

Assignment NO: 3

Name :Pathak Diksha Vilas

Rollno:58

Batch:3

Class:T.YBsc(Computer Science)

Slip 13(B)

Write a Python program to find the maximum and minimum value of a given flattened array

```
In [1]: import pandas as pd
import numpy as np
arr = np.array([3, 5, 7, 2, 8, 1, 4, 6, 9])
flattened_arr = arr.flatten()
max_value = np.max(flattened_arr)
min_value = np.min(flattened_arr)
print(f"Maximum value: {max_value}")
print(f"Minimum value: {min_value}")
```

Maximum value: 9
Minimum value: 1

Slips12(b)

Write a Python program to create data frame containing column name, salary, department add 10 rows with some missing and duplicate values to the data frame. Also drop all null and empty values. Print the modified data frame

```
In [2]: import pandas as pd
data = {
    'name': ['Ram', 'Payal', 'Rani', 'Joti', 'Sakshi', None, 'Shital', 'Nita', 'Joy', 'Bob'],
    'salary': [50000, 60000, 70000, 60000, None, 45000, None, 55000, 75000, 70000],
    'department': ['HR', 'IT', 'Finance', 'IT', 'HR', 'IT', None, 'Finance', 'HR', 'Finance']
}
```

```
In [3]: df = pd.DataFrame(data)
print("Original DataFrame:\n", df)
```

Original DataFrame:

	name	salary	department
0	Ram	50000.0	HR
1	Payal	60000.0	IT
2	Rani	70000.0	Finance
3	Joti	60000.0	IT
4	Sakshi	NaN	HR
5	None	45000.0	IT
6	Shital	NaN	None
7	Nita	55000.0	Finance
8	Joy	75000.0	HR
9	Bob	70000.0	Finance

```
In [4]: df_cleaned = df.dropna()
```

```
In [5]: df_cleaned = df_cleaned.drop_duplicates()
print("\nModified DataFrame (after dropping nulls and duplicates):\n", df_cleaned)
```

Modified DataFrame (after dropping nulls and duplicates):

	name	salary	department
0	Ram	50000.0	HR
1	Payal	60000.0	IT
2	Rani	70000.0	Finance
3	Joti	60000.0	IT
7	Nita	55000.0	Finance
8	Joy	75000.0	HR
9	Bob	70000.0	Finance

slips26,30(b)

