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

In [1]:
```
import numpy as np
```

In [3]:
```
import matplotlib as mpl
```

In [4]:
```
import matplotlib.pyplot as plt
```

In [5]:
```
import seaborn as sns
```

In [8]:
```
#1.1
data = pd.Series([164, 158,
172, 153, 144, 156, 189, 163, 134, 159, 143, 176, 177, 162, 141, 151, 182, 185
152])
data
```

Out[8]:
```
0     164
1     158
2     172
3     153
4     144
5     156
6     189
7     163
8     134
9     159
10    143
11    176
12    177
13    162
14    141
15    151
16    182
17    185
18    171
19    152
dtype: int64
```

In [10]:
```
#1.2
import pandas as pd

weight_lbs = ([164, 158,
172, 153, 144, 156, 189, 163, 134, 159, 143, 176, 177, 162, 141, 151, 182, 185
152])

weight_kg = [round(weight * 0.453592,2) for weight in weight_lbs]

weight_series = pd.Series(weight_kg)

print(weight_series)
```

```
0     74.39
1     71.67
2     78.02
3     69.40
4     65.32
5     70.76
6     85.73
7     73.94
8     60.78
9     72.12
10    64.86
11    79.83
12    80.29
13    73.48
14    63.96
15    68.49
16    82.55
17    83.91
18    77.56
19    68.95
dtype: float64
```

In [39]:
```python
#1.3
import pandas as pd

weight_lbs = ([164, 158,
172, 153, 144, 156, 189, 163, 134, 159, 143, 176, 177, 162, 141, 151, 182, 185
152])

weight_series_lbs = pd.Series(weight_lbs)

weight_kg = weight_series_lbs * 0.453592

mean_lbs = weight_series_lbs.mean()
median_lbs = weight_series_lbs.median()
std_dev_lbs = weight_series_lbs.std()

mean_kg = weight_kg.mean()
median_kg = weight_kg.median()
std_dev_kg = weight_kg.std()


print("weight in lbs")
print("Mean:",mean_lbs)
print("Median:", median_lbs)
print("Standard Deviation:", std_dev_lbs)
print("weight in kg:")
print("Mean:", mean_kg)
print("Median:", median_kg)
print("Standard Deviation:", std_dev_kg)
```

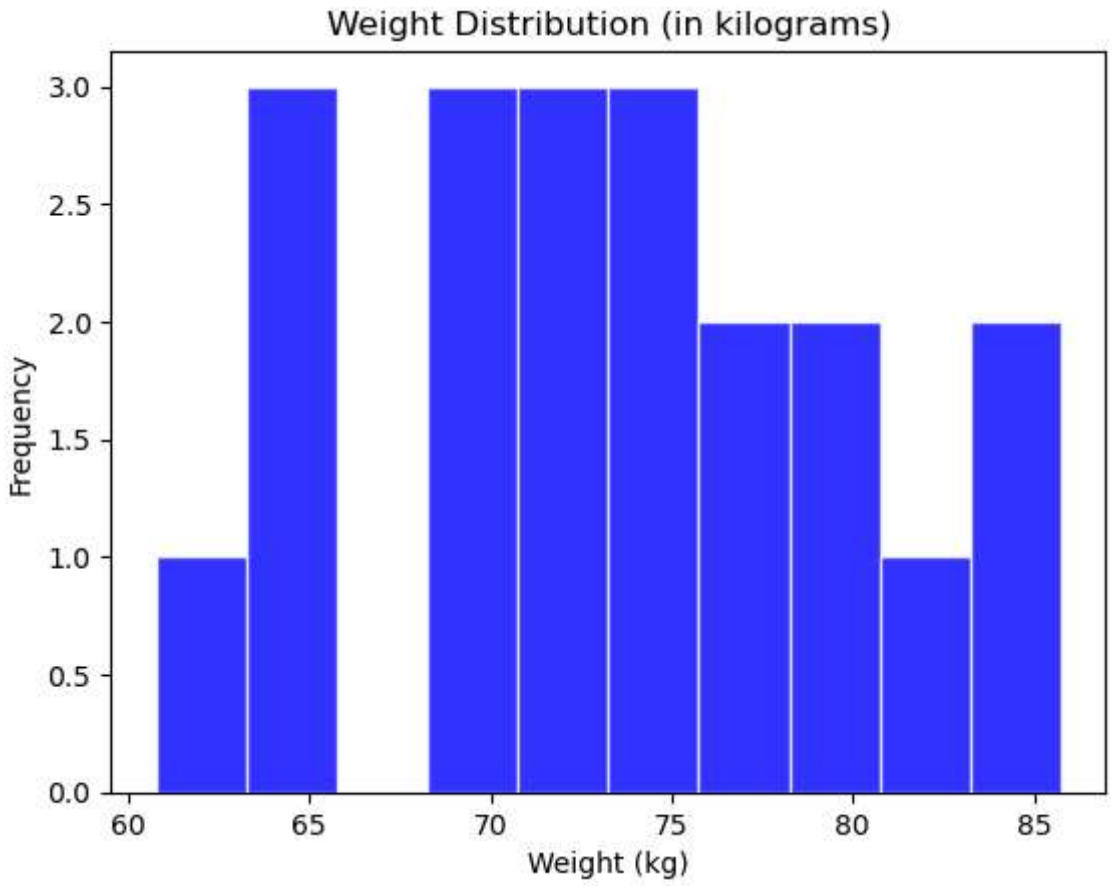
```
weight in lbs
Mean: 161.6
Median: 160.5
Standard Deviation: 15.44906742203316
weight in kg:
Mean: 73.30046720000001
Median: 72.80151599999999
Standard Deviation: 7.007573390094864
```

