CUBIC ZIRCONIA PRICE ANALYSIS

PROBLEM 1: Linear Regression

1.1. Read the data and do exploratory data analysis. Describe the data briefly. (Check the null values, Data types, shape, EDA). Perform Univariate and Bivariate Analysis.

Data Set:

- Read data from csv file and convert it into Data Frame
- For reading the data from csv file I use read_csv method of pandas

	carat	cut	color	clarity	depth	table	x	y	z	price
0	0.30	Ideal	E	SI1	62.1	58.0	4.27	4.29	2.66	499
1	0.33	Premium	G	IF	60.8	58.0	4.42	4.46	2.70	984
2	0.90	Very Good	E	VVS2	62.2	60.0	6.04	6.12	3.78	6289
3	0.42	Ideal	F	VS1	61.6	56.0	4.82	4.80	2.96	1082
4	0.31	Ideal	F	VVS1	60.4	59.0	4.35	4.43	2.65	779
...	...	...	...	...	...	...	...	...	...	...
26962	1.11	Premium	G	SI1	62.3	58.0	6.61	6.52	4.09	5408
26963	0.33	Ideal	H	IF	61.9	55.0	4.44	4.42	2.74	1114
26964	0.51	Premium	E	VS2	61.7	58.0	5.12	5.15	3.17	1656
26965	0.27	Very Good	F	VVS2	61.8	56.0	4.19	4.20	2.60	682
26966	1.25	Premium	J	SI1	62.0	58.0	6.90	6.88	4.27	5166

26967 rows × 10 columns

Data Dictionary:

- Data Dictionary means what all columns represent in dataset.
- Following table contain Column name and Description

Column Name	Description
Carat	Carat weight of the cubic zirconia.
Cut	Describe the cut quality of the cubic zirconia. Quality is increasing order Fair, Good, Very Good, Premium, Ideal.
Color	Color of the cubic zirconia. With D being the worst and the best.
Clarity	Cubic zirconia Clarity refers to the absence of the Inclusions and Blemishes. (In order from Best to Worst, IF = flawless, 11 = level 1 inclusion) IF, VVS1 VVS2, VS1, VS2. SI1, S12, 11
Depth	The Height of cubic zirconia, measured from the Culet to the table, divided by its average Girdle Diameter
Table	The Width of the cubic zirconia's Table expressed as a Percentage of its Average Diameter.
Price	The Price of the cubic zirconia.
X	Length of the cubic zirconia in mm.
Y	Width of the cubic zirconia in mm.
Z	Height of the cubic zirconia in mm.

Check the null values:

