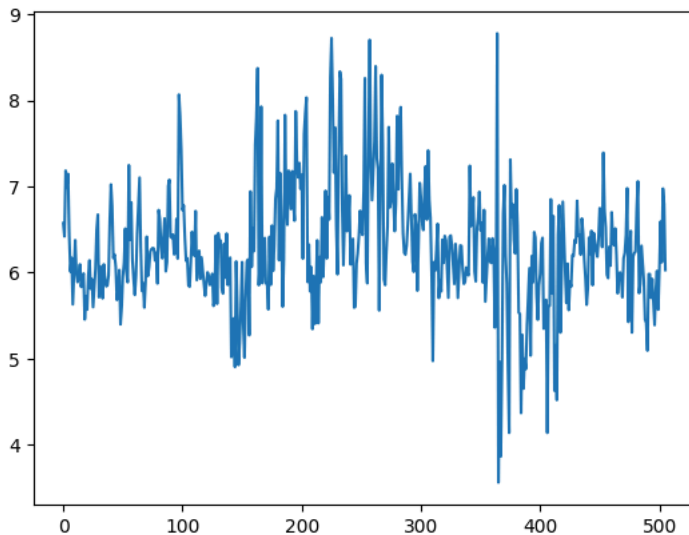```
!pip install matplotlib
```

```
Requirement already satisfied: matplotlib in /usr/local/lib/python3.12/dist-packages (3.10.0)
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.12/dist-packages (from matplotlib) (1.3.3)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.12/dist-packages (from matplotlib) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.12/dist-packages (from matplotlib) (4.61.1)
Requirement already satisfied: kiwisolver>=1.3.1 in /usr/local/lib/python3.12/dist-packages (from matplotlib) (1.4.9)
Requirement already satisfied: numpy>=1.23 in /usr/local/lib/python3.12/dist-packages (from matplotlib) (2.0.2)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.12/dist-packages (from matplotlib) (25.0)
Requirement already satisfied: pillow>=8 in /usr/local/lib/python3.12/dist-packages (from matplotlib) (11.3.0)
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.12/dist-packages (from matplotlib) (3.3.2)
Requirement already satisfied: python-dateutil>=2.7 in /usr/local/lib/python3.12/dist-packages (from matplotlib) (2.9.0.post0)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.12/dist-packages (from python-dateutil>=2.7->matplotlib) (1.17
```

```
#q1 task 1
import matplotlib.pyplot as plt
import pandas as pd
import kagglehub
from kagglehub import KaggleDatasetAdapter


#load the new version
boston_pd = kagglehub.dataset_load(
  KaggleDatasetAdapter.PANDAS,
  "altavish/boston-housing-dataset",
  "HousingData.csv")
boston_pd.head()

#line plot
plt.plot(boston_pd['RM'])
#Display
plt.show()
boston_pd.head()
```
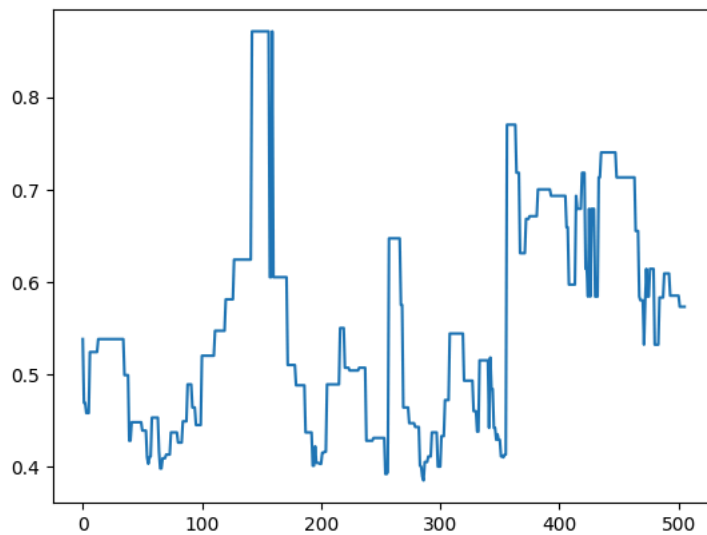
Using Colab cache for faster access to the 'boston-housing-dataset' dataset.

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV
0	0.00632	18.0	2.31	0.0	0.538	6.575	65.2	4.0900	1	296	15.3	396.90	4.98	24.0
1	0.02731	0.0	7.07	0.0	0.469	6.421	78.9	4.9671	2	242	17.8	396.90	9.14	21.6
2	0.02729	0.0	7.07	0.0	0.469	7.185	61.1	4.9671	2	242	17.8	392.83	4.03	34.7
3	0.03237	0.0	2.18	0.0	0.458	6.998	45.8	6.0622	3	222	18.7	394.63	2.94	33.4
4	0.06905	0.0	2.18	0.0	0.458	7.147	54.2	6.0622	3	222	18.7	396.90	NaN	36.2

