```
* Assignment No: 8
  Implement a program for maintaining a student records database using File Handling.
  Student has Student_id, name, Roll_no, Class, marks and address. Display the data
  for five  students.
*/
================================================================================

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


// ========================= FILE1 CLASS ========================//

class Database {
        static BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
        //creating bufferredReder class object

        // ---------------------- addRecords method ---------------------- //
        public void addRecords() throws IOException  {
                // Create or Modify a file for Database
                PrintWriter pw = new PrintWriter(new BufferedWriter(new
FileWriter("sample.txt",true)));
                //creating file with name sapmle.txt
                String studentname, address,s;//declaration of studentname , address ,s
                int studentid, rollno, Class;//declaration of studentid , rollno, Class
                float marks;//declaration of marks

                boolean addMore = false; //declaration of addmore
                do  {
                        System.out.print("\nEnter Student Name: "); //printing on console
                        studentname = br.readLine();  //taking input from user
                        System.out.print("Student Id: "); //printing on console
                        studentid = Integer.parseInt(br.readLine()); //taking input from user
                        System.out.print("Roll no: ");//printing on console
                        rollno = Integer.parseInt(br.readLine());  //taking input from user
                        System.out.print("Address: ");  //printing on console
                        address = br.readLine();  //taking input from user
                        System.out.print("Class: ");//printing on console
                        Class = Integer.parseInt(br.readLine());  //taking input from user
                        System.out.print("Marks : ");  //printing on console
                        marks = Float.parseFloat(br.readLine()); //taking input from user
                         pw.println(studentname+" "+studentid+" "+rollno+" "+address+" "+Class+"
"+marks);
                         //appending data into to file
```

```
                        System.out.print("\nRecords added successfully !\n\nDo you want to add
more records ? (y/n) : ");
                        s = br.readLine();//take input from user
                        if(s.equalsIgnoreCase("y")){
                                addMore = true;//modify addmore
                                System.out.println();
                                }
                        else
                                addMore = false; //modify addmore
                }
                while(addMore);
                 pw.close();
                }

        // ---------------------- addRecords method ---------------------- //
        public void readRecords() throws IOException  {
                try  {
                        // Open the file
                        BufferedReader file = new BufferedReader(new   FileReader("sample.txt"));
                        String name; //declaration of string name
                        int i=1;  //intizing value of i=1

                        // Read records from the file
                        while((name = file.readLine()) != null)   {
                                System.out.println(name); //printing on console
                                System.out.println("");
                                }   file.close();
                }
                catch(FileNotFoundException e){ //Exception handling
                        System.out.println("\nERROR : File not Found !!!");  //printing on console
                }
        }

        // ---------------------- addRecords method ---------------------- //
        public void searchRecords() throws IOException  {
                try  {   // Open the file
                        BufferedReader file = new BufferedReader(new   FileReader("sample.txt"));
                        String name;//declaration of string name
                        int flag=0; //intizing value of flag=0
                        Scanner sc=new Scanner(System.in); //creating obj of scanner class
                        System.out.print("Enter an id of the student you want to search: ");
                        //printing on console
                        String searchname=sc.next(); //taking input from user
                        // Read records from the file
```

```
                    while((name = file.readLine()) != null)  {
                            String[] line = name.split(" ");

                            if(searchname.equalsIgnoreCase(line[1])){
                                    System.out.println("Record found");    //printing on console
                                    System.out.println(name);   //printing record on console
                                    System.out.println("");
                                    flag=1;   //modify value
                                    break;
                                    }
                            }
                    if(flag==0)     //check condition
                            System.out.println("Record not found"); //printing on console
                    file.close();  //closing file
                    }
            catch(FileNotFoundException e)  {//Exception handling
                    System.out.println("\nERROR : File not Found !!!");//printing on console
                    }
            }

        // ---------------------- addRecords method ---------------------- //
        public void deleteRecords() throws IOException  {
                try  {  // Open the file
                        BufferedReader file1 = new BufferedReader(new FileReader("sample.txt"));
                        PrintWriter pw = new PrintWriter(new BufferedWriter(new
FileWriter("new.txt",true)));
                        String name;  //declaration of string name
                        int flag=0; //intizing value of flag=0
                        Scanner sc=new Scanner(System.in); //creating obj of scanner class
                        System.out.print("Enter the name of the student you want to delete: ");
                        String searchname=sc.next();  // Read records from the file
                        while((name = file1.readLine()) != null)  {
                                String[] line = name.split(" ");
                                if(!searchname.equalsIgnoreCase(line[0])){
                                        pw.println(name);
                                        flag=0; //modify value
                                }
                                else{
                                        System.out.println("Record found");  //printing on console
                                        flag=1;//modify value
                                }
                                }  file1.close();//closing file
                                pw.close();
```

```
                        File delName =  new File("sample.txt");//creating obj of sample.txt
                        File oldName =  new File("new.txt");   //creating obj of new.txt
                        File newName =   new File("sample.txt");  //creating obj of sample.txt

                        if(delName.delete())
                                System.out.println("deleted successfully");  //printing on
console
                        else
                                System.out.println("Error");//printing on console

                        if (oldName.renameTo(newName))
                                System.out.println("Renamed successfully");  //printing on
console
                        else
                                System.out.println("Error"); //printing on console

                }
                catch(FileNotFoundException e)  {//Exception handling
                        System.out.println("\nERROR : File not Found !!!");
                        }
                }

        // ---------------------- addRecords method ---------------------- //
        public void updateRecords() throws IOException  {
                try  {
                        // Open the file
                        BufferedReader file1 = new BufferedReader(new FileReader("sample.txt"));
                        PrintWriter pw = new PrintWriter(new BufferedWriter(new
FileWriter("new.txt",true)));
                        String name;//declaration of string name
                        int flag=0;   //intizing flag to 0
                        Scanner sc=new Scanner(System.in);  //creating obje of scanner class
                        System.out.print("Enter the name of the student you want to update: ");
//printing on console
                        String searchname=sc.next();   // Read records from the file

                        while((name = file1.readLine()) != null)   {  //check condition
                                String[] line = name.split(" ");

                                if(!searchname.equalsIgnoreCase(line[0])){  //check condition
                                        pw.println(name);
                                        flag=0;  //modify value of flag
                                        }
                                else
```

```
                                        {
                                        System.out.println("Record found");  //printing on console
                                        System.out.print("Enter updated marks: ");  //printing on
console
                                        String up_mark=sc.next();    //taking input from user
                                        pw.println(line[0]+" "+line[1]+" "+line[2]+" "+line[3]+"
"+line[4]+" "+up_mark);
                                        flag=1;    //modify value of flag
                                        }
                                }
                        file1.close(); //closing file
                                pw.close();
                                File delName =  new File("sample.txt");//creating obj of sample.txt
                                File oldName =  new File("new.txt");   //creating obj of new.txt
                                File newName =   new File("sample.txt");  //creating obj of sample.txt

                                if(delName.delete())        //check condition
                                        System.out.println("record updated successfully");   //printing
on console
                                else
                                        System.out.println("Error"); //printing on console

                                if (oldName.renameTo(newName))  //check condition
                                        System.out.println("Renamed successfully");    //printing on
console
                                else
                                        System.out.println("Error");    //printing on console

                                }
                catch(FileNotFoundException e)  {  //Exception handling
                        System.out.println("\nERROR : File not Found !!!"); //printing on console
                        }
                }

        // ---------------------- addRecords method ---------------------- //
        public void clear(String filename) throws IOException  {
                // Create a blank file
                PrintWriter pw = new PrintWriter(new BufferedWriter(new FileWriter(filename)));
                pw.close(); //closing PrintWriter object
                System.out.println("\nAll Records cleared successfully !");
                //printing on console

                }
```

```
}

// ======================== MAIN CLASS ========================//
public class Maulidemoprg8
{
	public static void main(String args[]) throws IOException  {
		Database f = new Database(); //creating obj of Database class
		Scanner sc =new Scanner(System.in);//creating object of scanner class
		System.out.println("");
		while(true) {
			//menu driven
		System.out.print("1. Add Records\n2. Display Records\n3. Clear All Records\n4.
Search Records"
				+ "\n5. Delete Records\n6. Update Records \n7. Exit\n\nEnter your
choice : ");
		int choice = sc.nextInt();//taking input from user
		System.out.println("");

		//switch Case
		switch(choice)  {
		case 1:
			f.addRecords();  //calling addRecords method

	System.out.println("\n=================================================\n");
			break;

		case 2:
			f.readRecords();   //calling readRecords method

	System.out.println("\n=================================================\n");
			break;

		case 3:
			f.clear("sample.txt"); //calling clear method

	System.out.println("\n=================================================\n");
			break;

		case 4:
			f.searchRecords(); //calling searchRecords method

	System.out.println("\n=================================================\n");
			break;
```

```
                case 5:
                        f.deleteRecords();//calling deleteRecords method

        System.out.println("\n=====================================================\n");
                        break;

                case 6:
                        f.updateRecords(); //calling updateRecords method

        System.out.println("\n=====================================================\n");
                        break;

                case 7:

        System.out.println("\n=====================================================\n");
                        System.exit(0);//stop execution of program
                        break;

                default:
                        System.out.println("\nInvalid Choice !"); //default case

        System.out.println("\n=====================================================\n");
                        break;
                        }
                }

        }

}
```

