Name : Adepu Shivani

College : JNTUH UNIVERSITY COLLEGE OF ENGINEERING JAGITYAL (JNTUH UCEJ)

Branch : Information Technology

Year : 4th year

Contact : 6300554876

URL :

https://www.kaggle.com/datasets/doaaalsenani/usa-cers-dataset

(OR)

https://colab.research.google.com/drive/1IKVupFkfUgik_oSuN8E0Ky6Af0KFoTpX#scrollTo=EYgw7tVdMvQs

(OR)

https://colab.research.google.com/drive/1q_K7JEs6AfWjPdStbo5RYiRrYUsDqMaE?usp=sharing

```
# MINIPROJECT1
# MACHINE LEARNING FROM DATASETS->USA CARS DATASETS-EDA
# Exploratary data analysis(EDA)
#1 create dataframe
import pandas as pd
df=pd.read_csv('/content/USA_cars_datasets.csv')
df
```

	Unnamed: 0	price	brand	model	year	title_status	mileage	color	vin	lot	state	country	condition
0	0	6300	toyota	cruiser	2008	clean vehicle	274117.0	black	jtezu11f88k007763	159348797	new jersey	usa	10 days left
1	1	2899	ford	se	2011	clean vehicle	190552.0	silver	2fmdk3gc4bbb02217	166951262	tennessee	usa	6 days left
2	2	5350	dodge	mpv	2018	clean vehicle	39590.0	silver	3c4pdcgg5jt346413	167655728	georgia	usa	2 days left
3	3	25000	ford	door	2014	clean vehicle	64146.0	blue	1ftfw1et4efc23745	167753855	virginia	usa	22 hours left
4	4	27700	chevrolet	1500	2018	clean vehicle	6654.0	red	3gcpcrec2jg473991	167763266	florida	usa	22 hours left
...	...	...	...	...	...	...	...	...	...	...	...	...	...
2494	2494	7800	nissan	versa	2019	clean vehicle	23609.0	red	3n1cn7ap9kl880319	167722715	california	usa	1 days left
2495	2495	9200	nissan	versa	2018	clean vehicle	34553.0	silver	3n1cn7ap5jl884088	167762225	florida	usa	21 hours left
2496	2496	9200	nissan	versa	2018	clean vehicle	31594.0	silver	3n1cn7ap9jl884191	167762226	florida	usa	21 hours left
2497	2497	9200	nissan	versa	2018	clean vehicle	32557.0	black	3n1cn7ap3jl883263	167762227	florida	usa	2 days left
2498	2498	9200	nissan	versa	2018	clean vehicle	31371.0	silver	3n1cn7ap4jl884311	167762228	florida	usa	21 hours left

2499 rows × 13 columns

```
df.shape #2499 rows and 13 columns
df.size #Total number of elements in my dataframe
# To check the null values or missing values
df.isnull().sum()
```

```
# To check the null values or missing values
df.isnull().sum()
```

```
Unnamed: 0      0
price           0
brand           0
model           0
year            0
title_status    0
mileage         0
color           0
vin             0
lot             0
state           0
country         0
condition       0
dtype: int64
```

```
df.info()
```

```
[5] df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2499 entries, 0 to 2498
Data columns (total 13 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   Unnamed: 0    2499 non-null   int64  
 1   price         2499 non-null   int64  
 2   brand         2499 non-null   object 
 3   model         2499 non-null   object 
 4   year          2499 non-null   int64  
 5   title_status  2499 non-null   object 
 6   mileage       2499 non-null   float64
 7   color         2499 non-null   object 
 8   vin           2499 non-null   object 
 9   lot           2499 non-null   int64  
 10  state         2499 non-null   object 
 11  country       2499 non-null   object 
 12  condition     2499 non-null   object 
dtypes: float64(1), int64(4), object(8)
memory usage: 253.9+ KB
```

```
#I want to find out the no of unique elements/values in
 each and every column
df.nunique()
```

```
#I want to find out the no of unique elements/values in each and every column
df.nunique()
```

