```
# Flight Price Prediction(EDA)
```

In [97]:
```
#importing basics libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

In [98]:
```
train_df=pd.read_excel('Data_Train.xlsx')
train_df.head()
```

Out[98]:

	Airline	Date_of_Journey	Source	Destination	Route	Dep_Time	Arrival_Time	Duration	Total_Stops	Additional_Info	Price
0	IndiGo	24/03/2019	Banglore	New Delhi	BLR → DEL	22:20	01:10 22 Mar	2h 50m	non-stop	No info	3897
1	Air India	1/05/2019	Kolkata	Banglore	CCU → IXR → BBI → BLR	05:50	13:15	7h 25m	2 stops	No info	7662
2	Jet Airways	9/06/2019	Delhi	Cochin	DEL → LKO → BOM → COK	09:25	04:25 10 Jun	19h	2 stops	No info	13882
3	IndiGo	12/05/2019	Kolkata	Banglore	CCU → NAG → BLR	18:05	23:30	5h 25m	1 stop	No info	6218
4	IndiGo	01/03/2019	Banglore	New Delhi	BLR → NAG → DEL	16:50	21:35	4h 45m	1 stop	No info	13302

In [99]:
```
test_df=pd.read_excel('Test_set.xlsx')
test_df.head()
```

Out[99]:

	Airline	Date_of_Journey	Source	Destination	Route	Dep_Time	Arrival_Time	Duration	Total_Stops	Additional_Info
0	Jet Airways	6/06/2019	Delhi	Cochin	DEL → BOM → COK	17:30	04:25 07 Jun	10h 55m	1 stop	No info
1	IndiGo	12/05/2019	Kolkata	Banglore	CCU → MAA → BLR	06:20	10:20	4h	1 stop	No info
2	Jet Airways	21/05/2019	Delhi	Cochin	DEL → BOM → COK	19:15	19:00 22 May	23h 45m	1 stop	In-flight meal not included
3	Multiple carriers	21/05/2019	Delhi	Cochin	DEL → BOM → COK	08:00	21:00	13h	1 stop	No info
4	Air Asia	24/06/2019	Banglore	Delhi	BLR → DEL	23:55	02:45 25 Jun	2h 50m	non-stop	No info

In [100]:
```
final_df=train_df.append(test_df)
final_df.head()
```

