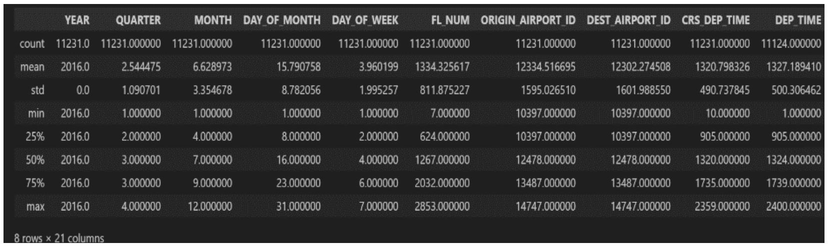
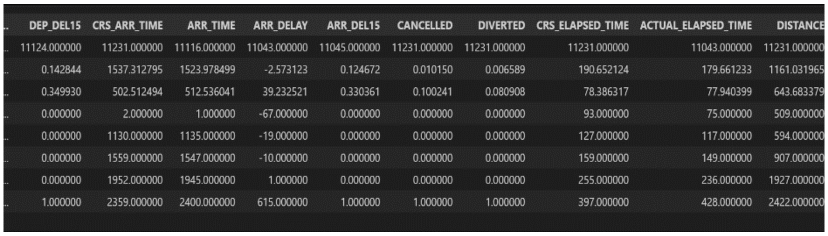
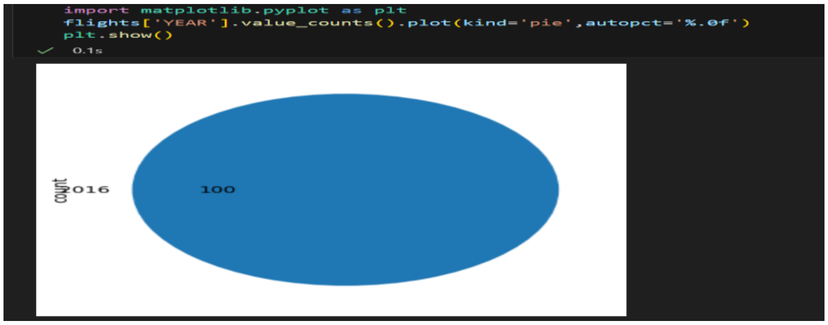
Data Collection and Preprocessing Phase

Date	15 July 2024
Team ID	740080
Project Title	Flight Delay Prediction using Machine Learning.
Maximum Marks	6 Marks

