In [66]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [67]:
```
m=pd.read_csv(r"C:\Users\manasa\Downloads\Data_Train.csv")
m
```

Out[67]:

	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
...	...	...	...	...	...	...	...	...	...	...	...
10678	Air Asia	9/04/2019	Kolkata	Banglore	CCU → BLR	19:55	22:25	2h 30m	non-stop	No info	4107
10679	Air India	27/04/2019	Kolkata	Banglore	CCU → BLR	20:45	23:20	2h 35m	non-stop	No info	4145
10680	Jet Airways	27/04/2019	Banglore	Delhi	BLR → DEL	08:20	11:20	3h	non-stop	No info	7229
10681	Vistara	01/03/2019	Banglore	New Delhi	BLR → DEL	11:30	14:10	2h 40m	non-stop	No info	12648
10682	Air India	9/05/2019	Delhi	Cochin	DEL → GOI → BOM → COK	10:55	19:15	8h 20m	2 stops	No info	11753

10683 rows × 11 columns

In [68]:
```
s=pd.read_csv(r"C:\Users\manasa\Downloads\Test_set.csv")
s
```

Out[68]:

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

2671 rows × 10 columns

In [69]:
```
m.head()
```

Out[69]:

	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 [70]:
```
s.head()
```

Out[70]:

	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 [71]:
```
m.tail()
```

Out[71]:

	Airline	Date_of_Journey	Source	Destination	Route	Dep_Time	Arrival_Time	Duration	Total_Stops	Additional_Info	Price
10678	Air Asia	9/04/2019	Kolkata	Banglore	CCU → BLR	19:55	22:25	2h 30m	non-stop	No info	4107
10679	Air India	27/04/2019	Kolkata	Banglore	CCU → BLR	20:45	23:20	2h 35m	non-stop	No info	4145
10680	Jet Airways	27/04/2019	Banglore	Delhi	BLR → DEL	08:20	11:20	3h	non-stop	No info	7229
10681	Vistara	01/03/2019	Banglore	New Delhi	BLR → DEL	11:30	14:10	2h 40m	non-stop	No info	12648
10682	Air India	9/05/2019	Delhi	Cochin	DEL → GOI → BOM → COK	10:55	19:15	8h 20m	2 stops	No info	11753

In [72]:
```
s.tail()
```

Out[72]:

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

In [73]:
```
m.shape
```

Out[73]:
```
(10683, 11)
```

In [74]:
```
s.shape
```

Out[74]:
```
(2671, 10)
```

In [75]:
```
m.isnull().sum()
```

Out[75]:
```
Airline            0
Date_of_Journey    0
Source             0
Destination        0
Route              1
Dep_Time           0
Arrival_Time       0
Duration           0
Total_Stops        1
Additional_Info    0
Price              0
dtype: int64
```

In [76]:
```
s.isnull().sum()
```

Out[76]:
```
Airline            0
Date_of_Journey    0
Source             0
Destination        0
Route              0
Dep_Time           0
Arrival_Time       0
Duration           0
Total_Stops        0
Additional_Info    0
dtype: int64
```

