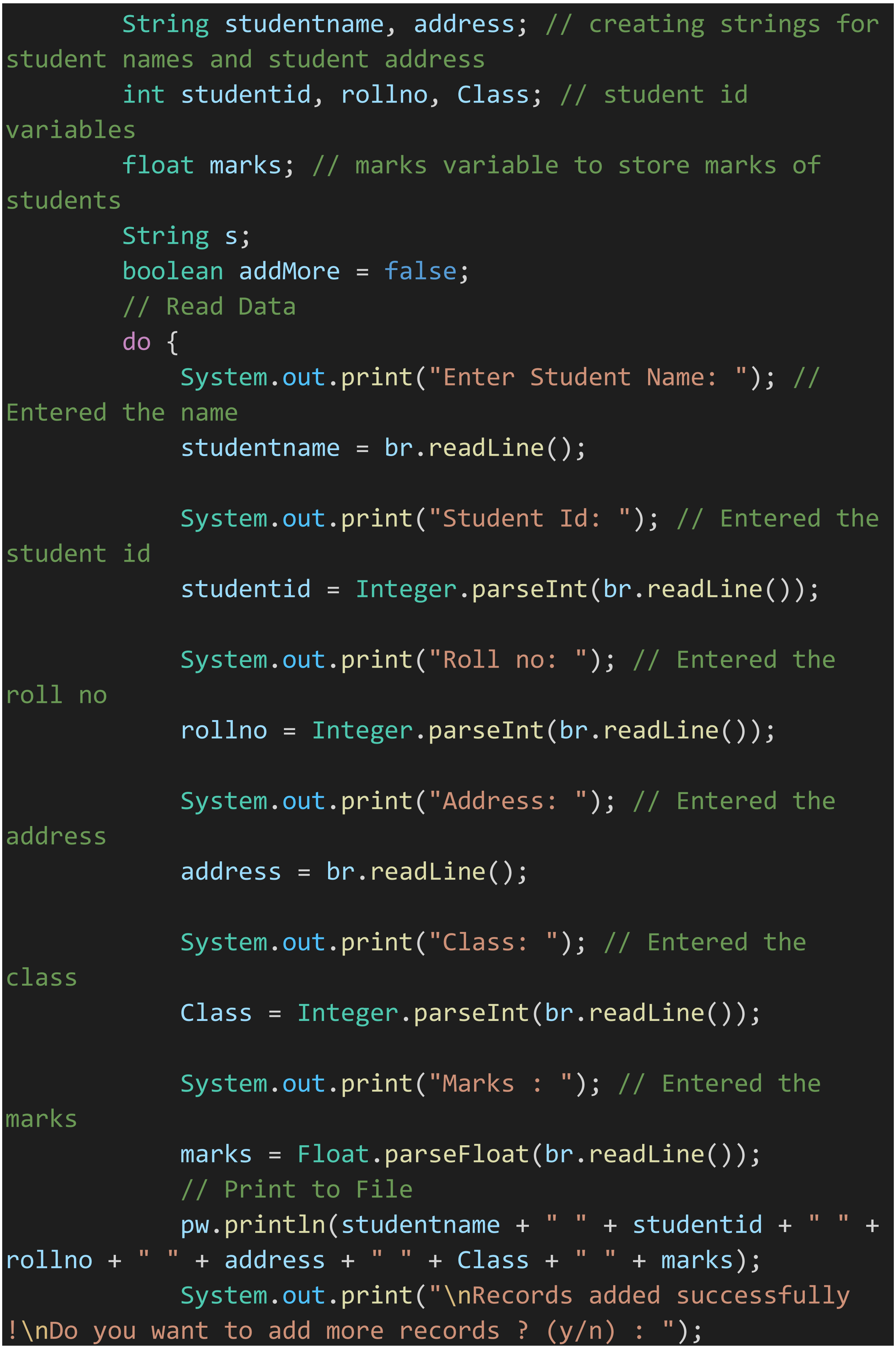
OOPL Assignment-8

File Handling

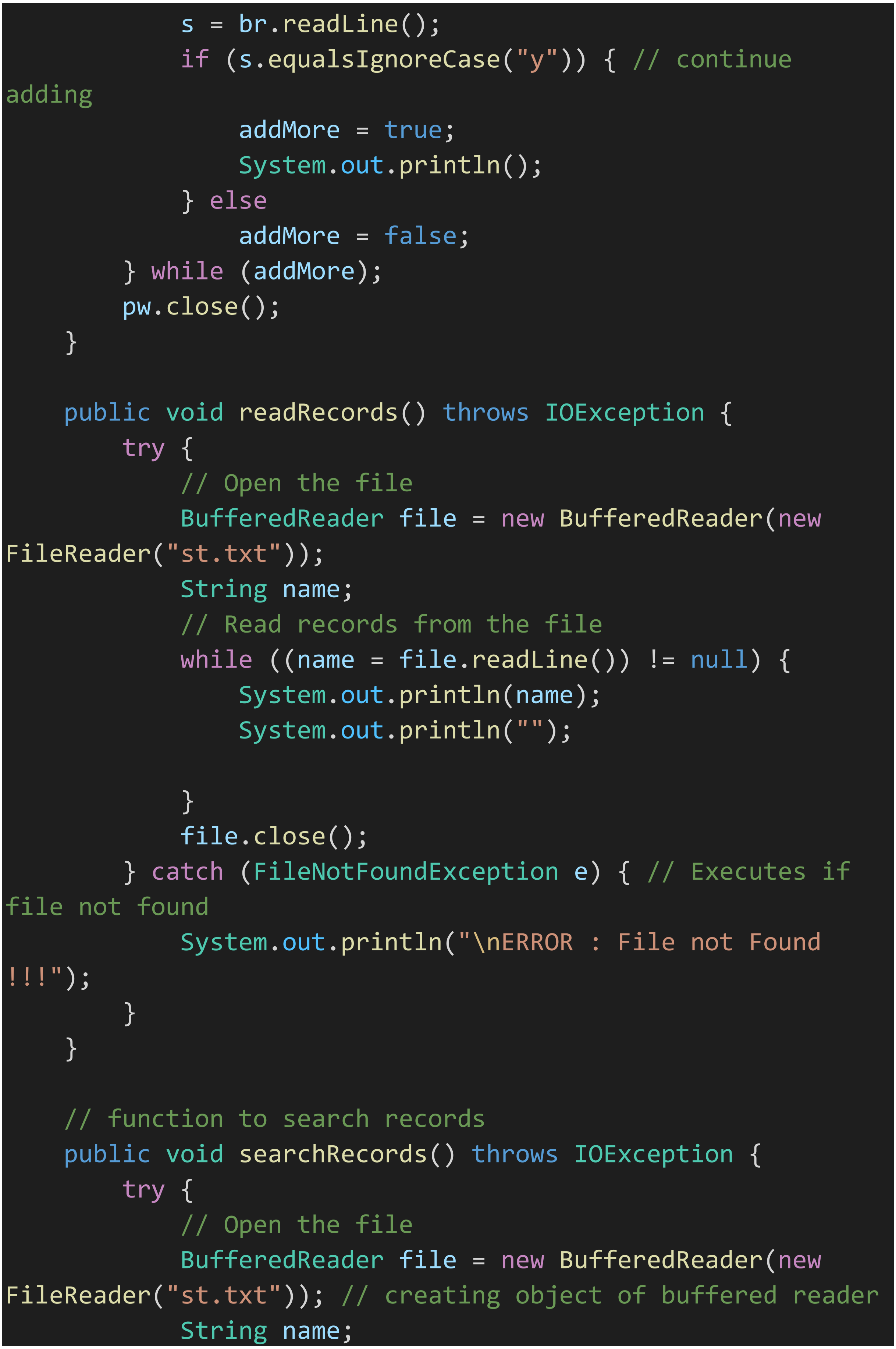
Name :- Atharva Kinikar
Div : SE 10
Batch : F 10
Roll No : 23241

Code :-

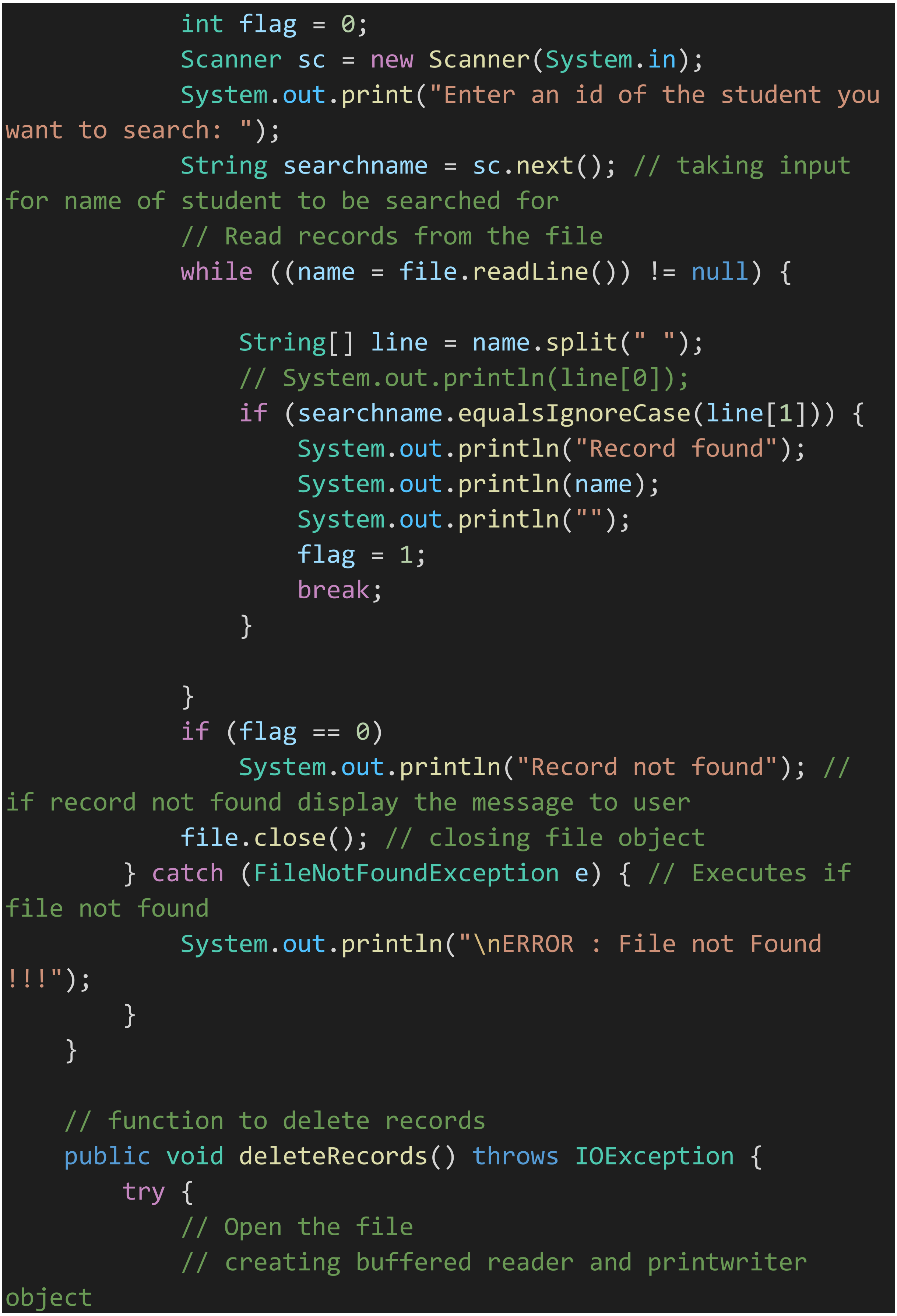
```
/*
Name: Atharva Kinikar
Div : SE 10
Batch : F10
Roll NO : 23241
*/

//importing java packages essential for our program
import java.io.*;
import java.io.File;
import java.util.*;

//creating a class student records
class StudentRecords {
    static BufferedReader br = new BufferedReader(new
InputStreamReader(System.in)); // creating an object of
buffererd

                         // reader

    public void addRecords() throws IOException {
        // Create or Modify a file for Database
        PrintWriter pw = new PrintWriter(new
BufferedWriter(new FileWriter("st.txt", true))); // creating
an object of

                                  // printwriter class
```



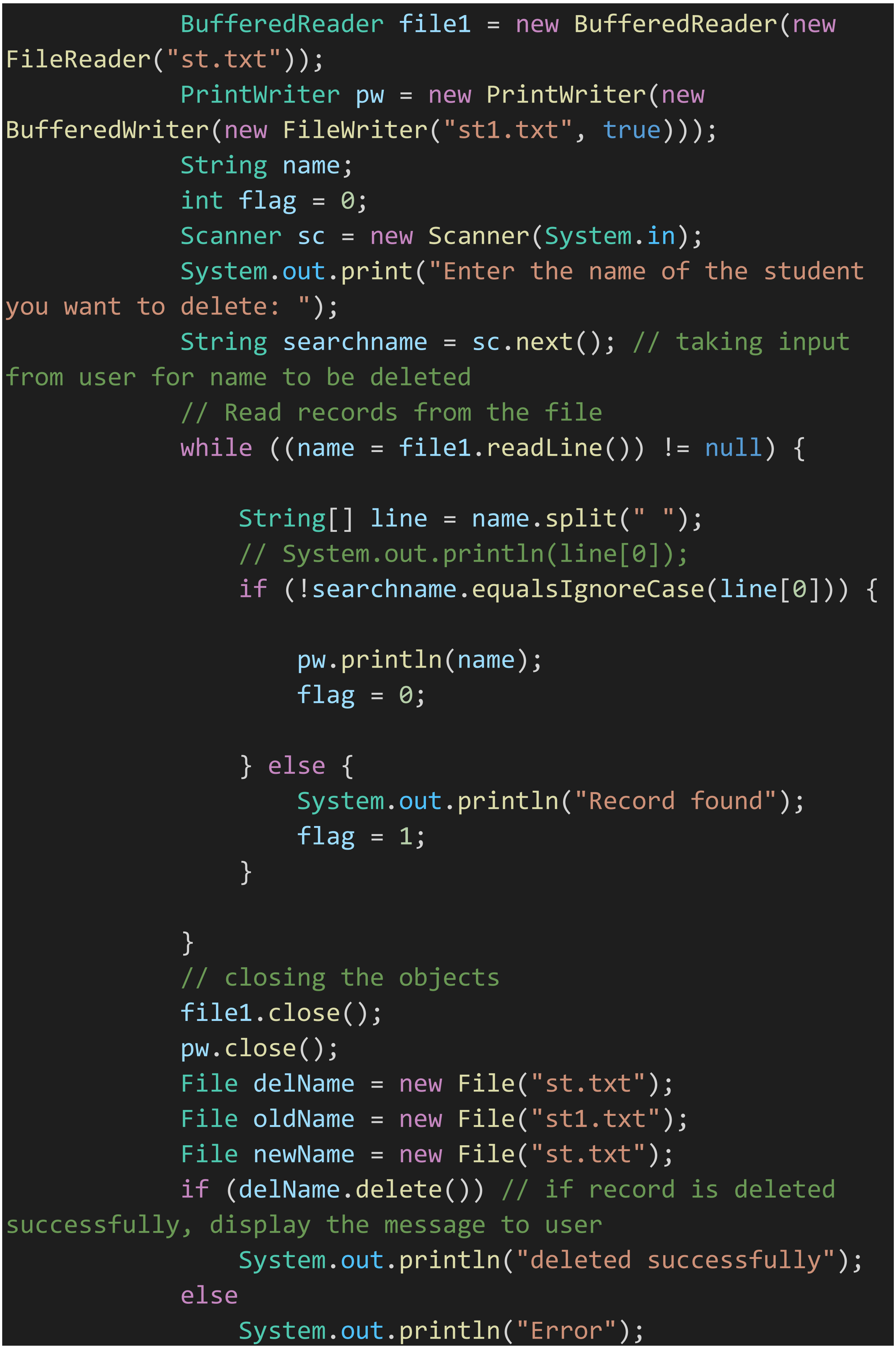
```
        String studentname, address; // creating strings for
student names and student address
        int studentid, rollno, Class; // student id
variables
        float marks; // marks variable to store marks of
students
        String s;
        boolean addMore = false;
        // Read Data
        do {
            System.out.print("Enter Student Name: "); //
Entered the name
            studentname = br.readLine();

            System.out.print("Student Id: "); // Entered the
student id
            studentid = Integer.parseInt(br.readLine());

            System.out.print("Roll no: "); // Entered the
roll no
            rollno = Integer.parseInt(br.readLine());

            System.out.print("Address: "); // Entered the
address
            address = br.readLine();

            System.out.print("Class: "); // Entered the
class
            Class = Integer.parseInt(br.readLine());