**OUTPUT:**

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

Enter your choice : 1


Enter Student Name: mauli
```

**Student Id: 1010**
**Roll no: 68**
**Address: ap-jalna**
**Class: 1**
**Marks : 90**

**Records added successfully !**

**Do you want to add more records ? (y/n) : y**

**Enter Student Name: karan**
**Student Id: 2020**
**Roll no: 67**
**Address: ap-jalna**
**Class: 2**
**Marks : 90**

**Records added successfully !**

**Do you want to add more records ? (y/n) : n**

**==========**

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

**Enter your choice : 2**

**mauli 1010 68 ap-jalna 1 90.0**

**karan 2020 67 ap-jalna 2 90.0**

**==========**

**1. Add Records**
**2. Display Records**
**3. Clear All Records**

**4. Search Records**
**5. Delete Records**
**6. Update Records**
**7. Exit**

**Enter your choice : 4**

**Enter an id of the student you want to search: 2020**
**Record found**
**karan 2020 67 ap-jalna 2 90.0**

**==========**

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

**Enter your choice : 6**

**Enter the name of the student you want to update: mauli**
**Record found**
**Enter updated marks: 95**
**record updated successfully**
**Renamed successfully**

**==========**

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

**Enter your choice : 3**

**All Records cleared successfully !**

**==========**

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

**Enter your choice : 5**

**Enter the name of the student you want to delete: 1010**
**deleted successfully**
**Renamed successfully**

**==========**

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

**Enter your choice : 7**

**==========**