

**CSC248**

**FUNDAMENTALS OF DATA  
STRUCTURE**

**LAB ASSIGNMENT 2**

NAME : MUHAMMAD REDZA BIN MAHAYADIN

STUDENT ID : 2022676696

GROUP : RCDCS1103B

LECTURER : SIR MOHD NIZAM BIN OSMAN

Question 1

import java.util.ArrayList;

import java.util.Scanner;

public class Q1 {

    public static void main(String[] args) {

        Scanner in = new Scanner(System.in);

        Scanner in1 = new Scanner(System.in);

        ArrayList<Integer> numbers = new ArrayList<Integer>();

        numbers.add(1);

        numbers.add(2);

        numbers.add(3);

        System.out.println("The elements are: " + numbers);

        System.out.println();

        System.out.print("1. Add new element into the list\n2. Delete element from the list\n3. The number of elements in the list\n4. Calculate the sum of all elements in the list\n6. Exit\n\nEnter your choice: ");

        int choice = in1.nextInt();

        while (choice != 6) {

            switch (choice) {

                case 1:

                    System.out.print("Enter the number of elements to be added: ");

                    int num = in1.nextInt();

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

                        System.out.print("Enter the number: ");

                        int number = in1.nextInt();

                        numbers.add(number);

                    }

                    System.out.println("The elements are: " + numbers);

                    break;

                case 2:

                    System.out.print("Enter the index of the element to be deleted: ");

                    int index = in1.nextInt();

                    try {

                        numbers.get(index);

                    } catch (IndexOutOfBoundsException e) {

                        System.out.println("Invalid index!");

                        break;

                    }

                    System.out.println("Before removing element at index " + index + ": " + numbers);

                    numbers.remove(index);

                    System.out.println("After removing element at index " + index + ": " + numbers);

                    break;

                case 3:

                    System.out.println("The number of elements in the list: " + numbers.size());

                    break;

                case 4:

                    int sum = 0;

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

                        sum += numbers.get(i);

                    }

                    System.out.println("The sum of all elements in the list: " + sum);

                    break;

                default:

                    System.out.println("Invalid choice!");

                    break;

            }

            System.out.print("\n1. Add new elements into the ArrayList\n2. Delete element from the ArrayList\n3. The number of elements in the list\n4. Calculate the sum of all elements in the list\n6. Exit\n\nEnter your choice: ");

            choice = in1.nextInt();

        }

        System.out.println("Thank you for using this program!");

        in.close();

        in1.close();

    }

}

Question 2

import java.util.Scanner;

import java.util.ArrayList;

public class Q2 {

    public static int choice;

    static Scanner in = new Scanner(System.in);

    static Scanner in1 = new Scanner(System.in);

    public static void main(String[] args) {

        ArrayList<String> names = new ArrayList<String>();

        names.add("Adam");

        names.add("Haikal");

        names.add("Lukman");

        names.add("Faris");

        names.add("Mail");

        System.out.println();

        System.out.println("The list: " + names);

        menuSelection();

        int ind;

        String ele;

        while (choice != 7) {

            switch (choice) {

                case 1:

                    System.out.print("Enter new name to be added: ");

                    ele = in.nextLine();

                    System.out.print("Enter index of the new name: ");

                    ind = in1.nextInt();

                    try {

                        names.add(ind, ele);

                    } catch (IndexOutOfBoundsException e) {

                        names.add(ele);

                        ind = names.indexOf(ele);

                        System.out.println(

                                "\nDue to index out of bounds, the new name was added at the end of the list instead, which is at index "

                                        + ind);

                    }

                    System.out.println();

                    System.out.println("The updated list: " + names);

                    break;

                case 2:

                    System.out.print("Enter index of the name to be deleted: ");

                    ind = in1.nextInt();

                    names.remove(ind);

                    System.out.println();

                    System.out.println("The updated list: " + names);

                    break;

                case 3:

                    System.out.println();

                    System.out.println("Size of the list: " + names.size());

                    break;

                case 4:

                    System.out.println();

                    System.out.print("Enter the name to find: ");

                    ele = in.nextLine();

                    if (names.contains(ele)) {

                        System.out.println("Name is found at index " + names.indexOf(ele));

                    } else

                        System.out.println("Name not found in the list.");

                    break;

                case 5:

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

                        for (int j = i + 1; j < names.size(); j++) {

                            if (names.get(i).compareTo(names.get(j)) > 0) {

                                String temp = names.get(i);

