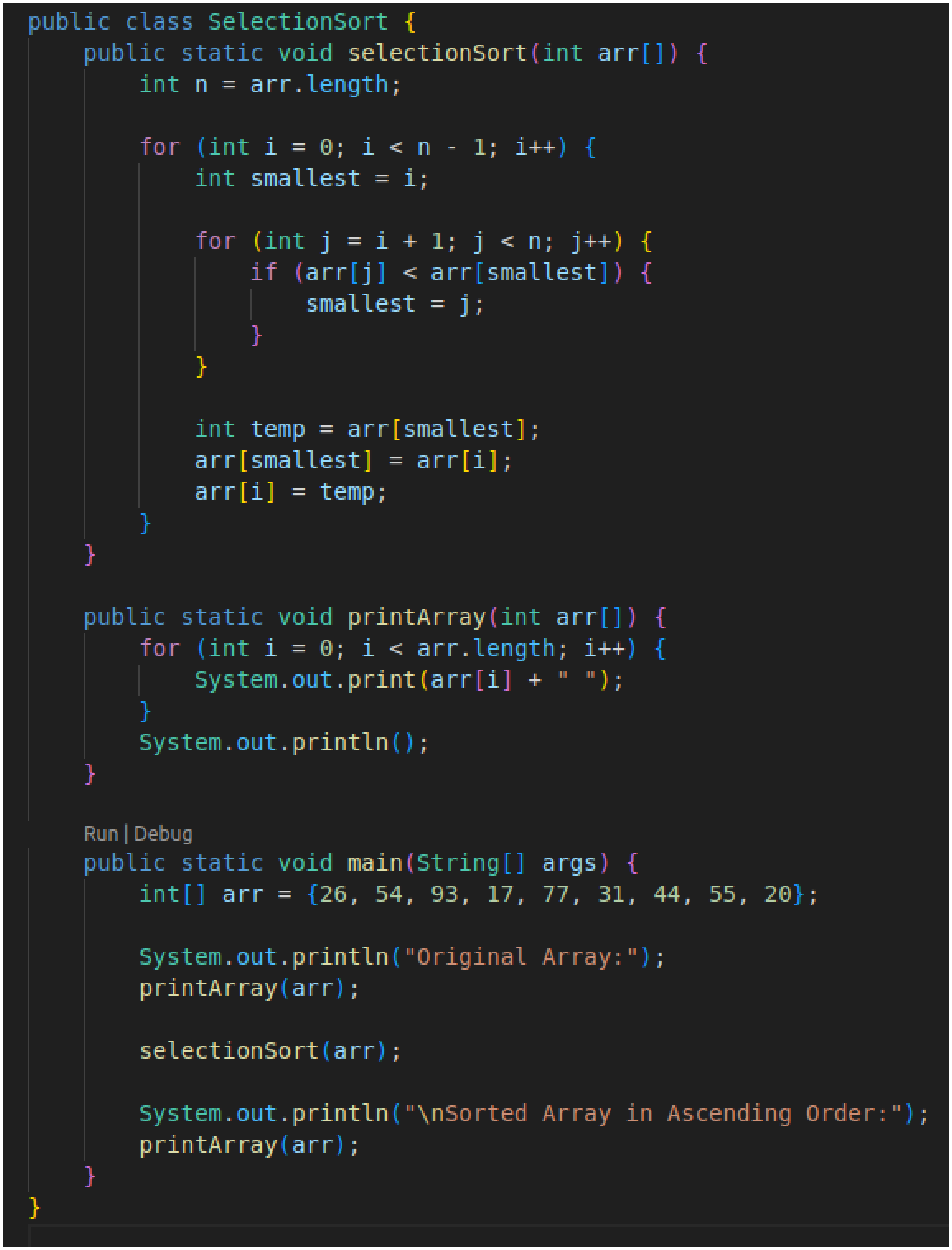
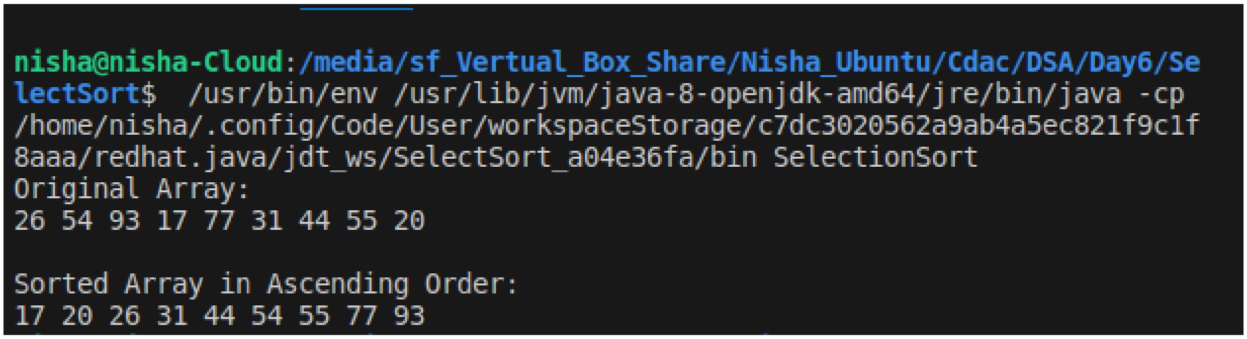
Lab Assignment [DSA] -6

Qn1. Create an array & insert the elements like 26 54 93 17 77 31 44 55 20 sort the array in Ascending order using Selection sort and display the result.