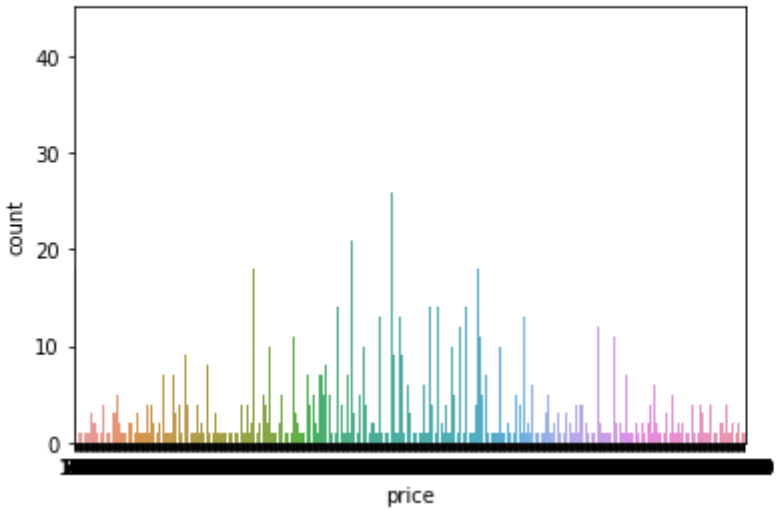
```
Unnamed: 0      2499
price            790
brand             28
model            127
year              30
title_status       2
mileage         2439
color             49
vin             2495
lot             2495
state             44
country            2
condition         47
dtype: int64
```

```
#VISUALISATION - SEABORN
# 1st Conclusion/fact
import seaborn as sns
sns.countplot(x = 'price',data = df)

#This count plot will tell us what's the price of all c
ars are there in usa car datasets
```

```
#This count plot will tell us what's the price of all cars are t
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f719a11b150>
```



```
#Finding out the exact count of prices aborad usa cars
df.groupby('price').size()
```

```
#Finding out the exact count o
df.groupby('price').size()
```

```
price
0        43
25       18
50        2
75        3
100       1
         ..
65500     1
67000     1
70000     1
74000     1
84900     1
Length: 790, dtype: int64
```

```
df['price'].value_counts()
```

```
0        43
16500    26
13900    21
15500    19
15000    19
         ..
12560     1
11760     1
7340      1
6530      1
30100     1
Name: price, Length: 790, dtype: int64
```

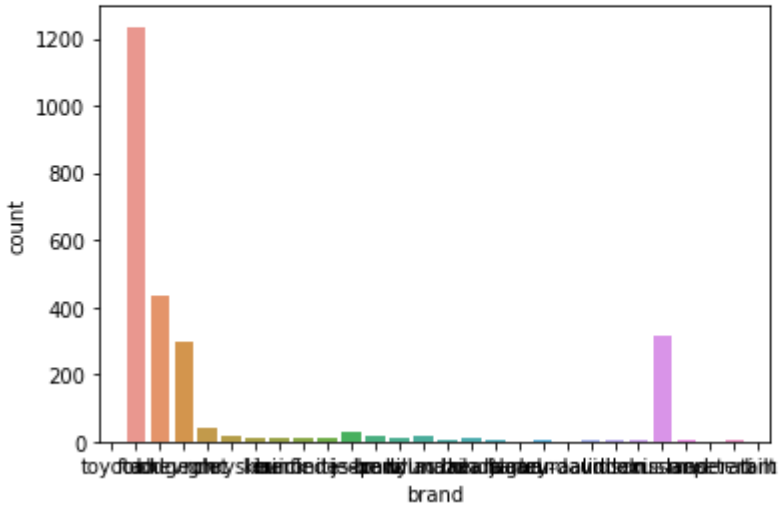
```
# 2nd Conclusion/fact

#This count plot will tell us types of brands are there in usa car datasets

sns.countplot(x = 'brand',data = df)
```

```
# 2nd Conclusion/fact
#This count plot will tell us types of brands are there in usa car datasets
sns.countplot(x = 'brand',data = df)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f719c3e8650>
```




```
#Finding out the exact count of no of brands aborad usa
 cars datasets
df.groupby('brand').size()
```

```
df.groupby('brand').size()

brand
acura                3
audi                 4
bmw                 17
buick               13
cadillac            10
chevrolet          297
chrysler            18
dodge              432
ford              1235
gmc                 42
harley-davidson      1
heartland            5
honda               12
hyundai             15
infiniti            12
jaguar               1
jeep                30
kia                 13
land                 4
lexus                2
lincoln              2
```

```
df['brand'].value_counts()
```

