

```
In [1]: import numpy as np
import pandas as pd
from matplotlib import pyplot as plt

In [2]: df = pd.read_csv("Daily_data_of_reservoir_level_of_Central_Water_Commission_(CWC)_Ag
df.head()
```

Out[2]:

	Reservoir_name	Basin	Subbasin	Agency_name	Lat	Long	Date	Year	Month
0	Aliyar Reservoir	West flowing rivers from Tadri to Kanyakumari ...	Varrar and others	CWC	10.472668	76.974597	01- 10- 23	2023	10
1	Aliyar Reservoir	West flowing rivers from Tadri to Kanyakumari ...	Varrar and others	CWC	10.472668	76.974597	02- 10- 23	2023	10
2	Aliyar Reservoir	West flowing rivers from Tadri to Kanyakumari ...	Varrar and others	CWC	10.472668	76.974597	03- 10- 23	2023	10
3	Aliyar Reservoir	West flowing rivers from Tadri to Kanyakumari ...	Varrar and others	CWC	10.472668	76.974597	04- 10- 23	2023	10
4	Aliyar Reservoir	West flowing rivers from Tadri to Kanyakumari ...	Varrar and others	CWC	10.472668	76.974597	05- 10- 23	2023	10

```
In [3]: df_revised=df.dropna()
df_revised
```

Out[3]:

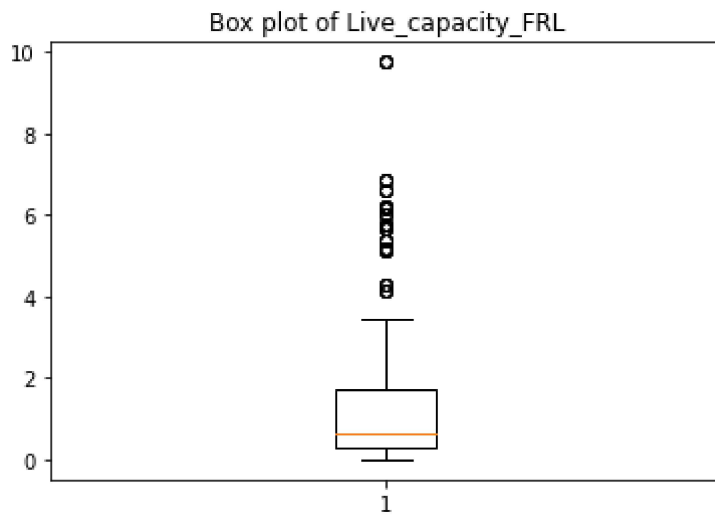
	Reservoir_name	Basin	Subbasin	Agency_name	Lat	Long	Date	Year	Mont
0	Aliyar Reservoir	West flowing rivers from Tadri to Kanyakumari ...	Varrar and others	CWC	10.472668	76.974597	01- 10- 23	2023	1
1	Aliyar Reservoir	West flowing rivers from Tadri to Kanyakumari ...	Varrar and others	CWC	10.472668	76.974597	02- 10- 23	2023	1

	Reservoir_name	Basin	Subbasin	Agency_name	Lat	Long	Date	Year	Mont
2	Aliyar Reservoir	West flowing rivers from Tadri to Kanyakumari ...	Varrar and others	CWC	10.472668	76.974597	03- 10- 23	2023	1
3	Aliyar Reservoir	West flowing rivers from Tadri to Kanyakumari ...	Varrar and others	CWC	10.472668	76.974597	04- 10- 23	2023	1
4	Aliyar Reservoir	West flowing rivers from Tadri to Kanyakumari ...	Varrar and others	CWC	10.472668	76.974597	05- 10- 23	2023	1
...
585	Yeldari Reservoir	Godavari Basin	Godavari Middle	CWC	19.752023	76.713746	01- 10- 23	2023	1
586	Yeldari Reservoir	Godavari Basin	Godavari Middle	CWC	19.752023	76.713746	02- 10- 23	2023	1
587	Yeldari Reservoir	Godavari Basin	Godavari Middle	CWC	19.752023	76.713746	03- 10- 23	2023	1
588	Yeldari Reservoir	Godavari Basin	Godavari Middle	CWC	19.752023	76.713746	04- 10- 23	2023	1
589	Yeldari Reservoir	Godavari Basin	Godavari Middle	CWC	19.752023	76.713746	05- 10- 23	2023	1