```
#q1 task 3
plt.plot(boston_pd['NOX'])
```
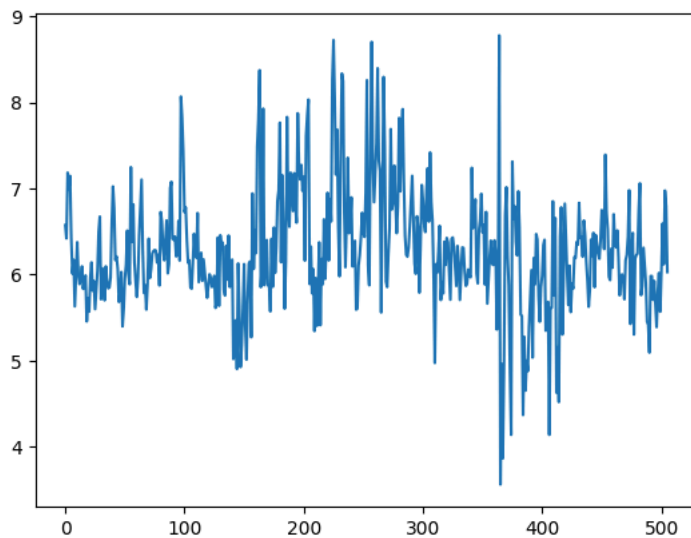
[<matplotlib.lines.Line2D at 0x7a854455e570>]


```
plt.plot(boston_pd['RM'])
```
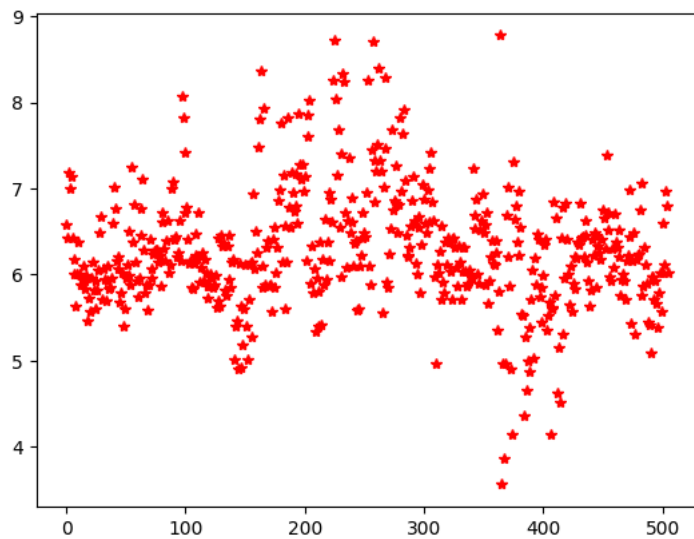
[<matplotlib.lines.Line2D at 0x7a854456f530>]