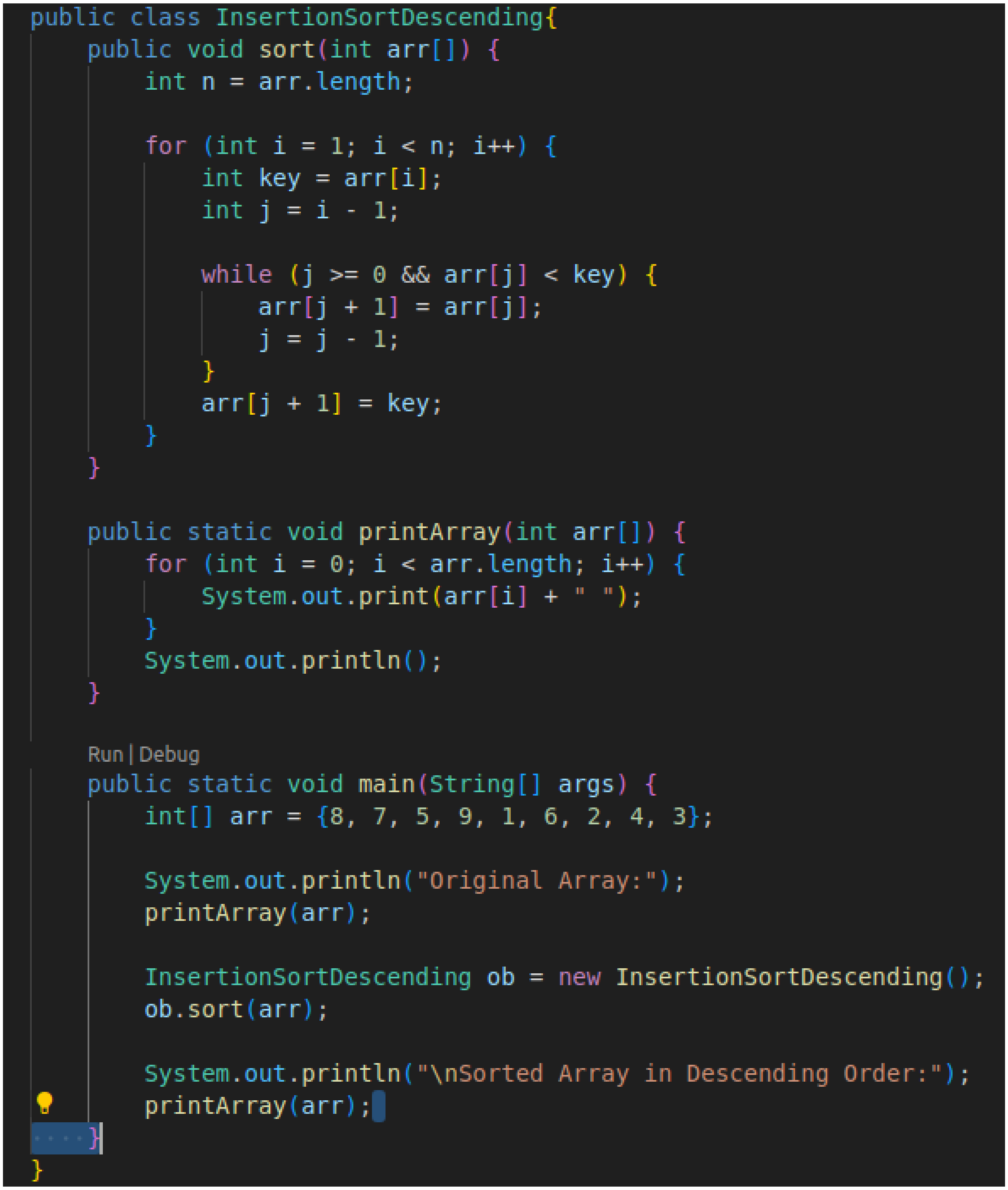


```java
public class SelectionSort {
    public static void selectionSort(int arr[]) {
        int n = arr.length;

        for (int i = 0; i < n - 1; i++) {
            int smallest = i;

            for (int j = i + 1; j < n; j++) {
                if (arr[j] < arr[smallest]) {
                    smallest = j;
                }
            }

            int temp = arr[smallest];
            arr[smallest] = arr[i];
            arr[i] = temp;
        }
    }

    public static void printArray(int arr[]) {
        for (int i = 0; i < arr.length; i++) {
            System.out.print(arr[i] + " ");
        }
        System.out.println();
    }

    public static void main(String[] args) {
        int[] arr = {26, 54, 93, 17, 77, 31, 44, 55, 20};