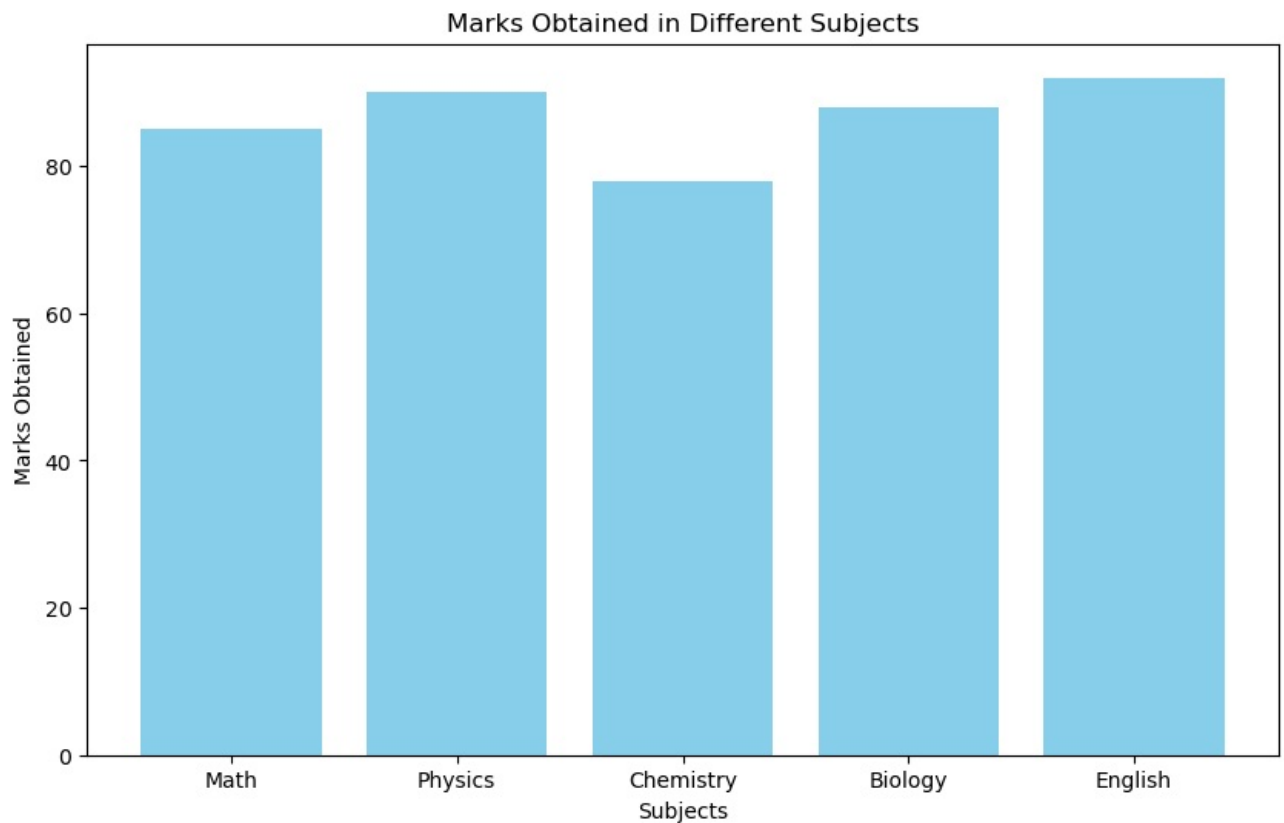
Create two lists, one representing subject names and the other representing marks obtained in those subjects. Display the data in bar chart

```
In [6]: import pandas as pd
```

```
import matplotlib.pyplot as plt
```

```
In [7]: subjects = ['Math', 'Physics', 'Chemistry', 'Biology', 'English']  
marks = [85, 90, 78, 88, 92]
```

```
In [8]: plt.figure(figsize=(10, 6))  
plt.bar(subjects, marks, color='skyblue')  
plt.xlabel('Subjects')  
plt.ylabel('Marks Obtained')  
plt.title('Marks Obtained in Different Subjects')  
plt.show()
```



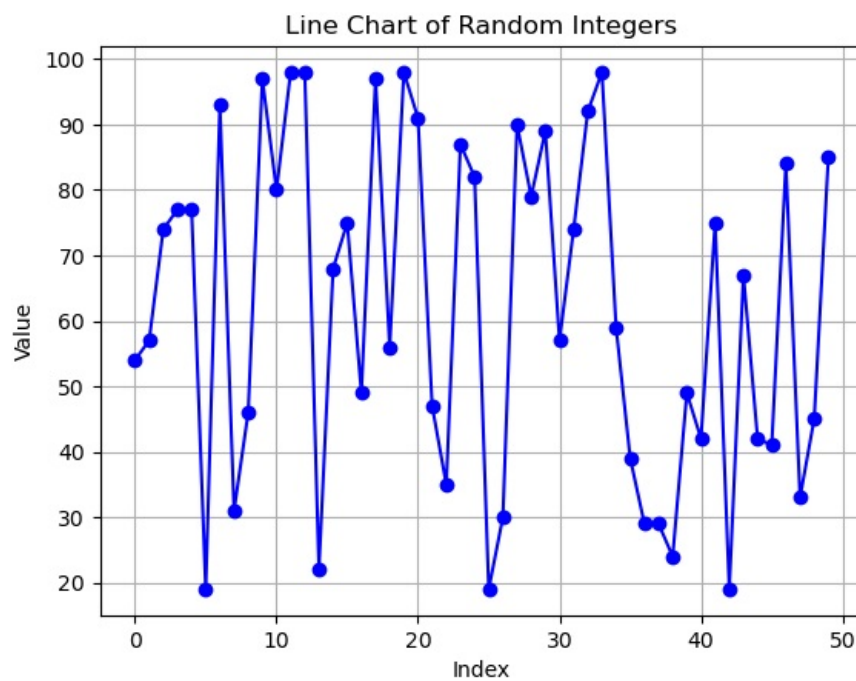
Slips4,5,12,15,20,25,26,30(a)