In [77]:
```
m.describe
```

Out[77]:
```
<bound method NDFrame.describe of            Airline Date_of_Journey    Source Destination
0           IndiGo      24/03/2019  Banglore   New Delhi  \
1        Air India       1/05/2019   Kolkata    Banglore
2      Jet Airways       9/06/2019     Delhi      Cochin
3           IndiGo      12/05/2019   Kolkata    Banglore
4           IndiGo      01/03/2019  Banglore   New Delhi
...            ...             ...       ...         ...
10678     Air Asia       9/04/2019   Kolkata    Banglore
10679    Air India      27/04/2019   Kolkata    Banglore
10680  Jet Airways      27/04/2019  Banglore       Delhi
10681      Vistara      01/03/2019  Banglore   New Delhi
10682    Air India       9/05/2019     Delhi      Cochin

                      Route Dep_Time  Arrival_Time Duration Total_Stops
0                   BLR → DEL    22:20  01:10 22 Mar   2h 50m    non-stop  \
1      CCU → IXR → BBI → BLR    05:50         13:15   7h 25m     2 stops
2      DEL → LKO → BOM → COK    09:25  04:25 10 Jun      19h     2 stops
3            CCU → NAG → BLR    18:05         23:30   5h 25m      1 stop
4            BLR → NAG → DEL    16:50         21:35   4h 45m      1 stop
...                      ...      ...           ...      ...         ...
10678              CCU → BLR    19:55         22:25   2h 30m    non-stop
10679              CCU → BLR    20:45         23:20   2h 35m    non-stop
10680              BLR → DEL    08:20         11:20       3h    non-stop
10681              BLR → DEL    11:30         14:10   2h 40m    non-stop
10682  DEL → GOI → BOM → COK    10:55         19:15   8h 20m     2 stops

      Additional_Info  Price
0             No info   3897
1             No info   7662
2             No info  13882
3             No info   6218
4             No info  13302
...               ...    ...
10678         No info   4107
10679         No info   4145
10680         No info   7229
10681         No info  12648
10682         No info  11753

[10683 rows x 11 columns]>
```

In [78]:
```
s.describe
```

Out[78]:
```
<bound method NDFrame.describe of                Airline Date_of_Journey    Source Destination  \
0           Jet Airways       6/06/2019     Delhi      Cochin
1                IndiGo      12/05/2019   Kolkata    Banglore
2           Jet Airways      21/05/2019     Delhi      Cochin
3     Multiple carriers      21/05/2019     Delhi      Cochin
4              Air Asia      24/06/2019  Banglore       Delhi
...                 ...             ...       ...         ...
2666         Air India       6/06/2019   Kolkata    Banglore
2667            IndiGo      27/03/2019   Kolkata    Banglore
2668        Jet Airways       6/03/2019     Delhi      Cochin
2669         Air India       6/03/2019     Delhi      Cochin
2670  Multiple carriers      15/06/2019     Delhi      Cochin

                Route Dep_Time  Arrival_Time Duration Total_Stops  \
0     DEL → BOM → COK    17:30  04:25 07 Jun  10h 55m      1 stop
1     CCU → MAA → BLR    06:20         10:20       4h      1 stop
2     DEL → BOM → COK    19:15  19:00 22 May  23h 45m      1 stop
3     DEL → BOM → COK    08:00         21:00      13h      1 stop
4           BLR → DEL    23:55  02:45 25 Jun   2h 50m    non-stop
...               ...      ...           ...      ...         ...
2666  CCU → DEL → BLR    20:30  20:25 07 Jun  23h 55m      1 stop
2667        CCU → BLR    14:20         16:55   2h 35m    non-stop
2668  DEL → BOM → COK    21:50  04:25 07 Mar   6h 35m      1 stop
2669  DEL → BOM → COK    04:00         19:15  15h 15m      1 stop
2670  DEL → BOM → COK    04:55         19:15  14h 20m      1 stop

                  Additional_Info
0                         No info
1                         No info
2     In-flight meal not included
3                         No info
4                         No info
...                           ...
2666                      No info
2667                      No info
2668                      No info
2669                      No info
2670                      No info

[2671 rows x 10 columns]>
```

In [79]:
```
m.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10683 entries, 0 to 10682
Data columns (total 11 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   Airline          10683 non-null  object
 1   Date_of_Journey  10683 non-null  object
 2   Source           10683 non-null  object
 3   Destination      10683 non-null  object
 4   Route            10682 non-null  object
 5   Dep_Time         10683 non-null  object
 6   Arrival_Time     10683 non-null  object
 7   Duration         10683 non-null  object
 8   Total_Stops      10682 non-null  object
 9   Additional_Info  10683 non-null  object
 10  Price            10683 non-null  int64
dtypes: int64(1), object(10)
memory usage: 918.2+ KB
```