In [20]:
```
#1.4
import matplotlib.pyplot as plt

weights_lbs = [164, 158, 172, 153, 144, 156, 189, 163, 134, 159, 143, 176, 177

weights_kg = [weight * 0.453592 for weight in weights_lbs]


plt.hist(weights_kg, bins=10, color='blue', edgecolor='white', alpha=0.8)
plt.title('Weight Distribution (in kilograms)')
plt.xlabel('Weight (kg)')
plt.ylabel('Frequency')
plt.show()
```



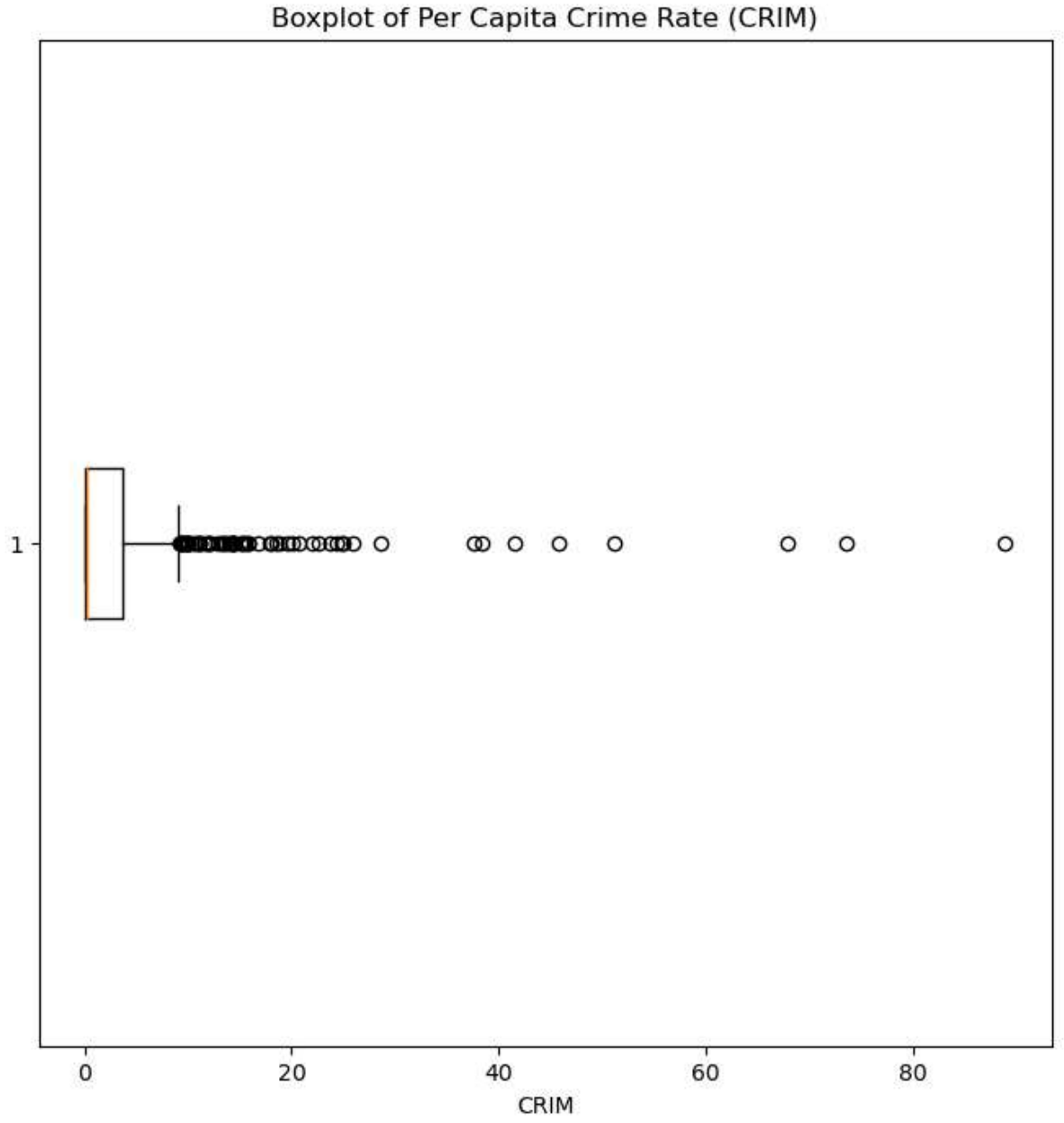
In [18]:
```
#2.1
import pandas as pd
df = pd.read_csv(r"C:\Users\gabed\.ipython\boston.csv")
num_rows, num_cols = df.shape
print("Number of rows:", num_rows)
print("Number of columns:", num_cols)
```

```
Number of rows: 506
Number of columns: 13
```