```
#task 2
## Draw line plot for column 'RM'
import matplotlib.pyplot as plt

# line plot
# the argument 'r*' plots each point as a red '*'
plt.plot(boston_pd['RM'], 'r*')

# display plot
plt.show()
```
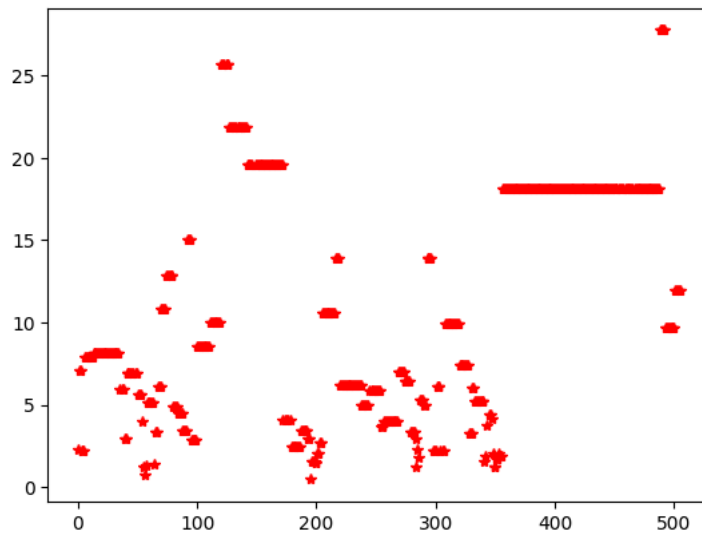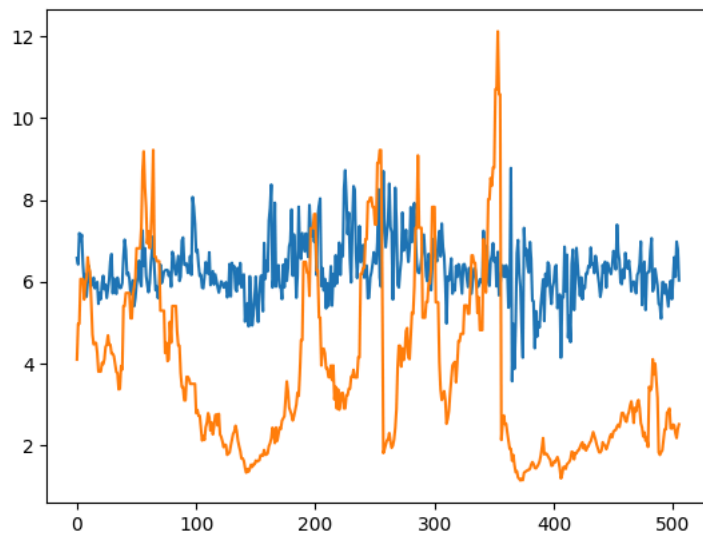
```
plt.plot(boston_pd['INDUS'], 'r*')
```
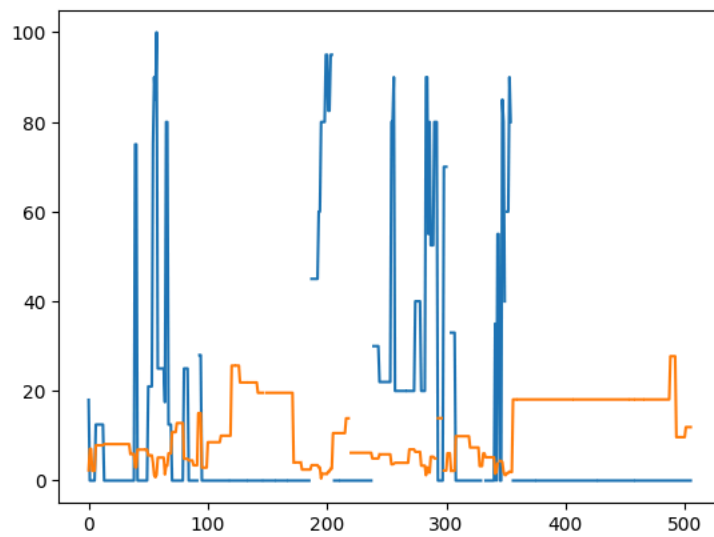
```
[<matplotlib.lines.Line2D at 0x7a85412bfb00>]
```
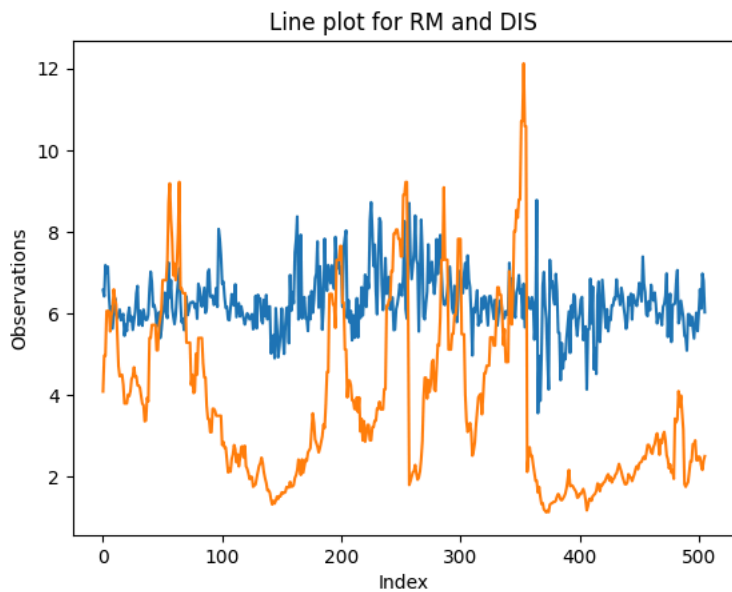


