```
import java.io.*;
import java.util.*;

public class StudentFileHandling {
    static final String FILE_NAME = "students.txt";

    // Create file if it doesn't exist
    public static void createFile() {
        try {
            File file = new File(FILE_NAME);
            if (file.createNewFile()) {
                System.out.println("File created.");
            }
        } catch (IOException e) {
            System.out.println("Error creating file.");
        }
    }

    // Write to file (overwrite)
    public static void writeFile(String data) {
        try (FileWriter fw = new FileWriter(FILE_NAME)) {
            fw.write(data);
            System.out.println("Data written.");
        } catch (IOException e) {
            System.out.println("Error writing to file.");
        }
    }

    // Append to file
    public static void appendFile(String data) {
        try (FileWriter fw = new FileWriter(FILE_NAME, true)) {
            fw.write(data);
            System.out.println("Data appended.");
        } catch (IOException e) {
            System.out.println("Error appending to file.");
        }
    }

    // Read file
    public static void readFile() {
        try (BufferedReader br = new BufferedReader(new FileReader(FILE_NAME))) {
            String line;
            System.out.println("Student Records:");
            while ((line = br.readLine()) != null) {
```

```
                System.out.println(line);
            }
        } catch (IOException e) {
            System.out.println("Error reading file.");
        }
    }

    // Update record by replacing entire file content
    public static void updateFile(String studentId, String newData) {
        File file = new File(FILE_NAME);
        List<String> lines = new ArrayList<>();
        try (BufferedReader br = new BufferedReader(new FileReader(file))) {
            String line;
            while ((line = br.readLine()) != null) {
                if (line.startsWith(studentId + ",")) {
                    lines.add(newData); // Replace line
                } else {
                    lines.add(line); // Keep line
                }
            }
        } catch (IOException e) {
            System.out.println("Error reading file during update.");
            return;
        }
try (FileWriter fw = new FileWriter(file)) {
            for (String l : lines) {
                fw.write(l + "\n");
            }
            System.out.println("Record updated.");
        } catch (IOException e) {
            System.out.println("Error updating file.");
        }
    }

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

        while (true) {
            System.out.println("\n1. Write Records");
            System.out.println("2. Append Record");
            System.out.println("3. Read Records");
            System.out.println("4. Update Record");
            System.out.println("5. Exit");
```

```
            System.out.print("Choose: ");

            int choice;
            try {
                choice = Integer.parseInt(sc.nextLine());
            } catch (NumberFormatException e) {
                System.out.println("Invalid input.");
                continue;
            }

            switch (choice) {
                case 1:
                    System.out.print("Enter number of students: ");
                    int n;
                    try {
                        n = Integer.parseInt(sc.nextLine());
                    } catch (NumberFormatException e) {
                        System.out.println("Invalid number.");
                        break;
                    }
                    StringBuilder sb = new StringBuilder();
                    for (int i = 0; i < n; i++) {
                        System.out.print("ID: ");
                        String id = sc.nextLine();
                        System.out.print("Name: ");
                        String name = sc.nextLine();
                        System.out.print("Age: ");
                        String age = sc.nextLine();

sb.append(id).append(",").append(name).append(",").append(age).append("\n");
                    }
                    writeFile(sb.toString());
                    break;
                case 2:
                    System.out.print("ID: ");
                    String id = sc.nextLine();
                    System.out.print("Name: ");
                    String name = sc.nextLine();
                    System.out.print("Age: ");
                    String age = sc.nextLine();
                    appendFile(id + "," + name + "," + age + "\n");
                    break;
                case 3:
                    readFile();
```

```
                break;
            case 4:
                System.out.print("Enter ID to update: ");
                String updateId = sc.nextLine();
                System.out.print("New Name: ");
                String newName = sc.nextLine();
                System.out.print("New Age: ");
                String newAge = sc.nextLine();
                updateFile(updateId, updateId + "," + newName + "," + newAge);
                break;
            case 5:
                System.out.println("Bye!");
                sc.close();
                return;
            default:
                System.out.println("Invalid choice.");
        }
    }
  }
}
```