In [20]:

```python
import pandas as pd
import numpy as np
```

In [21]:

```python
df=pd.read_csv('Datasets/auto-mpg.csv')
df.head()
```

Out[21]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model year	origin	car name
0	18.0	8	307.0	130	3504	12.0	70	1	chevrolet chevelle malibu
1	15.0	8	350.0	165	3693	11.5	70	1	buick skylark 320
2	18.0	8	318.0	150	3436	11.0	70	1	plymouth satellite
3	16.0	8	304.0	150	3433	12.0	70	1	amc rebel sst
4	17.0	8	302.0	140	3449	10.5	70	1	ford torino

In [22]:

```python
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 398 entries, 0 to 397
Data columns (total 9 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   mpg           398 non-null    float64
 1   cylinders     398 non-null    int64  
 2   displacement  398 non-null    float64
 3   horsepower    398 non-null    object 
 4   weight        398 non-null    int64  
 5   acceleration  398 non-null    float64
 6   model year    398 non-null    int64  
 7   origin        398 non-null    int64  
 8   car name      398 non-null    object 
dtypes: float64(3), int64(4), object(2)
memory usage: 28.1+ KB
```

In [23]:

```
df.describe()
```

Out[23]:

	mpg	cylinders	displacement	weight	acceleration	model year	origin
count	398.000000	398.000000	398.000000	398.000000	398.000000	398.000000	398.000000
mean	23.514573	5.454774	193.425879	2970.424623	15.568090	76.010050	1.572864
std	7.815984	1.701004	104.269838	846.841774	2.757689	3.697627	0.802055
min	9.000000	3.000000	68.000000	1613.000000	8.000000	70.000000	1.000000
25%	17.500000	4.000000	104.250000	2223.750000	13.825000	73.000000	1.000000
50%	23.000000	4.000000	148.500000	2803.500000	15.500000	76.000000	1.000000
75%	29.000000	8.000000	262.000000	3608.000000	17.175000	79.000000	2.000000
max	46.600000	8.000000	455.000000	5140.000000	24.800000	82.000000	3.000000

In [24]:

```
df.describe(exclude=np.number)
```

Out[24]:

	horsepower	car name
count	398	398
unique	94	305
top	150	ford pinto
freq	22	6

In [25]:

```
df.describe(include="all")
```

Out[25]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model year	origin	car name
count	398.000000	398.000000	398.000000	398	398.000000	398.000000	398.000000	398.000000	398
unique	NaN	NaN	NaN	94	NaN	NaN	NaN	NaN	305
top	NaN	NaN	NaN	150	NaN	NaN	NaN	NaN	ford pinto
freq	NaN	NaN	NaN	22	NaN	NaN	NaN	NaN	6
mean	23.514573	5.454774	193.425879	NaN	2970.424623	15.568090	76.010050	1.572864	NaN
std	7.815984	1.701004	104.269838	NaN	846.841774	2.757689	3.697627	0.802055	NaN
min	9.000000	3.000000	68.000000	NaN	1613.000000	8.000000	70.000000	1.000000	NaN
25%	17.500000	4.000000	104.250000	NaN	2223.750000	13.825000	73.000000	1.000000	NaN
50%	23.000000	4.000000	148.500000	NaN	2803.500000	15.500000	76.000000	1.000000	NaN
75%	29.000000	8.000000	262.000000	NaN	3608.000000	17.175000	79.000000	2.000000	NaN
max	46.600000	8.000000	455.000000	NaN	5140.000000	24.800000	82.000000	3.000000	NaN

In [26]:

```
df.describe(percentiles=[0.2,0.6,0.8,0.9,1,0.67])
```

Out[26]:

```
              mpg   cylinders  displacement       weight  acceleration  model year      origin
count  398.000000  398.000000    398.000000   398.000000    398.000000  398.000000  398.000000
 mean   23.514573    5.454774    193.425879  2970.424623     15.568090   76.010050    1.572864
  std    7.815984    1.701004    104.269838   846.841774      2.757689    3.697627    0.802055
  min    9.000000    3.000000     68.000000  1613.000000      8.000000   70.000000    1.000000
  20%   16.000000    4.000000     98.000000  2155.000000     13.500000   72.000000    1.000000
  50%   23.000000    4.000000    148.500000  2803.500000     15.500000   76.000000    1.000000
  60%   25.000000    6.000000    200.000000  3085.200000     16.000000   77.000000    1.000000
  67%   27.000000    6.000000    232.000000  3328.730000     16.500000   78.000000    2.000000
  80%   31.000000    8.000000    304.600000  3806.000000     17.760000   80.000000    2.000000
  90%   34.330000    8.000000    350.000000  4275.200000     19.000000   81.000000    3.000000
 100%   46.600000    8.000000    455.000000  5140.000000     24.800000   82.000000    3.000000
  max   46.600000    8.000000    455.000000  5140.000000     24.800000   82.000000    3.000000
```