Data Exploration and Preprocessing Template

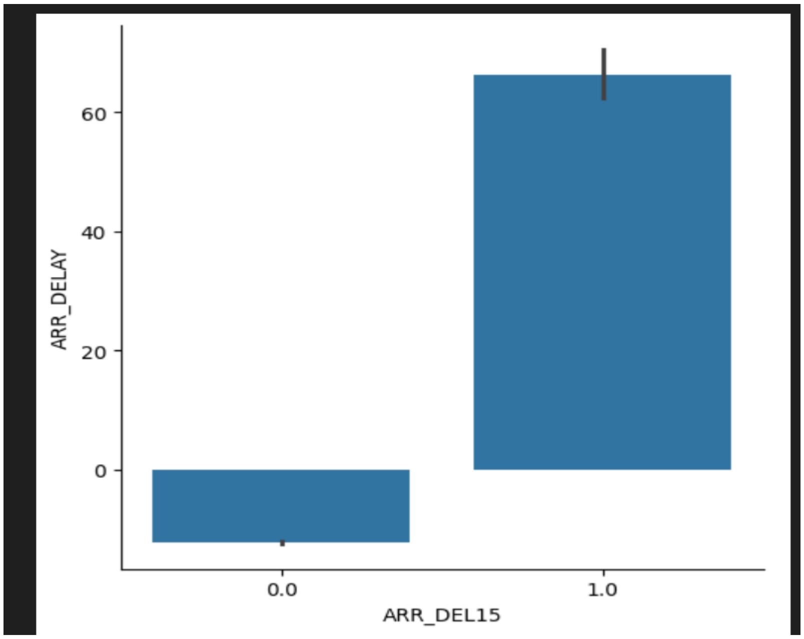
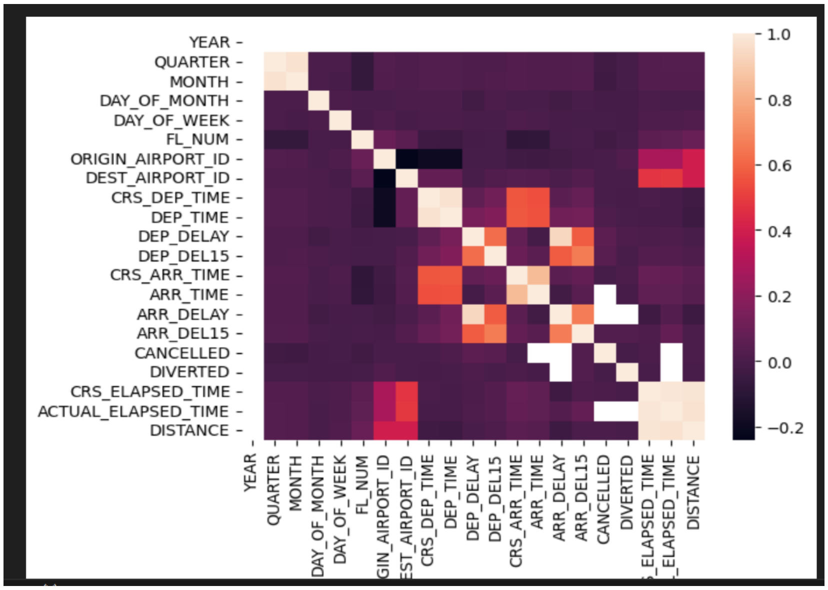
Dataset variables will be statistically analyzed to identify patterns and outliers, with Python employed for preprocessing tasks like normalization and feature engineering. Data cleaning will address missing values and outliers, ensuring quality for subsequent analysis and modeling, and forming a strong foundation for insights and predictions.

<table>
<tr><th>Section</th><th>Description</th></tr>
<tr><td>Data Overview</td><td>

<table>
<tr><th></th><th>YEAR</th><th>QUARTER</th><th>MONTH</th><th>DAY_OF_MONTH</th><th>DAY_OF_WEEK</th><th>FL_NUM</th><th>ORIGIN_AIRPORT_ID</th><th>DEST_AIRPORT_ID</th><th>CRS_DEP_TIME</th><th>DEP_TIME</th></tr>
<tr><td>count</td><td>11231.0</td><td>11231.000000</td><td>11231.000000</td><td>11231.000000</td><td>11231.000000</td><td>11231.000000</td><td>11231.000000</td><td>11231.000000</td><td>11231.000000</td><td>11124.000000</td></tr>
<tr><td>mean</td><td>2016.0</td><td>2.544475</td><td>6.628973</td><td>15.790758</td><td>3.960199</td><td>1334.325617</td><td>12334.516695</td><td>12302.274508</td><td>1320.798326</td><td>1327.189410</td></tr>
<tr><td>std</td><td>0.0</td><td>1.090701</td><td>3.354678</td><td>8.782056</td><td>1.995257</td><td>811.875227</td><td>1595.026510</td><td>1601.988550</td><td>490.737845</td><td>500.306462</td></tr>
<tr><td>min</td><td>2016.0</td><td>1.000000</td><td>1.000000</td><td>1.000000</td><td>1.000000</td><td>7.000000</td><td>10397.000000</td><td>10397.000000</td><td>10.000000</td><td>1.000000</td></tr>
<tr><td>25%</td><td>2016.0</td><td>2.000000</td><td>4.000000</td><td>8.000000</td><td>2.000000</td><td>624.000000</td><td>10397.000000</td><td>10397.000000</td><td>905.000000</td><td>905.000000</td></tr>
<tr><td>50%</td><td>2016.0</td><td>3.000000</td><td>7.000000</td><td>16.000000</td><td>4.000000</td><td>1267.000000</td><td>12478.000000</td><td>12478.000000</td><td>1320.000000</td><td>1324.000000</td></tr>
<tr><td>75%</td><td>2016.0</td><td>3.000000</td><td>9.000000</td><td>23.000000</td><td>6.000000</td><td>2032.000000</td><td>13487.000000</td><td>13487.000000</td><td>1735.000000</td><td>1739.000000</td></tr>
<tr><td>max</td><td>2016.0</td><td>4.000000</td><td>12.000000</td><td>31.000000</td><td>7.000000</td><td>2853.000000</td><td>14747.000000</td><td>14747.000000</td><td>2359.000000</td><td>2400.000000</td></tr>
</table>
8 rows × 21 columns