```
#7.2
import matplotlib.pyplot as plt
plt.plot(boston_pd['RM'])
plt.plot(boston_pd['DIS'])
plt.show()
```
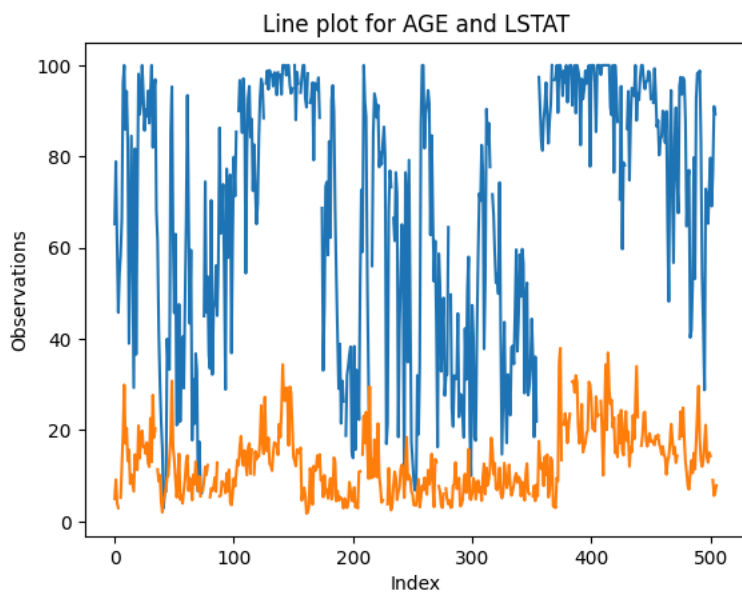
```
plt.plot(boston_pd['ZN'])
plt.plot(boston_pd['INDUS'])
plt.show()
```
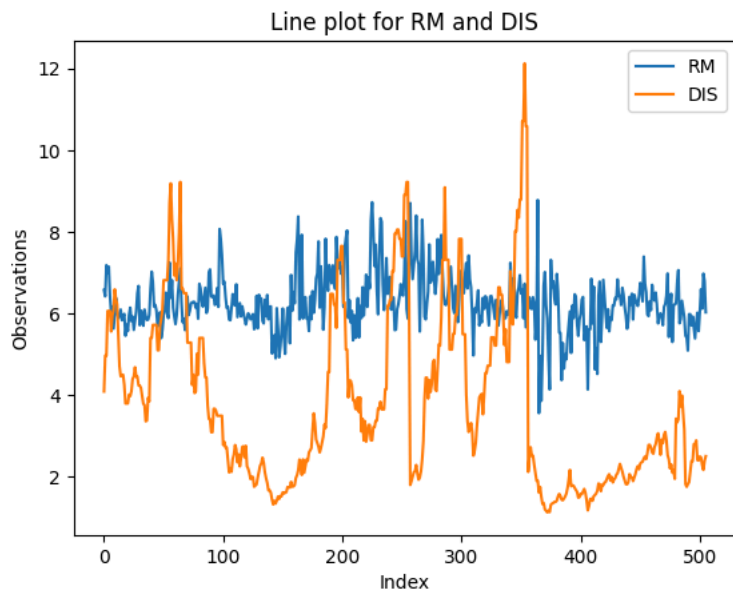


```
#7.3
import matplotlib.pyplot as plt
plt.plot(boston_pd['RM'])
plt.plot(boston_pd['DIS'])
plt.xlabel('Index')
plt.ylabel('Observations')
plt.title('Line plot for RM and DIS')
plt.show()
```
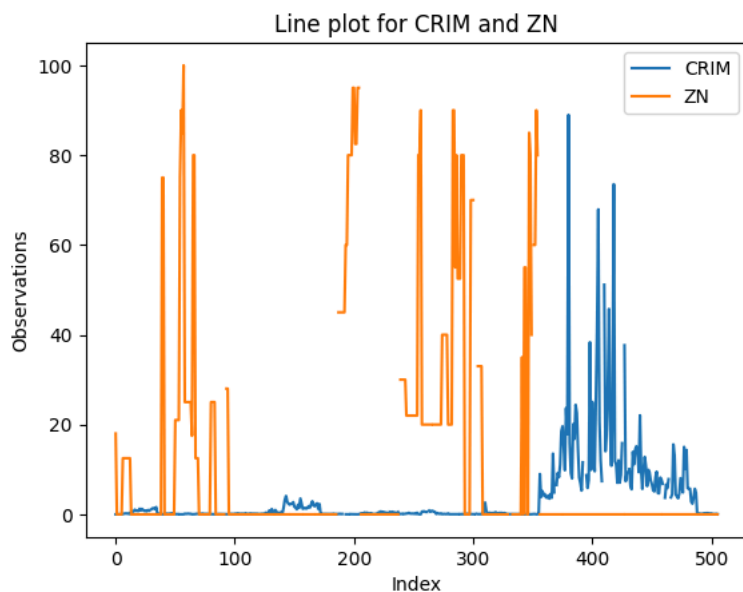
Line plot for RM and DIS


```
import matplotlib.pyplot as plt
plt.plot(boston_pd['AGE'])
plt.plot(boston_pd['LSTAT'])
plt.xlabel('Index')
plt.ylabel('Observations')
plt.title('Line plot for AGE and LSTAT')
plt.show()
```

Line plot for AGE and LSTAT