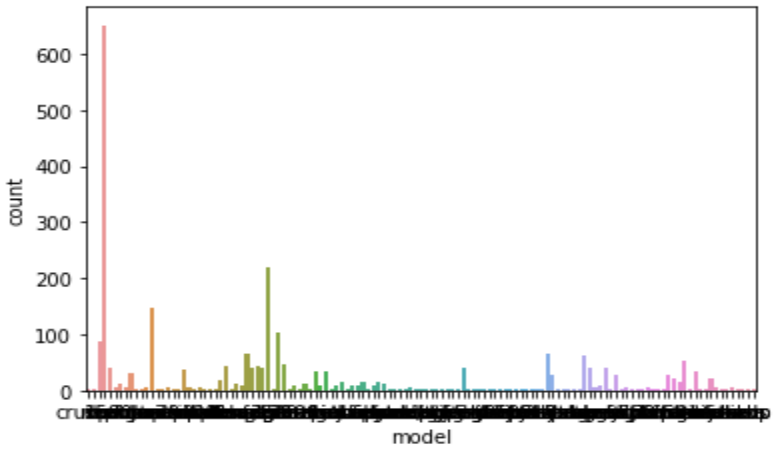
```
df['brand'].value_counts()

ford             1235
dodge             432
nissan            312
chevrolet         297
gmc                42
jeep               30
chrysler           18
bmw                17
hyundai            15
kia                13
buick              13
infiniti           12
honda              12
cadillac           10
mercedes-benz      10
heartland           5
land                4
peterbilt           4
audi                4
acura               3
lincoln             2
lexus               2
```

```
# 3rd Conclusion/fact
#This count plot will tell us types of models are there
 in usa car datasets
sns.countplot(x = 'model',data = df)
```

```
#This count plot will tell us types of models are there in us
sns.countplot(x = 'model',data = df)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f7198f88610>
```



```
#Finding out the exact count of no of models aborad usa car
s
df.groupby('model').size()
```

```
df.groupby('model').size()

model
1500      39
2500       8
2500hd     1
300        6
3500       4
          ..
wagon     30
x3         2
xd         1
xt5        1
xterra     1
Length: 127, dtype: int64
```

```
df['model'].value_counts()
```

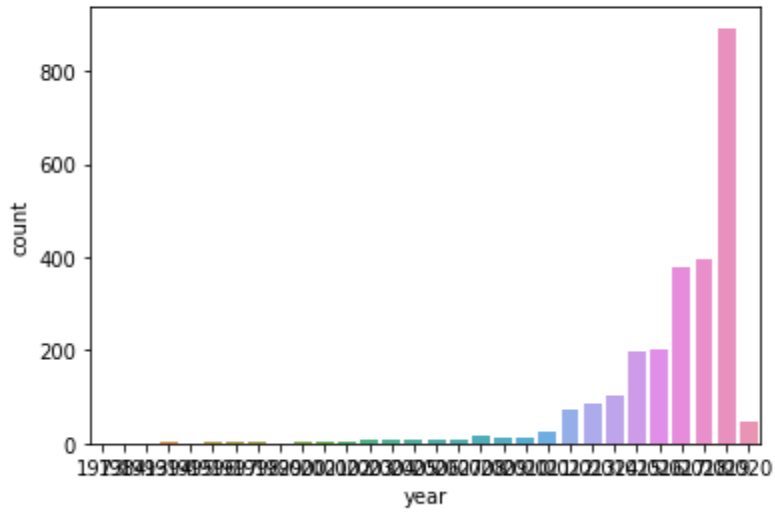
```
df['model'].value_counts()

door        651
f-150       219
doors       148
caravan     102
mpv          87
           ...
sl-class      1
cx-3          1
2500hd        1
mdx           1
nvp           1
Name: model, Length: 127, dtype: int64
```

```
# 4th Conclusion/fact
#This count plot will tell us about in which year how many
cars are prepared are there in usa car datasets
sns.countplot(x = 'year',data = df)
```

```
#This count plot will tell us about in which year how many c
sns.countplot(x = 'year',data = df)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f7199acd750>
```




```
#Finding out the exact count of no of years aborad usa cars
 datasets
df.groupby('year').size()
```

```
df.groupby('year').size()
```

```
year
1973      1
1984      1
1993      1
1994      2
1995      1
1996      2
1997      2
1998      4
1999      1
2000      4
2001      5
2002      2
2003      9
2004      6
2005      6
2006      8
2007      6
2008     18
2009     11
2010     13
2011     23
```

```
df['year'].value_counts()
```

