

COSC 223 FILE HANDLING ASSIGNMENT

Group Members:

1. Victor Mutugi Kathenya – EB1/56061/21
2. Agnes Mabeya -EB1/56047/21
3. Peter Maina – EB1/42996/19
4. Muchiri Timothy Maina– EB1/56037/21
5. Collins Mwaura Wambui-EB1/56094/21
6. James Maureen Kambua -EB1/56648/21
7. Purity Nafula – EB1/56079/21
8. Leah Achieng' – EB1/56068/21
9. Phoelix Otieno Mboya – EB1/56138/21
10. Felix Kibet Rotich – EB1/56106/21

```
1  import java.io.*;
2  import java.util.*;
3
4  public class Student {
5      private String regNo;
6      private String name;
7      private String program;
8      private String dateOfBirth;
9
10     //constructor for Student
11     public Student(String regNo, String name, String program, String dateOfBirth) {
12         this.regNo = regNo;
13         this.name = name;
14         this.program = program;
15         this.dateOfBirth = dateOfBirth;
16     }
17
18     public String getRegNo() {
19         return regNo;
20     }
21
22     public String getName() {
23         return name;
24     }
25
26     public String getProgram() {
27         return program;
28     }
29
30     public String getDateOfBirth() {
31         return dateOfBirth;
32     }
33
34     public String toString() {
35         return regNo + ", " + name + ", " + program + ", " + dateOfBirth;
36     } //overriding of the getObject() and toString() methods
37
38     public static void main(String[] args) {
39         Scanner input = new Scanner(System.in);
40
41         System.out.println("Enter student's registration no:");
42         String regNo = input.nextLine();
43
44         System.out.println("Enter student's name:");
45         String name = input.nextLine();
46
47         System.out.println("Enter student's program:");
48         String program = input.nextLine();
49
50         System.out.println("Enter student's date of birth (dd.mm.yyyy):");
51         String dateOfBirth = input.nextLine();
```

```

1      //creating an object Student with the inputs as the arguments and save to file
2      Student student = new Student(regNo, name, program, dateOfBirth);
3      writeToFile("student.txt", student);
4
5      System.out.println("Student's details saved");
6
7      //search for students taking a particular program
8      System.out.print("Enter program to search: ");
9      String searchProgram = input.nextLine();
10
11     List<Student> students = readFromFile("student.txt");
12     ArrayList<Student> matchingStudents = new ArrayList<Student>();
13     for (Student s : students) {
14         if (s.getProgram().equalsIgnoreCase(searchProgram)) {
15             matchingStudents.add(s);
16         }
17     }
18
19     System.out.println("Students taking " + searchProgram + ":");
20     for (Student s : matchingStudents) {
21         System.out.println(s);
22     }
23 }
24
25 //this method takes the arguments and writes to file using the filewriter
26 //catches an exception if error occurs
27 public static void writeToFile(String fileName, Student student) {
28     try {
29         FileWriter writer = new FileWriter(fileName, true);
30         writer.write(student.toString() + "\n");
31         writer.close();
32     } catch (IOException e) {
33         System.out.println("An error occurred while writing.");
34         e.printStackTrace();
35     }
36 }
37
38 //the readFromFile() method takes file name as parameter and reads the details of student
39 public static List<Student> readFromFile(String fileName) {
40     List<Student> students = new ArrayList<Student>();
41     try {
42         BufferedReader reader = new BufferedReader(new FileReader(fileName));
43         String line;
44         while ((line = reader.readLine()) != null) {
45             String[] parts = line.split(",");
46             String regNo = parts[0].trim();
47             String name = parts[1].trim();
48             String program = parts[2].trim();
49             String dateOfBirth = parts[3].trim();
50             Student student = new Student(regNo, name, program, dateOfBirth);
51             students.add(student);
52         }
53         reader.close();
54     } catch (IOException e) {
55         System.out.println("An error occurred while reading from file.");
56         e.printStackTrace();
57     }
58     return students;
59 }
60 }
61

```

```
Student.java x student.txt x Object.java x
1 EB1.56094.21, Collins Mwaura, Computer Science, 01.01.2002
2 EB1.56061.21, Victor Mutugi, Computer Science, 1.1.2002
3 EB3.57036.21, Jane Doe, Applied Computer Science, 29.02.2000
4 EB7.46723.20, Michelle Monroe, Maths and Stat, 11.09.2001
5 ASC1.24342.17, Lana Del Ray, Art and Music, 12.12.1992
6 |
```

```
import java.io.*;

import java.util.*;

public class Student {
    private String regNo;
    private String name;
    private String program;
    private String dateOfBirth;

    //constructor for Student
    public Student(String regNo, String name, String program, String
dateOfBirth) {
        this.regNo = regNo;
        this.name = name;
        this.program = program;
        this.dateOfBirth = dateOfBirth;
    }

    public String getRegNo() {
        return regNo;
    }

    public String getName() {
        return name;
    }

    public String getProgram() {
        return program;
    }

    public String getDateOfBirth() {
        return dateOfBirth;
    }

    public String toString() {
        return regNo + ", " + name + ", " + program + ", " + dateOfBirth;
    } //overriding of the getObject() and toString() methods

    public static void main(String[] args) {
```

```

Scanner input = new Scanner(System.in);

System.out.println("Enter student's registration no:");
String regNo = input.nextLine();

System.out.println("Enter student's name:");
String name = input.nextLine();

System.out.println("Enter student's program:");
String program = input.nextLine();

System.out.println("Enter student's date of birth (dd.mm.yyyy):");
String dateOfBirth = input.nextLine();

//creating an object Student with the inputs as the arguments and save
to file
Student student = new Student(regNo, name, program, dateOfBirth);
writeToFile("student.txt", student);

System.out.println("Student's details saved");

//search for students taking a particular program
System.out.print("Enter program to search: ");
String searchProgram = input.nextLine();

List<Student> students = readFromFile("student.txt");
ArrayList<Student> matchingStudents = new ArrayList<Student>();
for (Student s : students) {
    if (s.getProgram().equalsIgnoreCase(searchProgram)) {
        matchingStudents.add(s);
    }
}

System.out.println("Students taking " + searchProgram + ":");
for (Student s : matchingStudents) {
    System.out.println(s);
}
}

//this method takes the arguments and writes to file using the filewriter
//catches an exception if error occurs
public static void writeToFile(String fileName, Student student) {
    try {
        FileWriter writer = new FileWriter(fileName, true);
        writer.write(student.toString() + "\n");
        writer.close();
    } catch (IOException e) {
        System.out.println("An error occurred while writing.");
    }
}

```

```

        e.printStackTrace();
    }
}

//the readFromFile() method takes file name as parameter and reads the
details of student
public static List<Student> readFromFile(String fileName) {
    List<Student> students = new ArrayList<Student>();
    try {
        BufferedReader reader = new BufferedReader(new
FileReader(fileName));
        String line;
        while ((line = reader.readLine()) != null) {
            String[] parts = line.split(",");
            String regNo = parts[0].trim();
            String name = parts[1].trim();
            String program = parts[2].trim();
            String dateOfBirth = parts[3].trim();
            Student student = new Student(regNo, name, program,
dateOfBirth);
            students.add(student);
        }
        reader.close();
    } catch (IOException e) {
        System.out.println("An error occurred while reading from file.");
        e.printStackTrace();
    }
    return students;
}
}

```