**DAY-11**

**Date:23.07.2025**
**Name:Nancy M**

**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.**

```
package com.training.ooc;
import java.util.Scanner;
interface LibraryUser {
void registerAccount();
void requestBook();
}
class KidUser implements LibraryUser {
int age;
String bookType;
KidUser(int age, String bookType) {
   this.age = age;
   this.bookType = bookType;
}
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");
   }
}
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;
AdultUser(int age, String bookType) {
   this.age = age;
   this.bookType = bookType;
}
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");
   }
}
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");
    }
}
}
public class LibraryInterfaceDemo {
public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
       System.out.println("Enter age for Kid User:");
    int kidAge = sc.nextInt();
    sc.nextLine();     System.out.println("Enter book type for Kid User:");
    String kidBookType = sc.nextLine();
    KidUser kidUser = new KidUser(kidAge, kidBookType);
    kidUser.registerAccount();
    kidUser.requestBook();
    System.out.println();
      System.out.println("Enter age for Adult User:");
    int adultAge = sc.nextInt();
    sc.nextLine();     System.out.println("Enter book type for Adult User:");
    String adultBookType = sc.nextLine();
    AdultUser adultUser = new AdultUser(adultAge, adultBookType);
    adultUser.registerAccount();
    adultUser.requestBook();
    sc.close();
}
}
```

**Output:**

```
Enter age for Kid User:
10
Enter book type for Kid User:
Kids
You have successfully registered under a Kids Account
Book Issued successfully, please return the book within 10 days
Enter age for Adult User:
23
Enter book type for Adult User:
Fiction
You have successfully registered under an Adult Account
Book Issued successfully, please return the book within 7 days
```

**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.**

```
package com.training.ooc;
import java.util.*;
public class Merge{
  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 the first ArrayList:");
     for (int i = 0; i < 5; i++) {
        list1.add(sc.nextInt());
     }

     System.out.println("Enter 5 integers for the second ArrayList:");
     for (int i = 0; i < 5; i++) {
        list2.add(sc.nextInt());
     }
     ArrayList<Integer> result = mergeSortAndFetch(list1, list2);
     System.out.println("Final result (elements at 2nd, 6th, and 8th index): " + result);

     sc.close();
  }
  public static ArrayList<Integer> mergeSortAndFetch(ArrayList<Integer> list1, ArrayList<Integer> list2) {
     ArrayList<Integer> mergedList = new ArrayList<>();
     mergedList.addAll(list1);
     mergedList.addAll(list2);
     Collections.sort(mergedList);
     ArrayList<Integer> resultList = new ArrayList<>();
         int[] indexes = {2, 6, 8};
     for (int index : indexes) {
        if (index < mergedList.size()) {
           resultList.add(mergedList.get(index));
        } else {
           System.out.println("Index " + index + " is out of bounds in the merged list.");
        }
     }
     return resultList;
  }
}
```

**Output:**

```
Enter 5 integers for the first ArrayList:
3 7 8 9 3
Enter 5 integers for the second ArrayList:
4 98 67 8 9
Final result (elements at 2nd, 6th, and 8th index): [4, 9, 67]
```

**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.**

```
package com.training.ooc;
import java.util.*;
public class Studentgrade {
  public static void main(String[] args) {
     Scanner sc = new Scanner(System.in);
     HashMap<String, Float> studentMap = new HashMap<>();
     System.out.print("Enter number of students: ");
     int n = sc.nextInt();
     sc.nextLine();
        for (int i = 0; i < n; i++) {
        System.out.print("Enter student name: ");
        String name = sc.nextLine();
        System.out.print("Enter mark for " + name + ": ");
        float mark = sc.nextFloat();
        sc.nextLine(); // consume newline
        studentMap.put(name, mark);
     }
      System.out.print("Enter student name to get grade: ");
     String queryName = sc.nextLine();
     if (studentMap.containsKey(queryName)) {
        float mark = studentMap.get(queryName);
        String grade = (mark >= 60) ? "PASS" : "FAIL";
        System.out.println("Student: " + queryName + ", Mark: " + mark + ", Grade: " + grade);
     } else {
        System.out.println("Student not found.");
     }
     sc.close();
  }
}
```

**Output:**

```
Enter number of students: 2
Enter student name: ncy
Enter mark for ncy: 56
Enter student name: jcy
Enter mark for jcy: 82
Enter student name to get grade: jcy
Student: jcy, Mark: 82.0, Grade: PASS
```

**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.**

```
package com.training.ooc;
import java.util.*;
public class oddeven {
  public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);
      ArrayList<Integer> inputList = new ArrayList<>();
      ArrayList<Integer> evenNumbersList = new ArrayList<>();
      ArrayList<Integer> oddNumbersList = new ArrayList<>();
      System.out.print("Enter number of integers: ");
      int n = sc.nextInt();
       System.out.println("Enter " + n + " integers:");
      for (int i = 0; i < n; i++) {
          int num = sc.nextInt();
          inputList.add(num);
          if (num % 2 == 0) {
              evenNumbersList.add(num);
          } else {
              oddNumbersList.add(num);
          }
      }
          System.out.println("Even Numbers List: " + evenNumbersList);
      System.out.println("Odd Numbers List: " + oddNumbersList);
      sc.close();
  }
}
```

**Output:**

Enter number of integers: 5
Enter 5 integers:
6 8 9 7 43
Even Numbers List: [6, 8]
Odd Numbers List: [9, 7, 43]