        System.out.println("Original Array:");
        printArray(arr);

        selectionSort(arr);

        System.out.println("\nSorted Array in Ascending Order:");
        printArray(arr);
    }
}
```

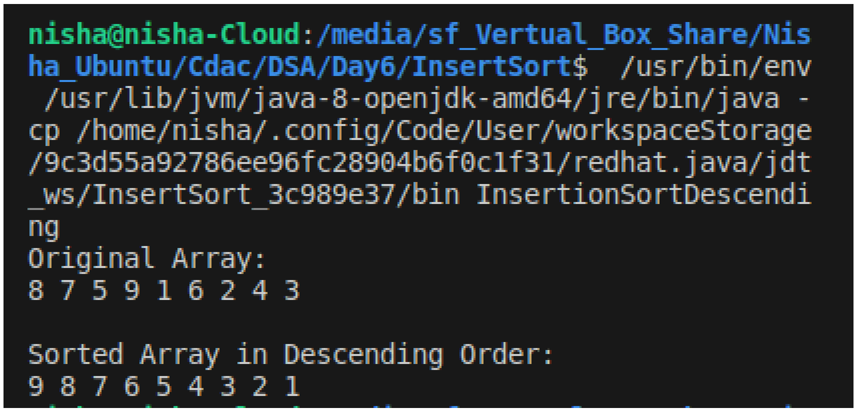


```
nisha@nisha-Cloud:/media/sf_Vertual_Box_Share/Nisha_Ubuntu/Cdac/DSA/Day6/Se
lectSort$  /usr/bin/env /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java -cp
/home/nisha/.config/Code/User/workspaceStorage/c7dc3020562a9ab4a5ec821f9c1f
8aaa/redhat.java/jdt_ws/SelectSort_a04e36fa/bin SelectionSort
Original Array:
26 54 93 17 77 31 44 55 20

Sorted Array in Ascending Order:
17 20 26 31 44 54 55 77 93
```

Qn2. Create an array & insert the elements like 8 7 5 9 1 6 2 4 3 sort the array in Descending order using Insertion sort and display the result.