In [14]:
```python
#2.2
import pandas as pd
df = pd.read_csv(r"C:\Users\gabed\.ipython\boston.csv")
index_lowest_nox = df['NOX'].idxmin()
medv_lowest_nox = df.loc[index_lowest_nox, 'MEDV']
print("owner-occupied home value (MEDV) for the lowest nitric oxide concentrat
```

```
owner-occupied home value (MEDV) for the lowest nitric oxide concentration (N
OX): 20.1
```

In [19]:
```
#2.3
import pandas as pd
import matplotlib.pyplot as plt
df = pd.read_csv(r"C:\Users\gabed\.ipython\boston.csv")
plt.figure(figsize=(8, 8))
plt.boxplot(df['CRIM'], vert=False)
plt.title('Boxplot of Per Capita Crime Rate (CRIM)')
plt.xlabel('CRIM')
plt.show()
Q1 = df['CRIM'].quantile(0.25)
Q3 = df['CRIM'].quantile(0.75)
IQR = Q3 - Q1
print("Interquartile Range (IQR) for Crime Rate (CRIM):", IQR)
```




```
Interquartile Range (IQR) for Crime Rate (CRIM): 3.5950375
```

In [ ]:
```
#2.4
```

In [ ]:
```
#2.5
```

In [ ]:
```
#2.6
```

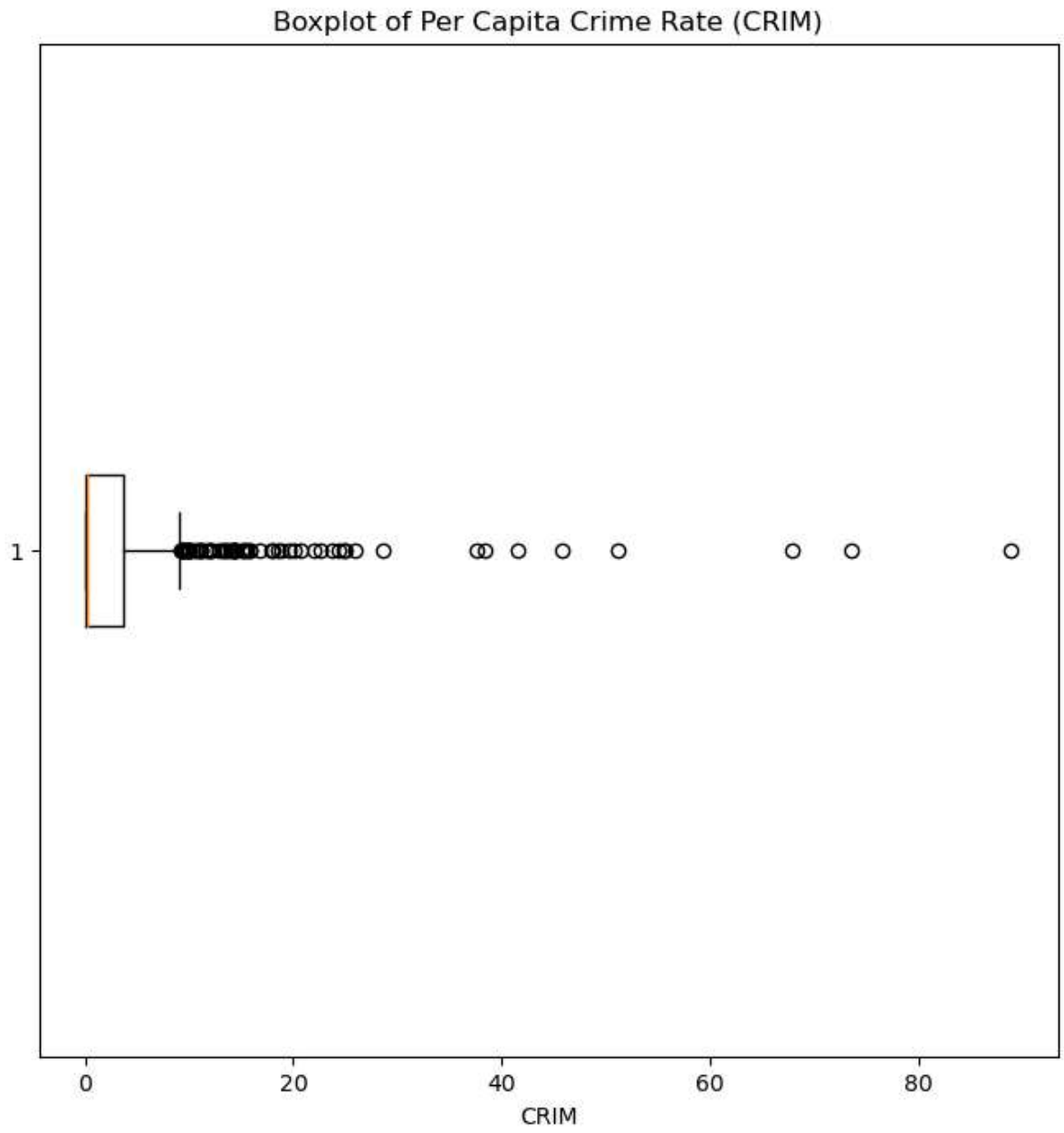
In [17]:
```
#3.1
import seaborn as sns
tips_df= sns.load_dataset('tips')
tips_df.head()
tips_df['tip_percent'] = (tips_df['tip'] / tips_df['total_bill']) * 100
tips_df['tip_percent'] = tips_df['tip_percent'].round(2)