            System.out.print("Marks : "); // Entered the
marks
            marks = Float.parseFloat(br.readLine());
            // Print to File
            pw.println(studentname + " " + studentid + " " +
rollno + " " + address + " " + Class + " " + marks);
            System.out.print("\nRecords added successfully
!\nDo you want to add more records ? (y/n) : ");
```



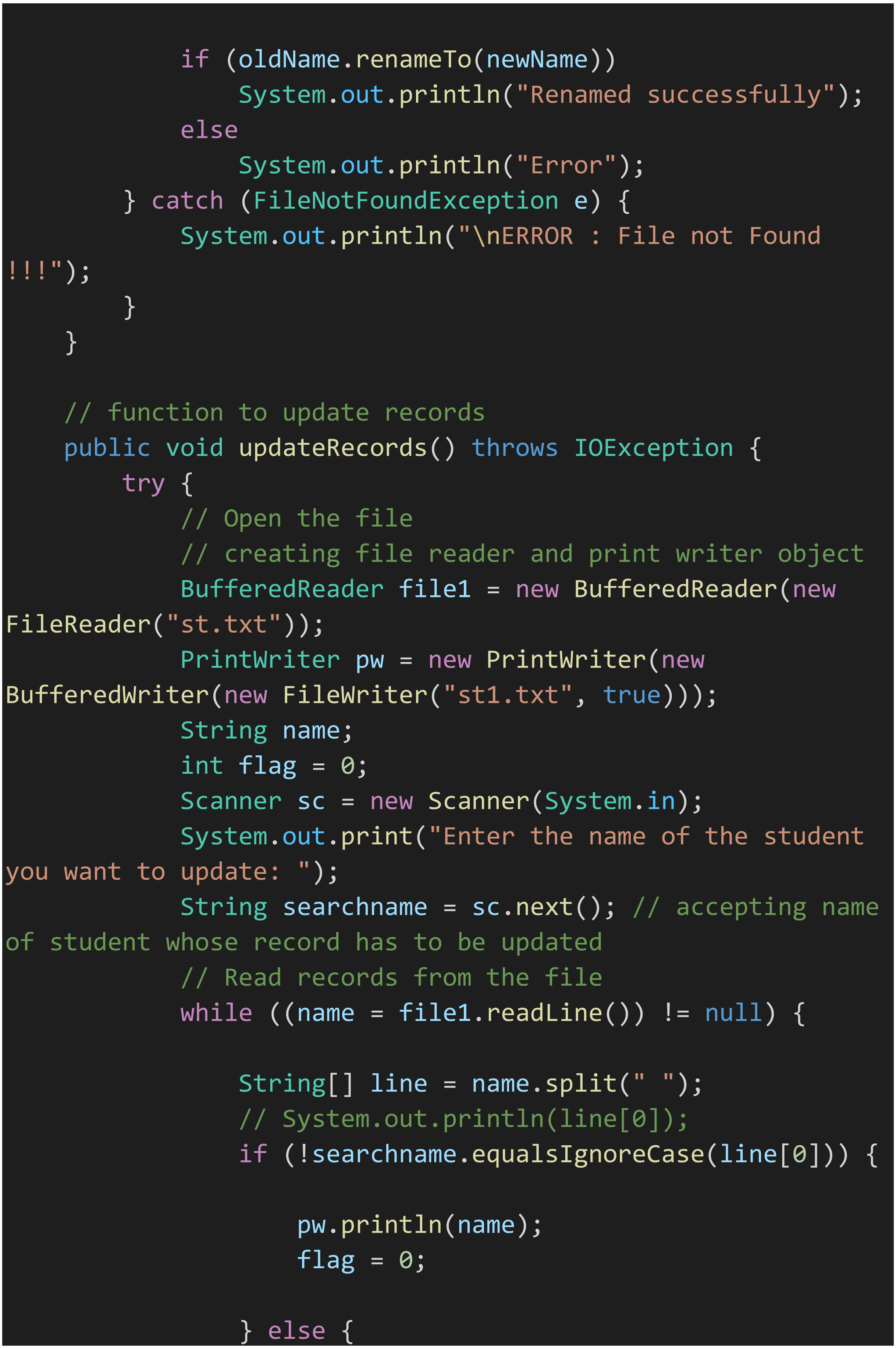
```
            s = br.readLine();
            if (s.equalsIgnoreCase("y")) { // continue
adding
                addMore = true;
                System.out.println();
            } else
                addMore = false;
        } while (addMore);
        pw.close();
    }

    public void readRecords() throws IOException {
        try {
            // Open the file
            BufferedReader file = new BufferedReader(new
FileReader("st.txt"));
            String name;
            // Read records from the file
            while ((name = file.readLine()) != null) {
                System.out.println(name);
                System.out.println("");

            }
            file.close();
        } catch (FileNotFoundException e) { // Executes if
file not found
            System.out.println("\nERROR : File not Found
!!!");
        }
    }

    // function to search records
    public void searchRecords() throws IOException {
        try {
            // Open the file
            BufferedReader file = new BufferedReader(new
FileReader("st.txt")); // creating object of buffered reader
            String name;
```



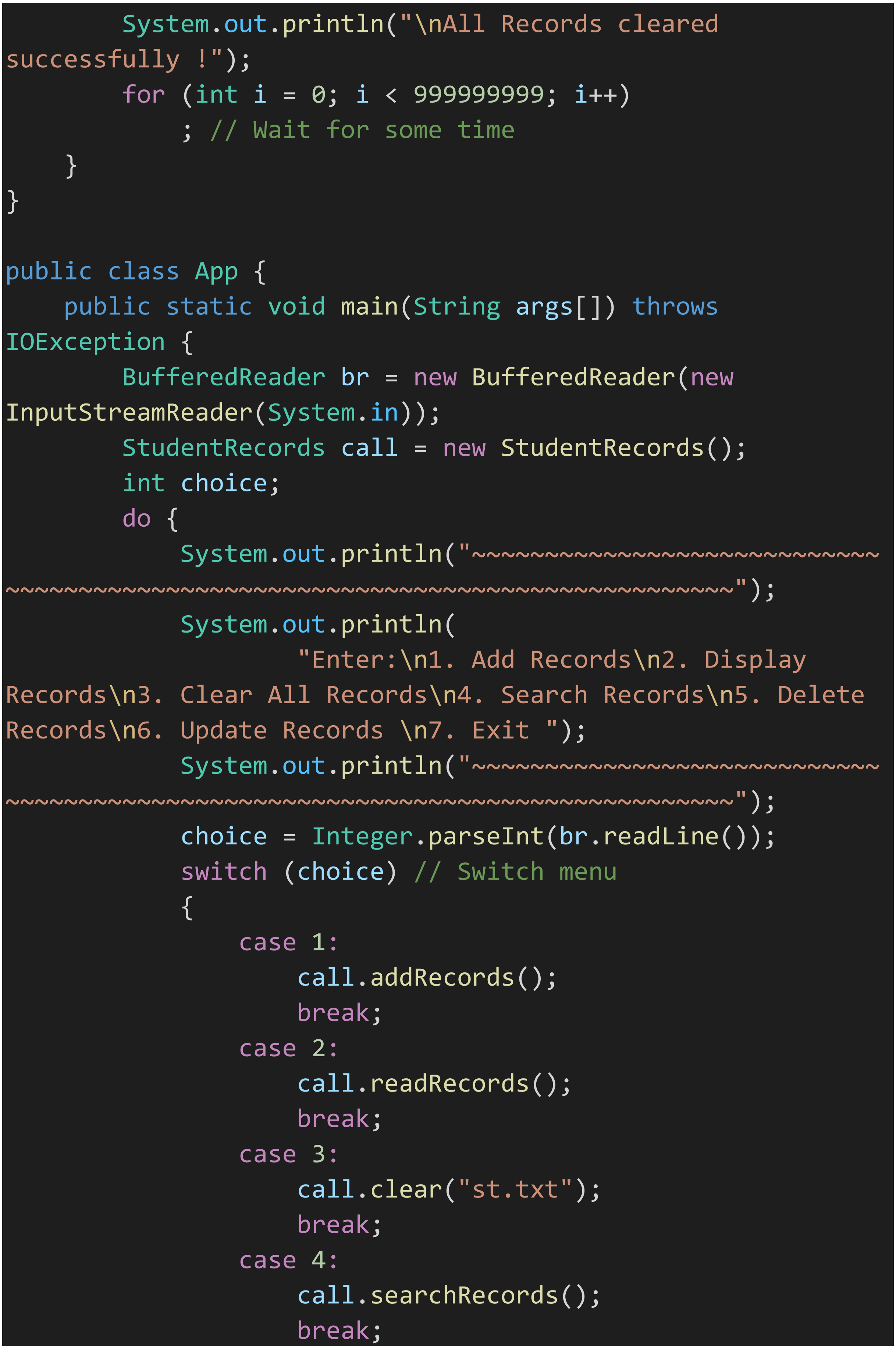
```
            int flag = 0;
            Scanner sc = new Scanner(System.in);
            System.out.print("Enter an id of the student you
want to search: ");
            String searchname = sc.next(); // taking input
for name of student to be searched for
            // Read records from the file