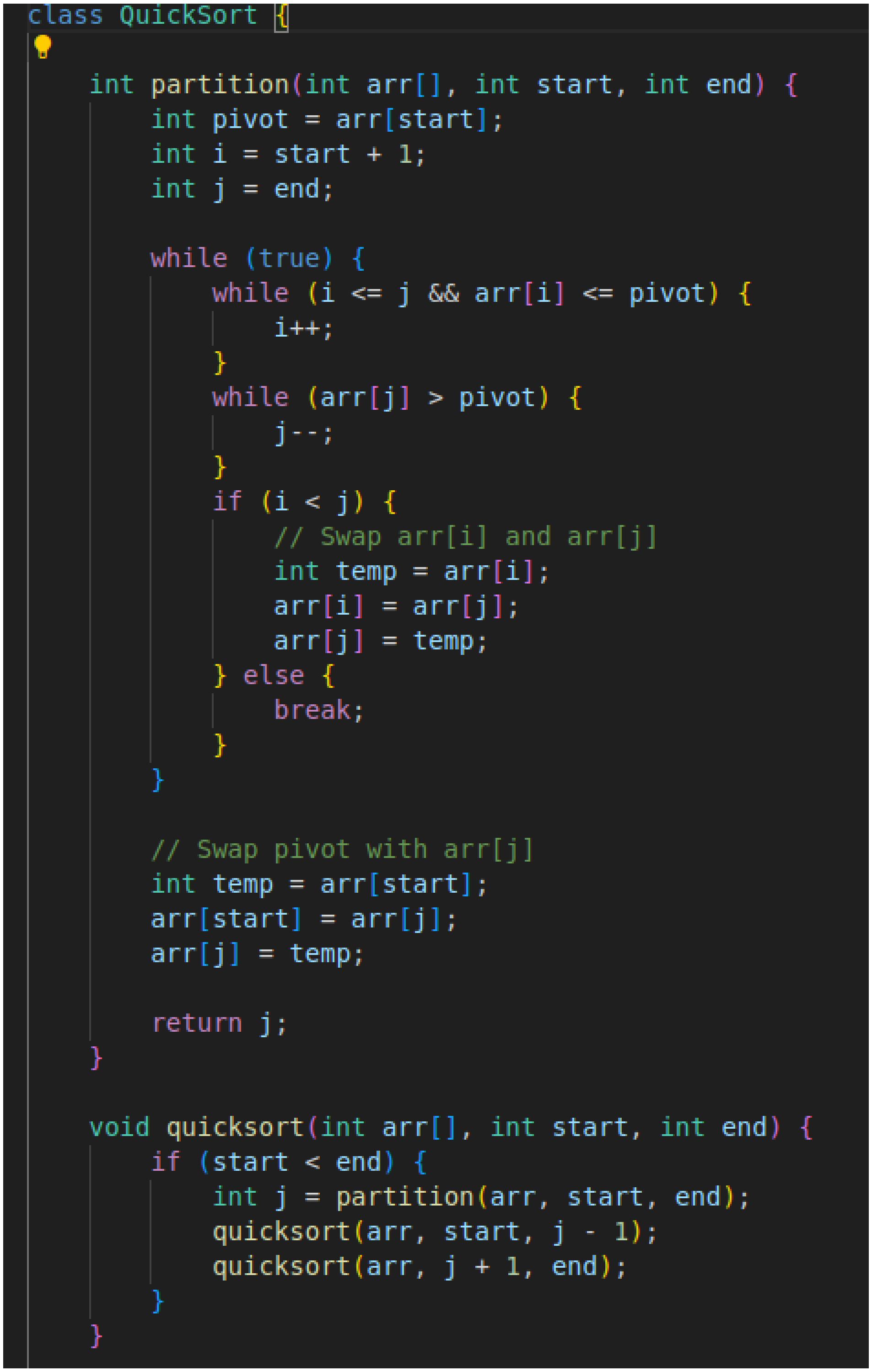


```java
public class InsertionSortDescending{
    public void sort(int arr[]) {
        int n = arr.length;

        for (int i = 1; i < n; i++) {
            int key = arr[i];
            int j = i - 1;

            while (j >= 0 && arr[j] < key) {
                arr[j + 1] = arr[j];
                j = j - 1;
            }
            arr[j + 1] = key;
        }
    }

    public static void printArray(int arr[]) {
        for (int i = 0; i < arr.length; i++) {
            System.out.print(arr[i] + " ");
        }
        System.out.println();
    }

    public static void main(String[] args) {
        int[] arr = {8, 7, 5, 9, 1, 6, 2, 4, 3};

        System.out.println("Original Array:");
        printArray(arr);

        InsertionSortDescending ob = new InsertionSortDescending();