print(tips_df.head())
```

```
   total_bill   tip     sex smoker  day    time  size  tip_percent
0       16.99  1.01  Female     No  Sun  Dinner     2         5.94
1       10.34  1.66    Male     No  Sun  Dinner     3        16.05
2       21.01  3.50    Male     No  Sun  Dinner     3        16.66
3       23.68  3.31    Male     No  Sun  Dinner     2        13.98
4       24.59  3.61  Female     No  Sun  Dinner     4        14.68
```

In [16]:
```
#3.2
tips_df = sns.load_dataset('tips')
mean_bill_per_day = tips_df.groupby('day')['total_bill'].mean()
day_highest_mean_bill = mean_bill_per_day.idxmax()
highest_mean_bill = mean_bill_per_day.max()
print("Days in the dataset:", mean_bill_per_day.index.tolist())
print("Day with the highest bill mean:", day_highest_mean_bill)
print("Highest mean bill amount:", highest_mean_bill)
```

```
Days in the dataset: ['Thur', 'Fri', 'Sat', 'Sun']
Day with the highest bill mean: Sun
Highest mean bill amount: 21.41
```

In [18]:
```
#3.3
```




```
Interquartile Range (IQR) for Crime Rate (CRIM): 3.5950375
```

In [ ]:
```
#3.4
```

In [ ]:
```
#3.5
```

In [ ]:
```
#4.1
```

In [ ]:
```
#4.2
```

In [ ]:
```
#4.3
```

In [ ]:
```
#4.4
```