In [1]:

```
def calculate_mean(numbers):
 return sum(numbers) / len(numbers)
```

In [2]:

```
def calculate_median(numbers):
 numbers.sort()
 n = len(numbers)
 middle = n // 2
 return (numbers[middle - 1] + numbers[middle]) / 2 if n % 2 == 0
```

In [3]:

```
def calculate_standard_deviation(numbers):
 mean = calculate_mean(numbers)
 variance = sum((x - mean) ** 2 for x in numbers) / len(numbers)
 return variance ** 0.5
```

In [4]:

```
def calculate_mode(numbers):
 from collections import Counter
 frequency = Counter(numbers)
 max_count = max(frequency.values())
 return [key for key, value in frequency.items() if value == max_c
```

In [5]:

```
# Input and Execution
numbers = list(map(int, input("Enter numbers separated by spaces:
```

```
Enter numbers separated by spaces: 1 2 3
```

In [6]:

```
print(f"Mean: {calculate_mean(numbers)}")
print(f"Median: {calculate_median(numbers)}")
print(f"Standard Deviation: {calculate_standard_deviation(numbers)
print(f"Mode: {calculate_mode(numbers)}")
```

```
Mean: 2.0
Median: 2
Standard Deviation: 0.816496580927726
Mode: [1, 2, 3]
```

In [ ]:

```

```