        ob.sort(arr);

        System.out.println("\nSorted Array in Descending Order:");
        printArray(arr);
    }
}
```

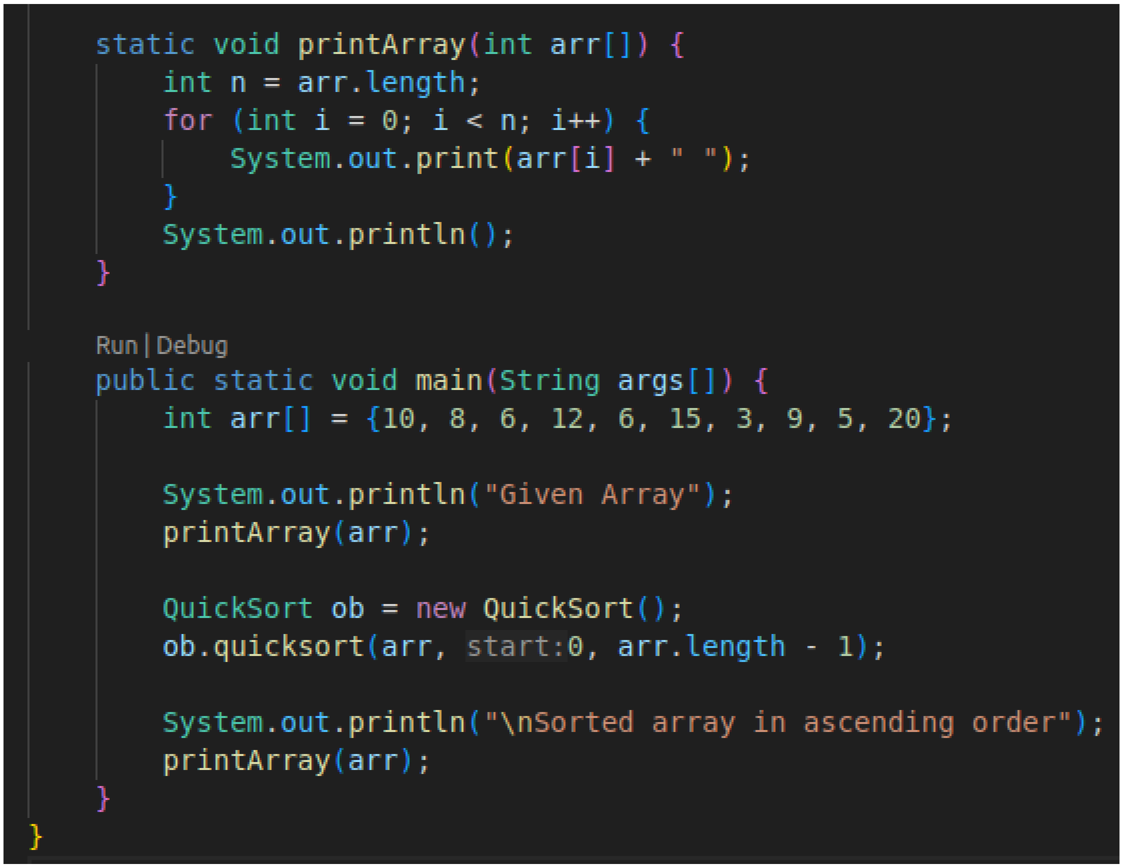


```
nisha@nisha-Cloud:/media/sf_Vertual_Box_Share/Nis
ha_Ubuntu/Cdac/DSA/Day6/InsertSort$  /usr/bin/env
 /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java -
cp /home/nisha/.config/Code/User/workspaceStorage
/9c3d55a92786ee96fc28904b6f0c1f31/redhat.java/jdt
_ws/InsertSort_3c989e37/bin InsertionSortDescendi
ng
Original Array:
8 7 5 9 1 6 2 4 3

Sorted Array in Descending Order:
9 8 7 6 5 4 3 2 1
```

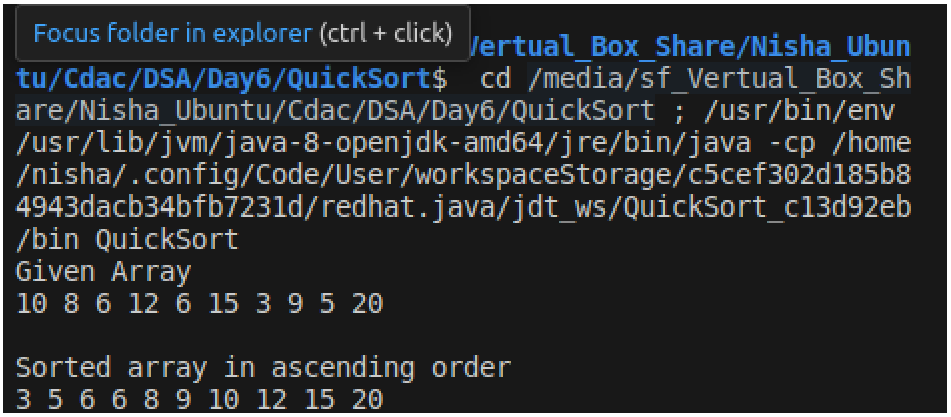
Qn3. Create an array & insert the elements like 10 , 8, 6 ,12,6,15,3 ,9 ,5 20 sort the array in Ascending order using Quick sort and display the result.