In [80]:
```
s.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2671 entries, 0 to 2670
Data columns (total 10 columns):
 #   Column           Non-Null Count  Dtype 
---  ------           --------------  ----- 
 0   Airline          2671 non-null   object
 1   Date_of_Journey  2671 non-null   object
 2   Source           2671 non-null   object
 3   Destination      2671 non-null   object
 4   Route            2671 non-null   object
 5   Dep_Time         2671 non-null   object
 6   Arrival_Time     2671 non-null   object
 7   Duration         2671 non-null   object
 8   Total_Stops      2671 non-null   object
 9   Additional_Info  2671 non-null   object
dtypes: object(10)
memory usage: 208.8+ KB
```

In [81]:
```
m.duplicated().sum()
```

Out[81]:
```
220
```

In [82]:
```
m['Airline'].value_counts()
```

Out[82]:
```
Airline
Jet Airways                          3849
IndiGo                               2053
Air India                            1752
Multiple carriers                    1196
SpiceJet                              818
Vistara                               479
Air Asia                              319
GoAir                                 194
Multiple carriers Premium economy      13
Jet Airways Business                    6
Vistara Premium economy                 3
Trujet                                  1
Name: count, dtype: int64
```

In [83]:
```
s.duplicated().sum()
```

Out[83]:
```
26
```

In [84]:
```
m['Destination'].value_counts()
```

Out[84]:
```
Destination
Cochin       4537
Banglore     2871
Delhi        1265
New Delhi     932
Hyderabad     697
Kolkata       381
Name: count, dtype: int64
```

In [85]:
```
m.columns
```

Out[85]:
```
Index(['Airline', 'Date_of_Journey', 'Source', 'Destination', 'Route',
       'Dep_Time', 'Arrival_Time', 'Duration', 'Total_Stops',
       'Additional_Info', 'Price'],
      dtype='object')
```

In [86]:
```
s.columns
```

Out[86]:
```
Index(['Airline', 'Date_of_Journey', 'Source', 'Destination', 'Route',
       'Dep_Time', 'Arrival_Time', 'Duration', 'Total_Stops',
       'Additional_Info'],
      dtype='object')
```

In [87]:
```
m['Total_Stops'].value_counts()
```

Out[87]:
```
Total_Stops
1 stop      5625
non-stop    3491
2 stops     1520
3 stops       45
4 stops        1
Name: count, dtype: int64
```

In [88]:
```
m.dropna(inplace=True)
```

In [89]:
```
m['Source'].value_counts()
```

Out[89]:
```
Source
Delhi       4536
Kolkata     2871
Banglore    2197
Mumbai       697
Chennai      381
Name: count, dtype: int64
```

In [90]:
```
m['Total_Stops'].value_counts()
```

Out[90]:
```
Total_Stops
1 stop      5625
non-stop    3491
2 stops     1520
3 stops       45
4 stops        1
Name: count, dtype: int64
```

In [91]:
```
flight={"Airline":{"Jet Airways":0,"IndiGo":1,"Air India":2,"Multiple carriers":3,"SpiceJet":4,"Vistara":5,"Ai
"Multiple carriers Premium economy":8,
"Jet Airways Business":9,"Vistara Premium economy":10,"Trujet":11}}
m=m.replace(flight)
m
```

Out[91]:

	Airline	Date_of_Journey	Source	Destination	Route	Dep_Time	Arrival_Time	Duration	Total_Stops	Additional_Info	Price
0	1	24/03/2019	Banglore	New Delhi	BLR → DEL	22:20	01:10 22 Mar	2h 50m	non-stop	No info	3897
1	2	1/05/2019	Kolkata	Banglore	CCU → IXR → BBI → BLR	05:50	13:15	7h 25m	2 stops	No info	7662
2	0	9/06/2019	Delhi	Cochin	DEL → LKO → BOM → COK	09:25	04:25 10 Jun	19h	2 stops	No info	13882
3	1	12/05/2019	Kolkata	Banglore	CCU → NAG → BLR	18:05	23:30	5h 25m	1 stop	No info	6218
4	1	01/03/2019	Banglore	New Delhi	BLR → NAG → DEL	16:50	21:35	4h 45m	1 stop	No info	13302
...	...	...	...	...	...	...	...	...	...	...	...
10678	6	9/04/2019	Kolkata	Banglore	CCU → BLR	19:55	22:25	2h 30m	non-stop	No info	4107
10679	2	27/04/2019	Kolkata	Banglore	CCU → BLR	20:45	23:20	2h 35m	non-stop	No info	4145
10680	0	27/04/2019	Banglore	Delhi	BLR → DEL	08:20	11:20	3h	non-stop	No info	7229
10681	5	01/03/2019	Banglore	New Delhi	BLR → DEL	11:30	14:10	2h 40m	non-stop	No info	12648
10682	2	9/05/2019	Delhi	Cochin	DEL → GOI → BOM → COK	10:55	19:15	8h 20m	2 stops	No info	11753

