

Main.java



Run

Output

Clear

```
1 import java.util.Arrays;
2 import java.util.HashMap;
3 import java.util.Map;
4 public class ArrayStatistics {
5     public static void main(String[] args) {
6         int[] arr = {16, 18, 27, 16, 23, 21, 19};
7         int mean = calculateMean(arr);
8         int median = calculateMedian(arr);
9         int mode = calculateMode(arr);
10        System.out.println("Mean = " + mean);
11        System.out.println("Median = " + median);
12        System.out.println("Mode = " + mode);
13    }
14    public static int calculateMean(int[] arr) {
15        int sum = 0;
16        for (int num : arr) {
17            sum += num;
18        }
19        return sum / arr.length;
20    }
21    public static int calculateMedian(int[] arr) {
22        Arrays.sort(arr);
```

```
^ java -cp /tmp/4gkcZ0WLp0 ArrayStatistics
Mean = 20Median = 19
Mode = 16
|
```

Main.java



Run

Output

Clear

```
21 public static int calculateMedian(int[] arr) {
22     Arrays.sort(arr);
23     int middle = arr.length / 2;
24     if (arr.length % 2 == 0) {
25         return (arr[middle - 1] + arr[middle]) / 2;
26     } else {
27         return arr[middle];
28     }
29 }
30 public static int calculateMode(int[] arr) {
31     Map<Integer, Integer> frequencyMap = new HashMap<>();
32     int mode = -1;
33     int maxFrequency = 0;
34     for (int num : arr) {
35         int frequency = frequencyMap.getOrDefault(num, 0) + 1;
36         frequencyMap.put(num, frequency);
37         if (frequency > maxFrequency) {
38             maxFrequency = frequency;
39             mode = num;
40         }
41     }
42 }
```

```
java -cp /tmp/4gkcZ0WLp0 ArrayStatistics
Mean = 20Median = 19
Mode = 16
```

Main.java



Run

Output

Clear

```
25         return (arr[middle - 1] + arr[middle]) / 2;
26     } else {
27         return arr[middle];
28     }
29 }
30 public static int calculateMode(int[] arr) {
31     Map<Integer, Integer> frequencyMap = new HashMap<>();
32     int mode = -1;
33     int maxFrequency = 0;
34     for (int num : arr) {
35         int frequency = frequencyMap.getOrDefault(num, 0) + 1;
36         frequencyMap.put(num, frequency);
37         if (frequency > maxFrequency) {
38             maxFrequency = frequency;
39             mode = num;
40         }
41     }
42
43     return mode;
44 }
45 }
46 }
```

```
java -cp /tmp/4gkcZOWLp0 ArrayStatistics
Mean = 20Median = 19
Mode = 16
```