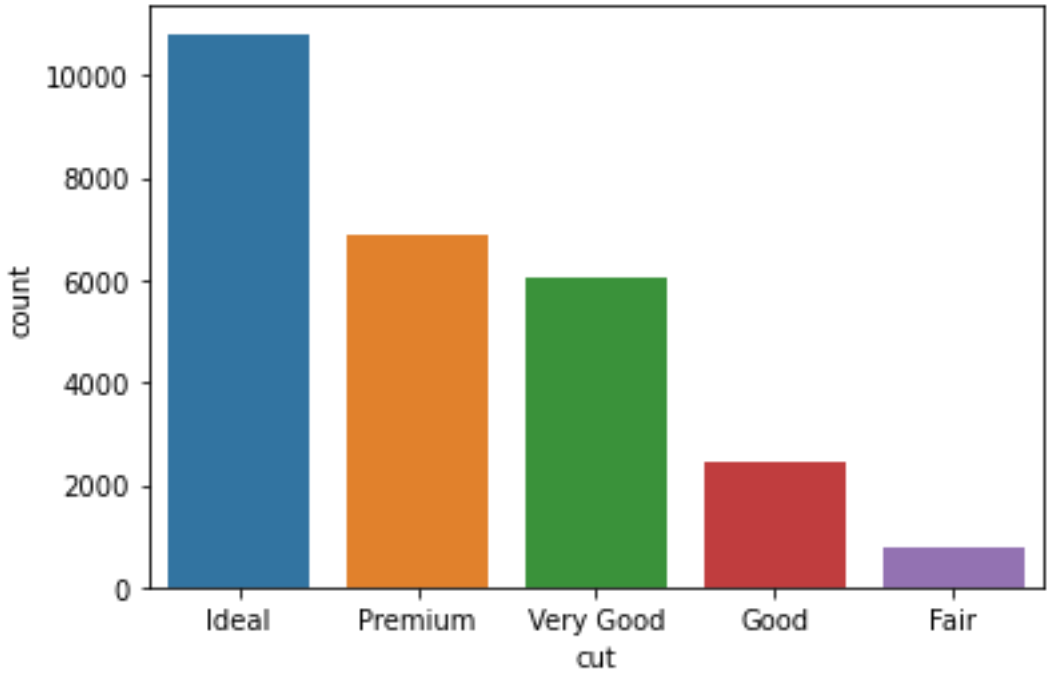
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 26967 entries, 0 to 26966
Data columns (total 10 columns):
 #   Column   Non-Null Count  Dtype  
---  ------   --------------  -----  
 0   carat    26967 non-null  float64
 1   cut      26967 non-null  object 
 2   color    26967 non-null  object 
 3   clarity  26967 non-null  object 
 4   depth    26270 non-null  float64
 5   table    26967 non-null  float64
 6   x        26967 non-null  float64
 7   y        26967 non-null  float64
 8   z        26967 non-null  float64
 9   price    26967 non-null  int64  
dtypes: float64(6), int64(1), object(3)
memory usage: 2.1+ MB
