**Lab Report**

**Course Tittle:** Object Oriented Programming Lab

**Course Code:** CSE 215

**Experiment No:** 09

**Experiment Name:** Designing Student Management System by utilizing Encapsulation, Dependency and Collection Framework of JAVA

**Submitted To:**

**Name:** Mst. Umme Ayman

**Designation:** Lecturer

**Department of CSE**

**Daffodil International University**

**Submitted By**

**Name:** Md Raduan Ahamed

**ID:** 0242220005101839

**Section:** 63\_O

**Department of CSE**

**Daffodil International University**

**Submission Date:** 29-08-2023

Managing student records is essential for educational institutions. Suppose you are a software developer working at DIU and have been tasked with creating an application to meet their specific needs. The system's primary functions include student registration and the ability to search for students by their student ID or mobile number. Furthermore, upon a student's successful completion of their studies, the system should provide the option to mark them as "completed," ensuring that they no longer appear in search results for active students.

import java.util.\*;  
import java.util.Scanner;  
  
class Student {  
 private String Id;  
 private String mobileNumber;  
 private boolean isCompleted;  
  
 public Student(String Id, String mobileNumber) {  
 this.Id = Id;  
 this.mobileNumber = mobileNumber;  
 this.isCompleted = false;  
 }  
  
 public String getId() {  
 return Id;  
 }  
  
 public String getMobileNumber() {  
 return mobileNumber;  
 }  
  
 public boolean isCompleted() {  
 return isCompleted;  
 }  
  
 public void markCompleted() {  
 this.isCompleted = true;  
 }  
  
 @Override  
 public String toString() {  
 return "Student ID: " + Id + ", Mobile Number: " + mobileNumber + ", Completed: " + isCompleted;  
 }  
}

import java.util.ArrayList;  
import java.util.Iterator;  
import java.util.Scanner;  
  
public class student\_management{  
 private ArrayList<Student> students;  
  
 public student\_management() {  
 this.students = new ArrayList<>();  
 }  
  
 public void registerStudent(String Id, String mobileNumber) {  
 students.add(new Student(Id, mobileNumber));  
 System.*out*.println("Student registered successfully.");  
 }  
  
 public Student searchStudent(String query) {  
 for (Student student : students) {  
 if (student.getId().equals(query) || student.getMobileNumber().equals(query)) {  
 return student;  
 }  
 }  
 return null; *// Return null if no student is found* }  
  
 public void markStudentCompleted(String Id) {  
 Iterator<Student> iterator = students.iterator();  
 while (iterator.hasNext()) {  
 Student student = iterator.next();  
 if (student.getId().equals(Id)) {  
 student.markCompleted();  
 System.*out*.println("Student marked as completed.");  
 return;  
 }  
 }  
 System.*out*.println("Student not found.");  
 }  
  
 public void displayAllStudents() {  
 System.*out*.println("All Students:");  
 for (Student student : students) {  
 System.*out*.println(student);  
 }  
 }  
 public static void main(String[] args) {  
 student\_management system = new student\_management();  
 Scanner scanner = new Scanner(System.*in*);  
  
 *// Example: Register students* system.registerStudent("1839", "1234567890");  
 system.registerStudent("1620", "1234568970");  
  
 *// Example: Search for a student* System.*out*.print("Enter student ID or mobile number to search: ");  
 String searchQuery = scanner.nextLine();  
 Student foundStudent = system.searchStudent(searchQuery);  
 if (foundStudent != null) {  
 System.*out*.println("Student found: " + foundStudent);  
 } else {  
 System.*out*.println("Student not found.");  
 }  
  
 *// Example: Mark a student as completed* System.*out*.print("Enter student ID to mark as completed: ");  
 String studentIdToComplete = scanner.nextLine();  
 system.markStudentCompleted(studentIdToComplete);  
  
  
 *// Example: Display all students* system.displayAllStudents();  
 }  
}

Output:

Student registered successfully.

Student registered successfully.

Enter student ID or mobile number to search: 1855

Student not found.

Enter student ID to mark as completed: 1839

Student marked as completed.

All Students:

Student ID: 1839, Mobile Number: 1234567890, Completed: true

Student ID: 1620, Mobile Number: 1234568970, Completed: false