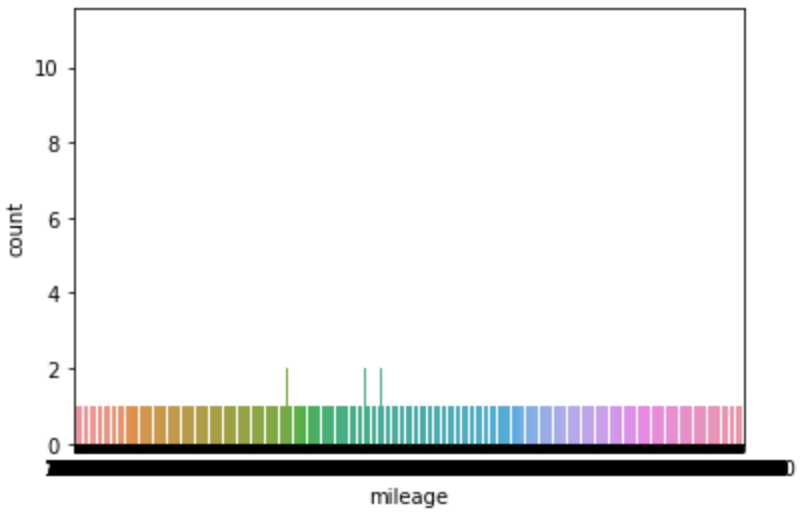
```
df['year'].value_counts()
```

```
2019    892
2018    395
2017    377
2016    203
2015    196
2014    104
2013     86
2012     72
2020     48
2011     23
2008     18
2010     13
2009     11
2003      9
2006      8
2004      6
2007      6
2005      6
2001      5
1998      4
2000      4
2002      2
```

```
# 5th Conclusion/fact
#This count plot will tell us about the mileage of cars are
 there in usa car datasets
sns.countplot(x = 'mileage',data = df)
```

```
# 5th Conclusion/fact
#This count plot will tell us about the mileage of cars are t
sns.countplot(x = 'mileage',data = df)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f71985bfed0>
```



```
#Finding out the exact count of no of mileages aborad usa c
ars
df.groupby('mileage').size()
```

```
df.groupby('mileage').size()

mileage
0.0           6
1.0          11
7.0           1
71.0          1
122.0         1
             ..
507985.0      1
902041.0      1
982486.0      1
999999.0      1
1017936.0     1
Length: 2439, dtype: int64
```

```
df['mileage'].value_counts()
```

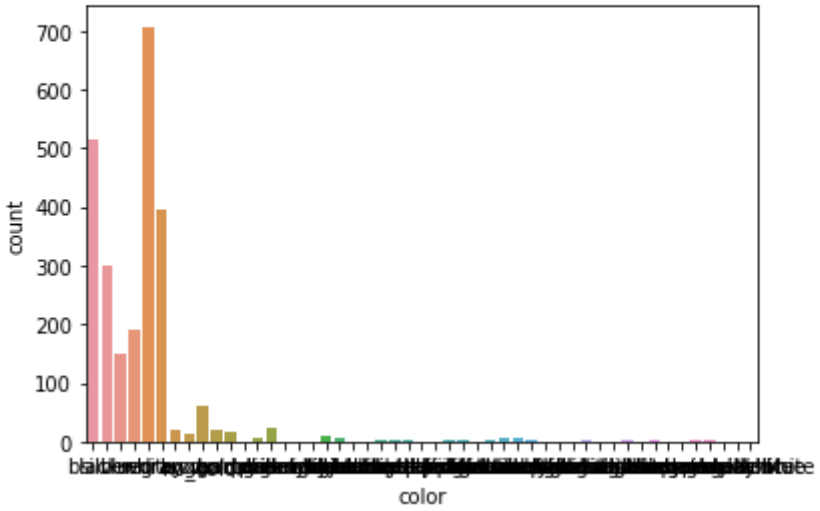
```
df['mileage'].value_counts()

1.0        11
0.0         6
31727.0     2
33808.0     2
21774.0     2
           ..
90685.0     1
54141.0     1
82240.0     1
66167.0     1
31371.0     1
Name: mileage, Length: 2439, dtype: int64
```

```
# 6 th Conclusion/fact
#This count plot will tell us about the types of colors are
 there in usa car datasets
sns.countplot(x = 'color',data = df)
```

```
sns.countplot(x = 'color',data = df)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f7198821a90>
```




```
#Finding out the exact count of no of colors aborad usa car datasets
df.groupby('color').size()
```

```
df.groupby('color').size()
```

```
color
beige                                5
billet silver metallic clearcoat     3
black                              516
black clearcoat                      2
blue                               151
bright white clearcoat               2
brown                               15
burgundy                             1
cayenne red                          2
charcoal                            18
color:                               5
competition orange                   1
dark blue                            1
glacier white                        1
gold                                19
gray                               395
green                               24
guard                                1
ingot silver                         1
ingot silver metallic                4
jazz blue pearlcoat                  1
```

```
df['color'].value_counts()
```