```

- We can see from upper image there is null value is present in depth column because not null count of that column is less than total number of rows.
- Cut, color and clarity have object datatype.
- Carat, depth, table, x, y and z have float64 datatype
- Only price column has integer datatype
- There is total 26967 rows and 10 columns present in data set.

Univariate Analysis:

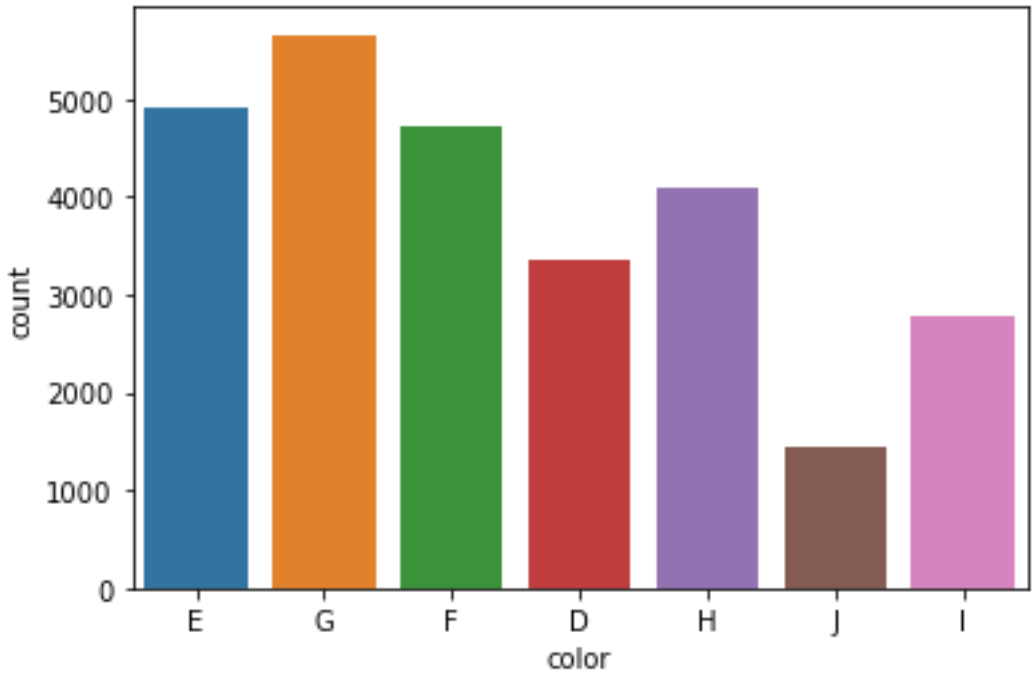
- **Cut Analysis:**



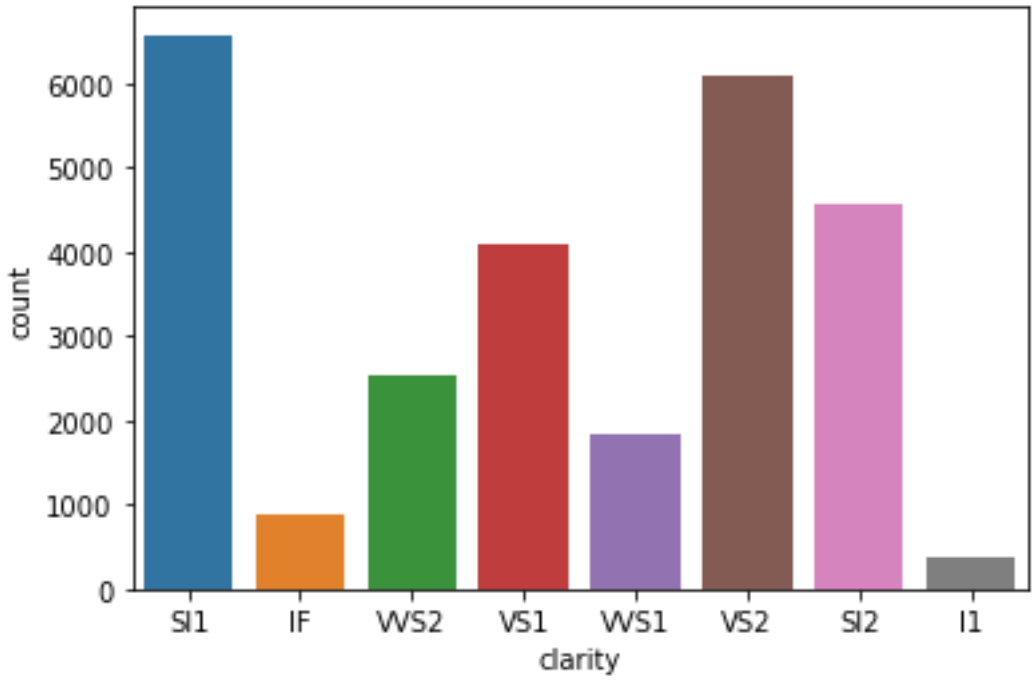
- From above graph we can see that people are mostly prefer to buy Ideal or Premium cube.
- People are not interested to buy Fair and Good quality zirconia cube.