10682 rows × 11 columns

In [92]:
```
city={"Source":{"Delhi":0,"Kolkata":1,"Banglore":2,
"Mumbai":3,"Chennai":4}}
m=m.replace(city)
m
```

Out[92]:

	Airline	Date_of_Journey	Source	Destination	Route	Dep_Time	Arrival_Time	Duration	Total_Stops	Additional_Info	Price
0	1	24/03/2019	2	New Delhi	BLR → DEL	22:20	01:10 22 Mar	2h 50m	non-stop	No info	3897
1	2	1/05/2019	1	Banglore	CCU → IXR → BBI → BLR	05:50	13:15	7h 25m	2 stops	No info	7662
2	0	9/06/2019	0	Cochin	DEL → LKO → BOM → COK	09:25	04:25 10 Jun	19h	2 stops	No info	13882
3	1	12/05/2019	1	Banglore	CCU → NAG → BLR	18:05	23:30	5h 25m	1 stop	No info	6218
4	1	01/03/2019	2	New Delhi	BLR → NAG → DEL	16:50	21:35	4h 45m	1 stop	No info	13302
...	...	...	...	...	...	...	...	...	...	...	...
10678	6	9/04/2019	1	Banglore	CCU → BLR	19:55	22:25	2h 30m	non-stop	No info	4107
10679	2	27/04/2019	1	Banglore	CCU → BLR	20:45	23:20	2h 35m	non-stop	No info	4145
10680	0	27/04/2019	2	Delhi	BLR → DEL	08:20	11:20	3h	non-stop	No info	7229
10681	5	01/03/2019	2	New Delhi	BLR → DEL	11:30	14:10	2h 40m	non-stop	No info	12648
10682	2	9/05/2019	0	Cochin	DEL → GOI → BOM → COK	10:55	19:15	8h 20m	2 stops	No info	11753

10682 rows × 11 columns

In [93]:
```
destination={"Destination":{"Cochin":0,"Banglore":1,"Delhi":2,"New Delhi":3,"Hyderabad":4,"Kolkata":5}}
m=m.replace(destination)
m
```

Out[93]:

	Airline	Date_of_Journey	Source	Destination	Route	Dep_Time	Arrival_Time	Duration	Total_Stops	Additional_Info	Price
0	1	24/03/2019	2	3	BLR → DEL	22:20	01:10 22 Mar	2h 50m	non-stop	No info	3897
1	2	1/05/2019	1	1	CCU → IXR → BBI → BLR	05:50	13:15	7h 25m	2 stops	No info	7662
2	0	9/06/2019	0	0	DEL → LKO → BOM → COK	09:25	04:25 10 Jun	19h	2 stops	No info	13882
3	1	12/05/2019	1	1	CCU → NAG → BLR	18:05	23:30	5h 25m	1 stop	No info	6218
4	1	01/03/2019	2	3	BLR → NAG → DEL	16:50	21:35	4h 45m	1 stop	No info	13302
...	...	...	...	...	...	...	...	...	...	...	...
10678	6	9/04/2019	1	1	CCU → BLR	19:55	22:25	2h 30m	non-stop	No info	4107
10679	2	27/04/2019	1	1	CCU → BLR	20:45	23:20	2h 35m	non-stop	No info	4145
10680	0	27/04/2019	2	2	BLR → DEL	08:20	11:20	3h	non-stop	No info	7229
10681	5	01/03/2019	2	3	BLR → DEL	11:30	14:10	2h 40m	non-stop	No info	12648
10682	2	9/05/2019	0	0	DEL → GOI → BOM → COK	10:55	19:15	8h 20m	2 stops	No info	11753