```
class QuickSort {

    int partition(int arr[], int start, int end) {
        int pivot = arr[start];
        int i = start + 1;
        int j = end;

        while (true) {
            while (i <= j && arr[i] <= pivot) {
                i++;
            }
            while (arr[j] > pivot) {
                j--;
            }
            if (i < j) {
                // Swap arr[i] and arr[j]
                int temp = arr[i];
                arr[i] = arr[j];
                arr[j] = temp;
            } else {
                break;
            }
        }

        // Swap pivot with arr[j]
        int temp = arr[start];
        arr[start] = arr[j];
        arr[j] = temp;

        return j;
    }

    void quicksort(int arr[], int start, int end) {
        if (start < end) {
            int j = partition(arr, start, end);
            quicksort(arr, start, j - 1);
            quicksort(arr, j + 1, end);
        }
    }
```

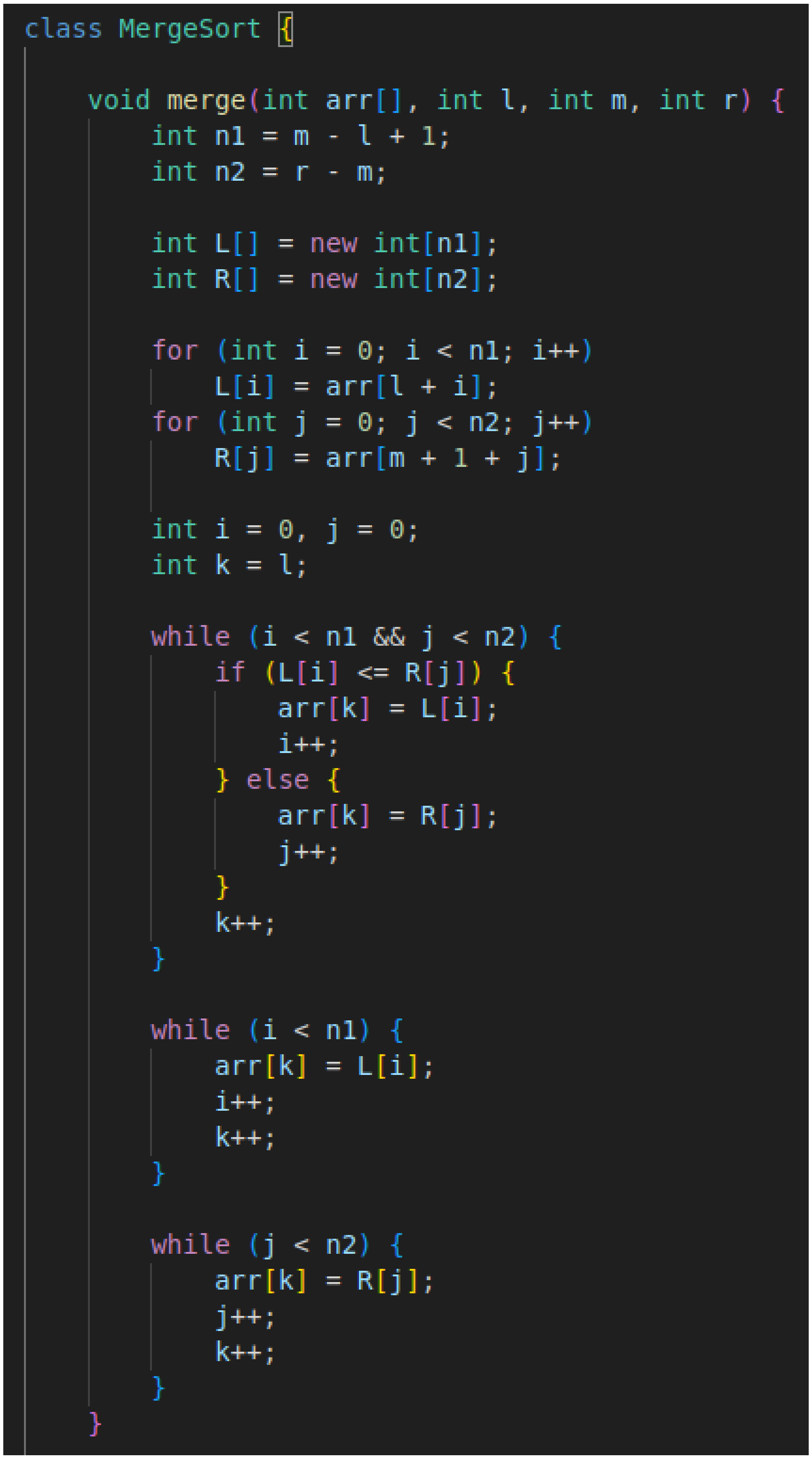


```
    static void printArray(int arr[]) {
        int n = arr.length;
        for (int i = 0; i < n; i++) {
            System.out.print(arr[i] + " ");
        }
        System.out.println();
    }

    public static void main(String args[]) {
        int arr[] = {10, 8, 6, 12, 6, 15, 3, 9, 5, 20};

        System.out.println("Given Array");
        printArray(arr);

        QuickSort ob = new QuickSort();