```
#7.4
import matplotlib.pyplot as plt
plt.plot(boston_pd['RM'], label = 'RM')
plt.plot(boston_pd['DIS'], label = 'DIS')
plt.xlabel('Index')
plt.ylabel('Observations')
plt.title('Line plot for RM and DIS')
plt.legend()
plt.show()
```
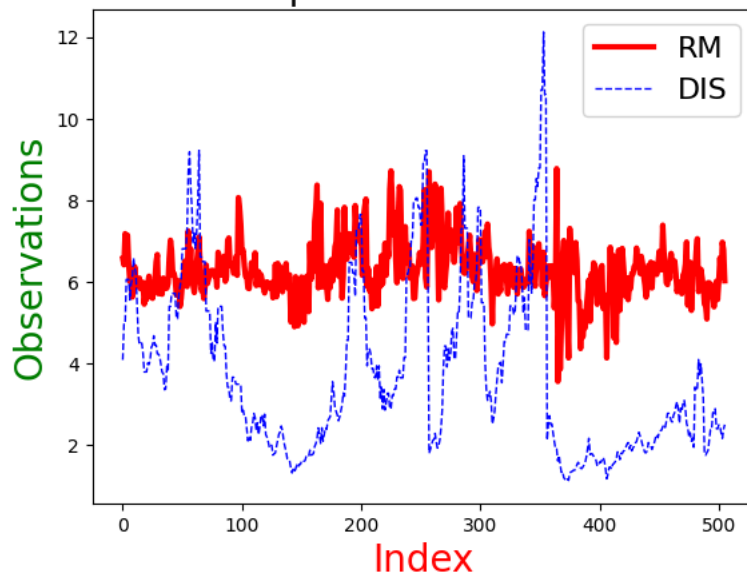
```python
import matplotlib.pyplot as plt
plt.plot(boston_pd['CRIM'], label = 'CRIM')
plt.plot(boston_pd['ZN'], label = 'ZN')
plt.xlabel('Index')
plt.ylabel('Observations')
plt.title('Line plot for CRIM and ZN')
plt.legend()
plt.show()
```



```python
#task 7.5
import matplotlib.pyplot as plt
plt.plot(boston_pd['RM'], label = 'RM',color='red',linewidth=3,linestyle='-')
plt.plot(boston_pd['DIS'], label = 'DIS',color='blue',linewidth=1,linestyle='--')
plt.xlabel('Index', fontsize=20,color="red")
plt.ylabel('Observations', fontsize=20,color="green")
plt.title('Line plot for RM and DIS', fontsize=20)
plt.legend(fontsize=16)
plt.show()
```
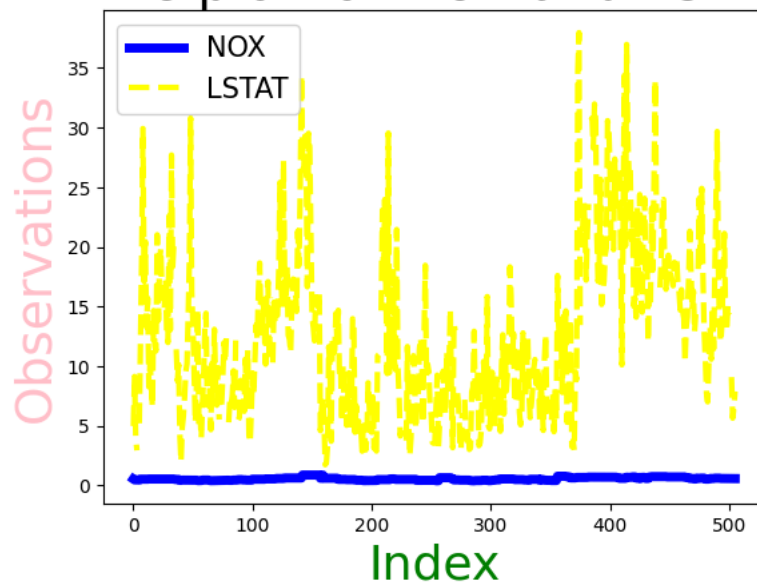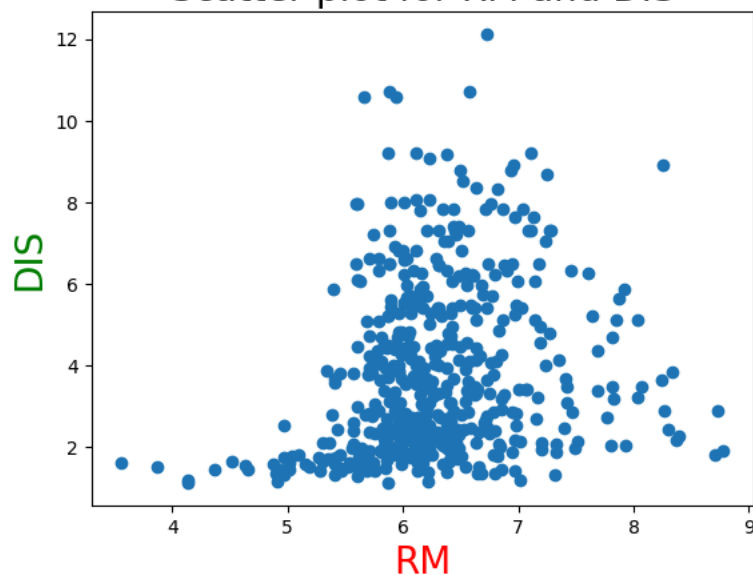
Line plot for RM and DIS


