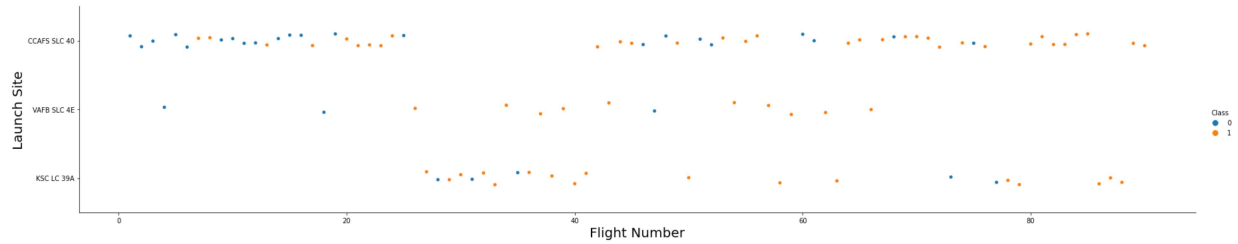
Out[2]:

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFi
0	1	2010-06-04	Falcon 9	6104.959412	LEO	CCAFS SLC 40	None None	1	Fal
1	2	2012-05-22	Falcon 9	525.000000	LEO	CCAFS SLC 40	None None	1	Fal
2	3	2013-03-01	Falcon 9	677.000000	ISS	CCAFS SLC 40	None None	1	Fal
3	4	2013-09-29	Falcon 9	500.000000	PO	VAFB SLC 4E	False Ocean	1	Fal
4	5	2013-12-03	Falcon 9	3170.000000	GTO	CCAFS SLC 40	None None	1	Fal

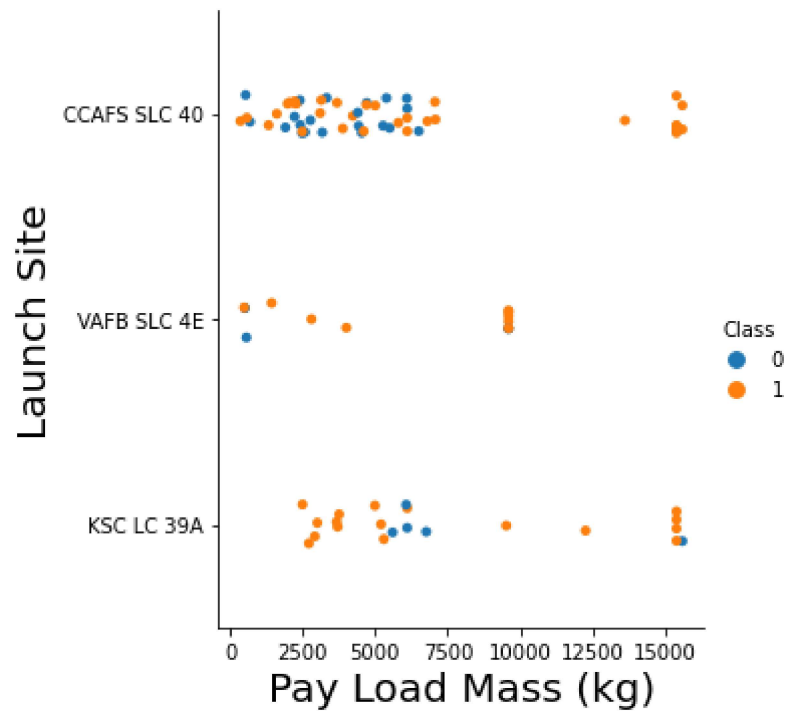
```
In [3]: sns.catplot(y="PayloadMass", x="FlightNumber", hue="Class", data=df, aspect = 5)
plt.xlabel("Flight Number",fontsize=20)
plt.ylabel("Pay load Mass (kg)",fontsize=20)
plt.show()
```



