**05/08/2024 JAVA-CSA0983**

**1.Implement the program by using map.**

**Program:**

**import java.util.HashMap;**

**public class map{**

**static void findelement(HashMap<String,String>Bowl,String fruit){**

**if(Bowl.containsKey(fruit))**

**System.out.println("The "+fruit+" is "+Bowl.get(fruit));**

**else**

**System.out.println("not found");**

**}**

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

**HashMap<String,String> Bowl=new HashMap<String,String>();**

**Bowl.put("Apple","Red");**

**Bowl.put("Berry","Blue");**

**Bowl.put("Orange","Orange");**

**Bowl.put("Banana","Yellow");**

**System.out.println(Bowl);**

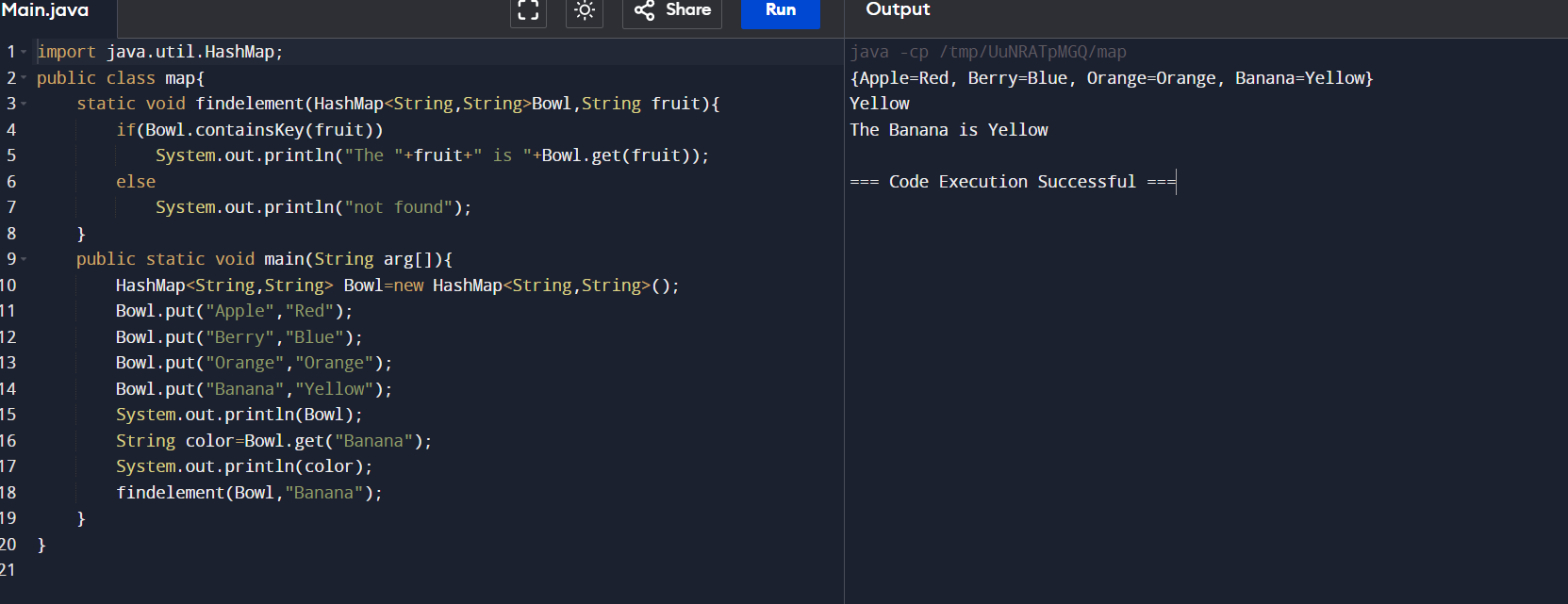