- **Color Analysis:**

- From below figure we can say people are more interested to buy E,F and G color quality cube and they are not interested to buy J, I and D color cube.

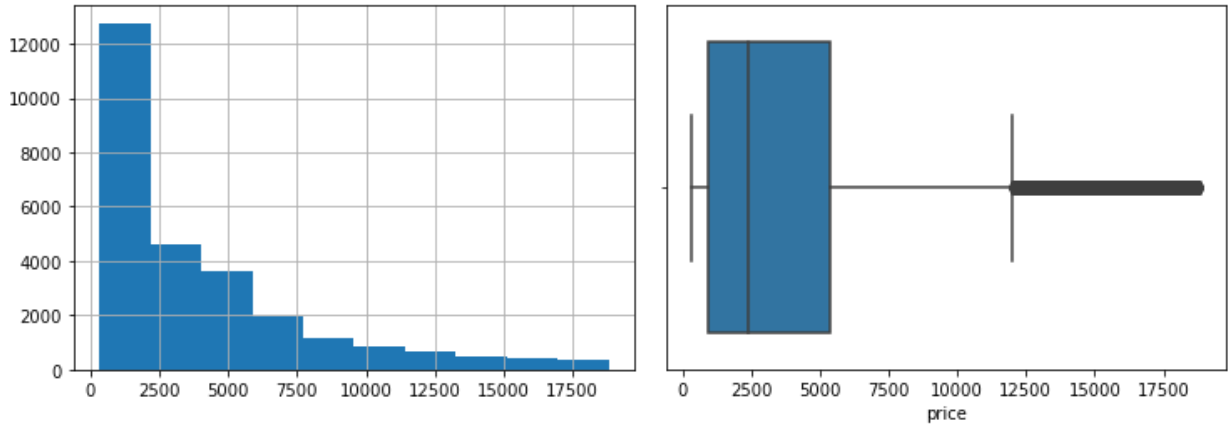


- **Clarity Analysis**

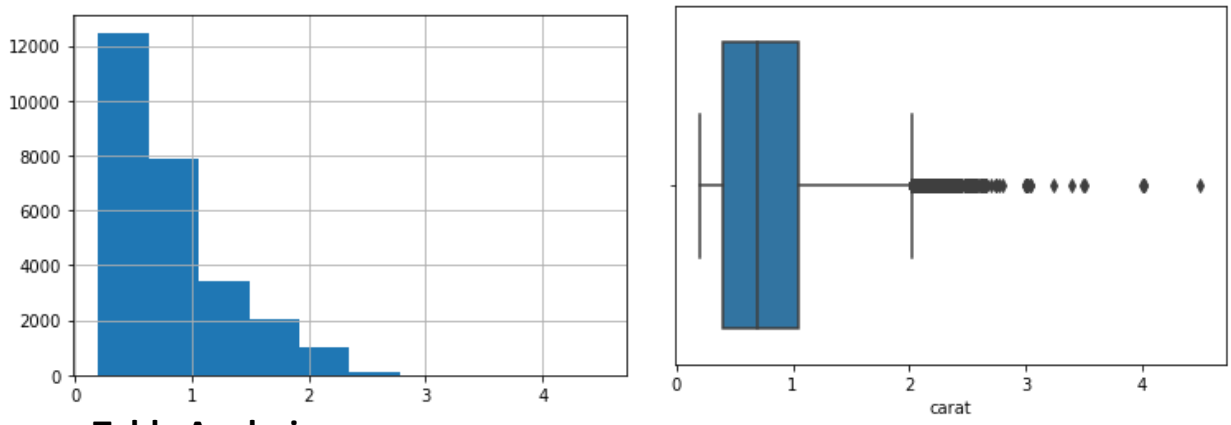


- From above figure we can say people are highly prefer to buy SL1, VS2 and SL2 clarity cube so company should focus on it.
- Only few people are interested to buy L1, IF VVS1 clarity cube.

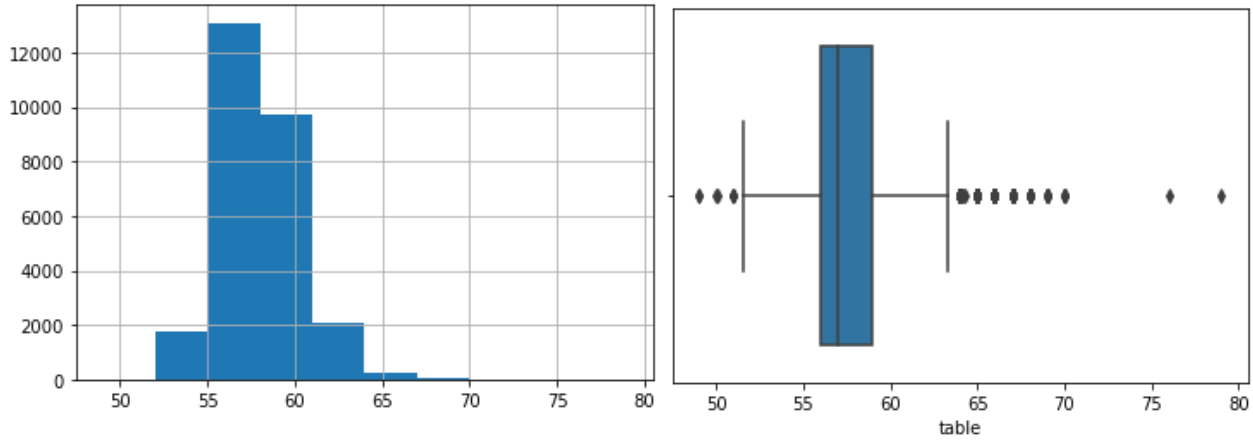
- **Price Analysis:**



- **Carat Analysis:**

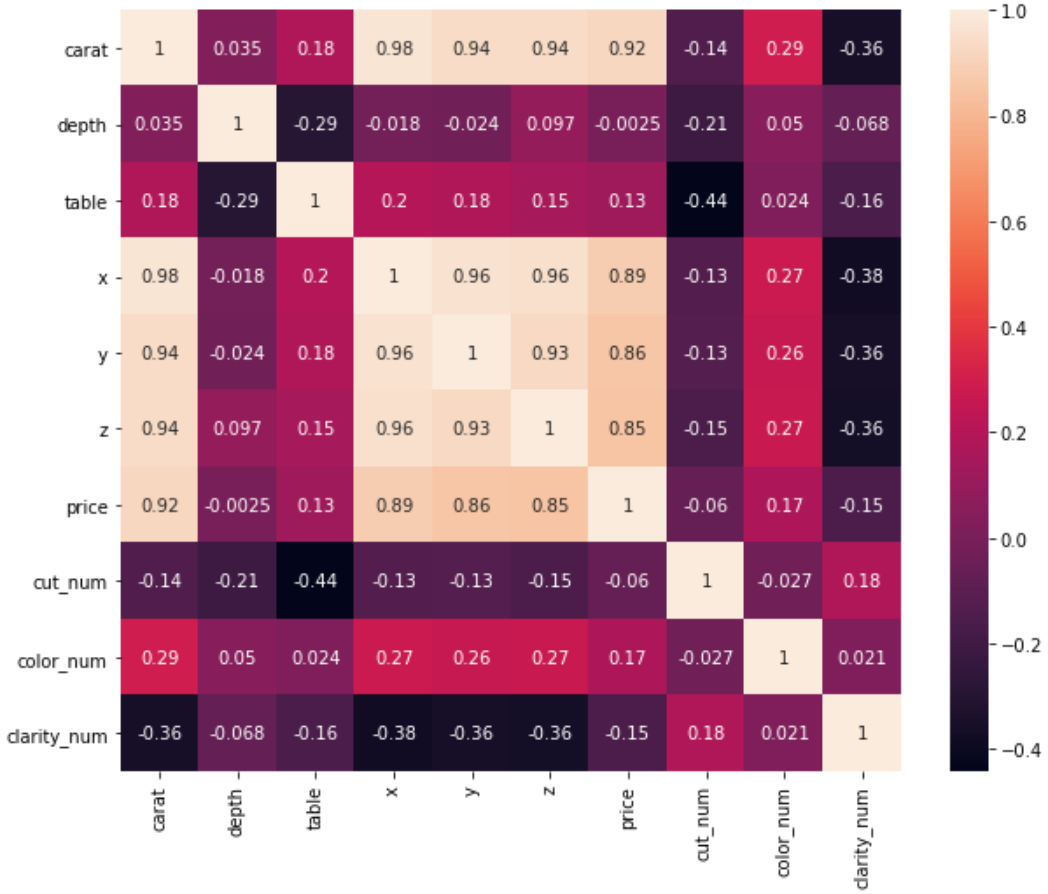


- **Table Analysis:**