<table>
<tr><th>...</th><th>DEP_DEL15</th><th>CRS_ARR_TIME</th><th>ARR_TIME</th><th>ARR_DELAY</th><th>ARR_DEL15</th><th>CANCELLED</th><th>DIVERTED</th><th>CRS_ELAPSED_TIME</th><th>ACTUAL_ELAPSED_TIME</th><th>DISTANCE</th></tr>
<tr><td>...</td><td>11124.000000</td><td>11231.000000</td><td>11116.000000</td><td>11043.000000</td><td>11045.000000</td><td>11231.000000</td><td>11231.000000</td><td>11231.000000</td><td>11043.000000</td><td>11231.000000</td></tr>
<tr><td>...</td><td>0.142844</td><td>1537.312795</td><td>1523.978499</td><td>-2.573123</td><td>0.124672</td><td>0.010150</td><td>0.006589</td><td>190.652124</td><td>179.661233</td><td>1161.031965</td></tr>
<tr><td>...</td><td>0.349930</td><td>502.512494</td><td>512.536041</td><td>39.232521</td><td>0.330361</td><td>0.100241</td><td>0.080908</td><td>78.386317</td><td>77.940399</td><td>643.683379</td></tr>
<tr><td>...</td><td>0.000000</td><td>2.000000</td><td>1.000000</td><td>-67.000000</td><td>0.000000</td><td>0.000000</td><td>0.000000</td><td>93.000000</td><td>75.000000</td><td>509.000000</td></tr>
<tr><td>...</td><td>0.000000</td><td>1130.000000</td><td>1135.000000</td><td>-19.000000</td><td>0.000000</td><td>0.000000</td><td>0.000000</td><td>127.000000</td><td>117.000000</td><td>594.000000</td></tr>
<tr><td>...</td><td>0.000000</td><td>1559.000000</td><td>1547.000000</td><td>-10.000000</td><td>0.000000</td><td>0.000000</td><td>0.000000</td><td>159.000000</td><td>149.000000</td><td>907.000000</td></tr>
<tr><td>...</td><td>0.000000</td><td>1952.000000</td><td>1945.000000</td><td>1.000000</td><td>0.000000</td><td>0.000000</td><td>0.000000</td><td>255.000000</td><td>236.000000</td><td>1927.000000</td></tr>
<tr><td>...</td><td>1.000000</td><td>2359.000000</td><td>2400.000000</td><td>615.000000</td><td>1.000000</td><td>1.000000</td><td>1.000000</td><td>397.000000</td><td>428.000000</td><td>2422.000000</td></tr>
</table>
</td></tr>
<tr><td>Univariate Analysis</td><td>


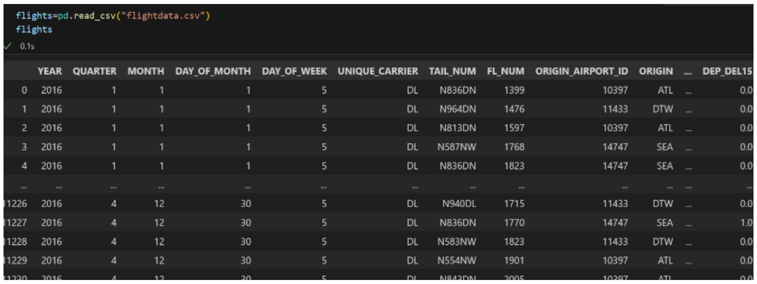
```
import matplotlib.pyplot as plt
flights['YEAR'].value_counts().plot(kind='pie',autopct='%.0f')
plt.show()
```

✓ 0.1s

</td></tr>
</table>

Bivariate Analysis	
Multivariate Analysis	
Outliers and Anomalies	
Data Preprocessing Code Screenshots	