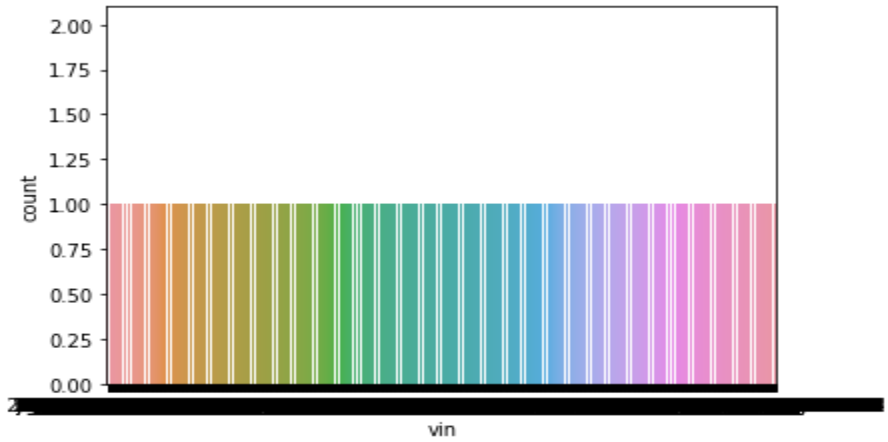
```
df['color'].value_counts()
```

```
white                                707
black                                516
gray                                 395
silver                               300
red                                  192
blue                                 151
no_color                              61
green                                 24
orange                                20
gold                                  19
charcoal                              18
brown                                 15
yellow                                 9
magnetic metallic                      6
shadow black                           5
color:                                 5
beige                                  5
oxford white                           4
ingot silver metallic                  4
super black                            3
billet silver metallic clearcoat       3
triple yellow tri-coat                 3
```

```
#7th Conclusion/fact
#This count plot will tell us about the different types of vehicle identification number (vin) are there in usa car da tasets
sns.countplot(x = 'vin',data = df)
```

```
#7th Conclusion/fact
#This count plot will tell us about the different types of v
sns.countplot(x = 'vin',data = df)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f71949a4a90>
```



```
#Finding out the exact count of no of vin aborad usa cars datasets
df.groupby('vin').size()
```

```
df.groupby('vin').size()

vin
  19uua96529a004646    1
  19xfb2f81fe252000    1
  1b7hc16x01s213315    1
  1b7hg38n62s587845    1
  1c3bc1fg1bn519076    1
                      ..
  wddwk4jb2jf613298    1
  wddzf4jb6ha277485    1
  wf0dp3th0g4113219    1
  wuac6bfr0fa901212    1
  zam57xslxh1248775    1
Length: 2495, dtype: int64
```