10682 rows × 11 columns

In [94]:
```
stops={"Total_Stops":{"non-stop":0,"1 stop":1,"2 stops":2,"3 stops":3,"4 stops":4}}
m=m.replace(stops)
m
```

Out[94]:

	Airline	Date_of_Journey	Source	Destination	Route	Dep_Time	Arrival_Time	Duration	Total_Stops	Additional_Info	Price
0	1	24/03/2019	2	3	BLR → DEL	22:20	01:10 22 Mar	2h 50m	0	No info	3897
1	2	1/05/2019	1	1	CCU → IXR → BBI → BLR	05:50	13:15	7h 25m	2	No info	7662
2	0	9/06/2019	0	0	DEL → LKO → BOM → COK	09:25	04:25 10 Jun	19h	2	No info	13882
3	1	12/05/2019	1	1	CCU → NAG → BLR	18:05	23:30	5h 25m	1	No info	6218
4	1	01/03/2019	2	3	BLR → NAG → DEL	16:50	21:35	4h 45m	1	No info	13302
...	...	...	...	...	...	...	...	...	...	...	...
10678	6	9/04/2019	1	1	CCU → BLR	19:55	22:25	2h 30m	0	No info	4107
10679	2	27/04/2019	1	1	CCU → BLR	20:45	23:20	2h 35m	0	No info	4145
10680	0	27/04/2019	2	2	BLR → DEL	08:20	11:20	3h	0	No info	7229
10681	5	01/03/2019	2	3	BLR → DEL	11:30	14:10	2h 40m	0	No info	12648
10682	2	9/05/2019	0	0	DEL → GOI → BOM → COK	10:55	19:15	8h 20m	2	No info	11753

10682 rows × 11 columns

Data Visualization

In [95]:
```
fdf=m[['Airline','Source','Destination','Total_Stops','Price']]
sns.heatmap(fdf.corr(),annot=True)
```

Out[95]: <Axes: >

In [96]:
```
x=fdf[['Airline','Source','Destination','Total_Stops']]
y=fdf['Price']
```

In [99]:
```
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test=train_test_split(x,y,test_size=0.3,random_state=100)
```

Linear Regression

In [100]:
```
from sklearn.linear_model import LinearRegression
regr=LinearRegression()
regr.fit(X_train,y_train)
print(regr.intercept_)
coeff_df=pd.DataFrame(regr.coef_,x.columns,columns=['coefficient'])
coeff_df
```

```
7211.098088897486
```

Out[100]:

	coefficient
Airline	-418.483922
Source	-3275.073380
Destination	2505.480291
Total_Stops	3541.798053

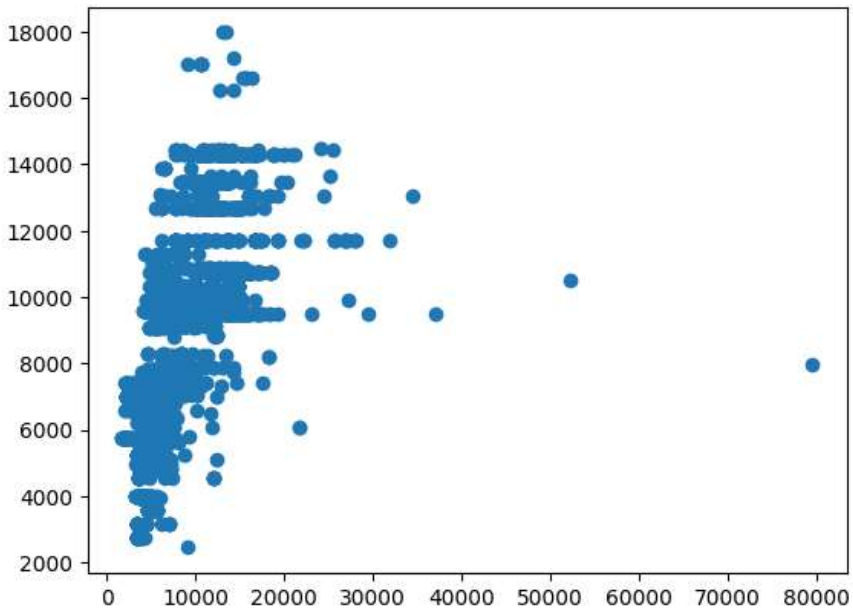
In [101]:
```
score=regr.score(X_test,y_test)
print(score)
```

```
0.41083048909283504
```