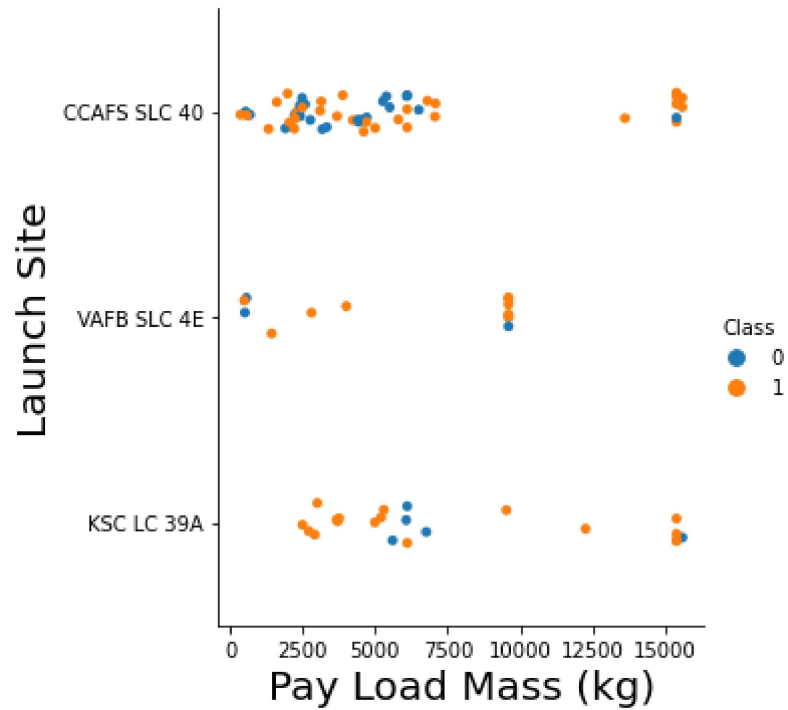
```
In [4]: # Plot a scatter point chart with x axis to be Flight Number and y axis to be the
sns.catplot(y="LaunchSite", x="FlightNumber", hue="Class", data=df, aspect = 5)
plt.xlabel("Flight Number",fontsize=20)
plt.ylabel("Launch Site",fontsize=20)
plt.show()
```



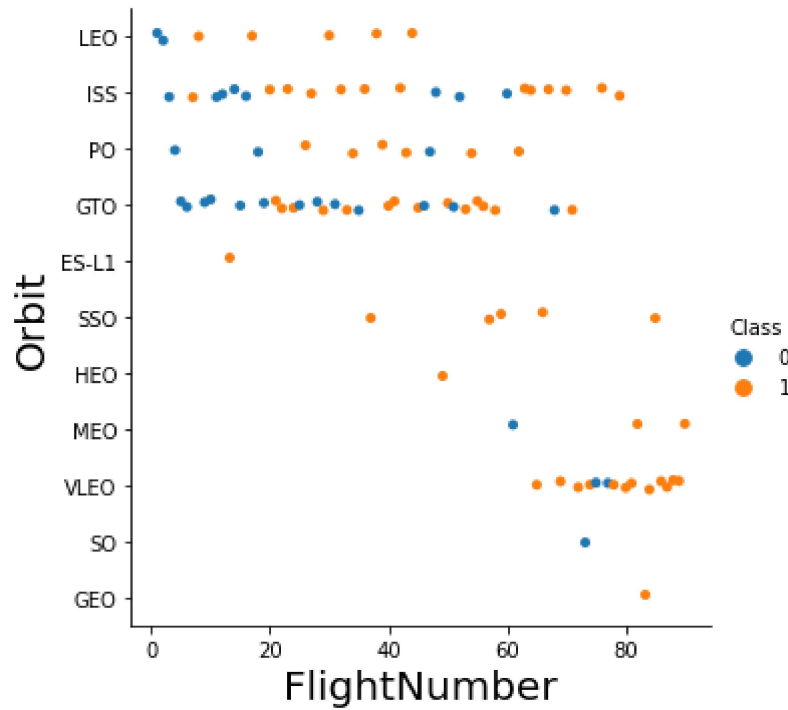
```
In [6]: # Plot a scatter point chart with x axis to be Pay Load Mass (kg) and y axis to be t
sns.catplot(y="LaunchSite", x="PayloadMass", hue="Class", data=df)
plt.xlabel("Pay Load Mass (kg)",fontsize=20)
plt.ylabel("Launch Site",fontsize=20)
plt.show()
```



