Java Assignment 2

Sahil Goyal

AIML-B1

22070126094

2022-26

Q1 Write a Java program that declares two arrays named 'even' and 'odd'. Accept numbers from the

user and move them to respective arrays depending on whether they are even or odd.

Code:

// UserInput.java

import java.io.BufferedReader;

import java.io.IOException;

import java.io.InputStreamReader;

public class UserInput {

// Method to get an array input from the user

public static double[] inputArray(int size) throws IOException {

// Create a BufferedReader to read user input

BufferedReader br = new BufferedReader(new InputStreamReader(System.in));

// Prompt the user to enter the array elements

System.out.println("Enter the array elements separated by spaces: ");

// Read the array input as a string

String array = br.readLine();

// Initialize an array to store the input elements

double[] arrayInput = new double[size];

// Split the input string and convert each element to double

String[] input = array.trim().split("\\s+");

// Populate the array with the converted elements

for (int i = 0; i < size; i++) {

arrayInput[i] = Double.parseDouble(input[i]);

}

// Return the array containing user-input elements

return arrayInput;

}

}

import java.io.IOException;

import java.util.Arrays;

import java.util.Scanner;

public class OddEven {

    public static void main(String[] args) throws IOException {

        // Create a Scanner object to read user input

        Scanner sc = new Scanner(System.in);

        // Declare arrays for even and odd numbers

        double[] even;

        int j = 0;

        int k = 0;

        double[] odd;

        // Prompt the user to enter the number of elements

        System.out.println("Enter the number of elements to enter:");

        // Read the size from user input

        int size = sc.nextInt();

        // Initialize arrays for even and odd numbers based on the user-defined size

        even = new double[size];

        odd = new double[size];

        // Get the array input from the user using the UserInput class

        double[] array = UserInput.inputArray(size);

        // Close the Scanner to avoid resource leaks

        sc.close();

        // Separate even and odd numbers into their respective arrays

        for (int i = 0; i < size; i++) {

            if (array[i] % 2 == 0) {

                even[j] = array[i];

                j++;

            } else {

                odd[k] = array[i];

                k++;

            }

        }

        System.out.println("Even numbers:");

        System.out.print(Arrays.toString(even));

        System.out.println("Odd numbers:");

        System.out.print(Arrays.toString(odd));

    }

       // Method to print elements of an array

       public static void print(double[] array, int size) {

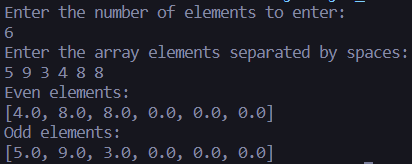
        for (int i = 0; i < size; i++) {

            System.out.println(array[i]);

        }

}

Output:



Q2 Implement a Java function that finds two neighbouring numbers in an array with the smallest distance to each. The function should return the index of the 1st number.

Code:

// Neighbours.java

//Sahil Goyal

//AIML-B1

//2022-26

//22070126094

package Assignment\_2;

import java.io.IOException;

import java.util.Scanner;

public class Neighbours {

public static void main(String[] args) throws IOException {

// Create a Scanner object to read user input

Scanner sc = new Scanner(System.in);

// Prompt the user to enter the size of the array

System.out.println("Enter the size of the array: ");

// Read the size of the array from user input

int size = sc.nextInt();

// Get the array input from the user using the UserInput class

double[] array = UserInput.inputArray(size);

sc.close();

// Find and print the index of the nearest neighbours in the array

System.out.println("Index of Nearest Neighbours: " + findNearestNeighbours(array));

}

// Method to find the index of nearest neighbours in the array

public static int findNearestNeighbours(double[] arr) {

double minDistance = Double.MAX\_VALUE;

int index = -1;

// Iterate through the array and calculate distances between adjacent elements

for (int i = 0; i < arr.length - 1; i++) {

double distance = Math.abs(arr[i] - arr[i + 1]);

// Update the index if the current distance is smaller than the minimum distance

if (distance < minDistance) {

minDistance = distance;

index = i;

}

}

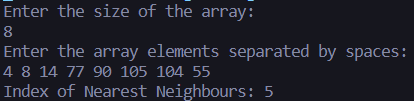
// Return the index of the nearest neighbours

return index;

}

}

Output:



Q3 Write a Java program to convert an array into ArrayList and vice versa.

Code:

// ArrayAndArrayList.java

//Sahil Goyal

//AIML-B1

//2022-26

//22070126094

package Assignment\_2;

import java.io.IOException;

import java.util.\*;

public class ArrayAndArrayList {

public static void main(String[] args) throws IOException {

// Create a Scanner object to read user input

Scanner sc = new Scanner(System.in);

// Prompt the user to enter the size of the array

System.out.println("Enter the size of array: ");

// Read the size from user input

int size = sc.nextInt();

// Get the primitive double array from user input using UserInput class

double[] array = UserInput.inputArray(size);

// Close the Scanner to avoid resource leaks

sc.close();

// Print the primitive double array using Arrays.toString

System.out.println(Arrays.toString(array));

// Convert the primitive double array to an ArrayList<Double>

System.out.println("Primitive array to ArrayList: ");

List<Double> arrlist = Arrays.asList(Arrays.stream(array).boxed().toArray(Double[]::new));

// Print the elements of the ArrayList using a for-each loop

for (Double element : arrlist) {

System.out.println(element);

}

// Convert ArrayList to array

System.out.println("ArrayList to array: ");

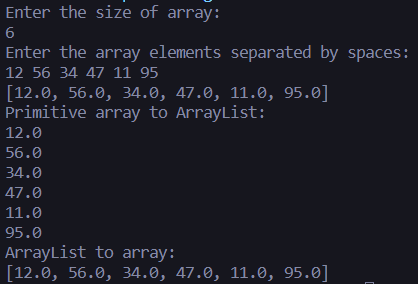
double[] arr = arrlist.stream().mapToDouble(Double::doubleValue).toArray();

System.out.println(Arrays.toString(arr));

}

}

Output:



Check my repo for all the assignments organized:

<https://github.com/sahilgoyal7214/programming-in-java/tree/main/Assignment_2>