In [102]:
```
predictions=regr.predict(X_test)
```

In [103]:
```
plt.scatter(y_test,predictions)
```

Out[103]:
```
<matplotlib.collections.PathCollection at 0x15ccc2a9f50>
```



In [104]:
```
x=np.array(fdf['Price']).reshape(-1,1)
y=np.array(fdf['Total_Stops']).reshape(-1,1)
fdf.dropna(inplace=True)
```

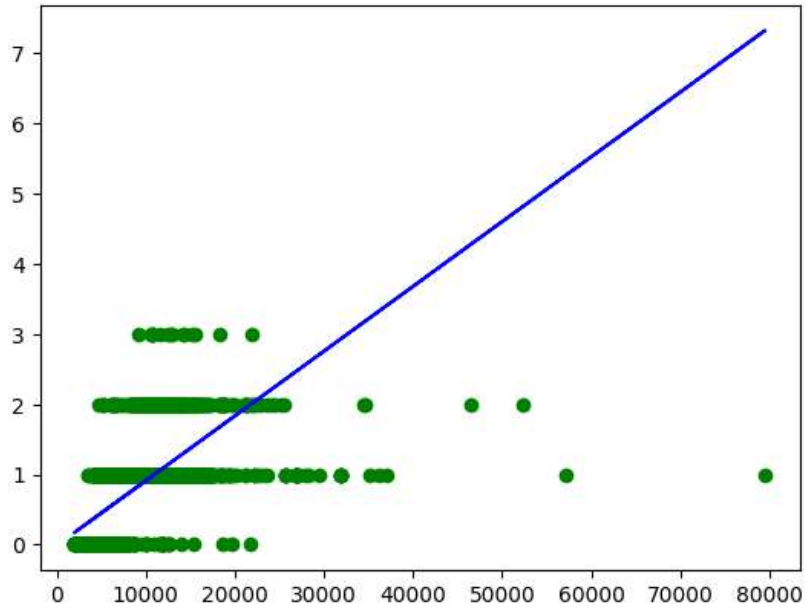
```
C:\Users\manasa\AppData\Local\Temp\ipykernel_11228\521034954.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)
  fdf.dropna(inplace=True)
```

In [107]:
```
X_train,X_test,y_train,y_test=train_test_split(x,y,test_size=0.3)
regr.fit(X_train,y_train)
regr.fit(X_train,y_train)
```

Out[107]:
```
LinearRegression
LinearRegression()
```

In [108]:
```
y_pred=regr.predict(X_test)
plt.scatter(X_test,y_test,color='g')
plt.plot(X_test,y_pred,color='b')
plt.show()
```



In [111]:
```
#Logistic Regression
x=np.array(fdf['Price']).reshape(-1,1)
y=np.array(fdf['Total_Stops']).reshape(-1,1)
fdf.dropna(inplace=True)
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3,random_state=1)
from sklearn.linear_model import LogisticRegression
lr=LogisticRegression(max_iter=10000)
```

```
C:\Users\manasa\AppData\Local\Temp\ipykernel_11228\3604832714.py:4: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)
  fdf.dropna(inplace=True)
```

In [112]:
```
lr.fit(x_train,y_train)
```

```
C:\Users\manasa\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\utils\validation.py:1143: D
ataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y
to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
```