                                names.set(i, names.get(j));

                                names.set(j, temp);

                            }

                        }

                    }

                    System.out.println("The updated list: " + names);

                    break;

                case 6:

                    System.out.println("The list: " + names);

                    break;

                case 7:

                    System.out.println("Thank you for using this program!");

                    return;

                default:

                    System.out.println("Invalid choice!");

                    break;

            }

            // System pause

            System.out.print("\nPress enter to continue...");

            in.nextLine();

            menuSelection();

        }

        in.close();

        in1.close();

    }

    public static void menuSelection() {

        System.out.print(

                "\n\n\tMenu Selection\n1. Add new element into the list\n2. Delete element from the list\n3. The number of elements in the list\n4. Find if a name exists in the list\n5. Sort the list of names in ascending order\n6. Print all the names in the list\n7. Exit\n\nEnter your choice: ");

        choice = in1.nextInt();

    }

}

Question 3

Product Class

public class Product{

    private String productName;

    private double price;

    private int quantity;

    public Product(String productName, double price, int quantity){

        this.productName = productName;

        this.price = price;

        this.quantity = quantity;

    }

    public String getProductName() {

        return this.productName;

    }

    public void setProductName(String productName) {

        this.productName = productName;

    }

    public double getPrice() {

        return this.price;

    }

    public void setPrice(double price) {

        this.price = price;

    }

    public int getQuantity() {

        return this.quantity;

    }

    public void setQuantity(int quantity) {

        this.quantity = quantity;

    }

    public String toString(){

        return "Product Name: " + this.productName + "\nPrice: " + this.price + "\nQuantity: " + this.quantity + "\n";

    }

}

Main Class

import java.util.\*;

public class Q3 {

    public static void main(String[] args) {

        Scanner in = new Scanner(System.in);

        ArrayList<Product> listProduct1 = new ArrayList<Product>();

        ArrayList<Product> listProduct2 = new ArrayList<Product>();

        listProduct1.add(new Product("Apple", 2.99, 10));

        listProduct1.add(new Product("Tesla Model S", 99123.99, 5));

        listProduct1.add(new Product("Banana", 0.99, 30));

        listProduct1.add(new Product("Grape", 3.99, 40));

        listProduct1.add(new Product("Boeing 747", 999999.99, 1));

        listProduct1.add(new Product("Pineapple", 5.99, 60));

        listProduct1.add(new Product("Pen", 6.99, 70));

        listProduct1.add(new Product("MSI GeForce RTX 4070ti Gaming X Trio", 829.99, 10));

        listProduct1.add(new Product("Blueberry", 8.99, 90));

        listProduct1.add(new Product("Cherry", 9.99, 100));

        // Find and display the record based on productName.

        System.out.print("Enter a product name to search: ");

        String search = in.nextLine();

        boolean found = false;

        for (Product l1 : listProduct1) {

            if (l1.getProductName().equalsIgnoreCase(search)) {

                l1.toString();

                found = true;

                System.out.println("Product found at index " + listProduct1.indexOf(l1));

            }

        }

        if (!found) {

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

        }

        // Update the record where the productName is equal to Pen.

        boolean found2 = false;

        for (Product l1 : listProduct1) {

            if (l1.getProductName().equalsIgnoreCase("Pen")) {

                l1.setPrice(1.00);

                l1.setQuantity(30);

                found2 = true;

                System.out.println("\nProduct updated at index " + listProduct1.indexOf(l1));

            }

        }

        if (!found2) {

            System.out.println("\nProduct not found.");

        }

        // Remove  all  records  for  total  price  is  more  than  RM  1000  and  store  them  into listProduct2.

        for (int i = 0; i < listProduct1.size(); i++) {

            double totalPrice = listProduct1.get(i).getPrice() \* listProduct1.get(i).getQuantity();

            if (totalPrice > 1000) {

                listProduct2.add(listProduct1.get(i));

                listProduct1.remove(i);

                i--;

            }

        }

        // Display the records in listProduct1 and listProduct2.

        System.out.println("\n\tList 1:");

        for (Product l1 : listProduct1) {

            System.out.println(l1.toString());

        }

        System.out.println("\n\tList 2:");

        for (Product l2 : listProduct2) {

            System.out.println(l2.toString());

        }

        in.close();

    } // end mains

} // end class