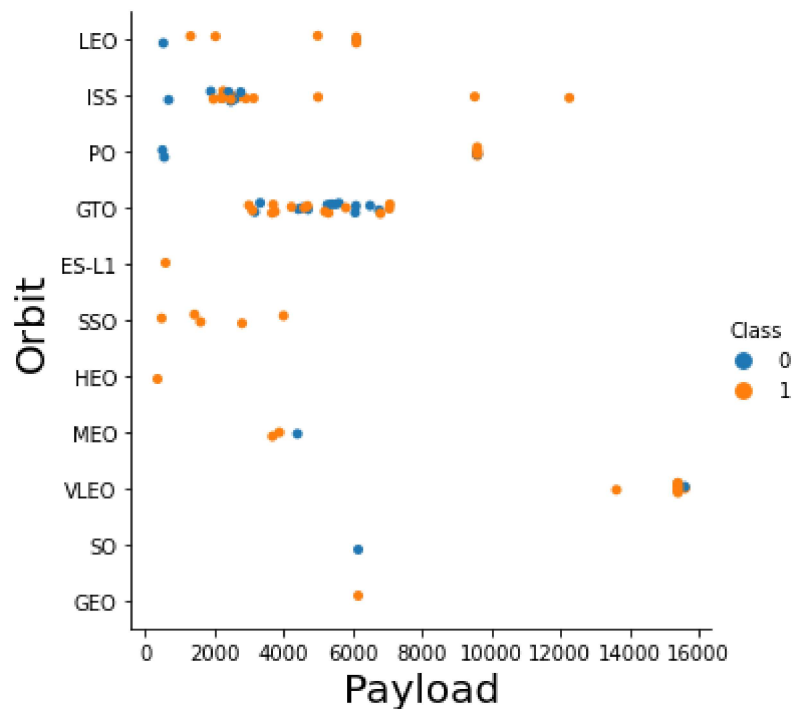
```
In [7]: # HINT use groupby method on Orbit column and get the mean of Class column
sns.catplot(y="LaunchSite", x="PayloadMass", hue="Class", data=df)
plt.xlabel("Pay Load Mass (kg)",fontsize=20)
plt.ylabel("Launch Site",fontsize=20)
plt.show()
```



```
In [8]: # Plot a scatter point chart with x axis to be FlightNumber and y axis to be the
sns.catplot(y="Orbit", x="FlightNumber", hue="Class", data=df)
plt.xlabel("FlightNumber",fontsize=20)
plt.ylabel("Orbit",fontsize=20)
plt.show()
```



```
In [9]: # Plot a scatter point chart with x axis to be Payload and y axis to be the Orbit
sns.catplot(y="Orbit", x="PayloadMass", hue="Class", data=df)
plt.xlabel("Payload", fontsize=20)
plt.ylabel("Orbit", fontsize=20)
plt.show()
```



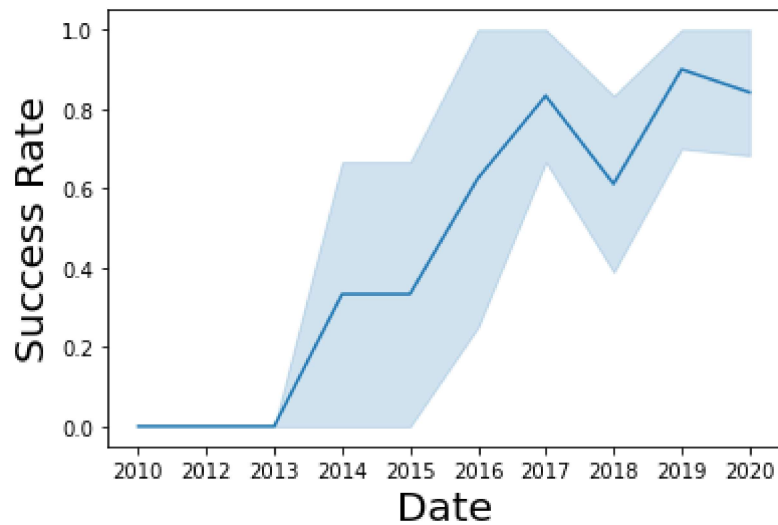
```
In [10]: # A function to Extract years from the date
def Extract_year():
    for i in df["Date"]:
        year.append(i.split("-")[0])
    return year
```

```
In [11]: year=[]
df1 = df.copy()
year = Extract_year()
df1["Date"] = year
df1.head()
```

Out[11]:

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFin
0	1	2010	Falcon 9	6104.959412	LEO	CCAFS SLC 40	None None	1	False
1	2	2012	Falcon 9	525.000000	LEO	CCAFS SLC 40	None None	1	False
2	3	2013	Falcon 9	677.000000	ISS	CCAFS SLC 40	None None	1	False
3	4	2013	Falcon 9	500.000000	PO	VAFB SLC 4E	False Ocean	1	False
4	5	2013	Falcon 9	3170.000000	GTO	CCAFS SLC 40	None None	1	False

```
In [12]: # Plot a line chart with x axis to be the extracted year and y axis to be the success rate
sns.lineplot(data=df1, x="Date", y="Class")
plt.xlabel("Date", fontsize=20)
plt.ylabel("Success Rate", fontsize=20)
plt.show()
```



```
In [13]: features = df[['FlightNumber', 'PayloadMass', 'Orbit', 'LaunchSite', 'Flights', 'GridFins', 'Reused', 'Legs', 'LandingPad', 'Block', 'ReusedCount', 'Orbit_ES-L1']]
features.head()
```