            while ((name = file.readLine()) != null) {

                String[] line = name.split(" ");
                // System.out.println(line[0]);
                if (searchname.equalsIgnoreCase(line[1])) {
                    System.out.println("Record found");
                    System.out.println(name);
                    System.out.println("");
                    flag = 1;
                    break;
                }

            }
            if (flag == 0)
                System.out.println("Record not found"); //
if record not found display the message to user
            file.close(); // closing file object
        } catch (FileNotFoundException e) { // Executes if
file not found
            System.out.println("\nERROR : File not Found
!!!");
        }
    }

    // function to delete records
    public void deleteRecords() throws IOException {
        try {
            // Open the file
            // creating buffered reader and printwriter
object
```



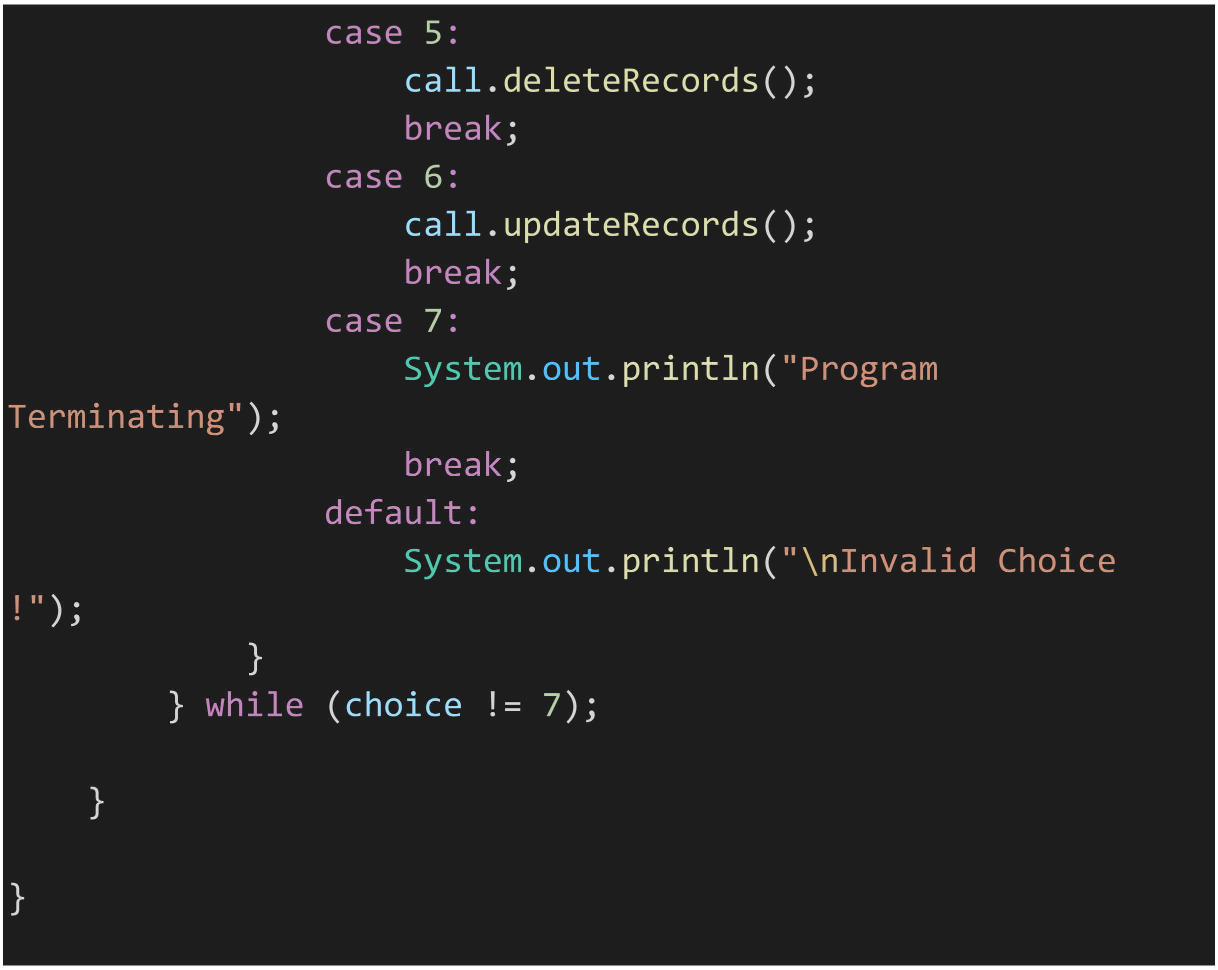
```
            BufferedReader file1 = new BufferedReader(new
FileReader("st.txt"));
            PrintWriter pw = new PrintWriter(new
BufferedWriter(new FileWriter("st1.txt", true)));
            String name;
            int flag = 0;
            Scanner sc = new Scanner(System.in);
            System.out.print("Enter the name of the student
you want to delete: ");
            String searchname = sc.next(); // taking input
from user for name to be deleted
            // Read records from the file
            while ((name = file1.readLine()) != null) {

                String[] line = name.split(" ");
                // System.out.println(line[0]);
                if (!searchname.equalsIgnoreCase(line[0])) {

                    pw.println(name);
                    flag = 0;

                } else {