**String color=Bowl.get("Banana");**

**System.out.println(color);**

**findelement(Bowl,"Banana");**

**}**

**}Output:**

****

**2.Create program using linkedlist in queue.**

**Program:**

**import java.util.LinkedList;**

**public class QueueExample {**

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

**LinkedList<String> queue = new LinkedList<>();**

**queue.add("ABI");**

**queue.add("AISHU");**

**queue.add("AKSHU");**

**System.out.println("Queue: " + queue);**

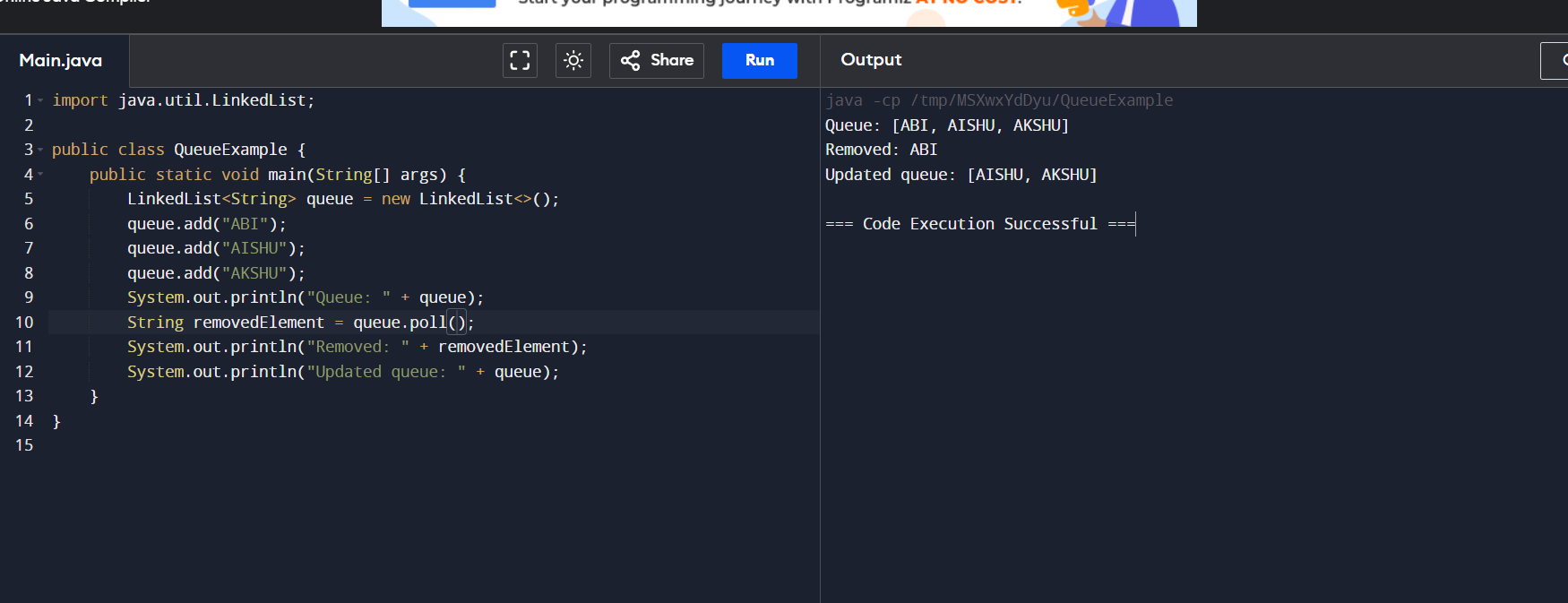
**String removedElement = queue.poll();**

