



Submitted By	Habib ur Rehman (116)
Subject	OOP
Assignment	Generics
Date	Dec 09 th , 2024

Submitted to:

Moderator	Ms, Sajida Kalsoom
------------------	---------------------------

Lab Task 01:

```
import java.util.*;
import java.util.ArrayList;
import java.util.Scanner;

class Contacts {
    private String f_name;
    private String l_name;
    private String contactNo;
    private String email;
    private static ArrayList<Contacts> contactList = new ArrayList<>();

    public Contacts() {
    }

    public Contacts(String f_name, String l_name, String contactNo, String email) {
        this.f_name = f_name;
        this.l_name = l_name;
        this.contactNo = contactNo;
        this.email = email;
    }

    public String getF_name() {
        return f_name;
    }
}
```

```
public void setF_name(String f_name) {  
    this.f_name = f_name;  
}
```

```
public String getL_name() {  
    return l_name;  
}
```

```
public void setL_name(String l_name) {  
    this.l_name = l_name;  
}
```

```
public String getContactNo() {  
    return contactNo;  
}
```

```
public void setContactNo(String contactNo) {  
    this.contactNo = contactNo;  
}
```

```
public String getEmail() {  
    return email;  
}
```

```
public void setEmail(String email) {  
    this.email = email;  
}
```

```
public static void addContactList(Contacts c1) {  
    contactList.add(c1);  
  
}
```

```
public static void displayContactList() {  
    if (contactList.isEmpty()) {  
        System.out.println("No contacts to display.");  
    } else {  
        for (Contacts contact : contactList) {  
            contact.display();  
        }  
    }  
}
```

```
public static void deleteContact(Contacts c1) {  
    boolean found = false;  
  
    for (Contacts contact : contactList) {  
  
        if (c1.f_name.equals(contact.f_name) &&  
            c1.l_name.equals(contact.l_name) &&  
            c1.email.equals(contact.email) &&  
            c1.contactNo.equals(contact.contactNo)) {  
  
            contactList.remove(contact);  
            System.out.println("Contact removed: " + contact.f_name + " " + contact.l_name);  
        }  
    }  
}
```

```

        found = true;
        break;
    }
}

if (!found) {
    System.out.println("Contact not found.");
}
}

public void display() {
    System.out.println("Name: " + f_name + " " + l_name);
    System.out.println("Phone: " + contactNo);
    System.out.println("Email: " + email);
    System.out.println("*****");
}

public static void searchContacts(String c1) {
    boolean found = false;
    for (Contacts contact : contactList) {

        if (contact.f_name.toLowerCase().contains(c1.toLowerCase()) ||
            contact.l_name.toLowerCase().contains(c1.toLowerCase()) ||
            contact.email.toLowerCase().contains(c1.toLowerCase()) ||
            contact.contactNo.contains(c1)) {
            contact.display();
            found = true;
        }
    }
}

```

```

    }
    if (!found) {
        System.out.println("No matching contacts found.");
    }
}
}

public class labtask1 {
    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);
        boolean run = true;

        Contacts C1 = new Contacts("Habib", "rehman", "123-456-7890", "hb@@example.com");
        Contacts c2 = new Contacts("hb", "Sdwdvv", "987-654-3210", "cxc@example.com");
        Contacts.addContactList(C1);
        Contacts.addContactList(c2);

        while (run) {
            System.out.println("Contact Menu:");
            System.out.println("1. Add a Contact");
            System.out.println("2. Display All Contacts");
            System.out.println("3. Search for a Contact");
            System.out.println("4. Delete a Contact");
            System.out.println("5. Exit");
            System.out.print("Choose an option: ");
            int choice = scanner.nextInt();
            scanner.nextLine();

```

```
switch (choice) {  
    case 1:  
  
        System.out.print("Enter first name: ");  
        String firstName = scanner.nextLine();  
        System.out.print("Enter last name: ");  
        String lastName = scanner.nextLine();  
        System.out.print("Enter phone number: ");  
        String phoneNumber = scanner.nextLine();  
        System.out.print("Enter email address: ");  
        String email = scanner.nextLine();  
  
        Contacts.addContactList((new Contacts(firstName, lastName, phoneNumber,  
email)));  
        System.out.println("Contact added.");  
        break;  
  
    case 2:  
        // Display all contacts  
        Contacts.displayContactList();  
        break;  
  
    case 3:  
        // Search for a contact  
        System.out.print("Enter search term (name, email, phone): ");  
        String search = scanner.nextLine();  
        Contacts.searchContacts(search);  
        break;
```

case 4:

```
Contacts.deleteContact(C1);
```

```
break;
```

case 5:

```
// Exit the program
```

```
System.out.println("Exiting the program.");
```

```
run = false;
```

```
break;
```

default:

```
System.out.println("Invalid choice. Please try again.");
```

```
}
```

```
}
```

```
scanner.close();
```

```
}
```

```
}
```


