

# SOFTWARE CONSTRUCTION AND DEVELOPEMNT

LAB REPORT #2



Umar Ali  
B22F0088SE094  
SE-F22 Red

**Task #1:**

Declare an array of 5 integers. Accept user input for all elements. Print the elements of the array.

**Code Solution:**

```
1  import java.util.Scanner;
2
3  public class task1 {
4      Run main | Debug main
5      public static void main(String[] args) {
6          int size;
7          Scanner scanner = new Scanner(System.in);
8          System.out.println("Enter the size of the array: ");
9          size = scanner.nextInt();
10         int[] arr = new int[size];
11
12         System.out.println("Array of size " + arr.length + " created");
13
14         for (int i=0; i<=arr.length-1; i++){
15             System.out.println("Enter the element at index " + i + ": ");
16             arr[i] = scanner.nextInt();
17         }
18
19         for(int i=0; i<=arr.length-1; i++){
20             System.out.println("Element at index " + i + " is " + arr[i]);
21         }
22     }
```

**Output:**

```
PS D:\6th\Construction and Development\Lab Work\Lab 2 SCD Umar Ali B22F0088SE094> java task1
Enter the size of the array:
5
Array of size 5 created
Enter the element at index 0:
42
Enter the element at index 1:
45
Enter the element at index 2:
87
Enter the element at index 3:
96
Enter the element at index 4:
12
Element at index 0 is 42
Element at index 1 is 45
Element at index 2 is 87
Element at index 3 is 96
Element at index 4 is 12
PS D:\6th\Construction and Development\Lab Work\Lab 2 SCD Umar Ali B22F0088SE094> |
```

### Explanation of Code:

This code takes input with help Scanner object which is size of array and then it will take inputs of array values with for loop and Scanner object and at the end the inputted values will be printed.

### Task2:

Reverse the Array:

### Code:

```
1  import java.util.Scanner;
2
3  public class task2{
4      Run main | Debug main
5      public static void main(String[] args) {
6          int size;
7          Scanner scanner = new Scanner(System.in);
8          System.out.println("Enter the size of the array: ");
9          size = scanner.nextInt();
10         int[] arr = new int[size];
11         System.out.println("Array of size " + arr.length + " created");
12         for (int i=0; i<=arr.length-1; i++){
13             System.out.println("Enter the element at index " + i + ": ");
14             arr[i] = scanner.nextInt();
15         }
16         for(int i=0; i<=arr.length-1; i++){
17             System.out.println("Element at index " + i + " is " + arr[i]);
18             System.out.println("/n");
19         }
20         System.out.println("Reversing the Array");
21
22         int left=0; int right=arr.length-1;
23         while(left<right){
24             int temp=arr[left];
25             arr[left]=arr[right];
26             arr[right]=temp;
27             left++;
28             right--;
29         }
30         for(int i=0; i<=arr.length-1; i++){
31             System.out.println("Element at index " + i + " is " + arr[i]);
32             System.out.println("/n");
33         }
34     }
35 }
```

### **Output:**

```
PS D:\6th\Construction and Development\Lab Work\Lab 2 SCD Umar Ali B22F0088SE094> java task2
Enter the size of the array:
5
Array of size 5 created
Enter the element at index 0:
1
Enter the element at index 1:
2
Enter the element at index 2:
3
Enter the element at index 3:
4
Enter the element at index 4:
5
Element at index 0 is 1
Element at index 1 is 2
Element at index 2 is 3
Element at index 3 is 4
Element at index 4 is 5
Reversing the Array
Element at index 0 is 5
Element at index 1 is 4
Element at index 2 is 3
Element at index 3 is 2
Element at index 4 is 1
PS D:\6th\Construction and Development\Lab Work\Lab 2 SCD Umar Ali B22F0088SE094> |
```

### **Explanation of Code:**

This code is a typical and basic DSA problem for Linked List and Array. In the code i have allowed the user to give input and then those inputs are printed after that i have used a double pointer method to reverse the array. The first pointer is `int left` which was initially declared to zero and second pointer is `right` whose value is **`int right = arr.length()-1`**; means it will allow us to access the last element of array and **`int left=0`** allows us to access the start of array. Using while loop we will keep swapping the value at left and right and when left and right becomes equal or left exceed right it will end and until then it will continue to swap values. **As Pointers are not supported in Java so it is done through simple variable.**

---