1. Implement an ArrayDequeue and all of its methods such as add(), addFirst(), addLast(), element(), poll(), push(), remove.

Solution

*package sba3;*

*import java.util.ArrayDeque;*

*public class Q1 {*

*public static void main(String[] args) {*

*ArrayDeque<String> animals = new ArrayDeque<>();*

*// Using add()*

*animals.add("Dog");*

*// Using addFirst()*

*animals.addFirst("Cat");*

*// Using addLast()*

*animals.addLast("Horse");*

*System.out.println("ArrayDeque: " + animals);*

*// Using poll()*

*String element = animals.poll();*

*System.out.println("Removed Element: " + element);*

*System.out.println("New ArrayDeque: " + animals);*

*// Using pollFirst()*

*String firstElement = animals.pollFirst();*

*System.out.println("Removed First Element: " + firstElement);*

*// Using pollLast()*

*String lastElement = animals.pollLast();*

*System.out.println("Removed Last Element: " + lastElement);*

*// using push()*

*animals.push("Rabbit");*

*animals.push("cow");*

*animals.push("goat");*

*System.out.println("After push method ArrayDeque: " + animals);*

*// using element()--returns element present in the head*

*System.out.println("Head element by element() method: " + animals.element());*

*// Using remove()*

*String element1 = animals.remove();*

*System.out.println("Removed Element: " + element1);*

*System.out.println("New ArrayDeque: " + animals);*

*// Using removeFirst()*

*String firstElement1 = animals.removeFirst();*

*System.out.println("Removed First Element: " + firstElement1);*

*// Using removeLast()*

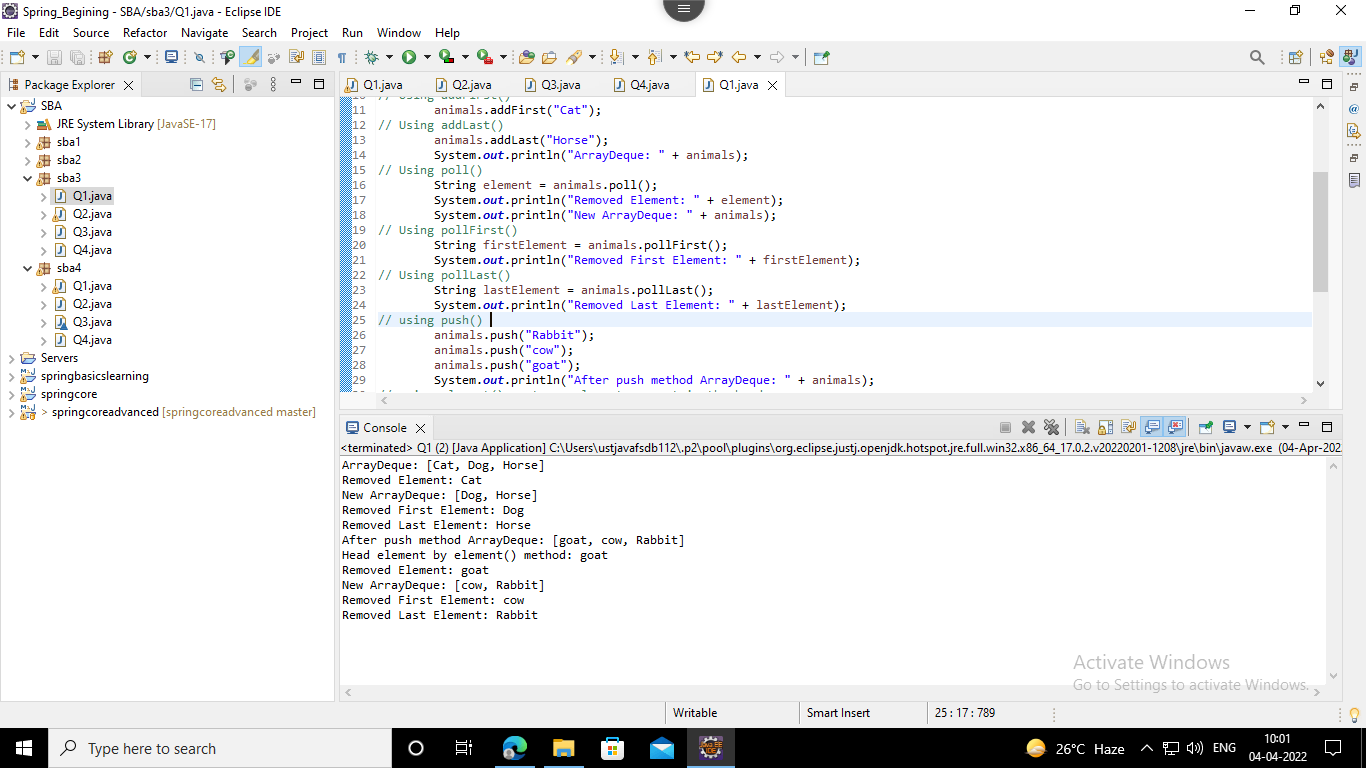
*String lastElement1 = animals.removeLast();*