        ob.quicksort(arr, start:0, arr.length - 1);

        System.out.println("\nSorted array in ascending order");
        printArray(arr);
    }
}
```

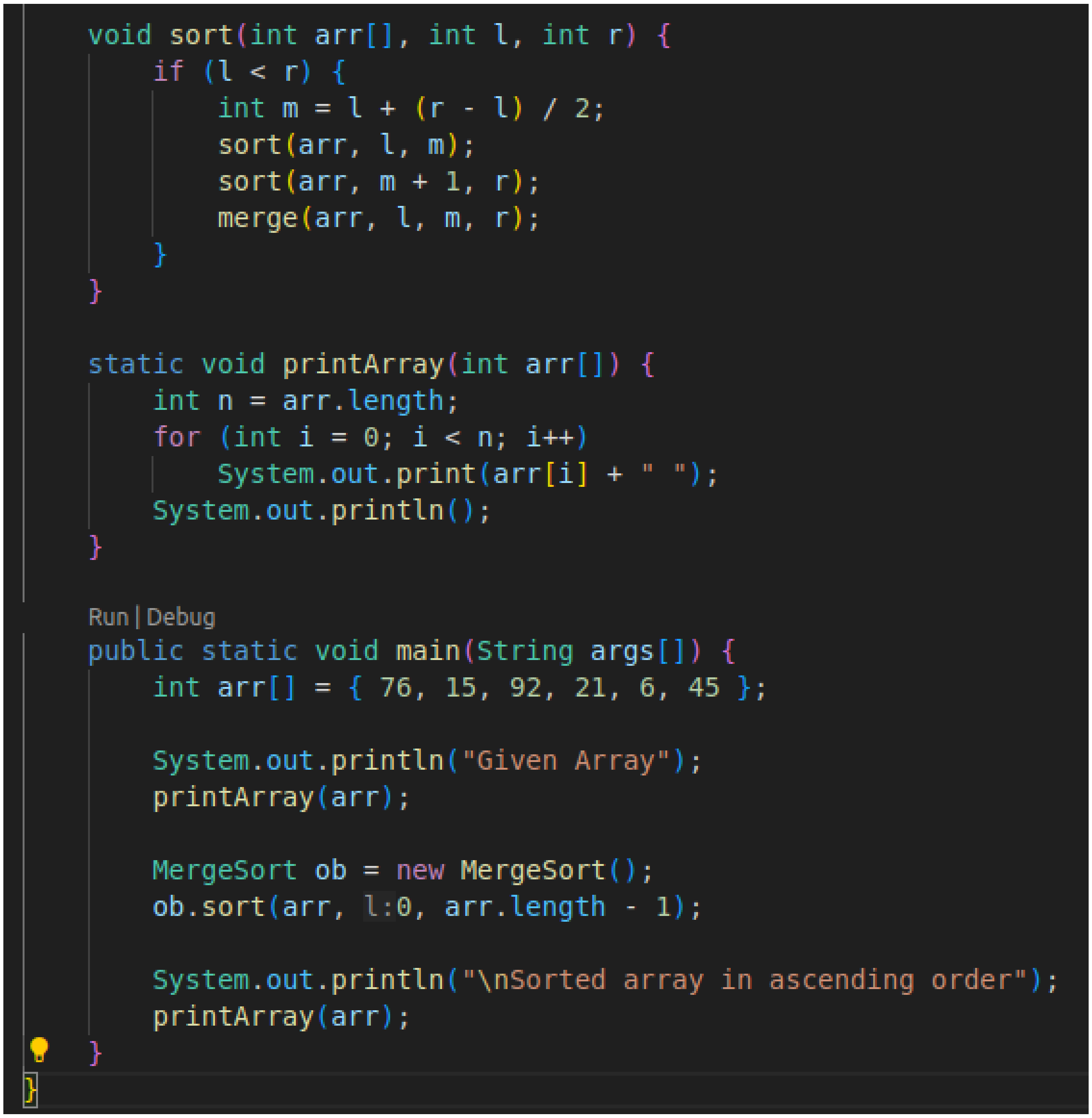


```
/ertual_Box_Share/Nisha_Ubun
tu/Cdac/DSA/Day6/QuickSort$  cd /media/sf_Vertual_Box_Sh
are/Nisha_Ubuntu/Cdac/DSA/Day6/QuickSort ; /usr/bin/env
/usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java -cp /home
/nisha/.config/Code/User/workspaceStorage/c5cef302d185b8
4943dacb34bfb7231d/redhat.java/jdt_ws/QuickSort_c13d92eb
/bin QuickSort
Given Array
10 8 6 12 6 15 3 9 5 20

Sorted array in ascending order
3 5 6 6 8 9 10 12 15 20
```

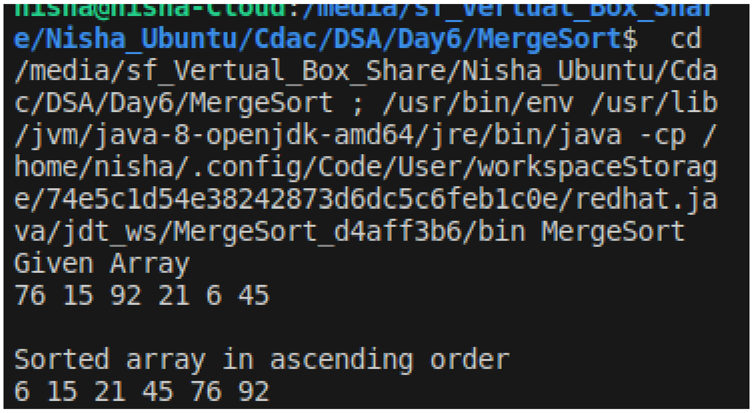
Qn4. Create an array & insert the elements like 76 ,15, 92, 21, 6, 45 sort the array in Ascending order using Merge sort and display the result.



```
class MergeSort {

    void merge(int arr[], int l, int m, int r) {
        int n1 = m - l + 1;
        int n2 = r - m;

        int L[] = new int[n1];
        int R[] = new int[n2];

        for (int i = 0; i < n1; i++)
            L[i] = arr[l + i];
        for (int j = 0; j < n2; j++)
            R[j] = arr[m + 1 + j];

        int i = 0, j = 0;
        int k = l;

        while (i < n1 && j < n2) {
            if (L[i] <= R[j]) {
                arr[k] = L[i];
                i++;
            } else {
                arr[k] = R[j];
                j++;
            }
            k++;
        }