Generate a random array of 50 integers and display them using a line chart, scatter plot, histogram and box plot. Apply appropriate color, labels and styling options. import pandas as pd

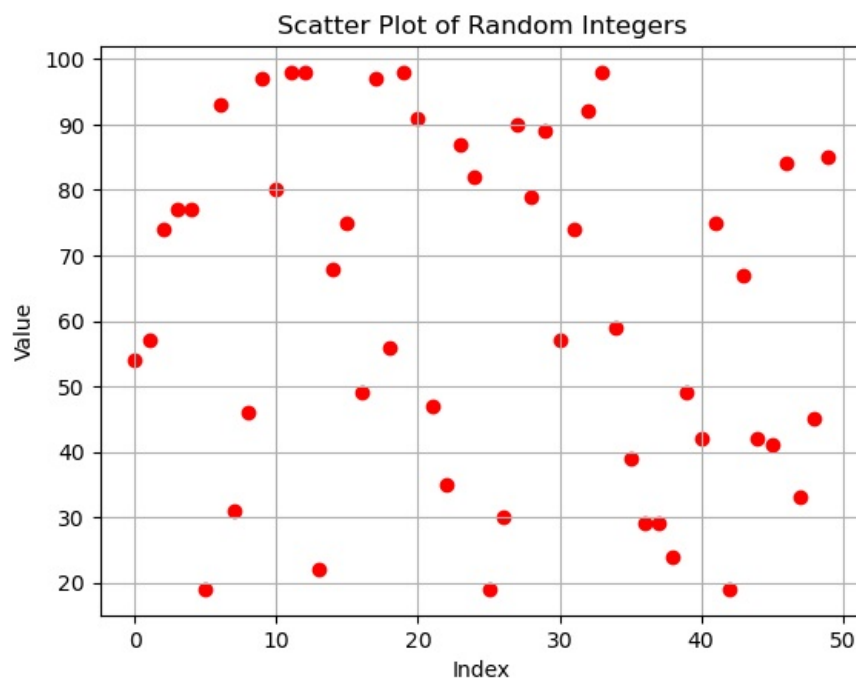
```
In [9]: import pandas as pd  
import numpy as np  
import matplotlib.pyplot as plt
```

```
In [10]: np.random.seed(0) # For reproducibility  
random_integers = np.random.randint(10, 100, 50)
```

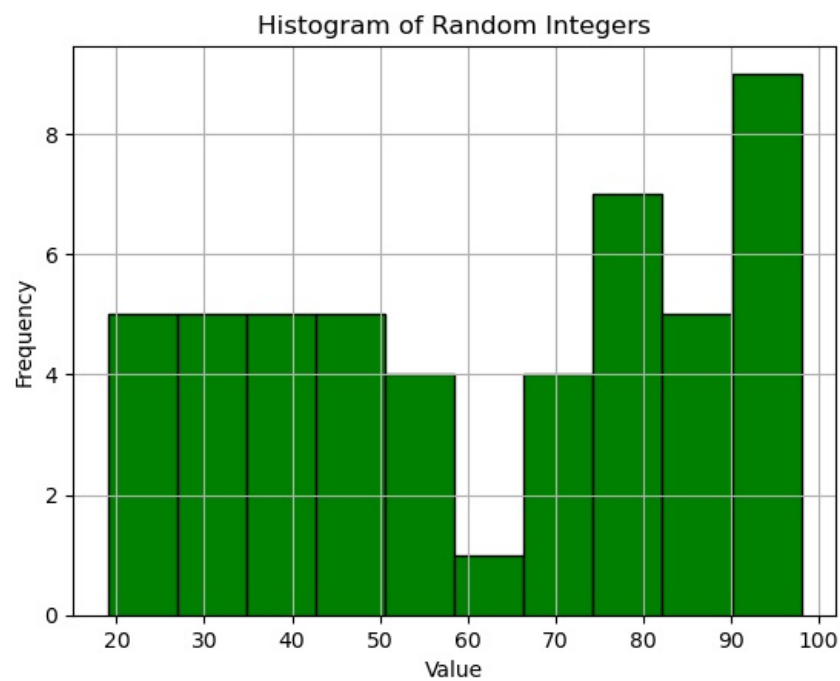
```
In [11]: plt.plot(random_integers, color='blue', linestyle='-', marker='o')  
plt.title('Line Chart of Random Integers')  
plt.xlabel('Index')  
plt.ylabel('Value')  
plt.grid(True)
```