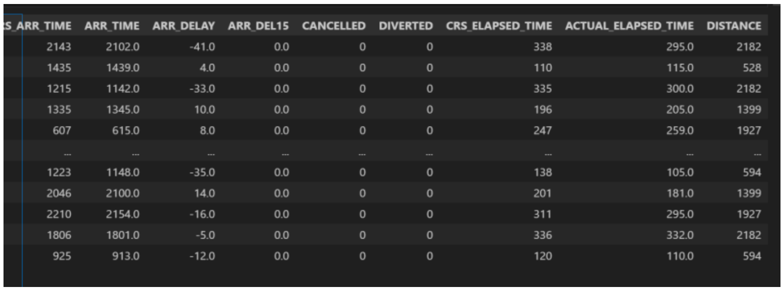
Loading Data



```
flights=pd.read_csv("flightdata.csv")
flights
```

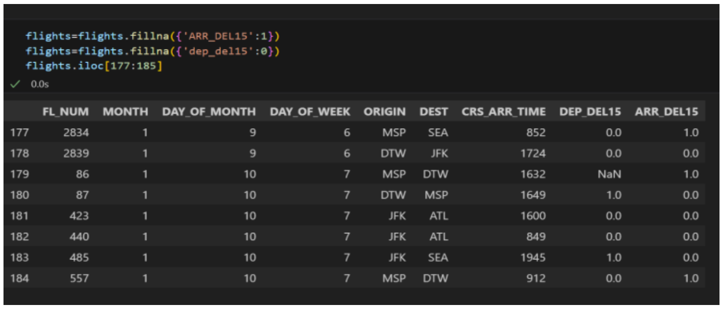
✓ 0.1s

	YEAR	QUARTER	MONTH	DAY_OF_MONTH	DAY_OF_WEEK	UNIQUE_CARRIER	TAIL_NUM	FL_NUM	ORIGIN_AIRPORT_ID	ORIGIN	...	DEP_DEL15
0	2016	1	1	1	5	DL	N836DN	1399	10397	ATL	...	0.0
1	2016	1	1	1	5	DL	N964DN	1476	11433	DTW	...	0.0
2	2016	1	1	1	5	DL	N813DN	1597	10397	ATL	...	0.0
3	2016	1	1	1	5	DL	N587NW	1768	14747	SEA	...	0.0
4	2016	1	1	1	5	DL	N836DN	1823	14747	SEA	...	0.0
...	...	...	...	...	...	...	...	...	...	...	...	...
11226	2016	4	12	30	5	DL	N940DL	1715	11433	DTW	...	0.0
11227	2016	4	12	30	5	DL	N836DN	1770	14747	SEA	...	1.0
11228	2016	4	12	30	5	DL	N583NW	1823	11433	DTW	...	0.0
11229	2016	4	12	30	5	DL	N554NW	1901	10397	ATL	...	0.0
11230	2016	4	12	30	5	DL	N843DN	2005	10397	ATL		0.0



S_ARR_TIME	ARR_TIME	ARR_DELAY	ARR_DEL15	CANCELLED	DIVERTED	CRS_ELAPSED_TIME	ACTUAL_ELAPSED_TIME	DISTANCE
2143	2102.0	-41.0	0.0	0	0	338	295.0	2182
1435	1439.0	4.0	0.0	0	0	110	115.0	528
1215	1142.0	-33.0	0.0	0	0	335	300.0	2182
1335	1345.0	10.0	0.0	0	0	196	205.0	1399
607	615.0	8.0	0.0	0	0	247	259.0	1927
...	...	...	...	...	...	...	...	...
1223	1148.0	-35.0	0.0	0	0	138	105.0	594
2046	2100.0	14.0	0.0	0	0	201	181.0	1399
2210	2154.0	-16.0	0.0	0	0	311	295.0	1927
1806	1801.0	-5.0	0.0	0	0	336	332.0	2182
925	913.0	-12.0	0.0	0	0	120	110.0	594

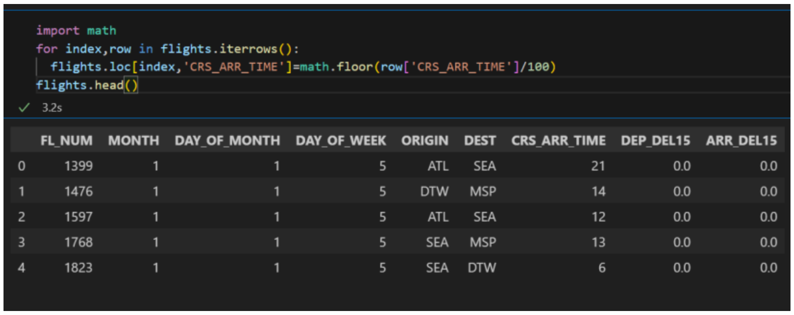
Handling Missing Data



```
flights=flights.fillna({'ARR_DEL15':1})
flights=flights.fillna({'dep_del15':0})
flights.iloc[177:185]
```