                    System.out.println("Record found");
                    flag = 1;
                }

            }
            // closing the objects
            file1.close();
            pw.close();
            File delName = new File("st.txt");
            File oldName = new File("st1.txt");
            File newName = new File("st.txt");
            if (delName.delete()) // if record is deleted
successfully, display the message to user
                System.out.println("deleted successfully");
            else
                System.out.println("Error");
```



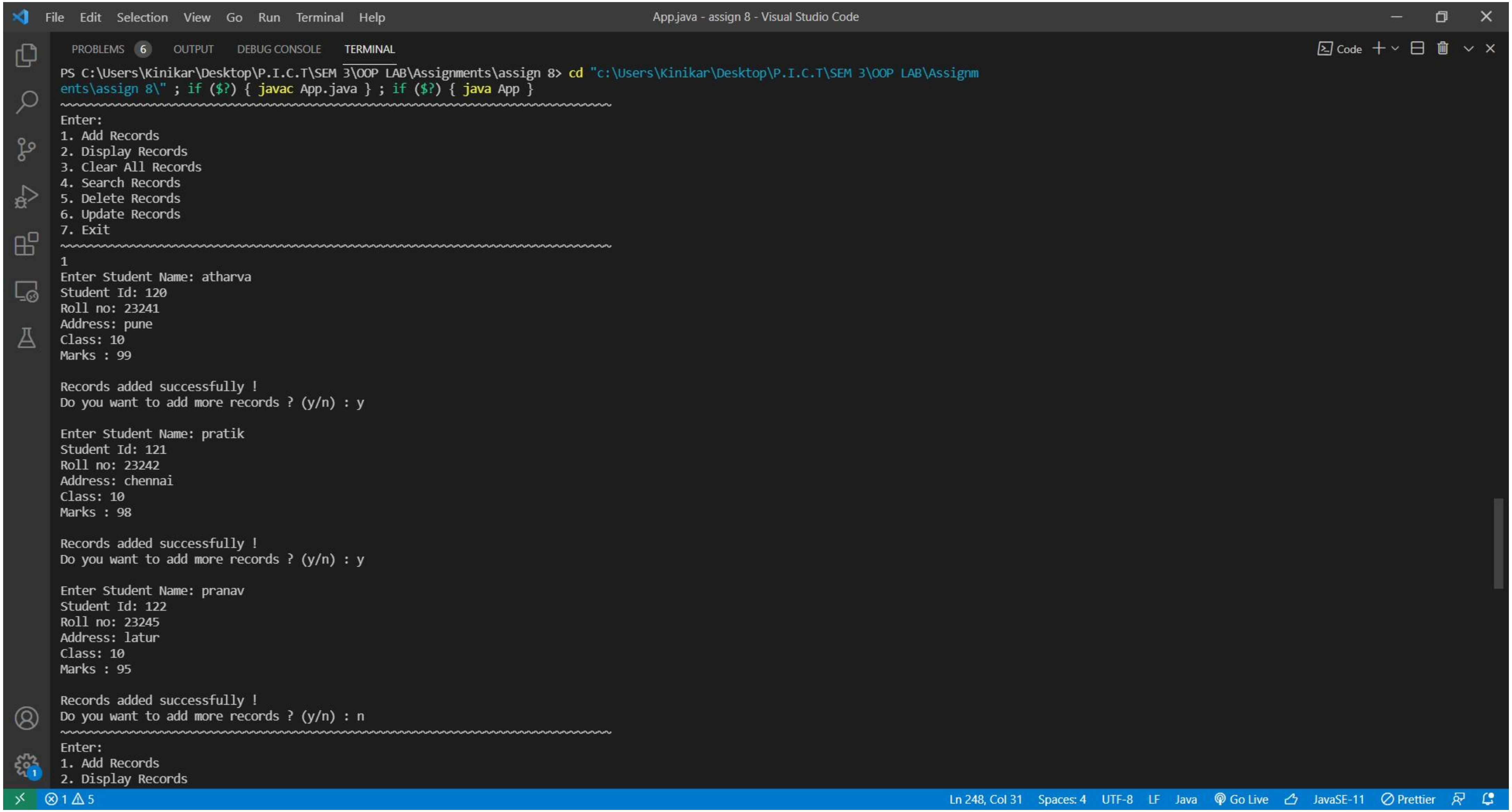
```
            if (oldName.renameTo(newName))
                System.out.println("Renamed successfully");
            else
                System.out.println("Error");
        } catch (FileNotFoundException e) {
            System.out.println("\nERROR : File not Found
!!!");
        }
    }

    // function to update records
    public void updateRecords() throws IOException {
        try {
            // Open the file
            // creating file reader and print writer object
            BufferedReader file1 = new BufferedReader(new
FileReader("st.txt"));
            PrintWriter pw = new PrintWriter(new
BufferedWriter(new FileWriter("st1.txt", true)));
            String name;
            int flag = 0;
            Scanner sc = new Scanner(System.in);
            System.out.print("Enter the name of the student
you want to update: ");
            String searchname = sc.next(); // accepting name
of student whose record has to be updated
            // Read records from the file
            while ((name = file1.readLine()) != null) {

                String[] line = name.split(" ");
                // System.out.println(line[0]);
                if (!searchname.equalsIgnoreCase(line[0])) {

                    pw.println(name);