        while (i < n1) {
            arr[k] = L[i];
            i++;
            k++;
        }

        while (j < n2) {
            arr[k] = R[j];
            j++;
            k++;
        }
    }
```



```java
    void sort(int arr[], int l, int r) {
        if (l < r) {
            int m = l + (r - l) / 2;
            sort(arr, l, m);
            sort(arr, m + 1, r);
            merge(arr, l, m, r);
        }
    }

    static void printArray(int arr[]) {
        int n = arr.length;
        for (int i = 0; i < n; i++)
            System.out.print(arr[i] + " ");
        System.out.println();
    }

    Run | Debug
    public static void main(String args[]) {
        int arr[] = { 76, 15, 92, 21, 6, 45 };

        System.out.println("Given Array");
        printArray(arr);

        MergeSort ob = new MergeSort();
        ob.sort(arr, l:0, arr.length - 1);

        System.out.println("\nSorted array in ascending order");
        printArray(arr);
    }
}
```



```
nisha@nisha-Cloud:/media/sf_Vertual_Box_Shar
e/Nisha_Ubuntu/Cdac/DSA/Day6/MergeSort$  cd
/media/sf_Vertual_Box_Share/Nisha_Ubuntu/Cda
c/DSA/Day6/MergeSort ; /usr/bin/env /usr/lib
/jvm/java-8-openjdk-amd64/jre/bin/java -cp /
home/nisha/.config/Code/User/workspaceStorag
e/74e5c1d54e38242873d6dc5c6feb1c0e/redhat.ja
va/jdt_ws/MergeSort_d4aff3b6/bin MergeSort
Given Array
76 15 92 21 6 45

Sorted array in ascending order
6 15 21 45 76 92
```