437 rows × 13 columns

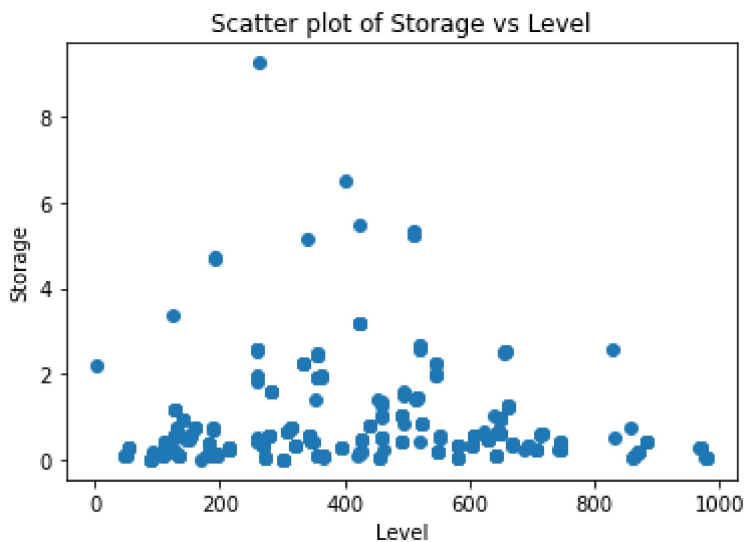
```
In [4]: # Creating box plot
plt.boxplot(df['Live_capacity_FRL'])
plt.title('Box plot of Live_capacity_FRL')
```

Out[4]: Text(0.5, 1.0, 'Box plot of Live_capacity_FRL')



```
In [5]: plt.scatter(df_revised.Level,df_revised.Storage)
plt.xlabel('Level')
plt.ylabel('Storage')
plt.title('Scatter plot of Storage vs Level')
```

Out[5]: Text(0.5, 1.0, 'Scatter plot of Storage vs Level')



```
In [6]: barWidth = 0.25
fig, ax = plt.subplots(figsize=(30, 200))

# Sample data for demonstration
reservoir_names = df_revised.Reservoir_name
level1 = df_revised.Level
level2 = df_revised.Full_reservoir_level # Replace 'AnotherLevel' with your second d

# Calculate the positions for the bars
pos1 = np.arange(len(reservoir_names))
pos2 = [x + barWidth for x in pos1]

# Plot the first set of bars
ax.barh(pos1, level1, barWidth, label='Current Level')

# Plot the second set of bars
ax.barh(pos2, level2, barWidth, label='Full or Maximum Reservoir level')

# Remove axes splines
for s in ['top', 'bottom', 'left', 'right']:
    ax.spines[s].set_visible(False)
```

```
# Show top values
ax.invert_yaxis()

# Add annotation to bars
for i, (val1, val2) in enumerate(zip(level1, level2)):
    plt.text(val1 + 0.2, pos1[i] + 0.5, str(round(val1, 2)), fontsize=20, fontweight
    plt.text(val2 + 0.2, pos2[i] + 0.5, str(round(val2, 2)), fontsize=20, fontweight

# Set Plot Title
ax.set_title('Reservoir name and their levels', loc='left')

# Set y-axis labels to the reservoir names
ax.set_yticks((pos1 + pos2) / 2)
ax.set_yticklabels(reservoir_names)

# Add a legend for the two sets of bars
ax.legend(loc='upper right', title='Legend')

# Show the plot
plt.show()
```