In [27]:

```
df[df["horsepower"]=="?"]
```

Out[27]:

```
      mpg  cylinders  displacement  horsepower  weight  acceleration  model year  origin            car name
 32  25.0          4          98.0           ?    2046          19.0          71       1          ford pinto
126  21.0          6         200.0           ?    2875          17.0          74       1        ford maverick
330  40.9          4          85.0           ?    1835          17.3          80       2  renault lecar deluxe
336  23.6          4         140.0           ?    2905          14.3          80       1   ford mustang cobra
354  34.5          4         100.0           ?    2320          15.8          81       2         renault 18i
374  23.0          4         151.0           ?    3035          20.5          82       1       amc concord dl
```

In [28]:

```
df['horsepower'].replace('?',100).iloc[32]
```

Out[28]:

```
100
```

In [29]:

```
df.drop(df[df["horsepower"]=="?"].index,axis=0)
```

Out[29]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model year	origin	car name
0	18.0	8	307.0	130	3504	12.0	70	1	chevrolet chevelle malibu
1	15.0	8	350.0	165	3693	11.5	70	1	buick skylark 320
2	18.0	8	318.0	150	3436	11.0	70	1	plymouth satellite
3	16.0	8	304.0	150	3433	12.0	70	1	amc rebel sst
4	17.0	8	302.0	140	3449	10.5	70	1	ford torino
...	...	...	...	...	...	...	...	...	...
393	27.0	4	140.0	86	2790	15.6	82	1	ford mustang gl
394	44.0	4	97.0	52	2130	24.6	82	2	vw pickup
395	32.0	4	135.0	84	2295	11.6	82	1	dodge rampage
396	28.0	4	120.0	79	2625	18.6	82	1	ford ranger
397	31.0	4	119.0	82	2720	19.4	82	1	chevy s-10

392 rows × 9 columns

In [30]:

```
df.drop('car name',axis=1)
```

Out[30]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model year	origin
0	18.0	8	307.0	130	3504	12.0	70	1
1	15.0	8	350.0	165	3693	11.5	70	1
2	18.0	8	318.0	150	3436	11.0	70	1
3	16.0	8	304.0	150	3433	12.0	70	1
4	17.0	8	302.0	140	3449	10.5	70	1
...	...	...	...	...	...	...	...	...
393	27.0	4	140.0	86	2790	15.6	82	1
394	44.0	4	97.0	52	2130	24.6	82	2
395	32.0	4	135.0	84	2295	11.6	82	1
396	28.0	4	120.0	79	2625	18.6	82	1
397	31.0	4	119.0	82	2720	19.4	82	1

398 rows × 8 columns

In [31]:

```
df['horsepower']=df["horsepower"].replace('?',100)
```

In [32]:

```
df.dtypes
```

Out[32]:

```
mpg             float64
cylinders         int64
displacement    float64
horsepower       object
weight            int64
acceleration    float64
model year        int64
origin            int64
car name         object
dtype: object
```

In [33]:

```
df['horsepower']=df['horsepower'].astype('int64')
```

In [34]:

```
df.dtypes
```

Out[34]:

```
mpg             float64
cylinders         int64
displacement    float64
horsepower        int64
weight            int64
acceleration    float64
model year        int64
origin            int64
car name         object
dtype: object
```

In [35]:

```
x=[[1,2,np.nan,np.nan,np.nan],
  [3,4,np.nan,5,6],
  [3,4,np.nan,7,9],
  [8,10,11,12,13],
  [8,10,11,12,13],
  [np.nan,np.nan,np.nan,np.nan,np.nan]]
df_x=pd.DataFrame(x)
print(df_x)
```

```
     0     1     2     3     4
0  1.0   2.0   NaN   NaN   NaN
1  3.0   4.0   NaN   5.0   6.0
2  3.0   4.0   NaN   7.0   9.0
3  8.0  10.0  11.0  12.0  13.0
4  8.0  10.0  11.0  12.0  13.0
5  NaN   NaN   NaN   NaN   NaN
```

