LOKESHVISWA M

23/07/2025

1. A library needs to develop an online application for two types of users/roles, Adults and children. Both of these users should be able to register an account. Any user who is less than 12 years of age will be registered as a child and they can borrow a "Kids" category book for 10 days, whereas an adult can borrow "Fiction" category books which need to be returned within 7 days.

1.Create an interface LibraryUser with the following methods declared,

Method Name

registerAccount

requestBook

2.Create 2 classes "KidUser" and "AdultUser" which implements the LibraryUser interface.

3.Both the classes should have two instance variables as specified below.

age	int
bookType	String

4.The methods in the KidUser class should perform the following logic.

1. registerAccount : if age < 12, a message displaying “You have successfully registered under a Kids Account” should be displayed in the console.

If(age>12), a message displaying, "Sorry, Age must be less than 12 to register as a kid" should be displayed in the console.

2. requestBook : if bookType is "Kids", a message displaying "Book Issued successfully, please return the book within 10 days" should be displayed in the console.

else, a message displaying, "You are allowed to take only kids books" should be displayed in the console.

5.The methods in the AdultUser class should perform the following logic.

1. registerAccount : if age > 12, a message displaying "You have successfully registered under an Adult Account" should be displayed in the console.

If age<12, a message displaying, "Sorry, Age must be greater than 12 to register as an adult" should be displayed in the console.

2. requestBook : if bookType is "Fiction", a message displaying "Book Issued successfully, please return the book within 7 days" should be displayed in the console.

else, a message displaying, "

You are allowed to take only adult Fiction books" should be displayed in the console.

6.Create a class LibraryInterfaceDemo with a main method which performs the below functions,

In the main method, test all the methods

CODE:

```
interface LibraryUser {
    void registerAccount();
    void requestBook();
}

class KidUser implements LibraryUser {
    int age;
    String bookType;
```

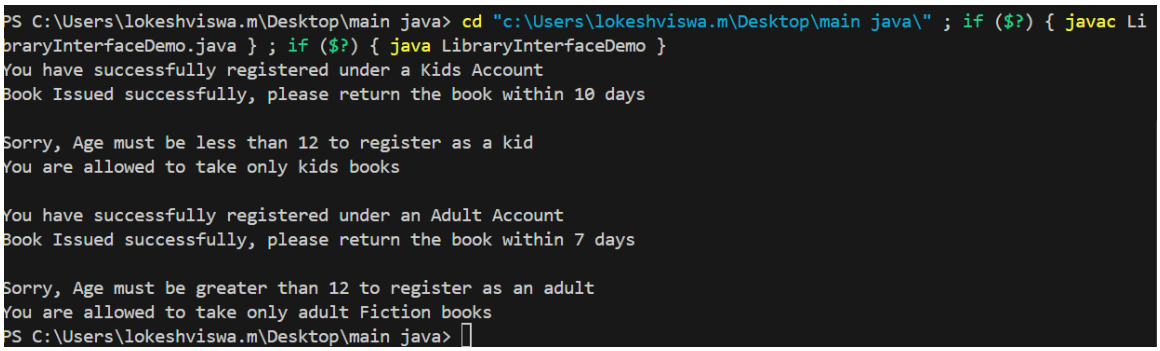
```
    public KidUser(int age, String bookType) {
        this.age = age;
        this.bookType = bookType;
    }

    @Override
    public void registerAccount() {
        if (age < 12) {
            System.out.println("You have successfully registered under a Kids Account");
        } else {
            System.out.println("Sorry, Age must be less than 12 to register as a kid");
        }
    }

    @Override
    public void requestBook() {
        if (bookType.equalsIgnoreCase("Kids")) {
            System.out.println("Book Issued successfully, please return the book within 10 days");
        } else {
            System.out.println("You are allowed to take only kids books");
        }
    }
}

class AdultUser implements LibraryUser {
```

```
    int age;
    String bookType;

    public AdultUser(int age, String bookType) {
        this.age = age;
        this.bookType = bookType;
    }

    @Override
    public void registerAccount() {
        if (age > 12) {
            System.out.println("You have successfully registered under an Adult Account");
        } else {
            System.out.println("Sorry, Age must be greater than 12 to register as an adult");
        }
    }

    @Override
    public void requestBook() {
        if (bookType.equalsIgnoreCase("Fiction")) {
            System.out.println("Book Issued successfully, please return the book within 7 days");
        } else {
            System.out.println("You are allowed to take only adult Fiction books");
        }
    }
}
```

```
// Main class
public class LibraryInterfaceDemo {
    public static void main(String[] args) {
        // Test case 1: Kid user valid
        LibraryUser kid1 = new KidUser(10, "Kids");
        kid1.registerAccount();
        kid1.requestBook();

        System.out.println();

        // Test case 2: Kid user invalid
        LibraryUser kid2 = new KidUser(14, "Fiction");
        kid2.registerAccount();
        kid2.requestBook();

        System.out.println();

        // Test case 3: Adult user valid
        LibraryUser adult1 = new AdultUser(23, "Fiction");
        adult1.registerAccount();
        adult1.requestBook();

        System.out.println();

        // Test case 4: Adult user invalid
```

```
        LibraryUser adult2 = new AdultUser(11, "Kids");
        adult2.registerAccount();
        adult2.requestBook();
    }
}
```

OUTPUT:



2.Write a program to read two integer array lists of size 5 each as input and to merge the two arrayLists, sort the merged arraylist in ascending order and fetch the elements at 2nd, 6th and 8th index into a new arrayList and return the final ArrayList.