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

**System.out.println("Updated queue: " + queue);**

**}**

**}**

**Output:** ****

**3.Create program using linkedlist in stack.**

**Program:**

**import java.util.LinkedList;**

**public class Stack{**

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

**LinkedList<String> stack = new LinkedList<>();**

**stack.push("PIG");**

**stack.push("DONKEY");**

**stack.push("MONKEY");**

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

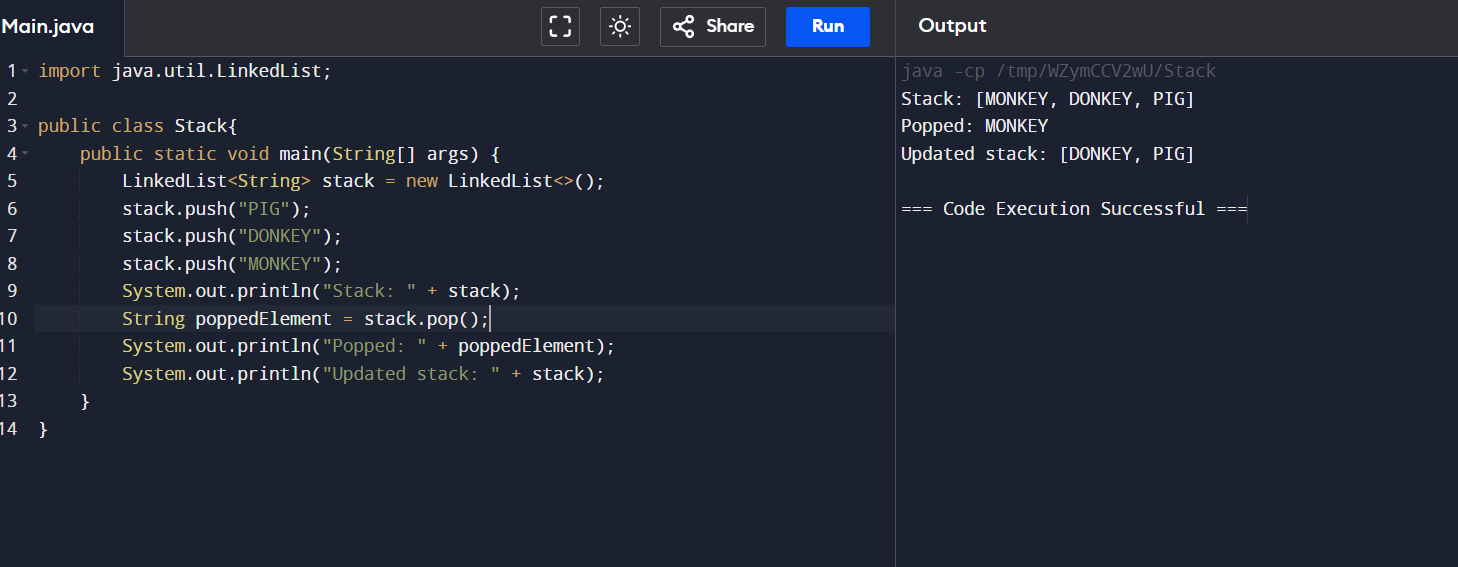
**String poppedElement = stack.pop();**

**System.out.println("Popped: " + poppedElement);**

**System.out.println("Updated stack: " + stack);**

**}**

**}**

**Output:** ****

**4. Create mobile class with field price brand and so on take any one field and sort the class.**

**Program:**

**import java.util.Arrays;**

**import java.util.Comparator;**

**class Mobile {**

**private double price;**

**private String brand;**

**private String model;**

**private int ram;**

**public Mobile(double price, String brand, String model, int ram) {**

**this.price = price;**

**this.brand = brand;**

**this.model = model;**

**this.ram = ram;**

**}**

**public double getPrice() {**

**return price;**

**}**

**public String getBrand() {**

**return brand;**

**}**

**public String getModel() {**

**return model;**

**}**

**public int getRam() {**

**return ram;**

**}**

**public String toString() {**

**return "Mobile{" +**

**"price=" + price +**

**", brand='" + brand + '\'' +**

**", model='" + model + '\'' +**

**", ram=" + ram +**

**'}';**

**}**

**}**

**public class MobileSorter {**

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

**Mobile[] mobiles = new Mobile[] {**

**new Mobile(15000, "Samsung", "Galaxy M31", 6),**

**new Mobile(20000, "Apple", "iPhone 12", 4),**

**new Mobile(10000, "Xiaomi", "Redmi Note 9", 4),**

**new Mobile(25000, "OnePlus", "Nord", 8),**

**new Mobile(12000, "Realme", "6 Pro", 6)**

**};**

**System.out.println("Before sorting:");**

**Arrays.stream(mobiles).forEach(System.out::println);**

**Arrays.sort(mobiles, Comparator.comparingDouble(Mobile::getPrice));**

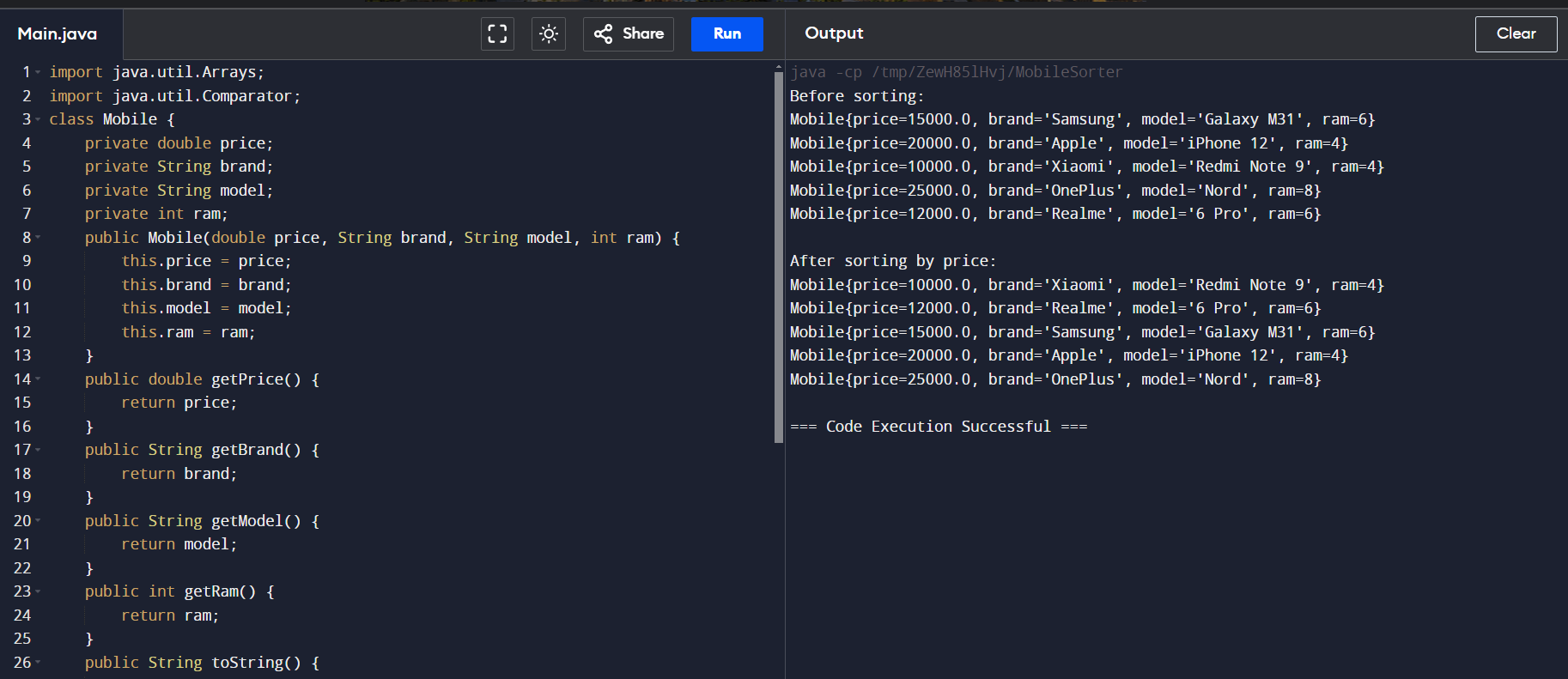
**System.out.println("\nAfter sorting by price:");**

**Arrays.stream(mobiles).forEach(System.out::println);**

**}**

**}**

**Output:**

****