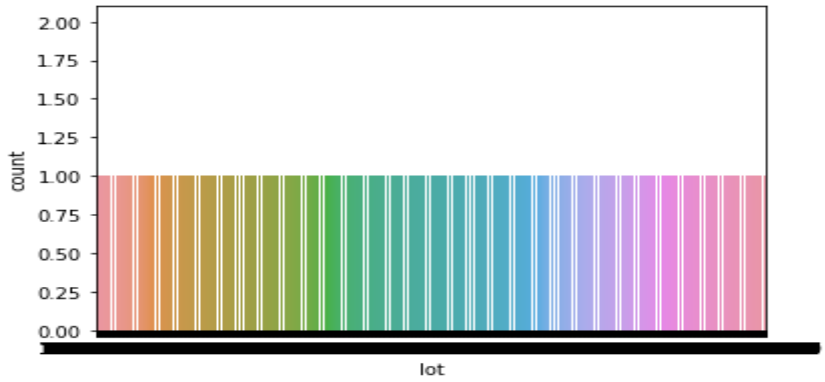
```
df['vin'].value_counts()
```

```
  1gnevhkw8jj148388    2
  1gndt13s632267445    2
  3gcrkse37ag234620    2
  1g1al58f787159241    2
  1fm5k8gt7kgb48943    1
                      ..
  2c3cdxbg5eh300547    1
  2c4rdgcg8jr208468    1
  3c4pdcab8ht507652    1
  3c4pdcgb7ht525941    1
  3n1cn7ap4jl884311    1
Name: vin, Length: 2495, dtype: int64
```

```
# 8th conclusion/fact
sns.countplot(x = 'lot',data = df)
```

```
# 8th conclusion/fact
sns.countplot(x = 'lot',data = df)

<matplotlib.axes._subplots.AxesSubplot at 0x7f7194de03d0>
```



```
#Finding out the exact count of no of lot aborad usa cars d
atasets
df.groupby('lot').size()
```

```
df.groupby('lot').size()

lot
159348797    1
166951262    1
167117726    1
167117732    1
167119104    1
            ..
167804714    1
167805479    1
167805483    1
167805497    1
167805500    1
Length: 2495, dtype: int64
```

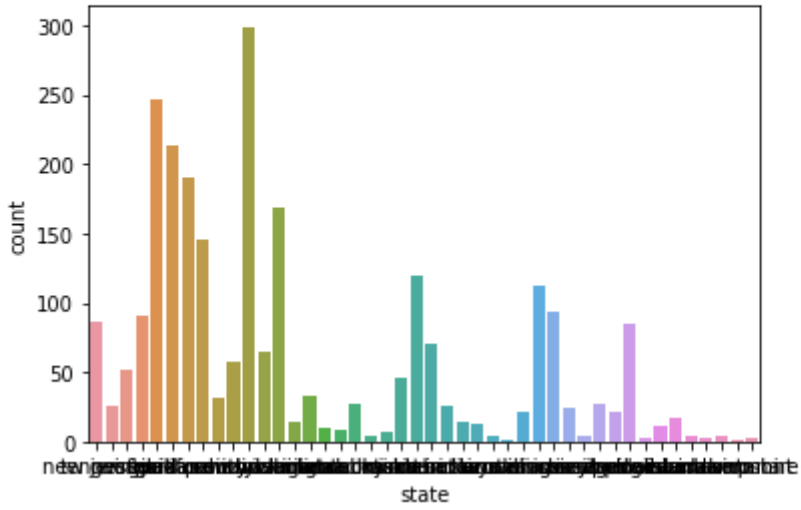
```
df['lot'].value_counts()
```

```
167781794    2
167650636    2
167650663    2
167650632    2
167749575    1
            ..
167771550    1
167772963    1
167772985    1
167772989    1
167762228    1
Name: lot, Length: 2495, dtype: int64
```

```
# 9th Conclusion/fact
#This count plot will tell us the about types of states are
 available in usa car datasets
sns.countplot(x = 'state',data = df)
```

```
# 9th Conclusion/fact
#This count plot will tell us the about types of states are a
sns.countplot(x = 'state',data = df)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f718f94ab90>
```




```
#Finding out the exact count of no of state aborad usa cars
 datasets
df.groupby('state').size()
```

```
df.groupby('state').size()

state
alabama          17
arizona          33
arkansas         12
california      190
colorado         21
connecticut      25
florida         246
georgia          51
idaho             2
illinois        113
indiana          14
kansas            4
kentucky          9
louisiana        11
maryland          4
massachusetts    27
michigan        169
minnesota       119
mississippi      24
missouri         46
montana           1
```

```
df['state'].value_counts()
```