In [36]:

```
df_x.dropna()
```

Out[36]:

	0	1	2	3	4
3	8.0	10.0	11.0	12.0	13.0
4	8.0	10.0	11.0	12.0	13.0

In [37]:

```python
df_x.dropna(axis=0)
```

Out[37]:

	0	1	2	3	4
3	8.0	10.0	11.0	12.0	13.0
4	8.0	10.0	11.0	12.0	13.0

In [38]:

```python
df_x.dropna(axis=1)
```

Out[38]:

0
1
2
3
4
5

In [39]:

```python

df_x=pd.DataFrame(x,columns=["A","B","C","D","E"])
```

In [40]:

```python
print(df_x)
```

```
      A     B     C     D     E
0   1.0   2.0   NaN   NaN   NaN
1   3.0   4.0   NaN   5.0   6.0
2   3.0   4.0   NaN   7.0   9.0
3   8.0  10.0  11.0  12.0  13.0
4   8.0  10.0  11.0  12.0  13.0
5   NaN   NaN   NaN   NaN   NaN
```

In [41]:

```
df_x.dropna(subset=['A'],axis=0)
```

Out[41]:

	A	B	C	D	E
0	1.0	2.0	NaN	NaN	NaN
1	3.0	4.0	NaN	5.0	6.0
2	3.0	4.0	NaN	7.0	9.0
3	8.0	10.0	11.0	12.0	13.0
4	8.0	10.0	11.0	12.0	13.0

In [42]:

```
df_x.dropna(subset=['A','C'],axis=0) #or condition
```

Out[42]:

	A	B	C	D	E
3	8.0	10.0	11.0	12.0	13.0
4	8.0	10.0	11.0	12.0	13.0

In [43]:

```
df_x.dropna(how='any') # signle bhi na hoi to drop kare dye. 0-row 1-column


```

Out[43]:

	A	B	C	D	E
3	8.0	10.0	11.0	12.0	13.0
4	8.0	10.0	11.0	12.0	13.0

In [44]:

```
df_x.dropna(how='all')
```

Out[44]:

	A	B	C	D	E
0	1.0	2.0	NaN	NaN	NaN
1	3.0	4.0	NaN	5.0	6.0
2	3.0	4.0	NaN	7.0	9.0
3	8.0	10.0	11.0	12.0	13.0
4	8.0	10.0	11.0	12.0	13.0

In [45]:

```
df_x.dropna(subset=['E','C'],axis=0,how='all') # and condition jo bey column ma na to j drop
```

Out[45]:

	A	B	C	D	E
1	3.0	4.0	NaN	5.0	6.0
2	3.0	4.0	NaN	7.0	9.0
3	8.0	10.0	11.0	12.0	13.0
4	8.0	10.0	11.0	12.0	13.0

In [46]:

```
df_x.dropna(thresh=2) #2 karta ochi fill value hoy to drop kareee.
```

Out[46]:

	A	B	C	D	E
0	1.0	2.0	NaN	NaN	NaN
1	3.0	4.0	NaN	5.0	6.0
2	3.0	4.0	NaN	7.0	9.0
3	8.0	10.0	11.0	12.0	13.0
4	8.0	10.0	11.0	12.0	13.0

In [47]:

```
df_x.dropna(thresh=3) #3 karta ochi fill value hoy to drop kareee.
```

Out[47]:

	A	B	C	D	E
1	3.0	4.0	NaN	5.0	6.0
2	3.0	4.0	NaN	7.0	9.0
3	8.0	10.0	11.0	12.0	13.0
4	8.0	10.0	11.0	12.0	13.0

In [ ]:

```

```

In [48]:

```
df_x.duplicated()
```

Out[48]:

```
0    False
1    False
2    False
3    False
4     True
5    False
dtype: bool
```

In [49]:

```
df_x.duplicated(subset=['A'])
```

Out[49]:

```
0    False
1    False
2     True
3    False
4     True
5    False
dtype: bool
```

In [50]:

```
1

df_x.duplicated(subset=['A','D'])
```

Out[50]:

```
0    False
1    False
2    False
3    False
4     True
5    False
dtype: bool
```

In [51]:

```
df_x.drop_duplicates()
```

Out[51]:

	A	B	C	D	E
0	1.0	2.0	NaN	NaN	NaN
1	3.0	4.0	NaN	5.0	6.0
2	3.0	4.0	NaN	7.0	9.0
3	8.0	10.0	11.0	12.0	13.0
5	NaN	NaN	NaN	NaN	NaN

