```
package datastructure.Sort;


public class QuickSort {
    int partition(int[] arr, int low, int high) {
        int pivot = arr[high];
        int i = (low - 1);
        for (int j = low; j < high; j++) {
            if (arr[j] <= pivot) {
                i++;
                int temp = arr[i];
                arr[i] = arr[j];
                arr[j] = temp;
            }
        }
        int temp = arr[i + 1];
        arr[i + 1] = arr[high];
        arr[high] = temp;
        return i + 1;
    }
    //  0   1   2   3
    //  5,  2,  1,  3, -->  2,  1,  3, 5


    void sort(int[] arr, int low, int high) {
        if (low < high) {
            int partitionIndex = partition(arr, low, high);
            sort(arr, low, partitionIndex - 1);
            sort(arr, partitionIndex + 1, high);
        }
```

```
    }

    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[] = { 5, 2, 1, 3 };
        QuickSort ob = new QuickSort();
        ob.sort(arr, 0, arr.length - 1);

        System.out.println("sorted array");
        printArray(arr);
    }

}
```