```
C:\Users\SAI\AppData\Local\Temp\ipykernel_15776\2632932177.py:1: FutureWarning: The frame.append method is deprecated and will
be removed from pandas in a future version. Use pandas.concat instead.
  final_df=train_df.append(test_df)
```

Out[100]:

	Airline	Date_of_Journey	Source	Destination	Route	Dep_Time	Arrival_Time	Duration	Total_Stops	Additional_Info	Price
0	IndiGo	24/03/2019	Banglore	New Delhi	BLR → DEL	22:20	01:10 22 Mar	2h 50m	non-stop	No info	3897.0
1	Air India	1/05/2019	Kolkata	Banglore	CCU → IXR → BBI → BLR	05:50	13:15	7h 25m	2 stops	No info	7662.0
2	Jet Airways	9/06/2019	Delhi	Cochin	DEL → LKO → BOM → COK	09:25	04:25 10 Jun	19h	2 stops	No info	13882.0
3	IndiGo	12/05/2019	Kolkata	Banglore	CCU → NAG → BLR	18:05	23:30	5h 25m	1 stop	No info	6218.0
4	IndiGo	01/03/2019	Banglore	New Delhi	BLR → NAG → DEL	16:50	21:35	4h 45m	1 stop	No info	13302.0

In [101]:
```
final_df.tail()
```

Out[101]:

	Airline	Date_of_Journey	Source	Destination	Route	Dep_Time	Arrival_Time	Duration	Total_Stops	Additional_Info	Price
2666	Air India	6/06/2019	Kolkata	Banglore	CCU → DEL → BLR	20:30	20:25 07 Jun	23h 55m	1 stop	No info	NaN
2667	IndiGo	27/03/2019	Kolkata	Banglore	CCU → BLR	14:20	16:55	2h 35m	non-stop	No info	NaN
2668	Jet Airways	6/03/2019	Delhi	Cochin	DEL → BOM → COK	21:50	04:25 07 Mar	6h 35m	1 stop	No info	NaN
2669	Air India	6/03/2019	Delhi	Cochin	DEL → BOM → COK	04:00	19:15	15h 15m	1 stop	No info	NaN
2670	Multiple carriers	15/06/2019	Delhi	Cochin	DEL → BOM → COK	04:55	19:15	14h 20m	1 stop	No info	NaN

In [102]:
```
final_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 13354 entries, 0 to 2670
Data columns (total 11 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   Airline          13354 non-null  object
 1   Date_of_Journey  13354 non-null  object
 2   Source           13354 non-null  object
 3   Destination      13354 non-null  object
 4   Route            13353 non-null  object
 5   Dep_Time         13354 non-null  object
 6   Arrival_Time     13354 non-null  object
 7   Duration         13354 non-null  object
 8   Total_Stops      13353 non-null  object
 9   Additional_Info  13354 non-null  object
 10  Price            10683 non-null  float64
dtypes: float64(1), object(10)
memory usage: 1.2+ MB
```

The split() method splits a string into a list.

In [103]:
```
final_df['Date_of_Journey'].str.split('/').str[0]
```

Out[103]:
```
0       24
1        1
2        9
3       12
4       01
        ..
2666     6
2667    27
2668     6
2669     6
2670    15
Name: Date_of_Journey, Length: 13354, dtype: object
```

In [104]:
```
##Fetaure Engineering Process
final_df['Date']=final_df['Date_of_Journey'].str.split('/').str[0]
final_df['Month']=final_df['Date_of_Journey'].str.split('/').str[1]
final_df['Year']=final_df['Date_of_Journey'].str.split('/').str[2]
```

In [105]:
```
final_df.head(2)
```

Out[105]:

	Airline	Date_of_Journey	Source	Destination	Route	Dep_Time	Arrival_Time	Duration	Total_Stops	Additional_Info	Price	Date	Month	Year
0	IndiGo	24/03/2019	Banglore	New Delhi	BLR → DEL	22:20	01:10 22 Mar	2h 50m	non-stop	No info	3897.0	24	03	2019

1	Air India	1/05/2019	Kolkata	Banglore	CCU → IXR → BBI → BLR	05:50	13:15	7h 25m	2 stops	No info	7662.0	1	05	2019

In [ ]:

Lambda functions are defined using the keyword lambda. They can have any number of arguments but only one expression. A lambda function cannot contain any statements, and it returns a function object which can be assigned to any variable. They are generally used for one-line expressions.

Normal functions are created using the def keyword. They can have any number of arguments, any number of expressions, and lines of code. They are generally used for large blocks of code

In [106]:
```
# Add "Date," "Month," and "Year" columns