In [52]:

```
df_x.drop_duplicates(keep='first')
```

Out[52]:

	A	B	C	D	E
0	1.0	2.0	NaN	NaN	NaN
1	3.0	4.0	NaN	5.0	6.0
2	3.0	4.0	NaN	7.0	9.0
3	8.0	10.0	11.0	12.0	13.0
5	NaN	NaN	NaN	NaN	NaN

In [53]:

```
df_x.drop_duplicates(keep=False)
```

Out[53]:

	A	B	C	D	E
0	1.0	2.0	NaN	NaN	NaN
1	3.0	4.0	NaN	5.0	6.0
2	3.0	4.0	NaN	7.0	9.0
5	NaN	NaN	NaN	NaN	NaN

In [54]:

```
df_x.drop_duplicates(keep='first',subset=['A','D'])
```

Out[54]:

	A	B	C	D	E
0	1.0	2.0	NaN	NaN	NaN
1	3.0	4.0	NaN	5.0	6.0
2	3.0	4.0	NaN	7.0	9.0
3	8.0	10.0	11.0	12.0	13.0
5	NaN	NaN	NaN	NaN	NaN

In [ ]:

```

```

In [55]:

```
print(df)
```

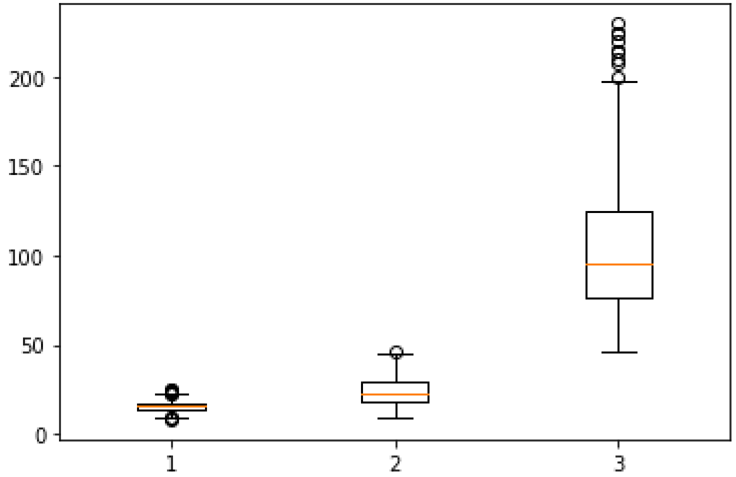
```
      mpg  cylinders  displacement  horsepower  weight  acceleration  \
0    18.0          8         307.0         130    3504          12.0   
1    15.0          8         350.0         165    3693          11.5   
2    18.0          8         318.0         150    3436          11.0   
3    16.0          8         304.0         150    3433          12.0   
4    17.0          8         302.0         140    3449          10.5   
..    ...        ...           ...         ...     ...           ...   
393  27.0          4         140.0          86    2790          15.6   
394  44.0          4          97.0          52    2130          24.6   
395  32.0          4         135.0          84    2295          11.6   
396  28.0          4         120.0          79    2625          18.6   
397  31.0          4         119.0          82    2720          19.4   

     model year  origin                   car name  
0            70       1  chevrolet chevelle malibu  
1            70       1          buick skylark 320  
2            70       1         plymouth satellite  
3            70       1              amc rebel sst  
4            70       1                ford torino  
..          ...     ...                        ...  
393          82       1            ford mustang gl  
394          82       2                  vw pickup  
395          82       1              dodge rampage  
396          82       1                ford ranger  
397          82       1                 chevy s-10  

[398 rows x 9 columns]
```

```
In [56]: df.dtypes
```

```
Out[56]: mpg               float64
         cylinders           int64
         displacement      float64
         horsepower          int64
         weight              int64
         acceleration      float64
         model year          int64
         origin              int64
         car name           object
         dtype: object
```

```
In [57]: import matplotlib.pyplot as plt
```

```
In [58]: plt.boxplot(df[['acceleration','mpg' ,'horsepower']])
         plt.show()
```