➢ From above 3 boxplot and histogram we can say price, carat and table columns contain outlier means some values which is not fit in boxplot range.

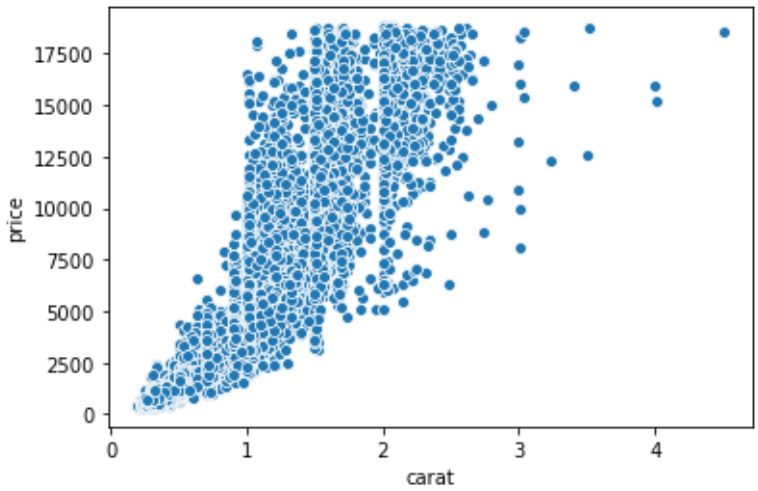
- Heatmap of Data Frame



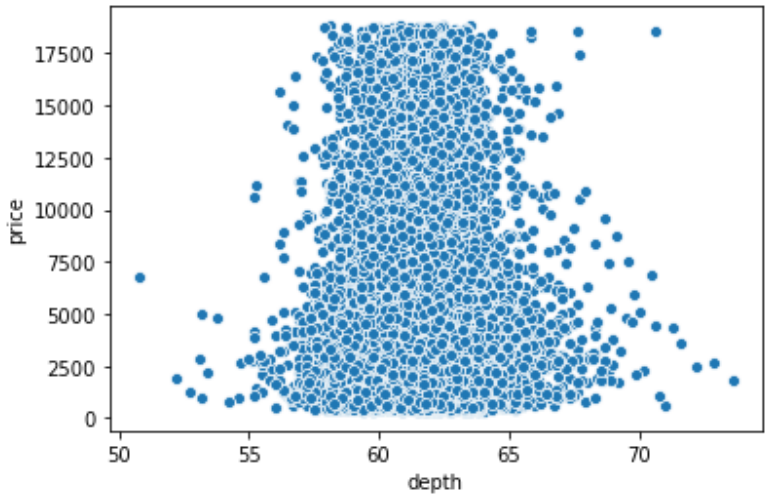
- In heat map all dark shad are highly correlate each other.
- From above observation we can say clarity_num is highly negative corelate with carat, x, y, and z.
- Cut_num is highly negative correlate with table.
- Table is highly negative correlate with depth.
- X, Y and Z are highly positive correlate with carat.
- X, Y and Z are also highly positive correlate with price.

Bivariate Analysis:

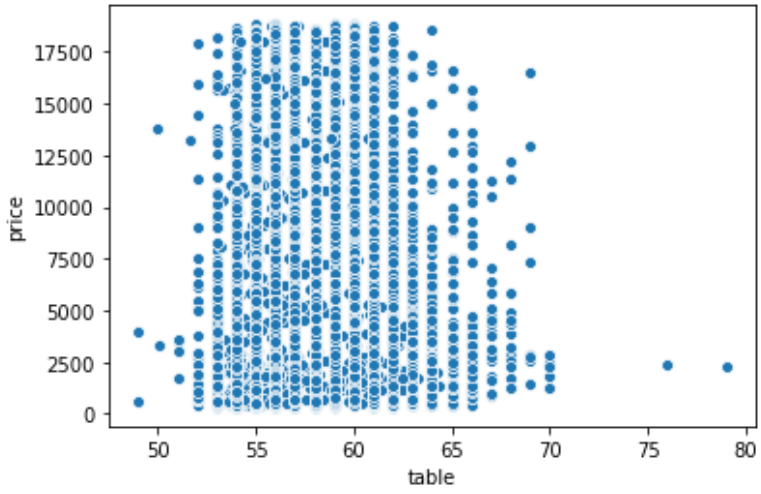
