This notebook contains the combined Flight and Weather Dataframe required to train the model

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

import necessary libraries

```
import pandas as pd
from sklearn.preprocessing import LabelEncoder
import seaborn as sns
```

```
df1=pd.read_csv('/content/drive/MyDrive/ML Project/flightinfo-3.csv')
```

```
df2=pd.read_csv('/content/drive/MyDrive/ML Project/WeatherDataF.csv')
```

```
df1
```

	Unnamed: 0	Date (MM/DD/YYYY)	Flight Number	Destination Airport	Scheduled elapsed time (Minutes)	Actual elapsed time (Minutes)	Departure delay (Minutes)	W
0	0	2/11/22	1094.0	SYR	109.0	109.0	29.0	
1	1	2/12/22	1094.0	SYR	109.0	98.0	59.0	
2	2	2/13/22	1094.0	SYR	109.0	104.0	-2.0	
3	3	2/14/22	1094.0	SYR	109.0	99.0	-2.0	
4	4	2/15/22	1094.0	SYR	109.0	105.0	-3.0	
...	...	...	...	...	...	...	...	
1017	1017	10/25/22	2198.0	SYR	71.0	67.0	-7.0	
1018	1018	10/26/22	2198.0	SYR	71.0	67.0	-4.0	
1019	1019	10/27/22	2198.0	SYR	71.0	68.0	-6.0	
1020	1020	10/28/22	2198.0	SYR	71.0	63.0	-13.0	
1021	1021	10/29/22	2198.0	SYR	71.0	66.0	-10.0	

1022 rows × 21 columns

```
df2
```

	Unnamed: 0	Origin Airport	tempmax	tempmin	temp	feelslikemax	feelslikemin	feelslike	dew
0	9	IAD	6.7	-1.7	1.6	2.4	-8.3	-3.6	-9.8
1	10	IAD	-1.8	-5.0	-3.5	-5.3	-12.7	-8.5	-15.0
2	11	IAD	8.3	-5.1	1.4	5.4	-10.5	-2.5	-8.5

Merge the data on Origin Airport and Date

```
df=pd.merge(df1,df2,on=['Origin Airport','day','month','year'])
```

```
df
```

	Unnamed: 0_x	Date (MM/DD/YYYY)	Flight Number	Destination Airport	Scheduled elapsed time (Minutes)	Actual elapsed time (Minutes)	Departure delay (Minutes)	Wheels-off time
0	0	2/11/22	1094.0	SYR	109.0	109.0	29.0	19:06
1	1	2/12/22	1094.0	SYR	109.0	98.0	59.0	19:25
2	2	2/13/22	1094.0	SYR	109.0	104.0	-2.0	18:26
3	3	2/14/22	1094.0	SYR	109.0	99.0	-2.0	18:24
4	4	2/15/22	1094.0	SYR	109.0	105.0	-3.0	18:36
...	...	...	...	...	...	...	...	...
1008	1017	10/25/22	2198.0	SYR	71.0	67.0	-7.0	22:28
1009	1018	10/26/22	2198.0	SYR	71.0	67.0	-4.0	22:30
1010	1019	10/27/22	2198.0	SYR	71.0	68.0	-6.0	22:31
1011	1020	10/28/22	2198.0	SYR	71.0	63.0	-13.0	22:25
1012	1021	10/29/22	2198.0	SYR	71.0	66.0	-10.0	22:25

1013 rows × 47 columns

```
df.columns
```

```
Index(['Unnamed: 0_x', 'Date (MM/DD/YYYY)', 'Flight Number',
       'Destination Airport', 'Scheduled elapsed time (Minutes)',
       'Actual elapsed time (Minutes)', 'Departure delay (Minutes)',
       'Wheels-off time', 'Taxi-Out time (Minutes)',
       'Delay Carrier (Minutes)_x', 'Delay Weather (Minutes)_x',
       'Delay National Aviation System (Minutes)_x',
       'Delay Security (Minutes)_x', 'Delay Late Aircraft Arrival (Minutes)_x',
       'Origin Airport', 'Arrival Delay (Minutes)', 'Wheels-on Time',
       'Taxi-In time (Minutes)', 'month', 'day', 'year', 'Unnamed: 0_y',
       'tempmax', 'tempmin', 'temp', 'feelslikemax', 'feelslikemin',
       'feelslike', 'dew', 'humidity', 'precip', 'precipprob', 'precipcover',
       'snow', 'snowdepth', 'windgust', 'windspeed', 'winddir',
       'sealevelpressure', 'cloudcover', 'visibility', 'solarradiation',
       'solarenergy', 'uvindex', 'severerisk', 'moonphase', 'icon'],
      dtype='object')
```

Perform label encoding on the columns which have string datatype

```
le=LabelEncoder()
```

```
df['Origin Airport'] = le.fit_transform(df['Origin Airport'])
df['Destination Airport'] = le.fit_transform(df['Destination Airport'])
```

```
df['icon'] = le.fit_transform(df['icon'])
```

```
df
```

	Unnamed: 0_x	Date (MM/DD/YYYY)	Flight Number	Destination Airport	Scheduled elapsed time (Minutes)	Actual elapsed time (Minutes)	Departure delay (Minutes)	Wheels-off time
0	0	2/11/22	1094.0	0	109.0	109.0	29.0	19:06
1	1	2/12/22	1094.0	0	109.0	98.0	59.0	19:25
2	2	2/13/22	1094.0	0	109.0	104.0	-2.0	18:26
3	3	2/14/22	1094.0	0	109.0	99.0	-2.0	18:24
4	4	2/15/22	1094.0	0	109.0	105.0	-3.0	18:36
...	...	...	...	...	...	...	...	...
1008	1017	10/25/22	2198.0	0	71.0	67.0	-7.0	22:28
1009	1018	10/26/22	2198.0	0	71.0	67.0	-4.0	22:30
1010	1019	10/27/22	2198.0	0	71.0	68.0	-6.0	22:31
1011	1020	10/28/22	2198.0	0	71.0	63.0	-13.0	22:25
1012	1021	10/29/22	2198.0	0	71.0	66.0	-10.0	22:25

1013 rows × 47 columns

Convert into a csv file

```
df.to_csv('Result.csv')
```