final_df["Date"]=final_df['Date_of_Journey'].apply(lambda x:x.split("/")[0])
final_df["Month"]=final_df['Date_of_Journey'].apply(lambda x:x.split("/")[1])
final_df["Year"]=final_df['Date_of_Journey'].apply(lambda x:x.split("/")[2])
```

In [107]:
```
final_df['Date']=final_df['Date'].astype(int)
final_df['Month']=final_df['Month'].astype(int)
final_df['Year']=final_df['Year'].astype(int)
```

In [108]:
```
final_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 13354 entries, 0 to 2670
Data columns (total 14 columns):
 #   Column           Non-Null Count  Dtype 
---  ------           --------------  ----- 
 0   Airline          13354 non-null  object
 1   Date_of_Journey  13354 non-null  object
 2   Source           13354 non-null  object
 3   Destination      13354 non-null  object
 4   Route            13353 non-null  object
 5   Dep_Time         13354 non-null  object
 6   Arrival_Time     13354 non-null  object
 7   Duration         13354 non-null  object
 8   Total_Stops      13353 non-null  object
 9   Additional_Info  13354 non-null  object
 10  Price            10683 non-null  float64
 11  Date             13354 non-null  int32 
 12  Month            13354 non-null  int32 
 13  Year             13354 non-null  int32 
dtypes: float64(1), int32(3), object(10)
memory usage: 1.4+ MB
```

In [109]:
```
final_df.drop('Date_of_Journey',axis=1,inplace=True)
```

In [110]:
```
final_df.head(10)
```

Out[110]:

	Airline	Source	Destination	Route	Dep_Time	Arrival_Time	Duration	Total_Stops	Additional_Info	Price	Date	Month	Year
0	IndiGo	Banglore	New Delhi	BLR → DEL	22:20	01:10 22 Mar	2h 50m	non-stop	No info	3897.0	24	3	2019
1	Air India	Kolkata	Banglore	CCU → IXR → BBI → BLR	05:50	13:15	7h 25m	2 stops	No info	7662.0	1	5	2019
2	Jet Airways	Delhi	Cochin	DEL → LKO → BOM → COK	09:25	04:25 10 Jun	19h	2 stops	No info	13882.0	9	6	2019
3	IndiGo	Kolkata	Banglore	CCU → NAG → BLR	18:05	23:30	5h 25m	1 stop	No info	6218.0	12	5	2019
4	IndiGo	Banglore	New Delhi	BLR → NAG → DEL	16:50	21:35	4h 45m	1 stop	No info	13302.0	1	3	2019
5	SpiceJet	Kolkata	Banglore	CCU → BLR	09:00	11:25	2h 25m	non-stop	No info	3873.0	24	6	2019
6	Jet Airways	Banglore	New Delhi	BLR → BOM → DEL	18:55	10:25 13 Mar	15h 30m	1 stop	In-flight meal not included	11087.0	12	3	2019
7	Jet Airways	Banglore	New Delhi	BLR → BOM → DEL	08:00	05:05 02 Mar	21h 5m	1 stop	No info	22270.0	1	3	2019
8	Jet Airways	Banglore	New Delhi	BLR → BOM → DEL	08:55	10:25 13 Mar	25h 30m	1 stop	In-flight meal not included	11087.0	12	3	2019
9	Multiple carriers	Delhi	Cochin	DEL → BOM → COK	11:25	19:15	7h 50m	1 stop	No info	8625.0	27	5	2019

In [111]:
```
final_df['Arrival_Time'].str.split(' ').str[0]
```

Out[111]:
```
0       01:10
1       13:15
2       04:25
3       23:30
4       21:35
        ...
2666    20:25
2667    16:55
2668    04:25
2669    19:15
2670    19:15
Name: Arrival_Time, Length: 13354, dtype: object
```

In [112]:
```
final_df['Arrival_Time']=final_df['Arrival_Time'].apply(lambda x : x.split(' ')[0])
```

In [113]:
```
final_df['Arrival_hour']=final_df['Arrival_Time'].str.split(':').str[0]
final_df['Arrival_min']=final_df['Arrival_Time'].str.split(':').str[1]
```

In [114]:
```
final_df.head(1)
```

Out[114]:

	Airline	Source	Destination	Route	Dep_Time	Arrival_Time	Duration	Total_Stops	Additional_Info	Price	Date	Month	Year	Arrival_hour	Arrival_min
0	IndiGo	Banglore	New Delhi	BLR → DEL	22:20	01:10	2h 50m	non-stop	No info	3897.0	24	3	2019	01	10

In [115]:
```
final_df['Arrival_hour']=final_df['Arrival_hour'].astype(int)
final_df['Arrival_min']=final_df['Arrival_min'].astype(int)
```

In [116]:
```
final_df.drop('Arrival_Time',axis=1,inplace=True)
```

In [117]:
```
final_df.head(5)
```

Out[117]:

	Airline	Source	Destination	Route	Dep_Time	Duration	Total_Stops	Additional_Info	Price	Date	Month	Year	Arrival_hour	Arrival_min
0	IndiGo	Banglore	New Delhi	BLR → DEL	22:20	2h 50m	non-stop	No info	3897.0	24	3	2019	1	10
1	Air India	Kolkata	Banglore	CCU → IXR → BBI → BLR	05:50	7h 25m	2 stops	No info	7662.0	1	5	2019	13	15
2	Jet Airways	Delhi	Cochin	DEL → LKO → BOM → COK	09:25	19h	2 stops	No info	13882.0	9	6	2019	4	25
3	IndiGo	Kolkata	Banglore	CCU → NAG → BLR	18:05	5h 25m	1 stop	No info	6218.0	12	5	2019	23	30
4	IndiGo	Banglore	New Delhi	BLR → NAG → DEL	16:50	4h 45m	1 stop	No info	13302.0	1	3	2019	21	35

In [118]:
```
final_df['Dept_hour']=final_df['Dep_Time'].str.split(':').str[0]
final_df['Dept_min']=final_df['Dep_Time'].str.split(':').str[1]
final_df['Dept_hour']=final_df['Dept_hour'].astype(int)