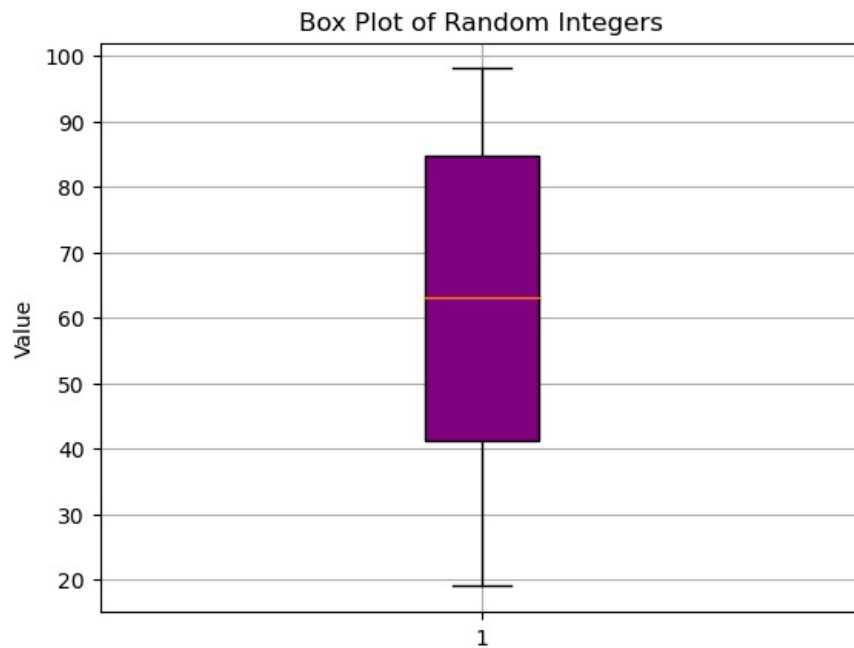
```
In [12]: plt.scatter(range(len(random_integers)), random_integers, color='red')
plt.title('Scatter Plot of Random Integers')
plt.xlabel('Index')
plt.ylabel('Value')
plt.grid(True)
```



```
In [13]: plt.hist(random_integers, bins=10, color='green', edgecolor='black')
plt.title('Histogram of Random Integers')
plt.xlabel('Value')
plt.ylabel('Frequency')
plt.grid(True)
```



```
In [14]: plt.boxplot(random_integers, patch_artist=True, boxprops=dict(facecolor='purple', color='black'))
plt.title('Box Plot of Random Integers')
plt.ylabel('Value')
plt.grid(True)
```