                    flag = 0;

                } else {
                    System.out.println("Record found");
                    System.out.println("Enter updated
marks:");
                    String up_mark = sc.next();
                    // if record is found updating the marks
                    pw.println(line[0] + " " + line[1] + " "
+ line[2] + " " + line[3] + " " + line[4] + " " + up_mark);
                    flag = 1;
                }

            }
            // clsoign the objects
            file1.close();
            pw.close();
            File delName = new File("st.txt");
            File oldName = new File("st1.txt");
            File newName = new File("st.txt");
            if (delName.delete())
                System.out.println("record updated
successfully");
            else
                System.out.println("Error");

            if (oldName.renameTo(newName))
                System.out.println("Renamed successfully");
            else
                System.out.println("Error");
        } catch (FileNotFoundException e) {
            System.out.println("\nERROR : File not Found
!!!");
        }
    }

    public void clear(String filename) throws IOException {
        // Create a blank file
        PrintWriter pw = new PrintWriter(new
BufferedWriter(new FileWriter(filename)));
        pw.close();
```



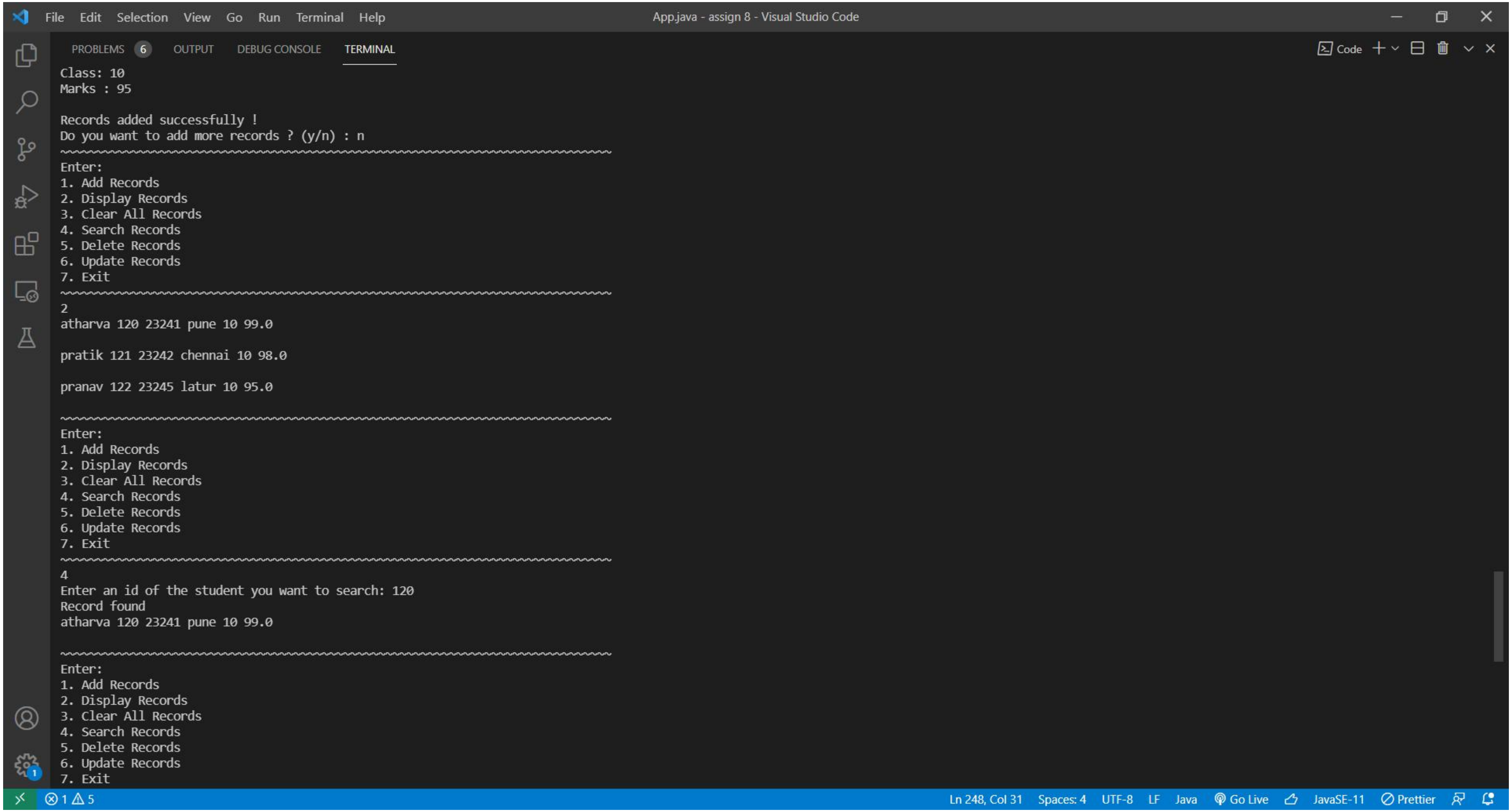
```java
        System.out.println("\nAll Records cleared 
successfully !");
        for (int i = 0; i < 999999999; i++)
            ; // Wait for some time
    }
}

public class App {
    public static void main(String args[]) throws 
IOException {
        BufferedReader br = new BufferedReader(new 
InputStreamReader(System.in));
        StudentRecords call = new StudentRecords();
        int choice;
        do {
            System.out.println("~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~");