Lab Task 02:

```
import java.lang.Number;
import java.util.ArrayList;

class MymathClass<T extends Number> {
    T obj;

    public MymathClass(T obj) {
        this.obj = obj;
    }

    public MymathClass() {

    }

    public T getobj() {
        return obj;
    }

    public void showType() {
        System.out.println("Type is: " + obj.getClass().getName());
    }

    public <T extends Number> double
    standardDeviation(ArrayList<T> list) {
```

```

        double sum = 0;
        for (T num : list) {
            sum += num.doubleValue();
        }
        // mean
        double mean = sum / list.size();

        // for sum of squared differences from the mean
        double sumSquaredDifferences = 0;
        for (T num : list) {
            sumSquaredDifferences += Math.pow(num.doubleValue() -
mean, 2);
        }

        // Calculate the variance
        double variance = sumSquaredDifferences / list.size();

        return Math.sqrt(variance);

    }

}

public class labtask2 {
    public static void main(String[] args) {
        ArrayList<Integer> l1 = new ArrayList<>();
        l1.add(10);
        l1.add(20);
        l1.add(30);
    }
}

```

```

        l1.add(40);
        l1.add(50);
        l1.add(60);
        l1.add(70);

        MymathClass<Integer> math = new MymathClass(null);
        double ans = math.standardDeviation(l1);
        System.out.println(ans);

        // ArrayList<String> stringList = new ArrayList<>();
        // stringList.add("Hello");
        // stringList.add("World");
        // stringList.add("Java");
        // math.standardDeviation(stringList);

        // The method standardDeviation(ArrayList<T>) in the type
        MymathClass<Integer>
        // is not applicable for the arguments
        (ArrayList<String>)Java(67108979)
        // <T> double MymathClass.standardDeviation(ArrayList<T>
        list)

    }
}

```

Lab Task 03:

```
import java.util.ArrayList;
import java.util.List;
import java.util.Random;

class Box<T> {
    private ArrayList<T> items;

    public Box() {
        items = new ArrayList<>();
    }

    public void add(T item) {
        items.add(item);
    }

    public boolean isEmpty() {
        return items.isEmpty();
    }

    public T drawItem() {
        if (isEmpty()) {
            return null;
        }

        Random random = new Random();
        int index = random.nextInt(items.size());
```

```

        return items.get(index);
    }

}

public class labtask3 {
    public static void main(String[] args) {

        Box<String> stringBox = new Box<>();
        stringBox.add("hb");
        stringBox.add("ah");
        stringBox.add("hfk");

        System.out.println("GEt a random name: " + stringBox.drawItem());

        Box<Integer> lotteryBox = new Box<>();
        lotteryBox.add(17);
        lotteryBox.add(123);
        lotteryBox.add(145);
        lotteryBox.add(33);

        System.out.println("getting a random lottery number: " + lotteryBox.drawItem());

    }

}

```

Lab Task 04:

```
import java.util.ArrayList;
import java.util.Collections;
import java.util.Scanner;

public class DivingScoreCalculator {

    public static void main(String[] args) {

        Scanner input = new Scanner(System.in);

        System.out.print("Enter the degree of difficulty (between 1.2 and 3.8): ");
        double degreeOfDifficulty = input.nextDouble();

        if (degreeOfDifficulty < 1.2 || degreeOfDifficulty > 3.8) {
            System.out.println("Invalid degree of difficulty. Must be between 1.2 and 3.8.");
            return;
        }

        ArrayList<Double> scores = new ArrayList<>();

        System.out.println("Enter 7 JUDGES' scores (between 0 and 10):");
        for (int i = 1; i <= 7; i++) {
            System.out.print("Judge's " + i + ": ");

            double score = input.nextDouble();

            if (score < 0 || score > 10) {
```

```
        System.out.println("Invalid entry! enter between 0 and 10.");
        return;
    }
    scores.add(score);
}

Collections.sort(scores);

scores.remove(0);
scores.remove(scores.size() - 1);

double sum = 0;
for (double score : scores) {
    sum += score;
}

double finalScore = sum * degreeOfDifficulty * 0.6;

System.out.println("The final score for the dive is: " + finalScore);
}
}
```