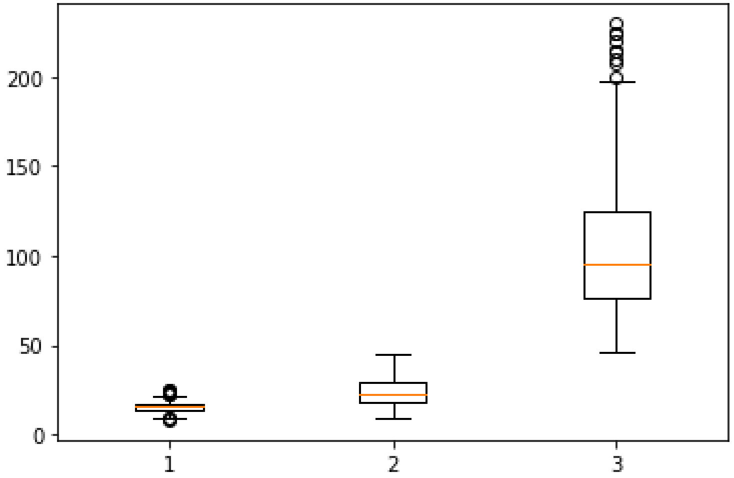


In [60]:

```
q1=df['mpg'].quantile(0.25)
q3=df['mpg'].quantile(0.75)
iqr=q3-q1
ul=q3+(1.5)*iqr
ll=q1-(1.5)*iqr
df=df[(df['mpg']>=ll) & (df['mpg']<=ul)]
```

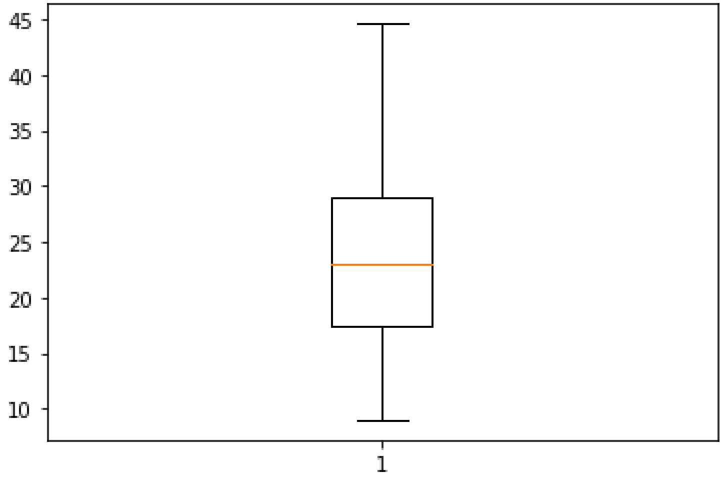
In [61]:

```
plt.boxplot(df[['acceleration','mpg' ,'horsepower']])
plt.show()
```



In [65]:

```
plt.boxplot(df[['mpg']])
plt.show()
```



In [64]:

```
df_x.corr() # laptop ma (numerical_only=True) karva nu
```

Out[64]:

	A	B	C	D	E
A	1.000000	0.999306	NaN	0.973329	0.933008
B	0.999306	1.000000	NaN	0.973329	0.933008
C	NaN	NaN	NaN	NaN	NaN
D	0.973329	0.973329	NaN	1.000000	0.990680
E	0.933008	0.933008	NaN	0.990680	1.000000

```
In [66]: df.corr()
```

Out[66]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model year	origin
mpg	1.000000	-0.778350	-0.806521	-0.774286	-0.834482	0.418997	0.578468	0.558579
cylinders	-0.778350	1.000000	0.950648	0.839695	0.895817	-0.504515	-0.347247	-0.561466
displacement	-0.806521	0.950648	1.000000	0.894364	0.932646	-0.542701	-0.368392	-0.608028
horsepower	-0.774286	0.839695	0.894364	1.000000	0.861096	-0.684646	-0.410906	-0.450870
weight	-0.834482	0.895817	0.932646	0.861096	1.000000	-0.416206	-0.304641	-0.579533
acceleration	0.418997	-0.504515	-0.542701	-0.684646	-0.416206	1.000000	0.286513	0.203070
model year	0.578468	-0.347247	-0.368392	-0.410906	-0.304641	0.286513	1.000000	0.176781
origin	0.558579	-0.561466	-0.608028	-0.450870	-0.579533	0.203070	0.176781	1.000000

```
In [2]: import pandas as pd
```

```
In [5]: pd.plotting.scatter_matrix(df_x,figsize=(20,20))
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-5-357d897f8776> in <module>
----> 1 pd.plotting.scatter_matrix(df_x,figsize=(20,20))

NameError: name 'df_x' is not defined
```

```
In [ ]:
```

```
In [ ]:

```