            System.out.println(
                    "Enter:\n1. Add Records\n2. Display 
Records\n3. Clear All Records\n4. Search Records\n5. Delete 
Records\n6. Update Records \n7. Exit ");
            System.out.println("~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~");
            choice = Integer.parseInt(br.readLine());
            switch (choice) // Switch menu
            {
                case 1:
                    call.addRecords();
                    break;
                case 2:
                    call.readRecords();
                    break;
                case 3:
                    call.clear("st.txt");
                    break;
                case 4:
                    call.searchRecords();
                    break;
```



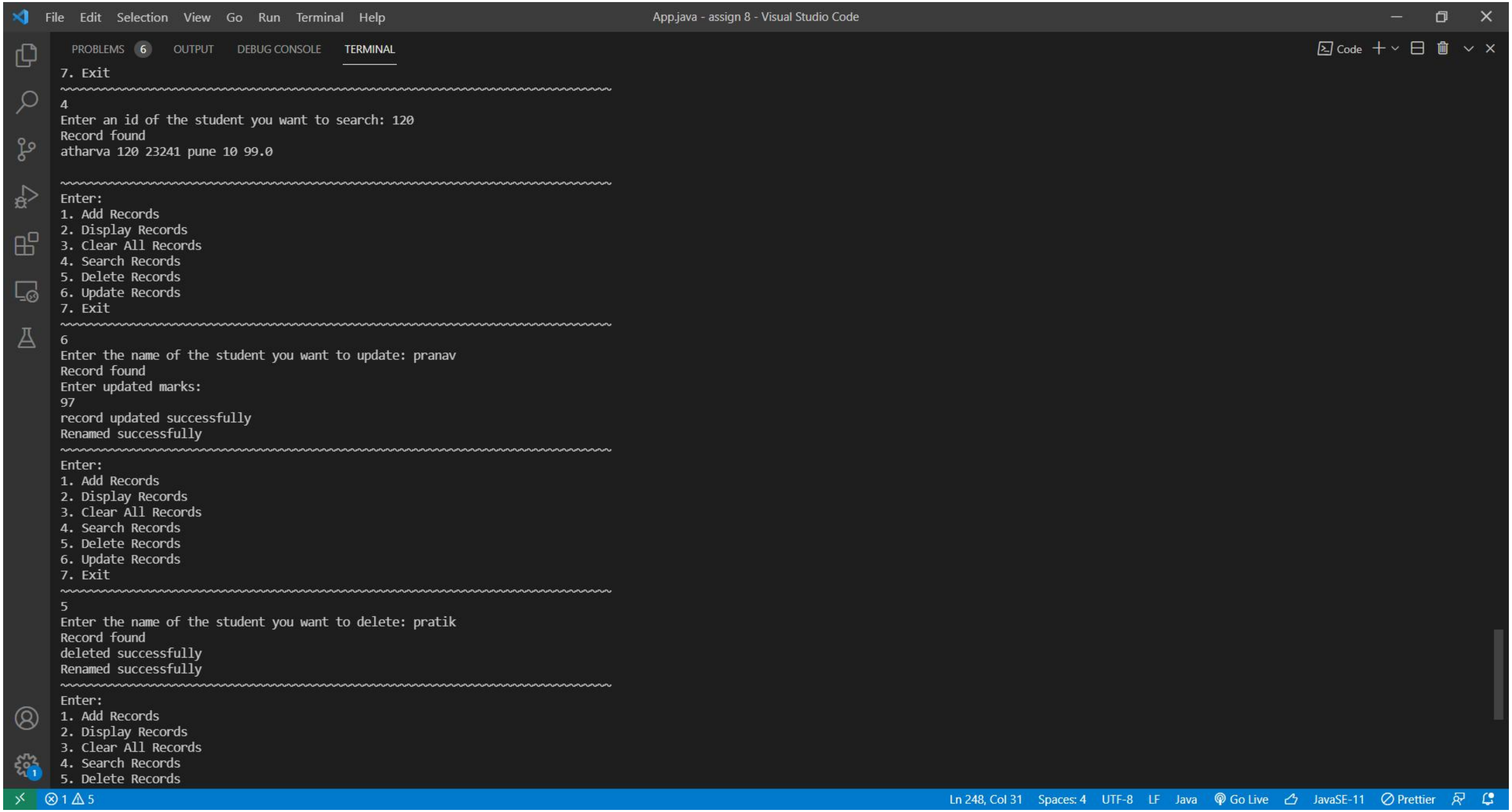
```
                case 5:
                    call.deleteRecords();
                    break;
                case 6:
                    call.updateRecords();
                    break;
                case 7:
                    System.out.println("Program 
Terminating");
                    break;
                default:
                    System.out.println("\nInvalid Choice 
!");
            }
        } while (choice != 7);

    }

}
```



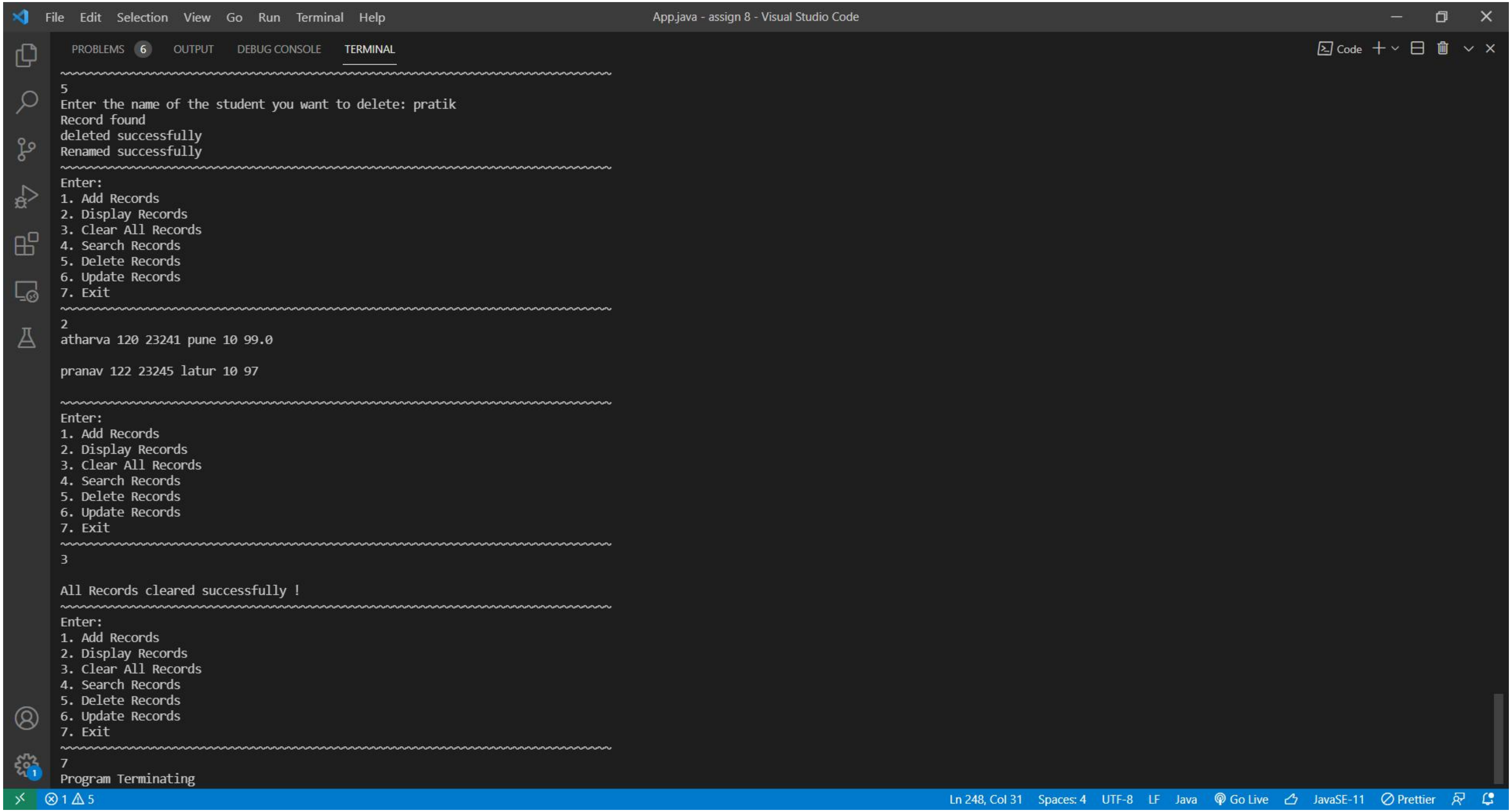
```
PS C:\Users\Kinikar\Desktop\P.I.C.T\SEM 3\OOP LAB\Assignments\assign 8> cd "c:\Users\Kinikar\Desktop\P.I.C.T\SEM 3\OOP LAB\Assignm
ents\assign 8\" ; if ($?) { javac App.java } ; if ($?) { java App }
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Enter:
1. Add Records
2. Display Records
3. Clear All Records
4. Search Records
5. Delete Records
6. Update Records
7. Exit
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
1
Enter Student Name: atharva
Student Id: 120
Roll no: 23241
Address: pune
Class: 10