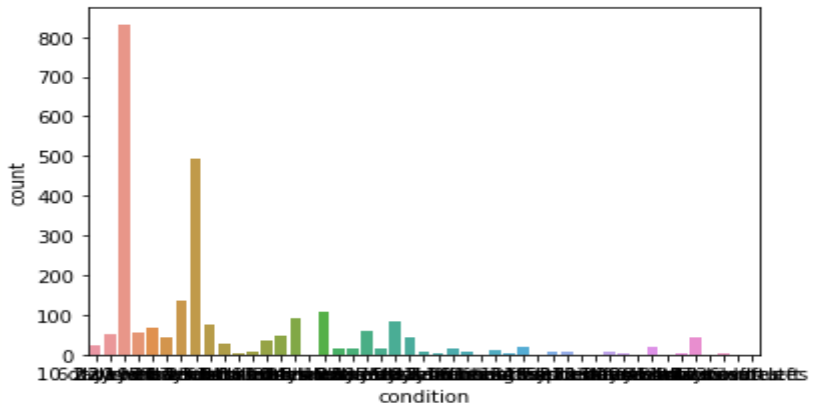
```
df['state'].value_counts()

pennsylvania      299
florida           246
texas             214
california        190
michigan          169
north carolina    146
minnesota         119
illinois          113
wisconsin          94
virginia           90
new jersey         87
nevada             85
oklahoma           71
south carolina     64
new york           58
georgia            51
missouri           46
arizona            33
ohio               31
massachusetts      27
oregon             27
tennessee          26
```

```
# 10th Conclusion/fact
#This count plot will tell us about conditions of cars are there in usa car datasets
sns.countplot(x = 'condition',data = df)
```

```
# 10th Conclusion/fact
#This count plot will tell us about conditions of cars are t
sns.countplot(x = 'condition',data = df)

<matplotlib.axes._subplots.AxesSubplot at 0x7f718d860050>
```



```
#Finding out the exact count of no of condition aborad usa
cars datasets
df.groupby('condition').size()
```

```
df.groupby('condition').size()

condition
1 days left          91
1 hours left          3
1 minutes            15
10 days left         23
11 days left         42
12 days left          8
12 hours left         1
13 days left          1
14 hours left       108
15 days left          4
15 hours left         8
16 hours left        36
16 minutes            1
17 hours left        76
18 hours left        48
19 hours left        45
2 days left         832
2 hours left         26
20 hours left        67
21 hours left       492
22 hours left        57
```

```
df['condition'].value_counts()
```

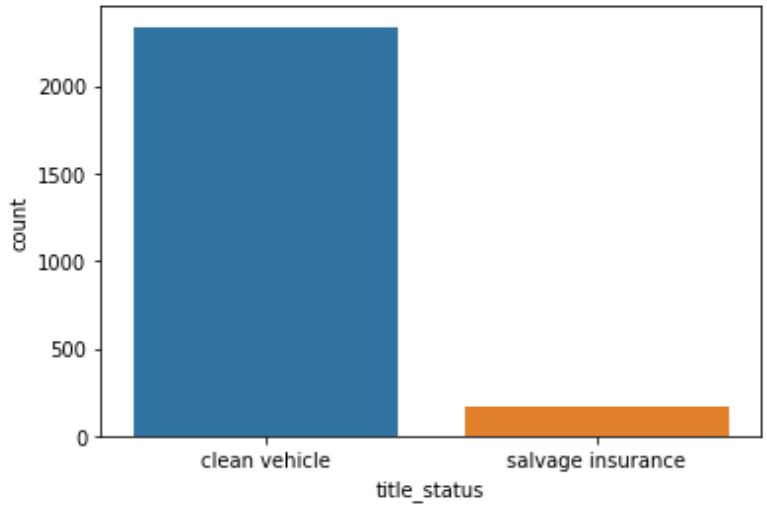
```
df['condition'].value_counts()

2 days left         832
21 hours left       492
3 days left         137
14 hours left       108
1 days left          91
8 days left          82
17 hours left        76
20 hours left        67
9 days left          58
22 hours left        57
6 days left          52
18 hours left        48
19 hours left        45
7 days left          43
11 days left         42
16 hours left        36
2 hours left         26
10 days left         23
Listing Expired      20
29 minutes           18
23 hours left        16
4 days left          16
```

```
# 11th Conclusion/fact
#This count plot will tell us how many clean vehicls and sl
avage insurance are there in usa car datasets
sns.countplot(x = 'title_status',data = df)
```

```
# 11th Conclusion/fact
#This count plot will tell us how many clean vehicls and slavage insurance are th
sns.countplot(x = 'title_status',data = df)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f718d750710>
```




```
#Finding out the exact count of no of clean vehicle and sal
vage insurance aborad usa cars datasets
df.groupby('title_status').size()
```

```
#Finding out the exact count of no of c
df.groupby('title_status').size()
```

```
title_status
clean vehicle        2336
salvage insurance     163
dtype: int64
```

```
df['title_status'].value_counts()
```