*System.out.println("Removed Last Element: " + lastElement1);*

*}*

*}*

Output



2. Implement a PriorityQueue and use all the methods.

Solution

*package sba3;*

*import java.util.Iterator;*

*import java.util.PriorityQueue;*

*public class Q2 {*

*public static void main(String args[]) {*

*PriorityQueue<String> pq = new PriorityQueue<>();*

*pq.add("Ajay");*

*pq.add("Vijay");*

*pq.add("Raj");*

*pq.add("Jagan");*

*System.out.println("head:" + pq.element());*

*System.out.println("head:" + pq.peek());*

*System.out.println("iterating the queue elements:");*

*Iterator itr = pq.iterator();*

*while (itr.hasNext()) {*

*System.out.println(itr.next());*

*}*

*pq.remove();// removes the head element*

*pq.poll(); // removes the head*

*System.out.println("after removing two elements:");*

*Iterator<String> itr2 = pq.iterator();*

*while (itr2.hasNext()) {*

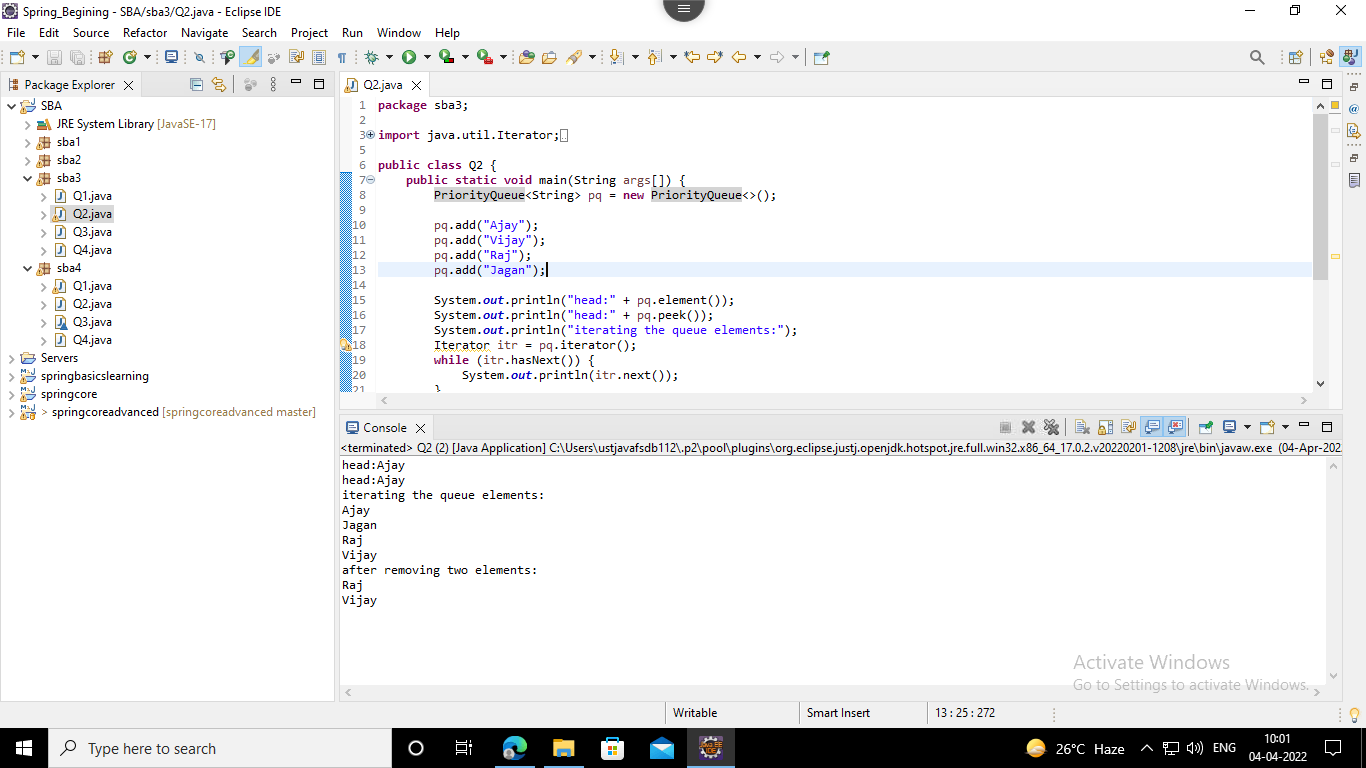
*System.out.println(itr2.next());*

*}*

*}*

*}*

Output

**

3. Implement a Stack and all of its methods peek(), push(), pop(), and to determine the size of the stack.

Solution

*package sba3;*

*import java.util.Stack;*

*public class Q3 {*

*public static void main(String[] args) {*

*// Creating an empty Stack*

*Stack<Integer> stk = new Stack<Integer>();*

*// Use add() method to add elements stk.push(10);*

*stk.push(15);*

*stk.push(30);*

*stk.push(20);*

*stk.push(5);*

*// Displaying the Stack*

*System.out.println("Initial Stack: " + stk);*

*// Removing elements using pop() method*

*System.out.println("Popped element: " + stk.pop());*

*System.out.println("Popped element: " + stk.pop());*

*// Displaying the Stack after pop operation*

*System.out.println("Stack after pop operation " + stk);*

*// Fetching the element at the head of the Stack*

*System.out.println("The element at the top of the" + " stack is: " + stk.peek());*