Marks : 99

Records added successfully !
Do you want to add more records ? (y/n) : y

Enter Student Name: pratik
Student Id: 121
Roll no: 23242
Address: chennai
Class: 10
Marks : 98

Records added successfully !
Do you want to add more records ? (y/n) : y

Enter Student Name: pranav
Student Id: 122
Roll no: 23245
Address: latur
Class: 10
Marks : 95

Records added successfully !
Do you want to add more records ? (y/n) : n
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Enter:
1. Add Records
2. Display Records
```



```
Class: 10
Marks : 95

Records added successfully !
Do you want to add more records ? (y/n) : n
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Enter:
1. Add Records
2. Display Records
3. Clear All Records
4. Search Records
5. Delete Records
6. Update Records
7. Exit
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
2
atharva 120 23241 pune 10 99.0

pratik 121 23242 chennai 10 98.0

pranav 122 23245 latur 10 95.0

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Enter:
1. Add Records
2. Display Records
3. Clear All Records
4. Search Records
5. Delete Records
6. Update Records
7. Exit
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
4
Enter an id of the student you want to search: 120
Record found
atharva 120 23241 pune 10 99.0

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Enter:
1. Add Records
2. Display Records
3. Clear All Records
4. Search Records
5. Delete Records
6. Update Records
7. Exit
```


File Edit Selection View Go Run Terminal Help
App.java - assign 8 - Visual Studio Code
PROBLEMS 6
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
Ln 248, Col 31 Spaces: 4 UTF-8 LF Java Go Live JavaSE-11 Prettier
⊗ 1 ⚠ 5

```
7. Exit
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
4
Enter an id of the student you want to search: 120
Record found
atharva 120 23241 pune 10 99.0