slip9,15,25

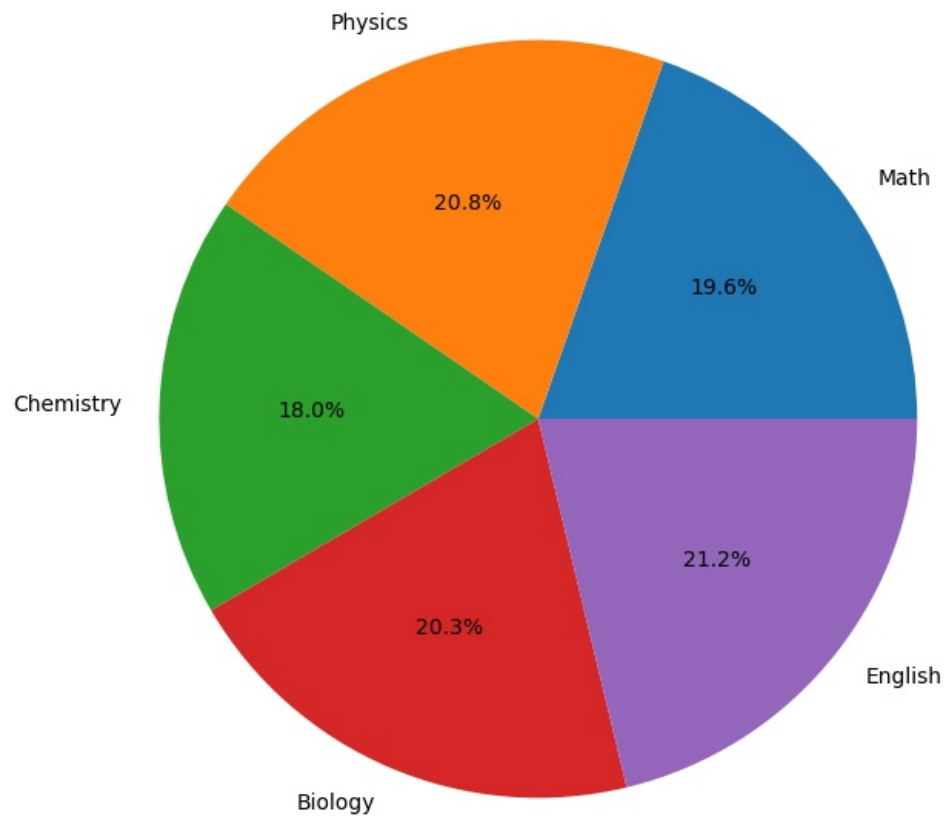
Create two lists, one representing subject names and the other representing marks obtained in those subjects. Display the data in a pie chart.

```
In [16]: import pandas as pd
import matplotlib.pyplot as plt
```

```
In [17]: subjects = ['Math', 'Physics', 'Chemistry', 'Biology', 'English']
marks = [85, 90, 78, 88, 92]
```

```
In [18]: plt.figure(figsize=(8, 8))
plt.pie(marks, labels=subjects, autopct='%1.1f%%')
plt.title('Marks Distribution in Different Subjects')
plt.show()
```

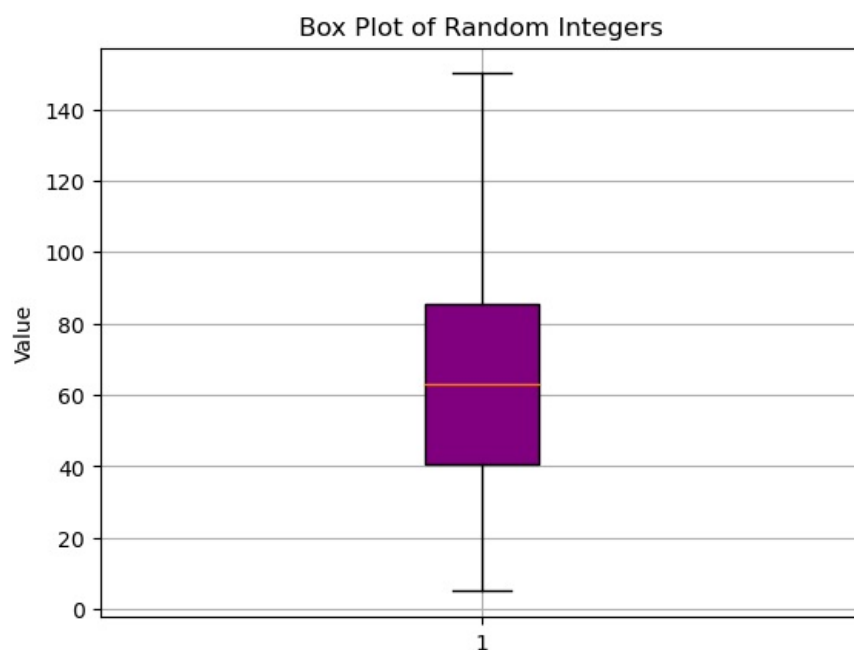
Marks Distribution in Different Subjects



Slips20(b)

B) Add two outliers to the above data and display the box plot.