Out[112]:
```
LogisticRegression
LogisticRegression(max_iter=10000)
```

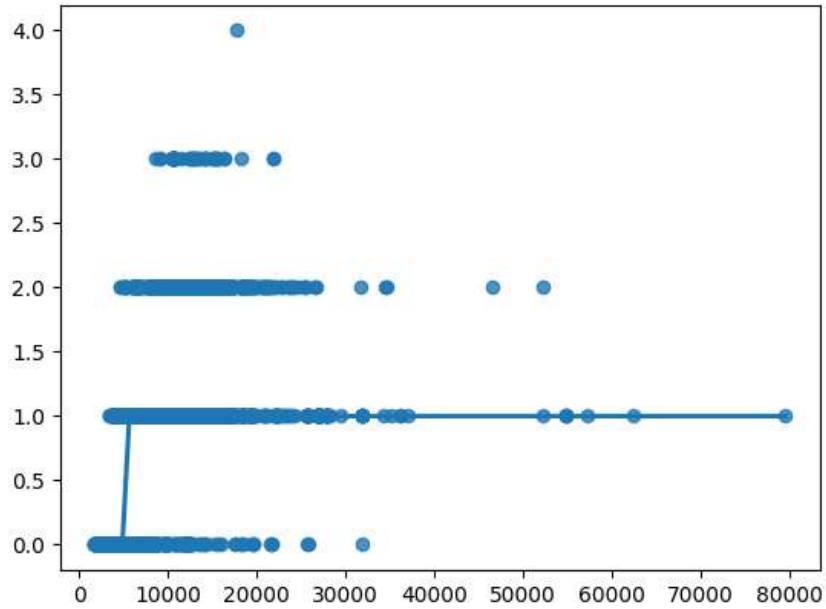
In [113]:
```
score=lr.score(x_test,y_test)
print(score)
```

```
0.7160686427457098
```

In [114]:
```
sns.regplot(x=x,y=y,data=fdf,logistic=True,ci=None)
```

```
C:\Users\manasa\AppData\Local\Programs\Python\Python311\Lib\site-packages\statsmodels\genmod\families\links.p
y:198: RuntimeWarning: overflow encountered in exp
  t = np.exp(-z)
```

Out[114]: <Axes: >



Decision Tree

In [115]:
```
from sklearn.tree import DecisionTreeClassifier
clf=DecisionTreeClassifier(random_state=0)
clf.fit(x_train,y_train)
```

Out[115]:
```
DecisionTreeClassifier
DecisionTreeClassifier(random_state=0)
```

In [116]:
```
score=clf.score(x_test,y_test)
print(score)
```

```
0.9369734789391576
```

Random Forest

In [117]:
```
#Random forest classifier
from sklearn.ensemble import RandomForestClassifier
rfc=RandomForestClassifier()
rfc.fit(X_train,y_train)
```

```
C:\Users\manasa\AppData\Local\Temp\ipykernel_11228\1232785509.py:4: DataConversionWarning: A column-vector y
was passed when a 1d array was expected. Please change the shape of y to (n_samples,), for example using rave
l().
  rfc.fit(X_train,y_train)
```

Out[117]:
```
▾ RandomForestClassifier
RandomForestClassifier()
```

In [120]:
```
params={'max_depth':[2,3,5,10,20],'min_samples_leaf':[5,10,20,50,100,200],'n_estimators':[10,25,30,50,100,200]
```

In [122]:
```
from sklearn.model_selection import GridSearchCV
grid_search=GridSearchCV(estimator=rfc,param_grid=params,cv=2,scoring="accuracy")
```

In [124]:
```
grid_search.fit(X_train,y_train)
```

```
C:\Users\manasa\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\model_selection\_validat
ion.py:686: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please chang
e the shape of y to (n_samples,), for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
C:\Users\manasa\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\model_selection\_validat
ion.py:686: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please chang
e the shape of y to (n_samples,), for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
C:\Users\manasa\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\model_selection\_validat
ion.py:686: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please chang
e the shape of y to (n_samples,), for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
C:\Users\manasa\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\model_selection\_validat
ion.py:686: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please chang
e the shape of y to (n_samples,), for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
C:\Users\manasa\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\model_selection\_validat
ion.py:686: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please chang
e the shape of y to (n_samples,), for example using ravel().
  estimator.fit(X_train, y_train, **fit_params)
```