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Enter:
1. Add Records
2. Display Records
3. Clear All Records
4. Search Records
5. Delete Records
6. Update Records
7. Exit
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
6
Enter the name of the student you want to update: pranav
Record found
Enter updated marks:
97
record updated successfully
Renamed successfully
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Enter:
1. Add Records
2. Display Records
3. Clear All Records
4. Search Records
5. Delete Records
6. Update Records
7. Exit
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
5
Enter the name of the student you want to delete: pratik
Record found
deleted successfully
Renamed successfully
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Enter:
1. Add Records
2. Display Records
3. Clear All Records
4. Search Records
5. Delete Records
```



```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
5
Enter the name of the student you want to delete: pratik
Record found
deleted successfully
Renamed successfully
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Enter:
1. Add Records
2. Display Records
3. Clear All Records
4. Search Records
5. Delete Records
6. Update Records
7. Exit
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
2
atharva 120 23241 pune 10 99.0

pranav 122 23245 latur 10 97

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Enter:
1. Add Records
2. Display Records
3. Clear All Records
4. Search Records
5. Delete Records
6. Update Records
7. Exit
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
3

All Records cleared successfully !
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Enter:
1. Add Records
2. Display Records
3. Clear All Records
4. Search Records
5. Delete Records
6. Update Records
7. Exit
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
7
Program Terminating
```