- Carat VS Price



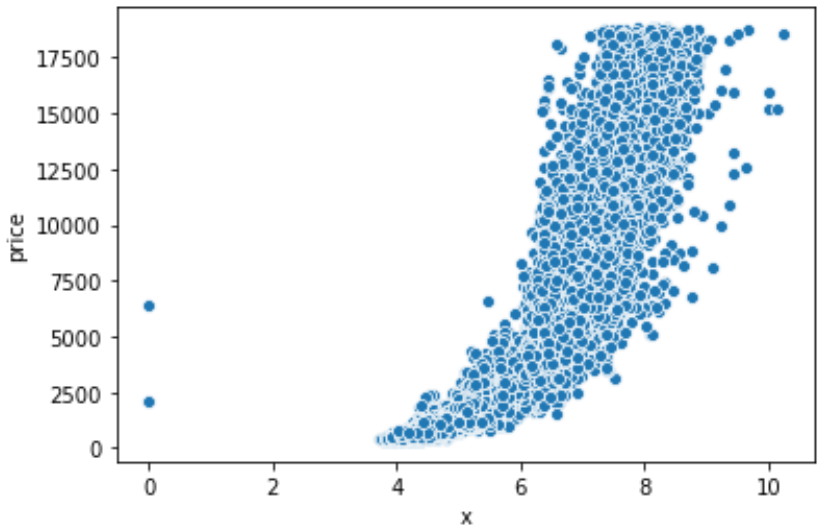
- Depth VS Price



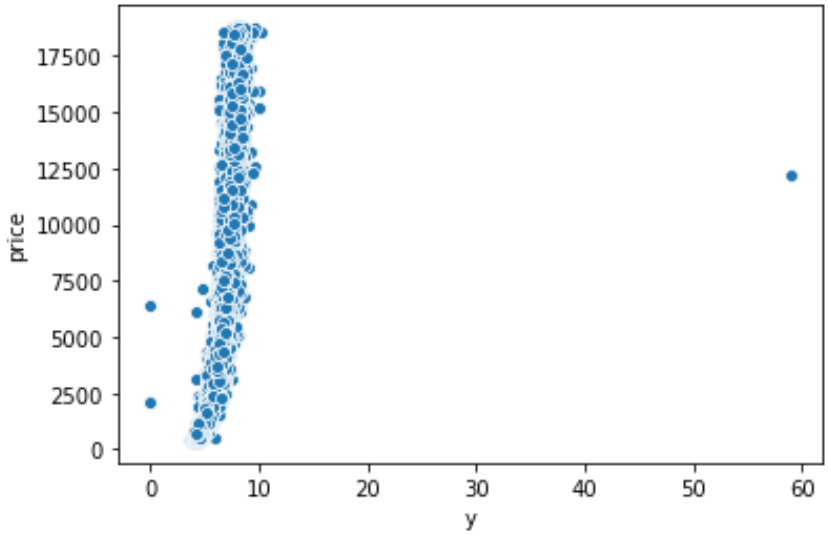
- Table VS Price



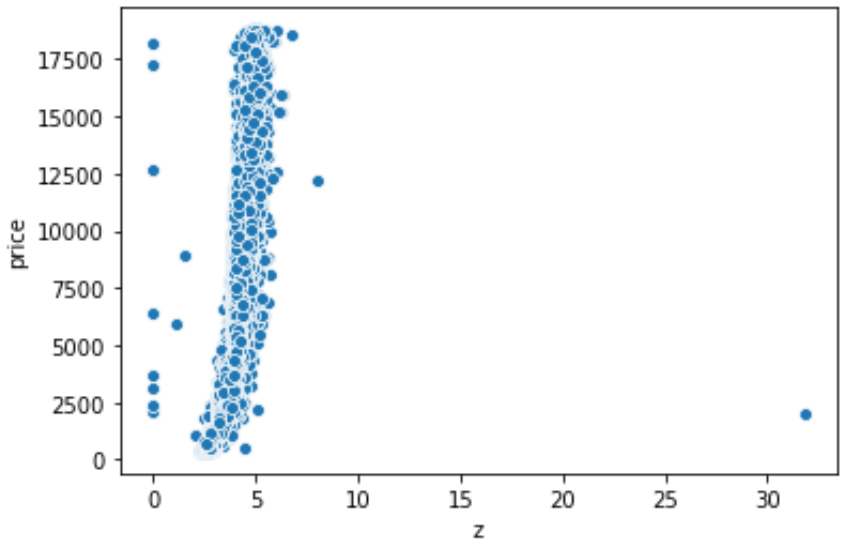
- X VS Price



- Y VS Price



- Z VS Price



Analysis from above scatter plot:

- We can see here price and carat are in linear relation.
- Depth and table are not in linear relationship with price.
- X, Y, and Z are linear relationship in relationship with price.

1.2. Input null values if present, also check for the values which are equal to zero. Do they have any meaning or do we need to change them or drop them? Do you think scaling is necessary in this case?