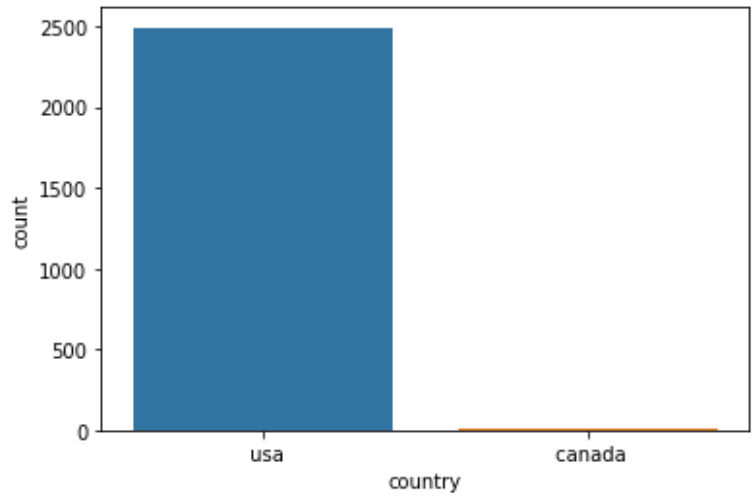
```
df['title_status'].value_counts()
```

```
clean vehicle        2336
salvage insurance     163
Name: title_status, dtype: int64
```

```
# 12th Conclusion/fact
#This count plot will tell us how many usa and canada are t
here in usa car datasets
sns.countplot(x = 'country',data = df)
```

```
# 12th Conclusion/fact
#This count plot will tell us how many usa and canada are there in usa car datasets
sns.countplot(x = 'country',data = df)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f718d697850>
```




```
#Finding out the exact count of no of usa and canada aborad
 usa cars
df.groupby('country').size()
```

```
#Finding out the exact count of no of usa and canada aborad usa cars
df.groupby('country').size()
```

```
country
 canada       7
 usa       2492
dtype: int64
```

```
df['country'].value_counts()
```

```
df['country'].value_counts()
```

```
 usa       2492
 canada       7
Name: country, dtype: int64
```

```
df.groupby(['country','title_status']).size()
```

```
df.groupby(['country','title_status']).size()
```

```
country  title_status
 canada  clean vehicle           7
 usa     clean vehicle        2329
         salvage insurance     163
dtype: int64
```

```
import numpy as np
#np.sum will tell the the sum of number of elements in the specific range
lowprice = np.sum((df['price']>=0)&(df['price']<30000))
mediumprice = np.sum((df['price']>=30000)&(df['price']<50000))
highprice = np.sum((df['price']>=50000)&(df['price']<90000))
print(lowprice)
print(mediumprice)
print(highprice)
```

```
highprice = np.sum((df
print(lowprice)
print(mediumprice)
print(highprice)
```

```
2121
326
52
```

```
#  The price column has 2499 values/rows
2121+326+52
```

```
# The price column has 2499 values/rows
2121+326+52
```

```
2499
```

```
np.min(df['price'])
np.max(df['price'])
```

```
title_status_m = np.sum((df['country']=='usa')&df['title_status']==1)
title_status_m
```

```
np.min(df['price'])
```

```
0
```

```
np.max(df['price'])
```

```
84900
```

```
title_status_m = np.sum((df['country']=='usa')&df['title_status']==1)
title_status_m
```

```
0
```

```
#distribution Plot
sns.distplot(df['price'])
```

```
#distribution Plot
sns.distplot(df['price'])
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:2619: FutureWar
  warnings.warn(msg, FutureWarning)
<matplotlib.axes._subplots.AxesSubplot at 0x7f718d7a9d10>
```