final_df['Dept_min']=final_df['Dept_min'].astype(int)
final_df.drop('Dep_Time',axis=1,inplace=True)
```

In [119]:
```
final_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 13354 entries, 0 to 2670
Data columns (total 15 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   Airline          13354 non-null  object 
 1   Source           13354 non-null  object 
 2   Destination      13354 non-null  object 
 3   Route            13353 non-null  object 
 4   Duration         13354 non-null  object 
 5   Total_Stops      13353 non-null  object 
 6   Additional_Info  13354 non-null  object 
 7   Price            10683 non-null  float64
 8   Date             13354 non-null  int32  
 9   Month            13354 non-null  int32  
 10  Year             13354 non-null  int32  
 11  Arrival_hour     13354 non-null  int32  
 12  Arrival_min      13354 non-null  int32  
 13  Dept_hour        13354 non-null  int32  
 14  Dept_min         13354 non-null  int32  
dtypes: float64(1), int32(7), object(7)
memory usage: 1.3+ MB
```

In [120]:
```
final_df['Total_Stops'].unique()
```

Out[120]:
```
array(['non-stop', '2 stops', '1 stop', '3 stops', nan, '4 stops'],
      dtype=object)
```

In [121]:
```
final_df['Total_Stops']=final_df['Total_Stops'].map({'non-stop':0,'1 stop':1,'2 stops':2,'3 stops':3,'4 stops':4,'nan':1})
```

In [122]:
```
final_df[final_df['Total_Stops'].isnull()]
```

Out[122]:

	Airline	Source	Destination	Route	Duration	Total_Stops	Additional_Info	Price	Date	Month	Year	Arrival_hour	Arrival_min	Dept_hour	Dept_min
9039	Air India	Delhi	Cochin	NaN	23h 40m	NaN	No info	7480.0	6	5	2019	9	25	9	45

In [123]:
```
final_df.drop('Route',axis=1,inplace=True)
```

In [124]:
```
final_df.head()
```

Out[124]:

	Airline	Source	Destination	Duration	Total_Stops	Additional_Info	Price	Date	Month	Year	Arrival_hour	Arrival_min	Dept_hour	Dept_min
0	IndiGo	Banglore	New Delhi	2h 50m	0.0	No info	3897.0	24	3	2019	1	10	22	20
1	Air India	Kolkata	Banglore	7h 25m	2.0	No info	7662.0	1	5	2019	13	15	5	50
2	Jet Airways	Delhi	Cochin	19h	2.0	No info	13882.0	9	6	2019	4	25	9	25
3	IndiGo	Kolkata	Banglore	5h 25m	1.0	No info	6218.0	12	5	2019	23	30	18	5
4	IndiGo	Banglore	New Delhi	4h 45m	1.0	No info	13302.0	1	3	2019	21	35	16	50

In [125]:
```
final_df['Additional_Info'].unique()
```

Out[125]:
```
array(['No info', 'In-flight meal not included',
       'No check-in baggage included', '1 Short layover', 'No Info',
       '1 Long layover', 'Change airports', 'Business class',
       'Red-eye flight', '2 Long layover'], dtype=object)
```

In [126]:
```
final_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 13354 entries, 0 to 2670
Data columns (total 14 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   Airline          13354 non-null  object 
 1   Source           13354 non-null  object 
 2   Destination      13354 non-null  object 
 3   Duration         13354 non-null  object 
 4   Total_Stops      13353 non-null  float64
 5   Additional_Info  13354 non-null  object 
 6   Price            10683 non-null  float64
 7   Date             13354 non-null  int32  
 8   Month            13354 non-null  int32  
 9   Year             13354 non-null  int32  
 10  Arrival_hour     13354 non-null  int32  
 11  Arrival_min      13354 non-null  int32  
 12  Dept_hour        13354 non-null  int32  
 13  Dept_min         13354 non-null  int32  
dtypes: float64(2), int32(7), object(5)
memory usage: 1.2+ MB
```

In [127]:
```
final_df['duration_hour']=final_df['Duration'].str.split(' ').str[0].str.split('h').str[0]
```

In [128]:
```
final_df[final_df['duration_hour']=='5m']
```

Out[128]:

	Airline	Source	Destination	Duration	Total_Stops	Additional_Info	Price	Date	Month	Year	Arrival_hour	Arrival_min	Dept_hour	Dept_min	duratio
6474	Air India	Mumbai	Hyderabad	5m	2.0	No info	17327.0	6	3	2019	16	55	16	50	
2660	Air India	Mumbai	Hyderabad	5m	2.0	No info	NaN	12	3	2019	16	55	16	50	

In [129]:
```
final_df.drop(6474,axis=0,inplace=True)
final_df.drop(2660,axis=0,inplace=True)
```

In [130]:
```
final_df['duration_hour']=final_df['duration_hour'].astype('int')
```

In [131]:
```
final_df.drop('Duration',axis=1,inplace=True)
```

In [132]:
```
final_df.head(1)
```

Out[132]:

	Airline	Source	Destination	Total_Stops	Additional_Info	Price	Date	Month	Year	Arrival_hour	Arrival_min	Dept_hour	Dept_min	duration_hour
0	IndiGo	Banglore	New Delhi	0.0	No info	3897.0	24	3	2019	1	10	22	20	2

In [133]:
```
final_df['Airline'].unique()
```

Out[133]:
```
array(['IndiGo', 'Air India', 'Jet Airways', 'SpiceJet',
```

