

STUDENT MANAGEMENT

Code:

```
2 package com.mycompany.studentmanagement;
3 import java.util.Scanner;
4 public class StudentManagement {
5
6     static String[][] studentStorage = new String[5][4];
7     static int currentStudentCount = 0;
8     static Scanner input = new Scanner(System.in);
9     public static void main(String[] args) {
10         int userChoice;
11         do {
12             showMenu();
13             System.out.print("Enter your choice (1-6): ");
14             userChoice = input.nextInt();
15             input.nextLine();
16             switch (userChoice) {
17                 case 1: addStudent(); break;
18                 case 2: updateStudent(); break;
19                 case 3: removeStudent(); break;
20                 case 4: findStudent(); break;
21                 case 5: showAllStudents(); break;
22                 case 6: System.out.println("Goodbye!"); break;
23                 default: System.out.println("Invalid choice. Try again.");
24             }
25             System.out.println("---");
26         } while (userChoice != 6);
27         input.close();
28     }
29
30     static void showMenu() {
31         System.out.println("\n--- Student Manager ---");
32         System.out.println("1. Add Student");
33         System.out.println("2. Update Student");
34         System.out.println("3. Remove Student");
35         System.out.println("4. Find Student");
36         System.out.println("5. Show All Students");
37         System.out.println("6. Exit");
38     }
39
40     static void addStudent() {
41         if (currentStudentCount == studentStorage.length) {
42             System.out.println("Storage full.");
43             return;
44         }
45         System.out.print("Student ID: ");
46         String id = input.nextLine();
47         if (findStudentIndex(id) != -1) {
48             System.out.println("ID already exists.");
49             return;
50         }
51         System.out.print("Name: ");
52         String name = input.nextLine();
53         System.out.print("Age: ");
54         String age = input.nextLine();
55         System.out.print("Semester: ");
56         String semester = input.nextLine();
57         studentStorage[currentStudentCount][0] = id;
58         studentStorage[currentStudentCount][1] = name;
```

```

57     studentStorage[currentStudentCount][2] = age;
58     studentStorage[currentStudentCount][3] = semester;
59     currentStudentCount++;
60     System.out.println("Student added.");
61 }
62 static void updateStudent() {
63     if (currentStudentCount == 0) {
64         System.out.println("No students to update."); return; }
65     System.out.print("Enter ID to update: ");
66     String idToUpdate = input.nextLine();
67     int foundIndex = findStudentIndex(idToUpdate);
68     if (foundIndex == -1) {
69         System.out.println("Student not found.");
70         return; }
71     System.out.print("New Name (current: " + studentStorage[foundIndex][1] + "): ");
72     String newName = input.nextLine();
73     if (!newName.isEmpty()) studentStorage[foundIndex][1] = newName;
74     System.out.print("New Age (current: " + studentStorage[foundIndex][2] + "): ");
75     String newAge = input.nextLine();
76     if (!newAge.isEmpty()) studentStorage[foundIndex][2] = newAge;
77     System.out.print("New Semester (current: " + studentStorage[foundIndex][3] + "): ");
78     String newSemester = input.nextLine();
79     if (!newSemester.isEmpty()) studentStorage[foundIndex][3] = newSemester;
80     System.out.println("Student updated.");
81 }
82 static void removeStudent() {
83     if (currentStudentCount == 0) { System.out.println("No students to remove."); return; }
84     System.out.print("Enter ID to remove: "); String idToRemove = input.nextLine();

85     int foundIndex = findStudentIndex(idToRemove);

86
87     if (foundIndex == -1) { System.out.println("Student not found."); return; }
88     for (int i = foundIndex; i < currentStudentCount - 1; i++) {
89         studentStorage[i][0] = studentStorage[i+1][0];
90         studentStorage[i][1] = studentStorage[i+1][1];
91         studentStorage[i][2] = studentStorage[i+1][2];
92         studentStorage[i][3] = studentStorage[i+1][3];
93     }
94     studentStorage[currentStudentCount - 1][0] = null;
95     studentStorage[currentStudentCount - 1][1] = null;
96     studentStorage[currentStudentCount - 1][2] = null;
97     studentStorage[currentStudentCount - 1][3] = null;
98     currentStudentCount--;
99     System.out.println("Student removed.");
100 }
101 static void findStudent() {
102     if (currentStudentCount == 0) { System.out.println("No students to find."); return; }
103     System.out.print("Enter ID to find: "); String searchId = input.nextLine();
104     int foundIndex = findStudentIndex(searchId);
105     if (foundIndex == -1) { System.out.println("Student not found."); return; }
106     System.out.println("ID: " + studentStorage[foundIndex][0] + ", Name: "
107         + studentStorage[foundIndex][1] +
108         ", Age: " + studentStorage[foundIndex][2] + ", Semester: "
109         + studentStorage[foundIndex][3]);
110 }
111 static void showAllStudents() {
112     if (currentStudentCount == 0) { System.out.println("No students in storage."); return; }

```

```

113         System.out.println("--- All Students ---");
114         for (int i = 0; i < currentStudentCount; i++) {
115             System.out.println(studentStorage[i][0] + "\t" + studentStorage[i][1] + "\t" +
116                               studentStorage[i][2] + "\t" + studentStorage[i][3]);
117         }
118     }
119     static int findStudentIndex(String targetId) {
120         for (int i = 0; i < currentStudentCount; i++) {
121             if (studentStorage[i][0].equalsIgnoreCase(targetId)) {
122                 return i;
123             }
124         }
125         return -1;
126     }
127 }

```

Output:

```

--- Student Manager ---
1. Add Student
2. Update Student
3. Remove Student
4. Find Student
5. Show All Students
6. Exit
Enter your choice (1-6): 1
Student ID: 1
Name: Samra
Age: 19
Semester: 3
Student added.
---

```

```

--- Student Manager ---
1. Add Student
2. Update Student
3. Remove Student
4. Find Student
5. Show All Students
6. Exit
Enter your choice (1-6): 1
Student ID: 2
Name: Ayesha
Age: 20
Semester: 4
Student added.
---

```

--- Student Manager ---

1. Add Student
2. Update Student
3. Remove Student
4. Find Student
5. Show All Students
6. Exit

Enter your choice (1-6): 5

--- All Students ---

1	Samra	19	3
2	Ayesha	20	4

--- Student Manager ---

1. Add Student
2. Update Student
3. Remove Student
4. Find Student
5. Show All Students
6. Exit

Enter your choice (1-6): 6

Goodbye!