```
import matplotlib.pyplot as plt
plt.plot(boston_pd['NOX'], label = 'NOX',color='blue',linewidth=5,linestyle='-')
plt.plot(boston_pd['LSTAT'], label = 'LSTAT',color='yellow',linewidth=3,linestyle='--')
plt.xlabel('Index', fontsize=26,color="green")
plt.ylabel('Observations', fontsize=27,color="pink")
plt.title('Line plot for NOX and LSTAT', fontsize=28)
plt.legend(fontsize=15)
plt.show()
```
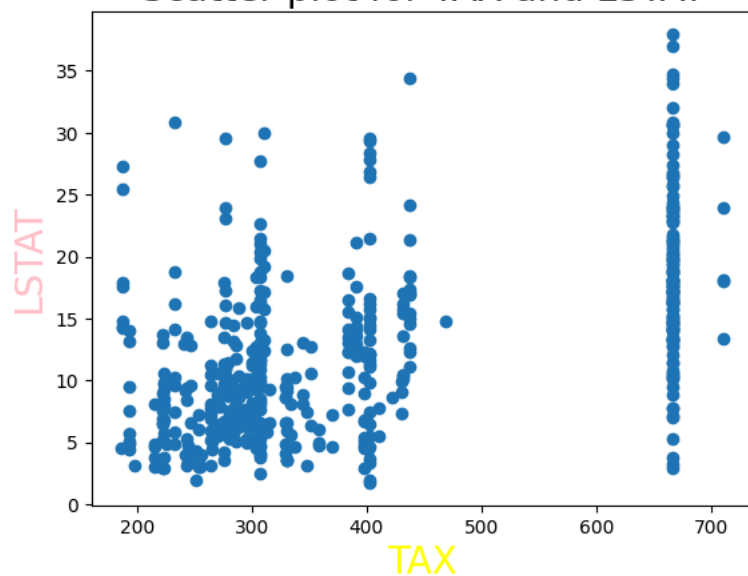
Line plot for NOX and LSTAT


```
#7.6
#7.61 scatter plot
## plot scatter plot
import matplotlib.pyplot as plt
plt.scatter(boston_pd['RM'], boston_pd['DIS'] )
plt.xlabel('RM', fontsize=20,color="red")
plt.ylabel('DIS', fontsize=20,color="green")
plt.title('Scatter plot for RM and DIS', fontsize=20)
plt.show()
```

Scatter plot for RM and DIS


```
import matplotlib.pyplot as plt
plt.scatter(boston_pd['TAX'], boston_pd['LSTAT'] )
plt.xlabel('TAX', fontsize=20,color="yelloW")
plt.ylabel('LSTAT', fontsize=20,color="pink")
plt.title('Scatter plot for TAX and LSTAT', fontsize=20)
plt.show()
```

Scatter plot for TAX and LSTAT


```
import pandas as pd
import kagglehub
from kagglehub import KaggleDatasetAdapter