```
       'Multiple carriers', 'GoAir', 'Vistara', 'Air Asia',
       'Vistara Premium economy', 'Jet Airways Business',
       'Multiple carriers Premium economy', 'Trujet'], dtype=object)
```

In [134]:
```
from sklearn.preprocessing import LabelEncoder
labelencoder=LabelEncoder()
```

In [135]:
```
final_df['Airline']=labelencoder.fit_transform(final_df['Airline'])
final_df['Source']=labelencoder.fit_transform(final_df['Source'])
final_df['Destination']=labelencoder.fit_transform(final_df['Destination'])
final_df['Additional_Info']=labelencoder.fit_transform(final_df['Additional_Info'])
```

In [136]:
```
final_df.shape
```

Out[136]:
```
(13351, 14)
```

In [137]:
```
final_df.head(2)
```

Out[137]:

	Airline	Source	Destination	Total_Stops	Additional_Info	Price	Date	Month	Year	Arrival_hour	Arrival_min	Dept_hour	Dept_min	duration_hour
0	3	0	5	0.0	8	3897.0	24	3	2019	1	10	22	20	2
1	1	3	0	2.0	8	7662.0	1	5	2019	13	15	5	50	7

In [138]:
```
final_df[['Airline']]
```

Out[138]:

	Airline
0	3
1	1
2	4
3	3
4	3
...	...
2666	1
2667	3
2668	4
2669	1
2670	6

13351 rows × 1 columns

In [139]:
```
import numpy as np
import pandas as pd
from sklearn.preprocessing import OneHotEncoder