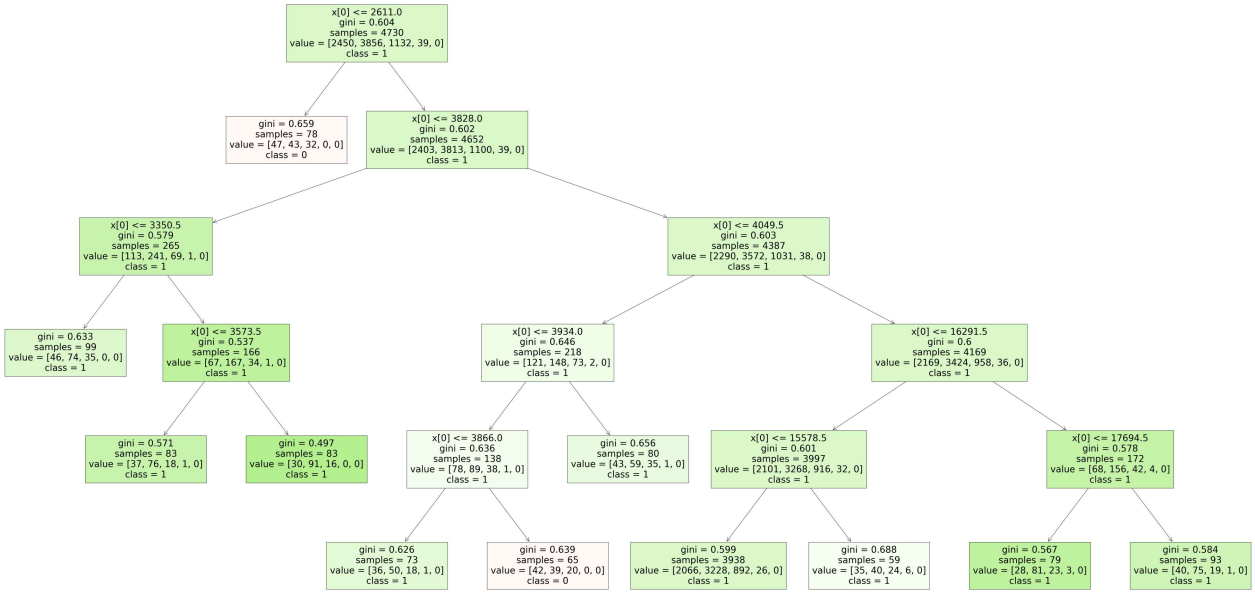
In [128]:
```
grid_search.best_score_
```

Out[128]:
```
0.5237394770692444
```

In [129]:
```
rf_best=grid_search.best_estimator_
rf_best
```

Out[129]:
```
▾ RandomForestClassifier
RandomForestClassifier(max_depth=5, min_samples_leaf=50, n_estimators=10)
```

In [133]:
```
from sklearn.tree import plot_tree
plt.figure(figsize=(80,40))
plot_tree(rf_best.estimators_[4],class_names=['0','1','2','3','4'],filled=True);
```



In [134]:
```
score=rfc.score(x_test,y_test)
print(score)
```

```
0.46333853354134164
```

Conclusion

In [137]:
```
*Here when we compare between Decision Tree and Random Forest, wecan confirm that Decision Tree has more accura
*It makes DecisionTree to perform better than Random Forest.*But it may vary for the other datasets where in m
implemented we can conclude that"Decision Tree" is the best modelforthe given dataset
```

```
  Cell In[137], line 1
    *Here when we compare between Decision Tree and Random Forest, wecan confirm that Decision Tree has more
accuracy than Random Forestwhich makesit the best model for this dataset.
           ^
SyntaxError: invalid syntax
```

In [ ]:
```
```