boston_pd = kagglehub.dataset_load(
    KaggleDatasetAdapter.PANDAS,
    "altavish/boston-housing-dataset",
    "HousingData.csv"
)
boston_pd.head()
```
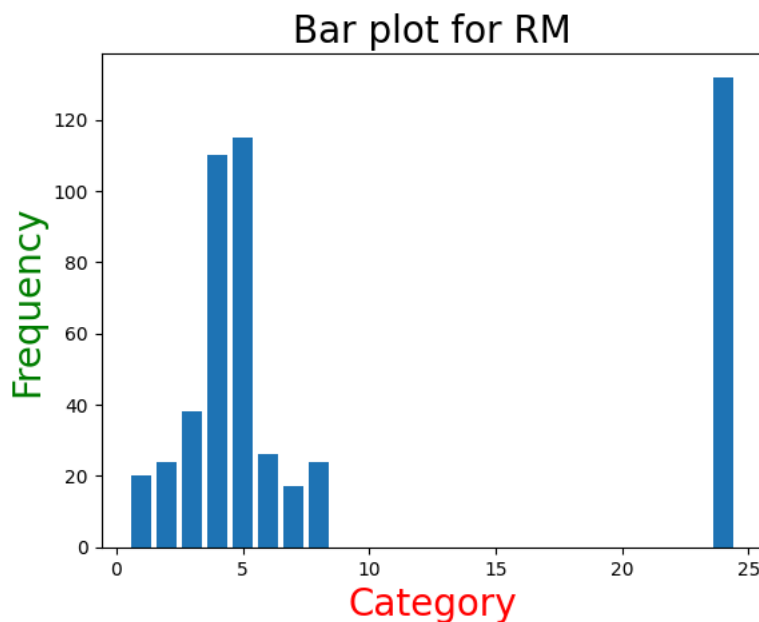
Warning: Looks like you're using an outdated `kagglehub` version (installed: 0.3.13), please consider upgrading to the latest ve

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV
0	0.00632	18.0	2.31	0.0	0.538	6.575	65.2	4.0900	1	296	15.3	396.90	4.98	24.0
1	0.02731	0.0	7.07	0.0	0.469	6.421	78.9	4.9671	2	242	17.8	396.90	9.14	21.6
2	0.02729	0.0	7.07	0.0	0.469	7.185	61.1	4.9671	2	242	17.8	392.83	4.03	34.7
3	0.03237	0.0	2.18	0.0	0.458	6.998	45.8	6.0622	3	222	18.7	394.63	2.94	33.4
4	0.06905	0.0	2.18	0.0	0.458	7.147	54.2	6.0622	3	222	18.7	396.90	NaN	36.2

```python
#task 7.6.2 bar chart
import matplotlib.pyplot as plt
import numpy as np
frequency_table = boston_pd['RAD'].value_counts()
category = frequency_table.index.tolist()
counts = frequency_table.values.tolist()
print(frequency_table)
plt.bar(category,counts)
plt.xlabel('Category', fontsize=20,color="red")
plt.ylabel('Frequency', fontsize=20,color="green")
plt.title('Bar plot for RM', fontsize=20)
plt.show()
```

```
RAD
24    132
5     115
4     110
3      38
6      26
8      24
2      24
1      20
7      17
Name: count, dtype: int64
```
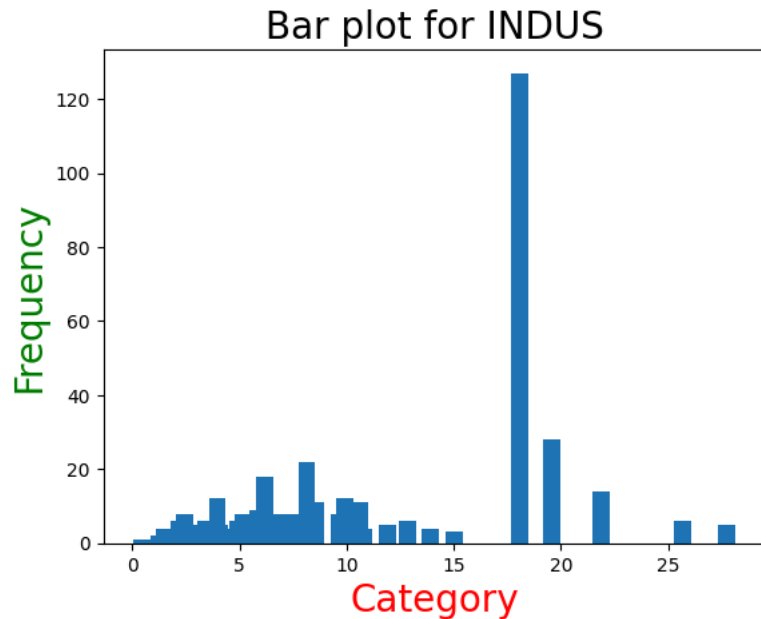