*// Displaying the Stack after the Operation*

*System.out.println("Final Stack: " + stk);*

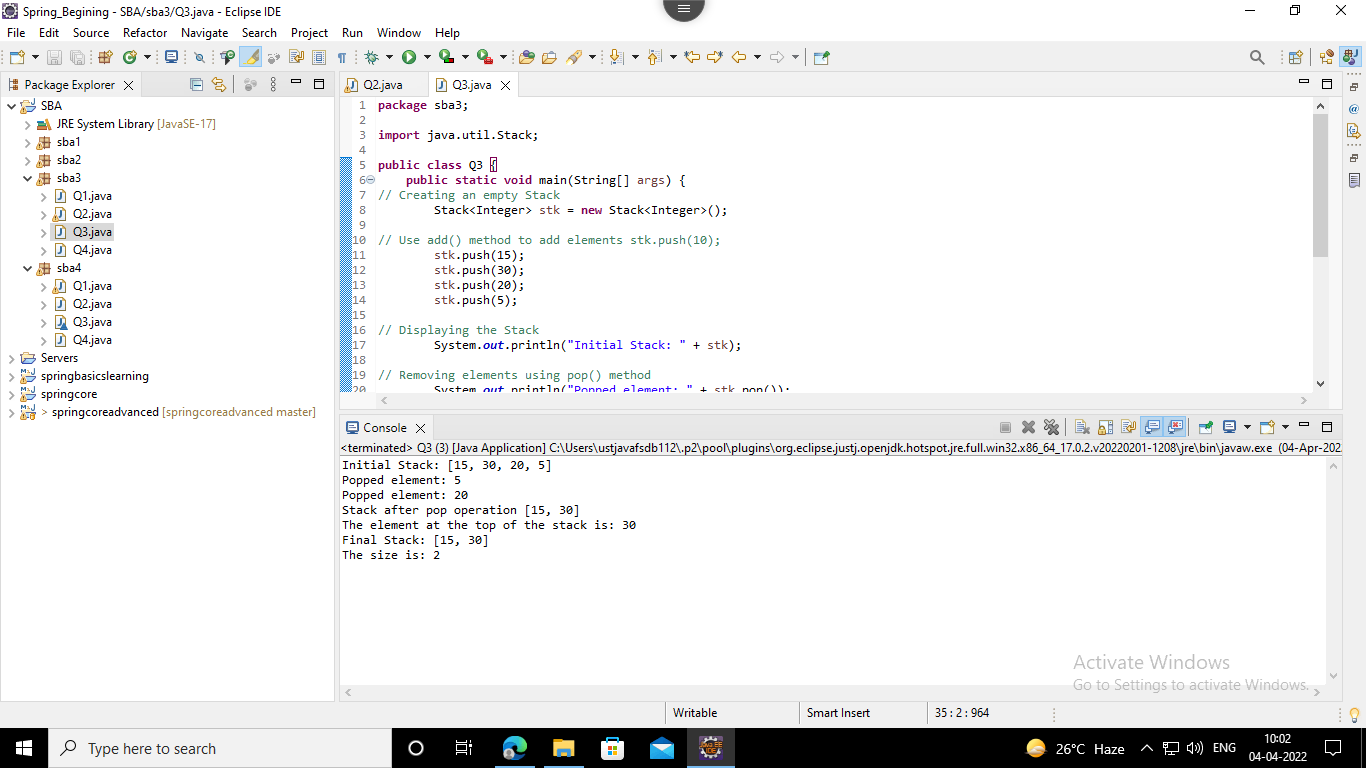
*// Displaying the size of stack*

*System.out.println("The size is: " + stk.size());*

*}*

*}*

Output



4. Write a program to implement insertion sort.

Solution

*package sba3;*

*public class Q4 {*

*/\* Function to sort array using insertion sort \*/*

*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;*