CODE:

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

public class MergeSortExtractList {

    public static ArrayList<Integer> mergeAndExtract(ArrayList<Integer> list1, ArrayList<Integer> list2) {

        ArrayList<Integer> mergedList = new ArrayList<>(list1);
        mergedList.addAll(list2);
```

```
        Collections.sort(mergedList);



        ArrayList<Integer> resultList = new ArrayList<>();
        if (mergedList.size() > 2) resultList.add(mergedList.get(2));
        if (mergedList.size() > 6) resultList.add(mergedList.get(6));
        if (mergedList.size() > 8) resultList.add(mergedList.get(8));

        return resultList;
}

public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        ArrayList<Integer> list1 = new ArrayList<>();
        ArrayList<Integer> list2 = new ArrayList<>();

        System.out.println("Enter 5 integers for List 1:");
        for (int i = 0; i < 5; i++) {
                list1.add(sc.nextInt());
        }

        System.out.println("Enter 5 integers for List 2:");
        for (int i = 0; i < 5; i++) {
                list2.add(sc.nextInt());
        }
```

```
        ArrayList<Integer> result = mergeAndExtract(list1, list2);


        System.out.println("Result List (2nd, 6th, 8th index values from merged sorted list):");
        System.out.println(result);


        sc.close();
    }
}
```

OUTPUT:

3.Read student details as input.The details would include name, mark in the given order. The datatype for name is string, mark is float.Create a hashmap that contains name as key and mark as value.Get student name as input and display the student grade.

1. If Mark is less than 60, then grade is FAIL.

2. If Mark is greater than or equal to 60, then grade is PASS.

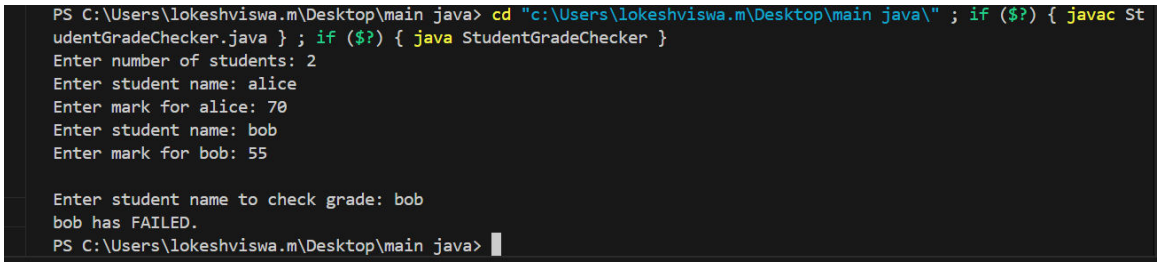
CODE:

```
import java.util.HashMap;

import java.util.Scanner;


public class StudentGradeChecker {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);
```

```
HashMap<String, Float> studentMap = new HashMap<>();



System.out.print("Enter number of students: ");
int n = scanner.nextInt();
scanner.nextLine();



for (int i = 0; i < n; i++) {
        System.out.print("Enter student name: ");
        String name = scanner.nextLine();

        System.out.print("Enter mark for " + name + ": ");
        float mark = scanner.nextFloat();
        scanner.nextLine();

        studentMap.put(name, mark);
}



System.out.print("\nEnter student name to check grade: ");
String searchName = scanner.nextLine();



if (studentMap.containsKey(searchName)) {
        float mark = studentMap.get(searchName);
        if (mark >= 60) {
                System.out.println(searchName + " has PASSED.");
```

```
            } else {
                System.out.println(searchName + " has FAILED.");
            }
        } else {
            System.out.println("Student not found.");
        }

        scanner.close();
    }
}
```

OUTPUT:



4. Write a program to get integers as input and store in the arraylist. Traverse the input list, if the number is even store in a arraylist called evenNumbersList and oddnumbers in oddNumberList. Print the input list and the lists containing even numbers and odd numbers.

CODE:

```
import java.util.ArrayList;

import java.util.Scanner;


public class EvenOddArrayList {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);



        ArrayList<Integer> inputList = new ArrayList<>();
```

```
ArrayList<Integer> evenNumbersList = new ArrayList<>();
ArrayList<Integer> oddNumbersList = new ArrayList<>();

System.out.print("Enter how many numbers you want to input: ");
int n = scanner.nextInt();


for (int i = 0; i < n; i++) {
	System.out.print("Enter number " + (i + 1) + ": ");
	int number = scanner.nextInt();
	inputList.add(number);


	if (number % 2 == 0) {
		evenNumbersList.add(number);
	} else {
		oddNumbersList.add(number);
	}
}


System.out.println("\nInput List: " + inputList);
System.out.println("Even Numbers List: " + evenNumbersList);
System.out.println("Odd Numbers List: " + oddNumbersList);

scanner.close();
```

```
    }
}
```

```
PS C:\Users\lokeshviswa.m\Desktop\main java> cd "c:\Users
enOddArrayList.java } ; if ($?) { java EvenOddArrayList }
Enter how many numbers you want to input: 5
Enter number 1: 10
Enter number 2: 3
Enter number 3: 7
Enter number 4: 8
Enter number 5: 4

Input List: [10, 3, 7, 8, 4]
Even Numbers List: [10, 8, 4]
Odd Numbers List: [3, 7]
PS C:\Users\lokeshviswa.m\Desktop\main java>
```