```
carat        0
cut          0
color        0
clarity      0
depth      697
table        0
x            0
y            0
z            0
price        0
dtype: int64
```

Here 697 null values are present in-depth column so we should fill them with mean of depth column.

```
df.depth.fillna(df.depth.mean(),inplace=True)
```

As shown in upper figure I used fillna method to fill missing values in depth column and I fill NaN values with mean of that perticular column.

We can also use interpolet method but here filling NaN values by mean is good for Linear Regression.

1.3. Encode the data (having string values) for Modelling. Data Split: Split the data. Data Split: Split the data into train and test (70:30). Apply Linear Regression. Performance Metrics: Check the performance of Predictions on Train and Test sets using Rsquare, RMSE.

Encode The Data:

- **Cut**

  Here in Dataset cut column contain categorical data and ML algorithm can't understand categorical data so we must convert that data into numerical data.
  Here quality of zirconia cube is incrise as order Fair, Good, Very Good, Premium, Ideal so we must encode that as shown below.

Fair	1
Good	2
Very Good	3
Premium	4
Ideal	5

- **Color**

  Here in Dataset color column contain categorical data and ML algorithm can't understand categorical data so we must convert that data into numerical data.
  Here color quality of zirconia cube is incrise as D,E,F,G,H,I,J alphabetic order incrise so I use labeencoder method to transform data into numbers and below table shows perticular number of alphabet.

D	0
E	1
F	2
G	3
H	4
I	5
J	6

- **Clarity**

  Here clarity of zirconia cube is incrising order from worst to best as following order L1,SL2,SL1,VS2,VS1,VVS2,VVS1,IF so I transfer categorical data into numeric data as following table.

L1	0

SL2	1
SL1	2
VS2	3
VS1	4
VVS2	5
VVS1	6
IF	7

After Encoding:

	carat	cut	color	clarity	depth	table	x	y	z	price	cut_num	color_num	clarity_num
0	0.30	Ideal	E	SI1	62.1	58.0	4.27	4.29	2.66	499	5	1	2
1	0.33	Premium	G	IF	60.8	58.0	4.42	4.46	2.70	984	4	3	7
2	0.90	Very Good	E	VVS2	62.2	60.0	6.04	6.12	3.78	6289	3	1	5
3	0.42	Ideal	F	VS1	61.6	56.0	4.82	4.80	2.96	1082	5	2	4
4	0.31	Ideal	F	VVS1	60.4	59.0	4.35	4.43	2.65	779	5	2	6
5	1.02	Ideal	D	VS2	61.5	56.0	6.46	6.49	3.99	9502	5	0	3
6	1.01	Good	H	SI1	63.7	60.0	6.35	6.30	4.03	4836	2	4	2
7	0.50	Premium	E	SI1	61.5	62.0	5.09	5.06	3.12	1415	4	1	2
8	1.21	Good	H	SI1	63.8	64.0	6.72	6.63	4.26	5407	2	4	2
9	0.35	Ideal	F	VS2	60.5	57.0	4.52	4.60	2.76	706	5	2	3

Split Data:

I use train_test_split method to split dataset into two parts 1) Training dataset 2) Testing dataset.

Using this method I devide dataset into 70:30 ratio for better model training

Following are dataset to train model.

	carat	depth	table	x	y	z	cut_num	color_num	clarity_num
7737	0.40	63.600000	57.0	4.69	4.65	2.97	2	1	2
11202	2.06	61.745147	55.0	8.23	8.19	5.08	5	6	1
5345	0.33	61.500000	57.0	4.46	4.49	2.75	5	5	2
16036	0.31	62.100000	55.0	4.35	4.32	2.69	5	0	3
10663	0.31	60.800000	56.0	4.38	4.41	2.67	5	5	2
...	...	...	...	...	...	...	...	...	...
21807	1.61	62.900000	56.0	7.52	7.46	4.71	5	2	3
21586	0.34	60.600000	60.0	4.53	4.48	2.73	4	0	1
17218	1.01	61.800000	60.0	6.37	6.41	3.95	3	0	1
24931	1.06	62.100000	58.0	6.50	6.52	4.04	4	2	2
26644	0.51	61.600000	58.0	5.09	5.14	3.15	3	6	3

18876 rows × 9 columns