*}*

*}*

*/\* A utility function to print array of size n \*/*

*static void printArray(int arr[]) {*

*int n = arr.length;*

*for (int i = 0; i < n; ++i)*

*System.out.print(arr[i] + " ");*

*System.out.println();*

*}*

*// Driver method*

*public static void main(String args[]) {*

*int arr[] = { 12, 11, 13, 5, 6 };*

*Q4 ob = new Q4();*

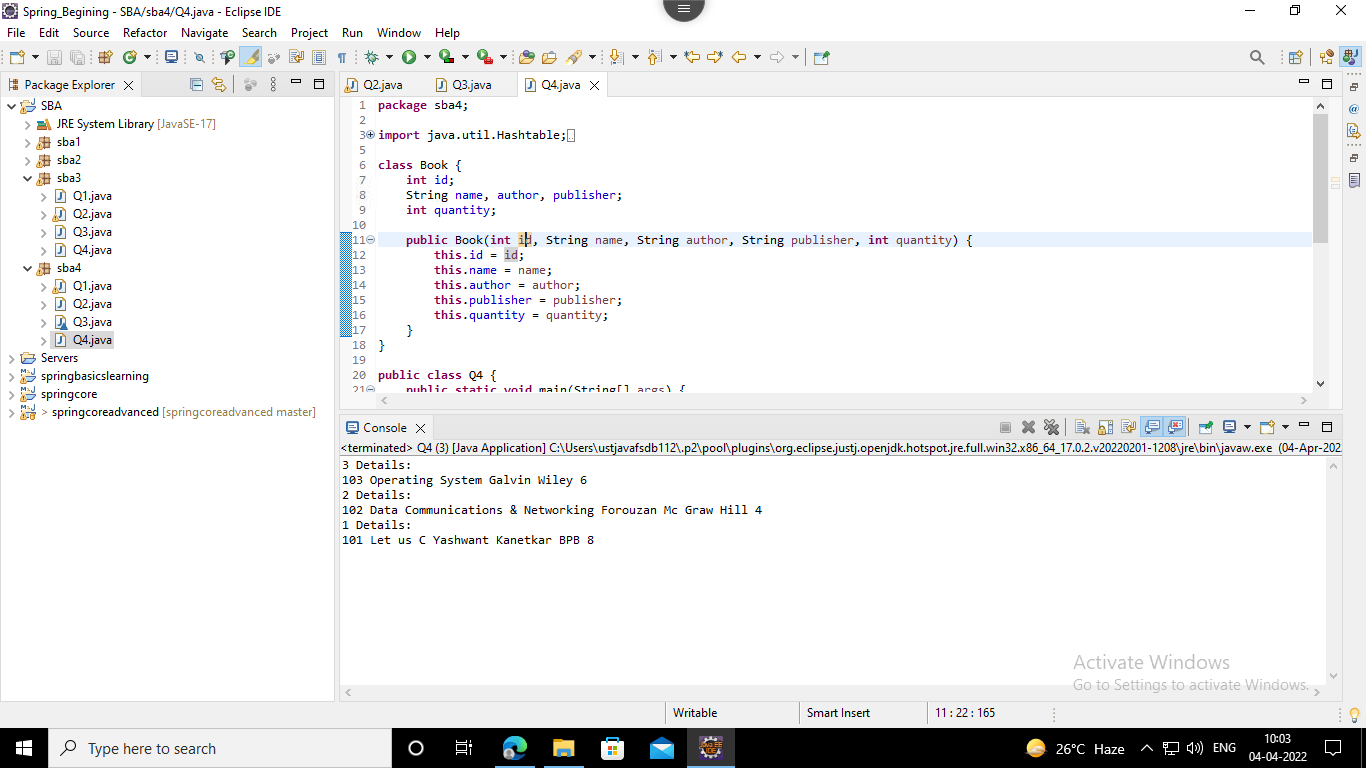
*ob.sort(arr);*

*printArray(arr);*

*}*

*}*

Output

**