# Step 1: Create the OneHotEncoder object
ohe = OneHotEncoder()

# Step 2: Fit and Transform the 'Airline' column using the OneHotEncoder
airline_encoded = ohe.fit_transform(np.array(final_df['Airline']).reshape(-1, 1))

# The result is a sparse matrix. If you want to convert it to a dense array, you can use the following:
airline_encoded_array = airline_encoded.toarray()

# Optionally, you can create a DataFrame with the encoded data
encoded_df = pd.DataFrame(airline_encoded_array, columns=ohe.get_feature_names_out(['Airline']))

# Display the encoded DataFrame
print(encoded_df)
```

```
       Airline_0  Airline_1  Airline_2  Airline_3  Airline_4  Airline_5  \
0            0.0        0.0        0.0        1.0        0.0        0.0
1            0.0        1.0        0.0        0.0        0.0        0.0
2            0.0        0.0        0.0        0.0        1.0        0.0
3            0.0        0.0        0.0        1.0        0.0        0.0
4            0.0        0.0        0.0        1.0        0.0        0.0
...          ...        ...        ...        ...        ...        ...
13346        0.0        1.0        0.0        0.0        0.0        0.0
13347        0.0        0.0        0.0        1.0        0.0        0.0
13348        0.0        0.0        0.0        0.0        1.0        0.0
13349        0.0        1.0        0.0        0.0        0.0        0.0
13350        0.0        0.0        0.0        0.0        0.0        0.0

       Airline_6  Airline_7  Airline_8  Airline_9  Airline_10  Airline_11
0            0.0        0.0        0.0        0.0         0.0         0.0
1            0.0        0.0        0.0        0.0         0.0         0.0
2            0.0        0.0        0.0        0.0         0.0         0.0
3            0.0        0.0        0.0        0.0         0.0         0.0
4            0.0        0.0        0.0        0.0         0.0         0.0
...          ...        ...        ...        ...         ...         ...
13346        0.0        0.0        0.0        0.0         0.0         0.0
13347        0.0        0.0        0.0        0.0         0.0         0.0
13348        0.0        0.0        0.0        0.0         0.0         0.0
13349        0.0        0.0        0.0        0.0         0.0         0.0
13350        1.0        0.0        0.0        0.0         0.0         0.0

[13351 rows x 12 columns]
```

In [140]:
```
final_df.head()
```

Out[140]:

	Airline	Source	Destination	Total_Stops	Additional_Info	Price	Date	Month	Year	Arrival_hour	Arrival_min	Dept_hour	Dept_min	duration_hour
0	3	0	5	0.0	8	3897.0	24	3	2019	1	10	22	20	2
1	1	3	0	2.0	8	7662.0	1	5	2019	13	15	5	50	7
2	4	2	1	2.0	8	13882.0	9	6	2019	4	25	9	25	19
3	3	3	0	1.0	8	6218.0	12	5	2019	23	30	18	5	5
4	3	0	5	1.0	8	13302.0	1	3	2019	21	35	16	50	4

In [141]:
```
final_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 13351 entries, 0 to 2670
Data columns (total 14 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   Airline          13351 non-null  int32
 1   Source           13351 non-null  int32
 2   Destination      13351 non-null  int32
 3   Total_Stops      13350 non-null  float64
 4   Additional_Info  13351 non-null  int32
 5   Price            10681 non-null  float64
 6   Date             13351 non-null  int32
 7   Month            13351 non-null  int32
 8   Year             13351 non-null  int32
 9   Arrival_hour     13351 non-null  int32
```

```
 9   Arrival_hour      13351 non-null  int32
 10  Arrival_min       13351 non-null  int32
 11  Dept_hour         13351 non-null  int32
 12  Dept_min          13351 non-null  int32
 13  duration_hour     13351 non-null  int32
dtypes: float64(2), int32(12)
memory usage: 938.7 KB
```

In [142]:
```
pd.get_dummies(final_df,columns=["Airline", "Source", "Destination"] ,drop_first = True)
```

Out[142]:

	Total_Stops	Additional_Info	Price	Date	Month	Year	Arrival_hour	Arrival_min	Dept_hour	Dept_min	...	Airline_11	Source_1	Source_2	Source_3
0	0.0	8	3897.0	24	3	2019	1	10	22	20	...	0	0	0	0
1	2.0	8	7662.0	1	5	2019	13	15	5	50	...	0	0	0	1
2	2.0	8	13882.0	9	6	2019	4	25	9	25	...	0	0	1	0
3	1.0	8	6218.0	12	5	2019	23	30	18	5	...	0	0	0	1
4	1.0	8	13302.0	1	3	2019	21	35	16	50	...	0	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
2666	1.0	8	NaN	6	6	2019	20	25	20	30	...	0	0	0	1
2667	0.0	8	NaN	27	3	2019	16	55	14	20	...	0	0	0	1
2668	1.0	8	NaN	6	3	2019	4	25	21	50	...	0	0	1	0
2669	1.0	8	NaN	6	3	2019	19	15	4	0	...	0	0	1	0
2670	1.0	8	NaN	15	6	2019	19	15	4	55	...	0	0	1	0

13351 rows × 31 columns

In [ ]: