**EXPLORATORY DATA ANALYSIS FROM SCRATCH**

**Typical work environment for a data analyst:**

Most data analysts do their jobs at desk in office settings. They use computers to analyze data, prepare reports and share their findings with colleagues and stakeholders.

**Steps to follow for EDA project:**

1. Choose a dataset
2. Chose an IDE
3. List down activities clearly
4. Take up tasks one by one
5. Prepare a summary
6. Share it on open platforms

**Exploratory Data Analysis:**

The main purpose of EDA is to detect any errors, outliers and to understand different patterns in the data. It allows Analysts to understand the data better before making any assumptions.

**Parts of EDA:**

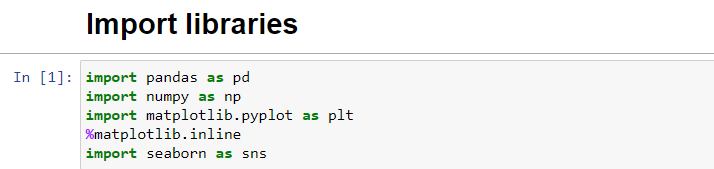
1. Get maximum insights from a data set
2. Uncover underlying structure
3. Extract important variables from the data set
4. Detect outliers and anomalies
5. Test underlying assumptions
6. Determine the optional factor settings

**Performing EDA using Automobile data set:**

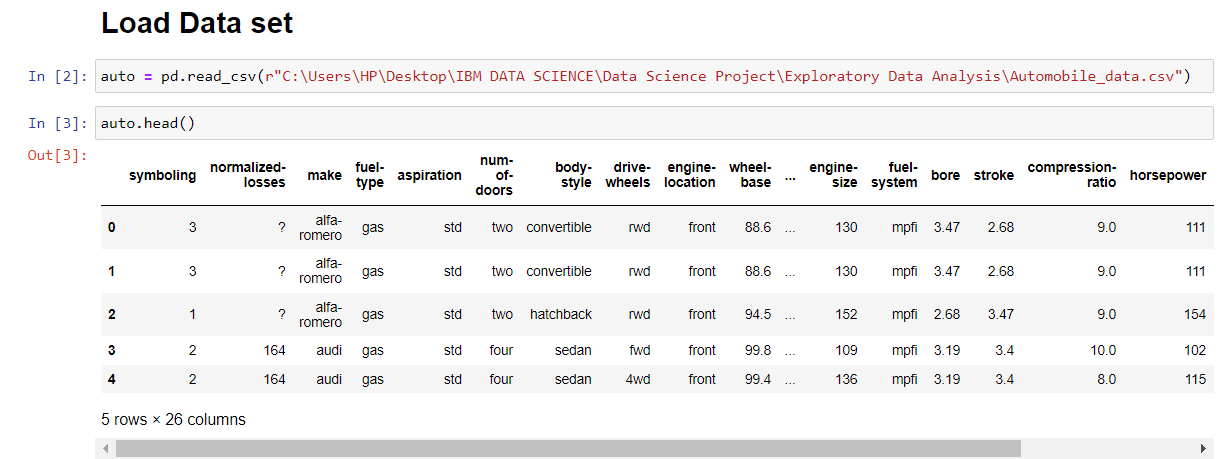
Here is the csv file used for this EDA project - [**https://www.kaggle.com/toramky/automobile-dataset**](https://www.kaggle.com/toramky/automobile-dataset)

**Step by step procedure for Exploratory Data Analysis:**

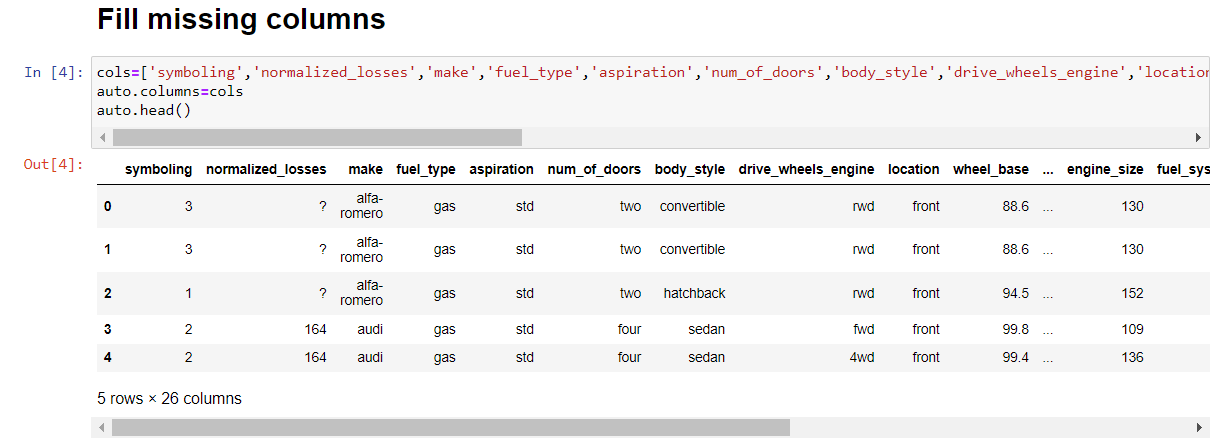
**Step 1:** Import necessary libraries such as Pandas, Numpy, Matplotlib and Seaborn



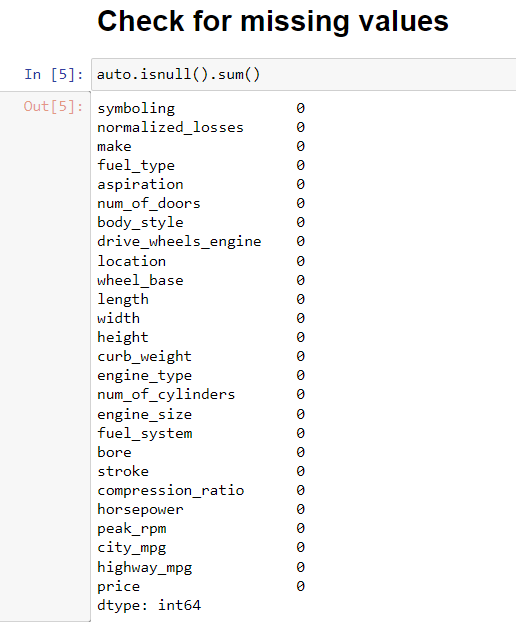
**Step 2:** Load the dataset and view the first 5 rows of the dataset



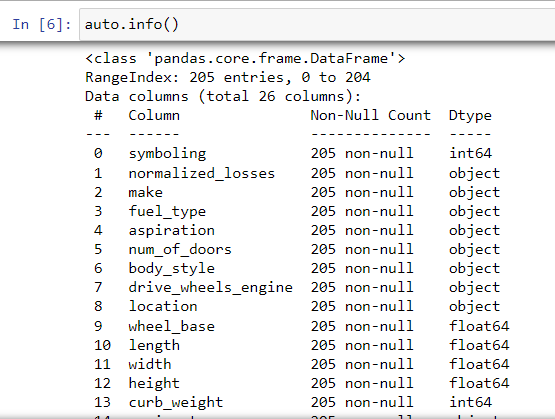
**Step 3:** The dataset has 26 attributes and some data are missing in some columns. In this step we’ll be filling those missing columns.



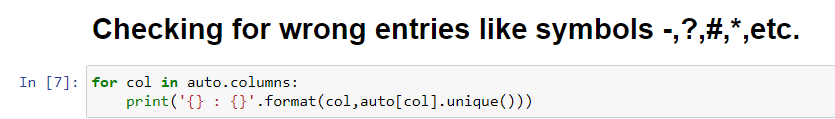
**Step 4:** Now check for the null values in the data set using the command ***auto.isnull().sum()***



**Step 5:** Now check the data types of each attribute using the command ***auto.info()***



**Step 6:** We can observe that those columns that have symbols are in object form as well as some columns should be of an integer type but are of an object type. Now let us detect which columns have symbols and if there are any other symbols too.



symboling : [ 3 1 2 0 -1 -2]

normalized\_losses : ['?' '164' '158' '192' '188' '121' '98' '81' '118' '148' '110' '145' '137'

'101' '78' '106' '85' '107' '104' '113' '150' '129' '115' '93' '142'

'161' '153' '125' '128' '122' '103' '168' '108' '194' '231' '119' '154'

'74' '186' '83' '102' '89' '87' '77' '91' '134' '65' '197' '90' '94'

'256' '95']

make : ['alfa-romero' 'audi' 'bmw' 'chevrolet' 'dodge' 'honda' 'isuzu' 'jaguar'

'mazda' 'mercedes-benz' 'mercury' 'mitsubishi' 'nissan' 'peugot'

'plymouth' 'porsche' 'renault' 'saab' 'subaru' 'toyota' 'volkswagen'

'volvo']

fuel\_type : ['gas' 'diesel']

aspiration : ['std' 'turbo']

num\_of\_doors : ['two' 'four' '?']

body\_style : ['convertible' 'hatchback' 'sedan' 'wagon' 'hardtop']

drive\_wheels\_engine : ['rwd' 'fwd' '4wd']

location : ['front' 'rear']

wheel\_base : [ 88.6 94.5 99.8 99.4 105.8 99.5 101.2 103.5 110. 88.4 93.7 103.3

95.9 86.6 96.5 94.3 96. 113. 102. 93.1 95.3 98.8 104.9 106.7

115.6 96.6 120.9 112. 102.7 93. 96.3 95.1 97.2 100.4 91.3 99.2

107.9 114.2 108. 89.5 98.4 96.1 99.1 93.3 97. 96.9 95.7 102.4

102.9 104.5 97.3 104.3 109.1]

length : [168.8 171.2 176.6 177.3 192.7 178.2 176.8 189. 193.8 197. 141.1 155.9

158.8 157.3 174.6 173.2 144.6 150. 163.4 157.1 167.5 175.4 169.1 170.7

172.6 199.6 191.7 159.1 166.8 169. 177.8 175. 190.9 187.5 202.6 180.3

208.1 199.2 178.4 173. 172.4 165.3 170.2 165.6 162.4 173.4 181.7 184.6

178.5 186.7 198.9 167.3 168.9 175.7 181.5 186.6 156.9 157.9 172. 173.5

173.6 158.7 169.7 166.3 168.7 176.2 175.6 183.5 187.8 171.7 159.3 165.7

180.2 183.1 188.8]

width : [64.1 65.5 66.2 66.4 66.3 71.4 67.9 64.8 66.9 70.9 60.3 63.6 63.8 64.6

63.9 64. 65.2 62.5 66. 61.8 69.6 70.6 64.2 65.7 66.5 66.1 70.3 71.7

70.5 72. 68. 64.4 65.4 68.4 68.3 65. 72.3 66.6 63.4 65.6 67.7 67.2

68.9 68.8]

height : [48.8 52.4 54.3 53.1 55.7 55.9 52. 53.7 56.3 53.2 50.8 50.6 59.8 50.2

52.6 54.5 58.3 53.3 54.1 51. 53.5 51.4 52.8 47.8 49.6 55.5 54.4 56.5

58.7 54.9 56.7 55.4 54.8 49.4 51.6 54.7 55.1 56.1 49.7 56. 50.5 55.2

52.5 53. 59.1 53.9 55.6 56.2 57.5]

curb\_weight : [2548 2823 2337 2824 2507 2844 2954 3086 3053 2395 2710 2765 3055 3230

3380 3505 1488 1874 1909 1876 2128 1967 1989 2191 2535 2811 1713 1819

1837 1940 1956 2010 2024 2236 2289 2304 2372 2465 2293 2734 4066 3950

1890 1900 1905 1945 1950 2380 2385 2500 2410 2443 2425 2670 2700 3515

3750 3495 3770 3740 3685 3900 3715 2910 1918 1944 2004 2145 2370 2328

2833 2921 2926 2365 2405 2403 1889 2017 1938 1951 2028 1971 2037 2008

2324 2302 3095 3296 3060 3071 3139 3020 3197 3430 3075 3252 3285 3485

3130 2818 2778 2756 2800 3366 2579 2460 2658 2695 2707 2758 2808 2847

2050 2120 2240 2190 2340 2510 2290 2455 2420 2650 1985 2040 2015 2280

3110 2081 2109 2275 2094 2122 2140 2169 2204 2265 2300 2540 2536 2551

2679 2714 2975 2326 2480 2414 2458 2976 3016 3131 3151 2261 2209 2264

2212 2319 2254 2221 2661 2563 2912 3034 2935 3042 3045 3157 2952 3049

3012 3217 3062]

engine\_type : ['dohc' 'ohcv' 'ohc' 'l' 'rotor' 'ohcf' 'dohcv']

num\_of\_cylinders : ['four' 'six' 'five' 'three' 'twelve' 'two' 'eight']

engine\_size : [130 152 109 136 131 108 164 209 61 90 98 122 156 92 79 110 111 119

258 326 91 70 80 140 134 183 234 308 304 97 103 120 181 151 194 203

132 121 146 171 161 141 173 145]

fuel\_system : ['mpfi' '2bbl' 'mfi' '1bbl' 'spfi' '4bbl' 'idi' 'spdi']

bore : ['3.47' '2.68' '3.19' '3.13' '3.5' '3.31' '3.62' '2.91' '3.03' '2.97'

'3.34' '3.6' '2.92' '3.15' '3.43' '3.63' '3.54' '3.08' '?' '3.39' '3.76'

'3.58' '3.46' '3.8' '3.78' '3.17' '3.35' '3.59' '2.99' '3.33' '3.7'

'3.61' '3.94' '3.74' '2.54' '3.05' '3.27' '3.24' '3.01']

stroke : ['2.68' '3.47' '3.4' '2.8' '3.19' '3.39' '3.03' '3.11' '3.23' '3.46' '3.9'

'3.41' '3.07' '3.58' '4.17' '2.76' '3.15' '?' '3.16' '3.64' '3.1' '3.35'

'3.12' '3.86' '3.29' '3.27' '3.52' '2.19' '3.21' '2.9' '2.07' '2.36'

'2.64' '3.08' '3.5' '3.54' '2.87']

compression\_ratio : [ 9. 10. 8. 8.5 8.3 7. 8.8 9.5 9.6 9.41 9.4 7.6

9.2 10.1 9.1 8.1 11.5 8.6 22.7 22. 21.5 7.5 21.9 7.8

8.4 21. 8.7 9.31 9.3 7.7 22.5 23. ]

horsepower : ['111' '154' '102' '115' '110' '140' '160' '101' '121' '182' '48' '70'

'68' '88' '145' '58' '76' '60' '86' '100' '78' '90' '176' '262' '135'

'84' '64' '120' '72' '123' '155' '184' '175' '116' '69' '55' '97' '152'

'200' '95' '142' '143' '207' '288' '?' '73' '82' '94' '62' '56' '112'

'92' '161' '156' '52' '85' '114' '162' '134' '106']

peak\_rpm : ['5000' '5500' '5800' '4250' '5400' '5100' '4800' '6000' '4750' '4650'

'4200' '4350' '4500' '5200' '4150' '5600' '5900' '5750' '?' '5250' '4900'

'4400' '6600' '5300']

city\_mpg : [21 19 24 18 17 16 23 20 15 47 38 37 31 49 30 27 25 13 26 36 22 14 45 28

32 35 34 29 33]

highway\_mpg : [27 26 30 22 25 20 29 28 53 43 41 38 24 54 42 34 33 31 19 17 23 32 39 18

16 37 50 36 47 46]

price : ['13495' '16500' '13950' '17450' '15250' '17710' '18920' '23875' '?'

'16430' '16925' '20970' '21105' '24565' '30760' '41315' '36880' '5151'

'6295' '6575' '5572' '6377' '7957' '6229' '6692' '7609' '8558' '8921'

'12964' '6479' '6855' '5399' '6529' '7129' '7295' '7895' '9095' '8845'

'10295' '12945' '10345' '6785' '11048' '32250' '35550' '36000' '5195'

'6095' '6795' '6695' '7395' '10945' '11845' '13645' '15645' '8495'

'10595' '10245' '10795' '11245' '18280' '18344' '25552' '28248' '28176'

'31600' '34184' '35056' '40960' '45400' '16503' '5389' '6189' '6669'

'7689' '9959' '8499' '12629' '14869' '14489' '6989' '8189' '9279' '5499'

'7099' '6649' '6849' '7349' '7299' '7799' '7499' '7999' '8249' '8949'

'9549' '13499' '14399' '17199' '19699' '18399' '11900' '13200' '12440'

'13860' '15580' '16900' '16695' '17075' '16630' '17950' '18150' '12764'

'22018' '32528' '34028' '37028' '9295' '9895' '11850' '12170' '15040'

'15510' '18620' '5118' '7053' '7603' '7126' '7775' '9960' '9233' '11259'

'7463' '10198' '8013' '11694' '5348' '6338' '6488' '6918' '7898' '8778'

'6938' '7198' '7788' '7738' '8358' '9258' '8058' '8238' '9298' '9538'

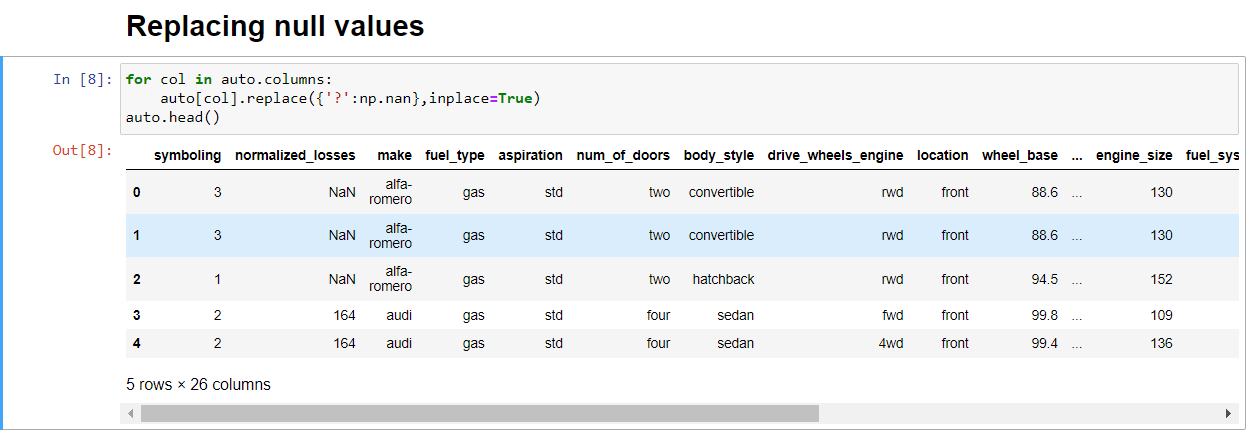
'8449' '9639' '9989' '11199' '11549' '17669' '8948' '10698' '9988'

'10898' '11248' '16558' '15998' '15690' '15750' '7975' '7995' '8195'

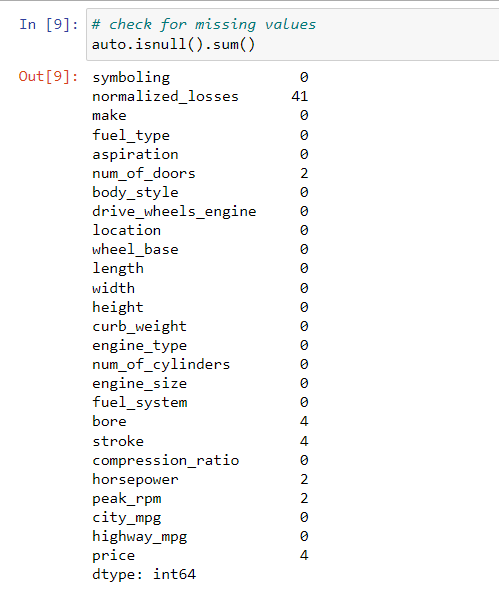
'9495' '9995' '11595' '9980' '13295' '13845' '12290' '12940' '13415'

'15985' '16515' '18420' '18950' '16845' '19045' '21485' '22470' '22625']

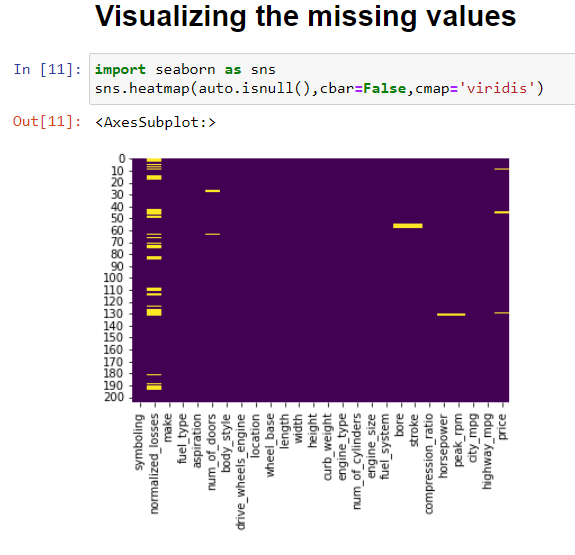
**Step 7:** There are null values in our dataset in form of ‘?’ only but pandas are not reading them so we will replace them into *np.nan* form.



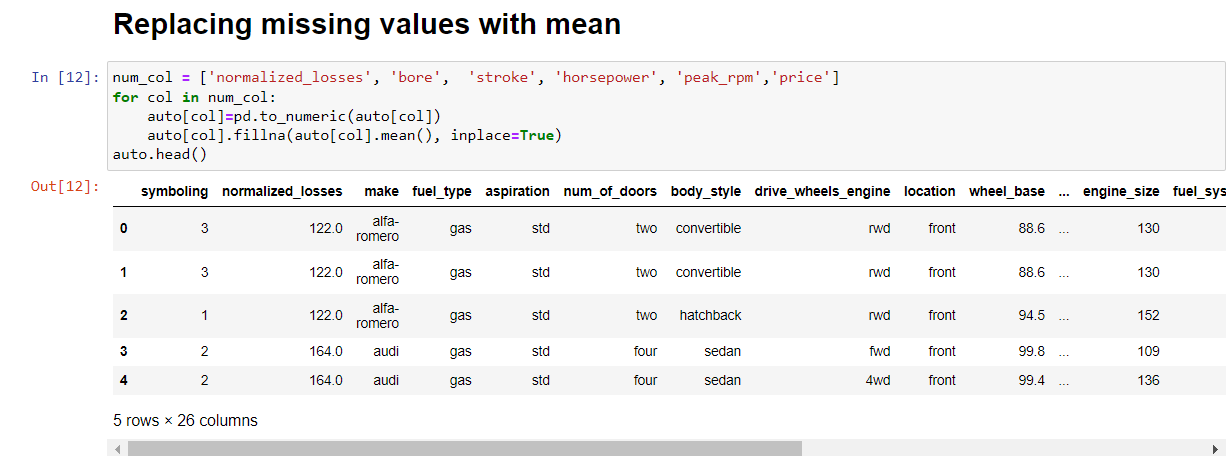
**Step 8:** Check for missing values after replacing null values.



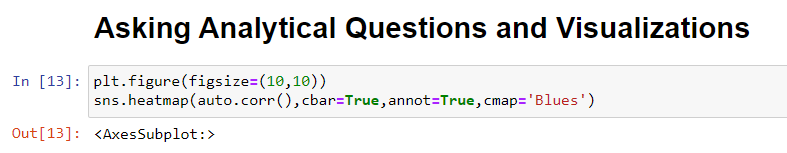
**Step 9:** With the help of heat map, we can see the amount of data that is missing from the attribute. With this, we can make decisions whether to drop these missing values or to replace them. Usually dropping the missing values is not advisable but sometimes it may be helpful too.

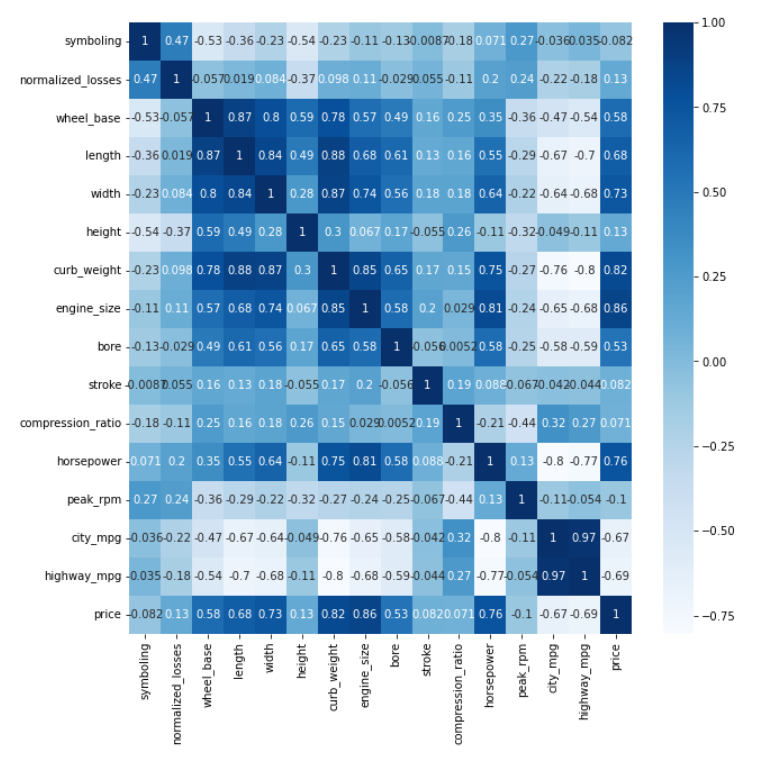


**Step 10:** Replace missing values with mean because the number of missing values is less



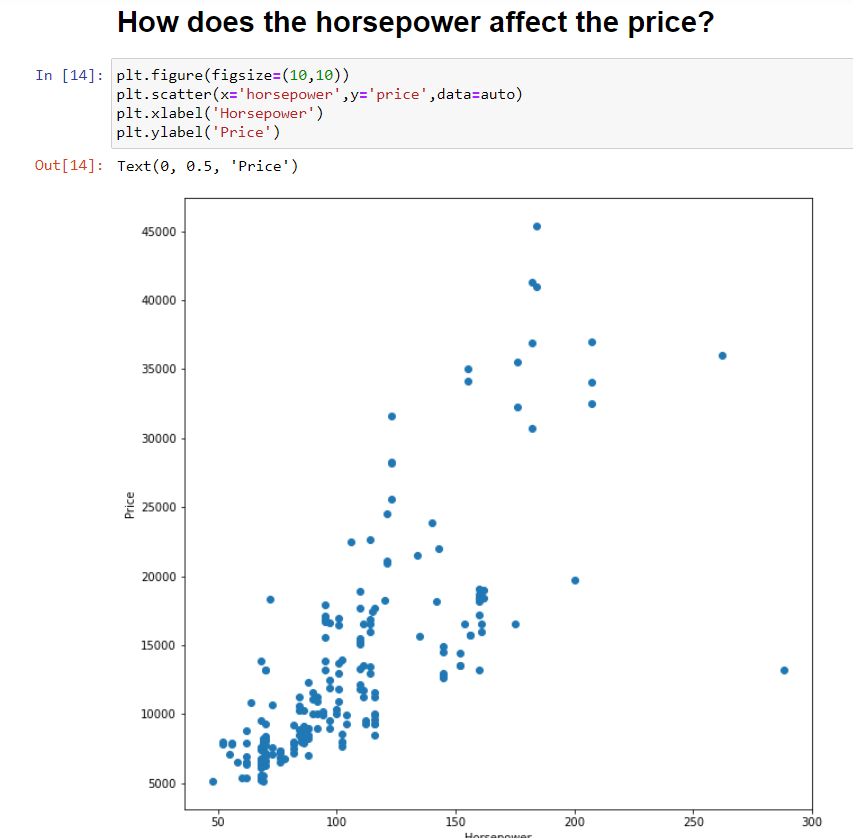
**Step 11:** This is the most important step in EDA. Try to ask questions related to independent variables and the target variable. Before this let us check the correlation between different variables, this will give us a roadmap on how to proceed further.

****

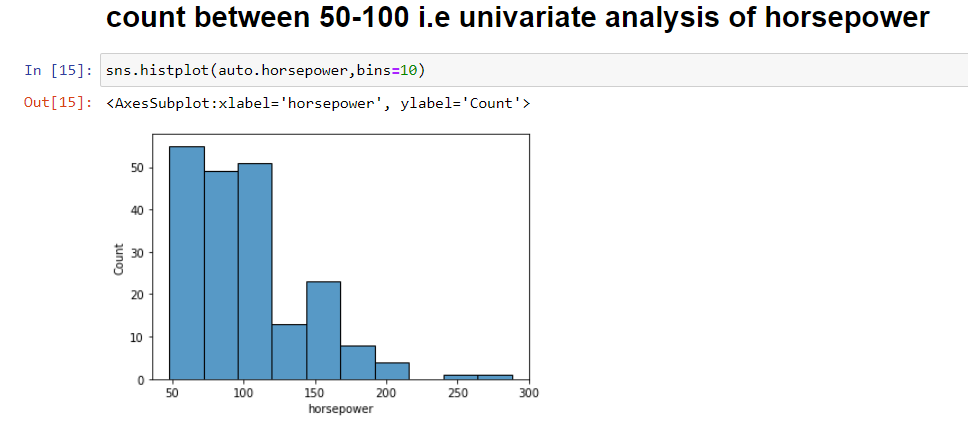


This heat map has given us great insights into the data.

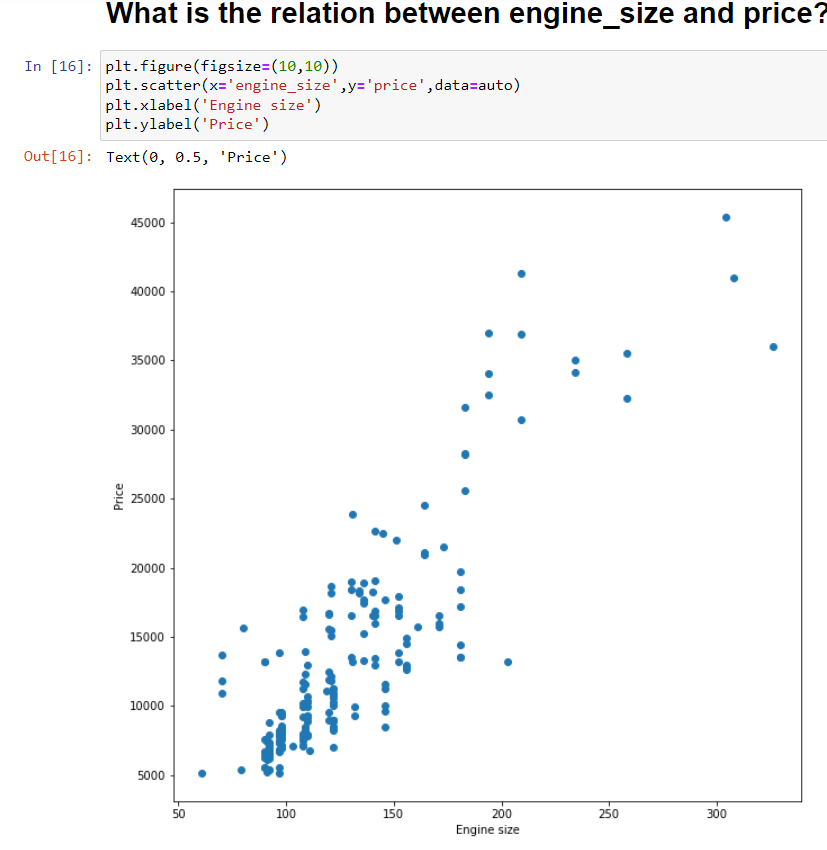
**Step 12:** Now let’s apply domain knowledge and ask the questions which will affect the price of the automobile.