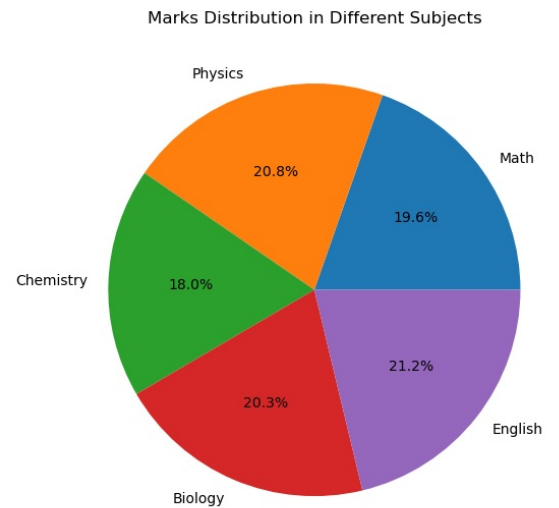
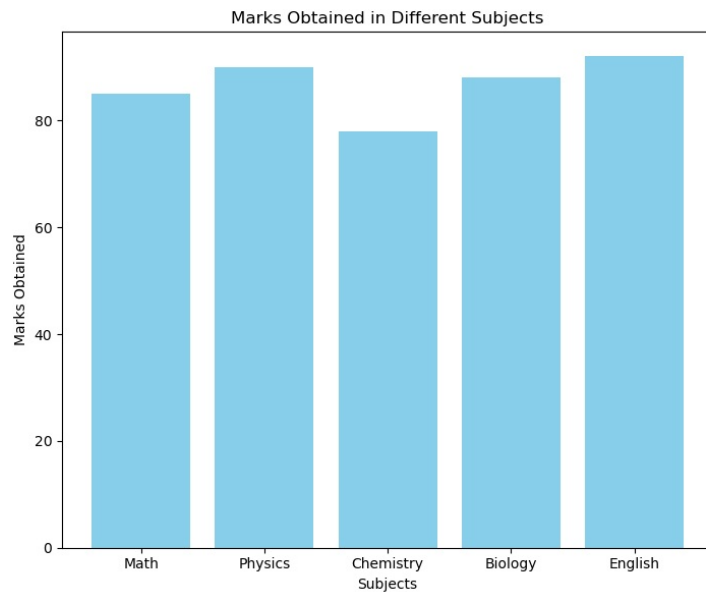
```
In [19]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
np.random.seed(0)
random_integers = np.random.randint(10, 100, size=50)
random_integers = np.append(random_integers, [5, 150])
plt.boxplot(random_integers, patch_artist=True, boxprops=dict(facecolor='purple', color='black'))
plt.title('Box Plot of Random Integers')
plt.ylabel('Value')
plt.grid(True)
```



slip16(a)

Write a python program to create two lists, one representing subject names and the other representing marks obtained in those subjects.
Display the data in a pie chart and bar chart

```
In [20]: import pandas as pd
import matplotlib.pyplot as plt
subjects = ['Math', 'Physics', 'Chemistry', 'Biology', 'English']
marks = [85, 90, 78, 88, 92]
plt.figure(figsize=(14, 6))
plt.subplot(1, 2, 1)
plt.bar(subjects, marks, color='skyblue')
plt.xlabel('Subjects')
plt.ylabel('Marks Obtained')
plt.title('Marks Obtained in Different Subjects')
plt.subplot(1, 2, 2)
plt.pie(marks, labels=subjects, autopct='%1.1f%%')
plt.title('Marks Distribution in Different Subjects')
plt.tight_layout()
plt.show()
```



In []:

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js