```
===== NumPy Analyzer =====
1.Create Array
2.Indexing & Slicing
3.Math Operations
4.Combine / Split
5.Search / Sort / Filter
6.Aggregates & Statistics
7.Exit
Enter choice: 1
Enter array dimension (1 or 2): 1
Enter number of elements: 5
Enter elements: 11 12 13 14 15
Array created:
 [11 12 13 14 15]

===== NumPy Analyzer =====
1.Create Array
2.Indexing & Slicing
3.Math Operations
4.Combine / Split
5.Search / Sort / Filter
6.Aggregates & Statistics
7.Exit
Enter choice: 1
Enter array dimension (1 or 2): 2
Enter rows: 2
Enter columns: 3
Enter 6 elements: 11 22 33 44 55 66
Array created:
 [[11 22 33]
 [44 55 66]]
```

```
===== NumPy Analyzer =====
1.Create Array
2.Indexing & Slicing
3.Math Operations
4.Combine / Split
5.Search / Sort / Filter
6.Aggregates & Statistics
7.Exit
Enter choice: 2

1. Indexing
2. Slicing
Choose option: 1
Enter index: 0
Element: [11 22 33]

===== NumPy Analyzer =====
1.Create Array
2.Indexing & Slicing
3.Math Operations
4.Combine / Split
5.Search / Sort / Filter
6.Aggregates & Statistics
7.Exit
Enter choice: 2

1. Indexing
2. Slicing
Choose option: 2
Row range (start:end): 0:2
Column range (start:end): 1:3
Sliced Array:
 [[22 33]
 [55 66]]
```

```
===== NumPy Analyzer =====
1.Create Array
2.Indexing & Slicing
3.Math Operations
4.Combine / Split
5.Search / Sort / Filter
6.Aggregates & Statistics
7.Exit
Enter choice: 3

1.Addition
2.Subtraction
3.Multiplication
4.Division
Choose operation: 1
Enter same size array elements: 66 55 44 33 22 11
Result:
 [[77 77 77]
 [77 77 77]]

===== NumPy Analyzer =====
1.Create Array
2.Indexing & Slicing
3.Math Operations
4.Combine / Split
5.Search / Sort / Filter
6.Aggregates & Statistics
7.Exit
Enter choice: 3

1.Addition
2.Subtraction
3.Multiplication
4.Division
Choose operation: 2
Enter same size array elements: 11 22 33 44 55 66
Result:
 [[0 0 0]
 [0 0 0]]
```

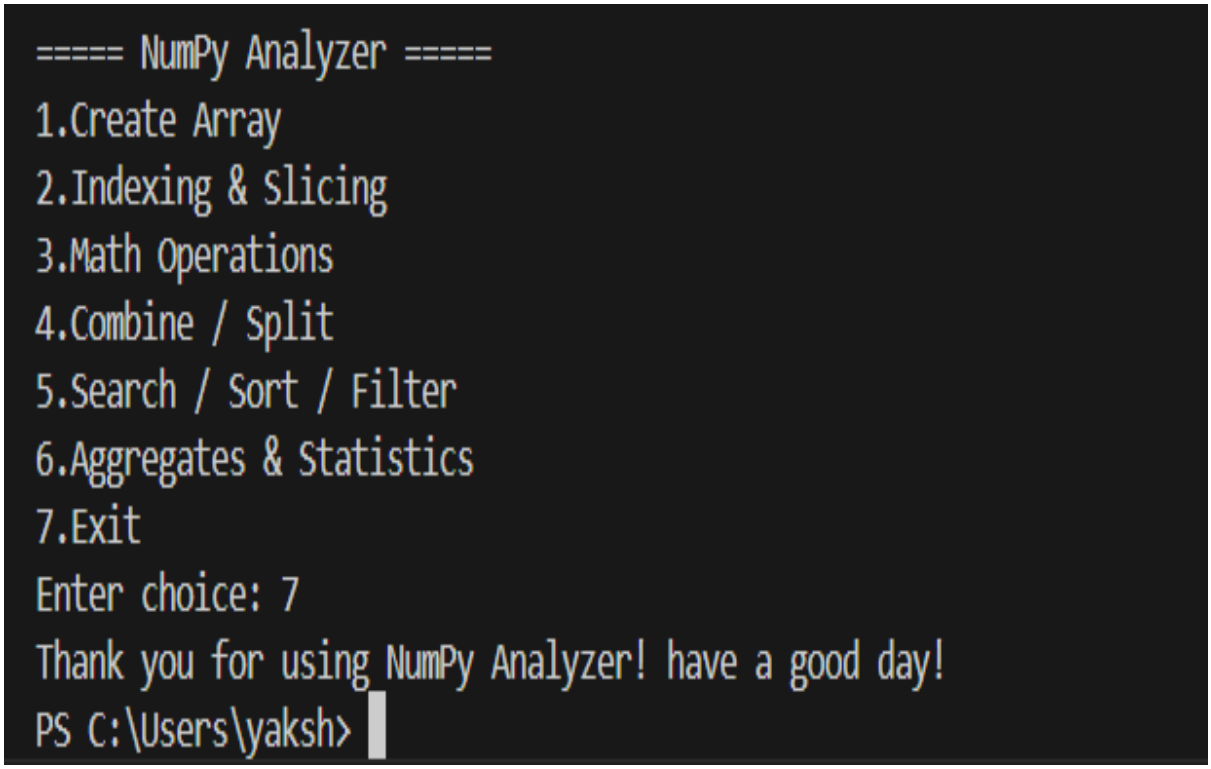
```
===== NumPy Analyzer =====
1.Create Array
2.Indexing & Slicing
3.Math Operations
4.Combine / Split
5.Search / Sort / Filter
6.Aggregates & Statistics
7.Exit
Enter choice: 3

1.Addition
2.Subtraction
3.Multiplication
4.Division
Choose operation: 3
Enter same size array elements: 11 22 33 44 55 66
Result:
 [[ 121  484 1089]
 [1936 3025 4356]]

===== NumPy Analyzer =====