```python
import matplotlib.pyplot as plt
import numpy as np
frequency_table = boston_pd['INDUS'].value_counts()
category = frequency_table.index.tolist()
counts = frequency_table.values.tolist()
print(frequency_table)
plt.bar(category,counts)
plt.xlabel('Category', fontsize=20,color="red")
plt.ylabel('Frequency', fontsize=20,color="green")
plt.title('Bar plot for INDUS', fontsize=20)
plt.show()
```
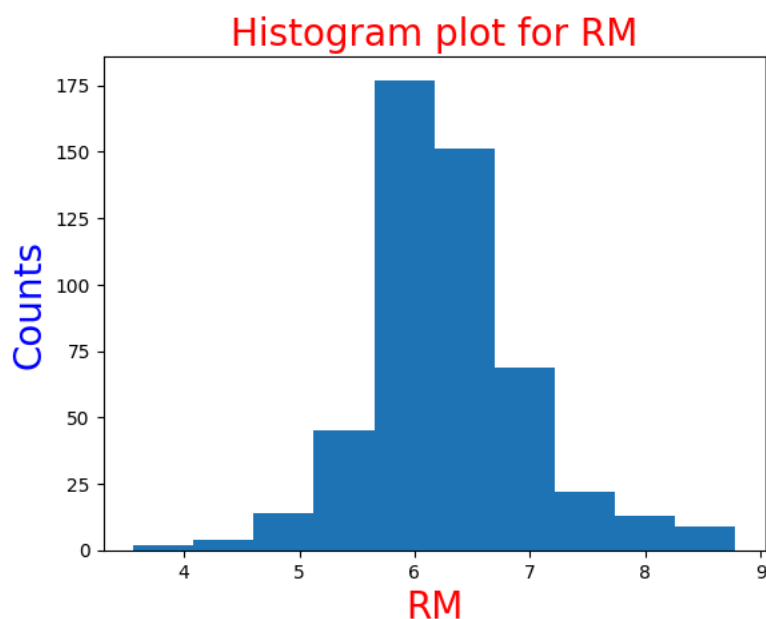
```
        INDUS
        18.10    127
        19.58     28
        8.14      22
        6.20      18
        21.89     14
                 ...
        1.89       1
        2.01       1
        4.15       1
        3.78       1
        2.02       1
        Name: count, Length: 76, dtype: int64
```
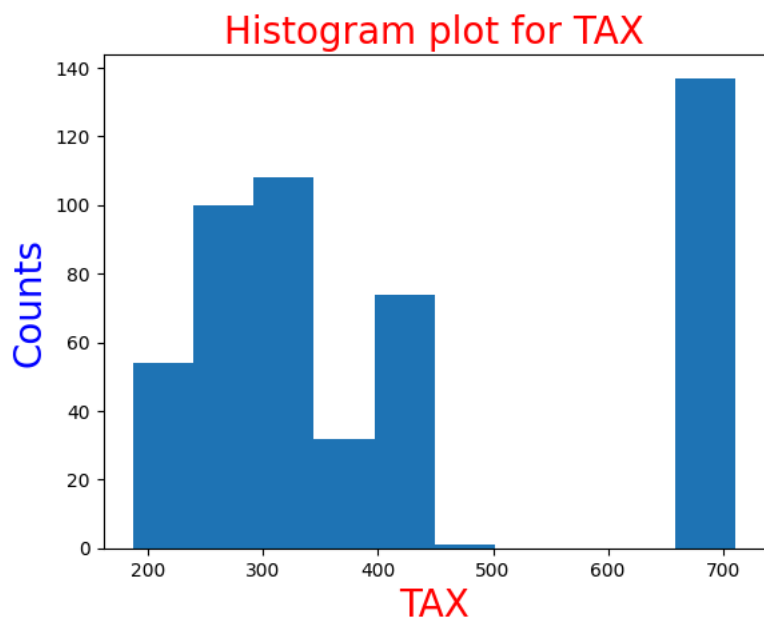
Bar plot for INDUS


```
#Task 7.6.3
import matplotlib.pyplot as plt
plt.hist(boston_pd['RM'])
plt.xlabel('RM', fontsize=20,color='red')
plt.ylabel('Counts', fontsize=20,color='blue')
plt.title('Histogram plot for RM', fontsize=20,color='red')
plt.show()
```

Histogram plot for RM


```
import matplotlib.pyplot as plt
plt.hist(boston_pd['TAX'])
plt.xlabel('TAX', fontsize=20,color='red')
```

```
plt.ylabel('Counts', fontsize=20,color='blue')
plt.title('Histogram plot for TAX', fontsize=20,color='red')
plt.show()
```



```
#task 7.7 multiple graphs in one figure
import matplotlib.pyplot as plt

plt.figure(figsize=(12,4))
## first graph
plt.subplot(1,2,1)
plt.hist(boston_pd['RM'])
plt.xlabel('RM', fontsize=20,color="red")
plt.ylabel('Counts', fontsize=20,color="green")
plt.title('Histogram plot for RM', fontsize=20)

## second graph
plt.subplot(1,2,2)

plt.scatter(boston_pd['RM'], boston_pd['DIS'] )
plt.xlabel('RM', fontsize=20,color="red")
plt.ylabel('DIS', fontsize=20,color="green")
plt.title('Scatter plot for RM and DIS', fontsize=20)
plt.tight_layout()
plt.show()
```