```
y_train
```

```
7737        882
11202     11337
5345        445
16036       942
10663       436
          ...
21807     15426
21586       650
17218      4588
24931      5142
26644      1008
Name: price, Length: 18876, dtype: int64
```

- Here 9 columns and 18875 rows are present in x_train dataset.
- In y_train dataset here 18876 length series present.
- This both x_train and y_train are contain 70% of original dataset because we use 70:30 ratio for split dataset.

Following are testing dataset for model.

```
x_test
```

	carat	depth	table	x	y	z	cut_num	color_num	clarity_num
6376	0.53	60.9	60.0	5.22	5.16	3.16	4	1	3
21323	1.03	62.2	54.0	6.46	6.50	4.03	5	5	2
10761	0.42	62.1	59.0	4.77	4.79	2.97	4	0	2
12716	1.22	61.8	57.0	6.90	6.83	4.23	5	5	1
20717	0.91	58.0	57.0	6.36	6.47	3.72	2	2	2
...	...	...	...	...	...	...	...	...	...
18209	0.77	62.1	57.0	5.88	5.84	3.64	5	2	4
8349	0.73	63.1	59.0	5.76	5.72	3.62	3	1	1
55	1.02	62.4	58.0	6.42	6.47	4.02	3	2	4
2089	1.02	63.3	58.0	6.42	6.38	4.05	3	3	4
19813	1.45	60.6	61.0	7.32	7.41	4.46	3	5	7

8091 rows × 9 columns

```
y_test
```

```
6376     1813
21323    4782
10761     810
12716    4612
20717    4067
         ...
18209    3387
8349     1975
55       7587
2089     6861
19813    9683
Name: price, Length: 8091, dtype: int64
```

- Here 9 columns and 8091 rows are present in x_test dataset.
- In y_test dataset here 18876 length series present.
- This both x_test and y_test are contain 30% of original dataset because we use 70:30 ratio for split dataset.

Apply LinearRegression:

```
from sklearn.linear_model import LinearRegression
lr = LinearRegression()
```

```
lr.fit(x_train,y_train)
```

```
LinearRegression()
```

- From sklearn library first import LinearRegression.
- Then train model using x_train and y_train dataset as shown in upper figure.

Perfomance Check:

- **Score Of Model**

```
lr.score(x_test,y_test)
```

```
0.9102317749435431
```

- **R*R Score of Model**

```
r2_score(y_test,y_predicted)
```

```
0.9102317749435431
```

- **Mean Square Error**

```
mean_squared_error(y_test,y_predicted)
```

```
1484645.567671571
```

Score	91%
R*R	91%
Mean Square Error	1484645

- Here Mean square error is very large because dataset have around 27000 colums.
- Accuracy and R*R accuracy are around 91% and this is good for model.

1.4. Inference: Basis on these predictions, what are the business insights and recommendations.

- In cut category Ideal and Premium cubes are contain more than 50% of totall cell that means people are more preffer to buy Ideal or Premium category cube so companey should more focus on this two category for advertisement or production.

- There is very low chance for Fair and good category to sold because people don't preffer to buy this two category cube.
- In color column if cube are in E, F or G color than there is high chance for selling and which cube contain J or I color they are low chance of selling. So companey shoud focus on E, F or G color cube Production rather than producing I and J color Production.

- If cube's clarity is SL1, VS2 or SL2 then there is high chance for selling and cube's clarity is L1, IF or VVS1 then there is very low chance for selling so companey should focus on buldind or pramoting cubes which are contain SL1, VS2 or SL2 clarity.

- We can see clearly from price histogram, if cube price is incrise than chancee of selling that perticular cube is decrise. So companey shoud focus on buildind low price cube.

- We can see clearly from carat histogram, if carat is more than 1 means low chance of cube selling.

- It is clearly show in table histogram around 90% cube have table value in range 55-61 so companey shoud focus on it.