**Step 13:** We can see that most of the horsepower value lies between 50-150 has price mostly between 5000-25000, there are outliers also (between 200-300). Let’s see a count between 50-100 i.e. univariate analysis of horsepower.

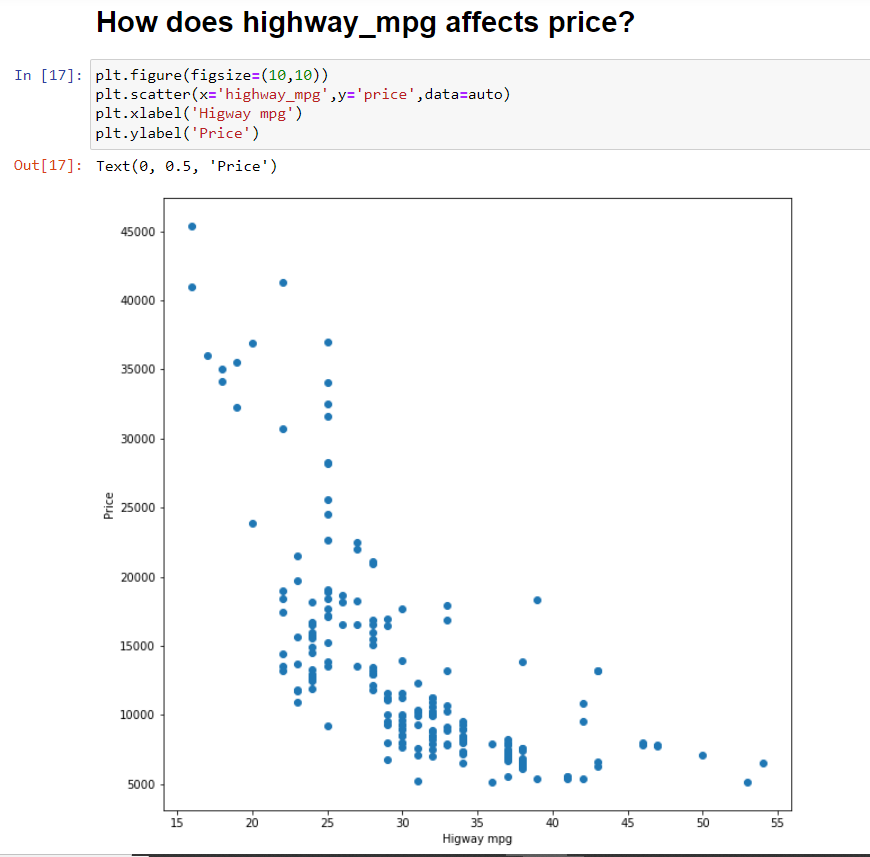


**Step 14:** Here we can view relation between engine size and price via graph.



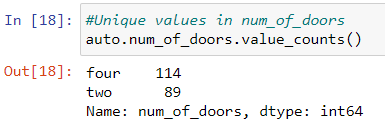
We can observe that the pattern is similar to horsepower vs price.

**Step 15:** Here we can view how does highway-mpg affects price via graph.

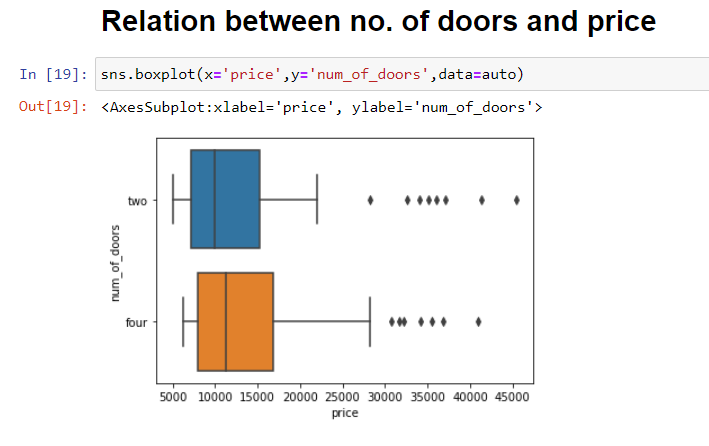


From the above graph we can see price decreases with an increase in higway\_mpg.

**Step 16:** lets’ check the number of doors using value\_counts() command.



**Step 17:** In this analysis step we’ll view the relation between number of doors and price using box plot.



**Step 18:** With this box plot, we can conclude that the average price of a vehicle with two doors is 10000, and the average price of a vehicle with four doors is 12000. With this plot, we have gained enough insights from data and our data is ready to build a model.