1.Create Array
2.Indexing & Slicing
3.Math Operations
4.Combine / Split
5.Search / Sort / Filter
6.Aggregates & Statistics
7.Exit
Enter choice: 3

1.Addition
2.Subtraction
3.Multiplication
4.Division
Choose operation: 4
Enter same size array elements: 11 22 33 44 55 66
Result:
 [[1. 1. 1.]
 [1. 1. 1.]]
```

```
===== NumPy Analyzer =====
1.Create Array
2.Indexing & Slicing
3.Math Operations
4.Combine / Split
5.Search / Sort / Filter
6.Aggregates & Statistics
7.Exit
Enter choice: 4

1. Combine
2. Split
Choose: 1
Enter elements: 1 2 3 4 5 6
Combined:
 [[11 22 33]
 [44 55 66]
 [ 1  2  3]
 [ 4  5  6]]

===== NumPy Analyzer =====
1.Create Array
2.Indexing & Slicing
3.Math Operations
4.Combine / Split
5.Search / Sort / Filter
6.Aggregates & Statistics
7.Exit
Enter choice: 4

1. Combine
2. Split
Choose: 2
Number of splits: 2
Splitted Arrays: [array([[11, 22, 33]]), array([[44, 55, 66]])]
```

```
===== NumPy Analyzer =====
1.Create Array
2.Indexing & Slicing
3.Math Operations
4.Combine / Split
5.Search / Sort / Filter
6.Aggregates & Statistics
7.Exit
Enter choice: 5

1.Search
2.Sort
3.Filter
Choose: 1
Value to search: 5
Found at positions: (array([], dtype=int64), array([], dtype=int64))

===== NumPy Analyzer =====
1.Create Array
2.Indexing & Slicing
3.Math Operations
4.Combine / Split
5.Search / Sort / Filter
6.Aggregates & Statistics
7.Exit
Enter choice: 5

1.Search
2.Sort
3.Filter
Choose: 2
Sorted Array:
 [[11 22 33]
 [44 55 66]]
```

```
===== NumPy Analyzer =====
1.Create Array
2.Indexing & Slicing
3.Math Operations
4.Combine / Split
5.Search / Sort / Filter
6.Aggregates & Statistics
7.Exit
Enter choice: 5

1.Search
2.Sort
3.Filter
Choose: 3
Filter values greater than: 33
Filtered:
 [44 55 66]
```

```
===== NumPy Analyzer =====
1.Create Array
2.Indexing & Slicing
3.Math Operations
4.Combine / Split
5.Search / Sort / Filter
6.Aggregates & Statistics
7.Exit
Enter choice: 6
Sum: 231
Mean: 38.5
Median: 38.5
Std Dev: 18.786076404259266
Variance: 352.9166666666667
Min: 11
Max: 66
Percentile (50%): 38.5
```


===== NumPy Analyzer =====
1.Create Array
2.Indexing & Slicing
3.Math Operations
4.Combine / Split
5.Search / Sort / Filter
6.Aggregates & Statistics
7.Exit
Enter choice: 7
Thank you for using NumPy Analyzer! have a good day!
PS C:\Users\yaksh>