Out[13]:

	FlightNumber	PayloadMass	Orbit	LaunchSite	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Orbit_ES-L1
0	1	6104.959412	LEO	CCAFS SLC 40	1	False	False	False	NaN	1	0	0
1	2	525.000000	LEO	CCAFS SLC 40	1	False	False	False	NaN	1	0	0
2	3	677.000000	ISS	CCAFS SLC 40	1	False	False	False	NaN	1	0	0
3	4	500.000000	PO	VAFB SLC 4E	1	False	False	False	NaN	1	0	0
4	5	3170.000000	GTO	CCAFS SLC 40	1	False	False	False	NaN	1	0	0

```
In [14]: # HINT: Use get_dummies() function on the categorical columns
features_one_hot = pd.get_dummies(features, columns=['Orbit', 'LaunchSite', 'LandingPad'])
features_one_hot.head()
```

Out[14]:

	FlightNumber	PayloadMass	Flights	GridFins	Reused	Legs	Block	ReusedCount	Orbit_ES-L1
0	1	6104.959412	1	False	False	False	1.0	0	0
1	2	525.000000	1	False	False	False	1.0	0	0
2	3	677.000000	1	False	False	False	1.0	0	0
3	4	500.000000	1	False	False	False	1.0	0	0
4	5	3170.000000	1	False	False	False	1.0	0	0

5 rows × 80 columns

```
In [15]: # HINT: use astype function
features_one_hot.astype(float)
```

Out[15]:

	FlightNumber	PayloadMass	Flights	GridFins	Reused	Legs	Block	ReusedCount	Orbit_ES-L1
0	1.0	6104.959412	1.0	0.0	0.0	0.0	1.0	0.0	0.0
1	2.0	525.000000	1.0	0.0	0.0	0.0	1.0	0.0	0.0
2	3.0	677.000000	1.0	0.0	0.0	0.0	1.0	0.0	0.0
3	4.0	500.000000	1.0	0.0	0.0	0.0	1.0	0.0	0.0
4	5.0	3170.000000	1.0	0.0	0.0	0.0	1.0	0.0	0.0
...	...	...	...	...	...	...	...	...	...
85	86.0	15400.000000	2.0	1.0	1.0	1.0	5.0	2.0	0.0
86	87.0	15400.000000	3.0	1.0	1.0	1.0	5.0	2.0	0.0
87	88.0	15400.000000	6.0	1.0	1.0	1.0	5.0	5.0	0.0
88	89.0	15400.000000	3.0	1.0	1.0	1.0	5.0	2.0	0.0
89	90.0	3681.000000	1.0	1.0	0.0	1.0	5.0	0.0	0.0

90 rows × 80 columns

```
In [16]: task_1 = '''
Select Distinct Launchsite
From SpaceX
'''
create_pandas_df(task_1,database=conn)
```

```
-----
NameError                                Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_14152\977993585.py in <module>
      3 From SpaceX
      4 '''
----> 5 create_pandas_df(task_1,database=conn)

NameError: name 'create_pandas_df' is not defined
```



```
In [17]: task_2 = '''
Select *
From SpaceX
Where LaunchSite Like 'CCA'
Limit 5
'''

create_pandas_df(task_2,database=conn)
```

```
-----
NameError                                Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_14152\3081930513.py in <module>
      6 '''
      7
----> 8 create_pandas_df(task_2,database=conn)

NameError: name 'create_pandas_df' is not defined
```

```
In [18]: task_3 = '''

SELECT SUM(payloadmassKG) AS Total_Payloadmass
from SpaceX
Where Customer Like 'NASA(CRS)'
'''

create_pandas_df(task_3,database=conn)
```

```
-----
NameError                                Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_14152\3062920854.py in <module>
      6 '''
      7
----> 8 create_pandas_df(task_3,database=conn)

NameError: name 'create_pandas_df' is not defined
```

```
In [19]: features_one_hot.to_csv('dataset_part_3.csv', index=False)
```