✓ 0.0s

	FL_NUM	MONTH	DAY_OF_MONTH	DAY_OF_WEEK	ORIGIN	DEST	CRS_ARR_TIME	DEP_DEL15	ARR_DEL15
177	2834	1	9	6	MSP	SEA	852	0.0	1.0
178	2839	1	9	6	DTW	JFK	1724	0.0	0.0
179	86	1	10	7	MSP	DTW	1632	NaN	1.0
180	87	1	10	7	DTW	MSP	1649	1.0	0.0
181	423	1	10	7	JFK	ATL	1600	0.0	0.0
182	440	1	10	7	JFK	ATL	849	0.0	0.0
183	485	1	10	7	JFK	SEA	1945	1.0	0.0
184	557	1	10	7	MSP	DTW	912	0.0	1.0

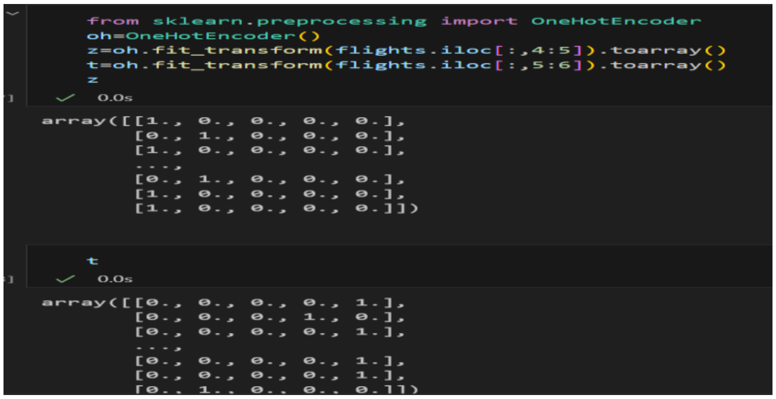


```
import math
for index,row in flights.iterrows():
  flights.loc[index,'CRS_ARR_TIME']=math.floor(row['CRS_ARR_TIME']/100)
flights.head()
```

✓ 3.2s

	FL_NUM	MONTH	DAY_OF_MONTH	DAY_OF_WEEK	ORIGIN	DEST	CRS_ARR_TIME	DEP_DEL15	ARR_DEL15
0	1399	1	1	5	ATL	SEA	21	0.0	0.0
1	1476	1	1	5	DTW	MSP	14	0.0	0.0
2	1597	1	1	5	ATL	SEA	12	0.0	0.0
3	1768	1	1	5	SEA	MSP	13	0.0	0.0
4	1823	1	1	5	SEA	DTW	6	0.0	0.0

<table>
<tr>
<td>Data Transformation</td>
<td>



```
from sklearn.preprocessing import OneHotEncoder
oh=OneHotEncoder()
z=oh.fit_transform(flights.iloc[:,4:5]).toarray()
t=oh.fit_transform(flights.iloc[:,5:6]).toarray()
z
 ✓  0.0s

array([[1., 0., 0., 0., 0.],
       [0., 1., 0., 0., 0.],
       [1., 0., 0., 0., 0.],
       ...,
       [0., 1., 0., 0., 0.],
       [1., 0., 0., 0., 0.],
       [1., 0., 0., 0., 0.]])

t
 ✓  0.0s

array([[0., 0., 0., 0., 1.],
       [0., 0., 0., 1., 0.],
       [0., 0., 0., 0., 1.],
       ...,
       [0., 0., 0., 0., 1.],
       [0., 0., 0., 0., 1.],
       [0., 1., 0., 0., 0.]])
```

</td>
</tr>
<tr>
<td>Feature Engineering</td>
<td>Attached the codes in the final submission.</td>
</tr>
<tr>
<td>Save Processed Data</td>
<td>_</td>
</tr>
</table>