

**University of Rajshahi**  
Department of Computer Science and Engineering  
Part- II, Semester – Odd, Final Exam – 2019  
Course: CSE2142 (Software Development Lab)

Download the following Project or StudentList.java and students.txt data file from the following link

<https://github.com/omarfaruge/CSE2142SDLEExam2019>

if you have ssh configured

git clone [git@github.com:omarfaruge/CSE2142SDLEExam2019.git](https://github.com/omarfaruge/CSE2142SDLEExam2019.git)

if you don't have ssh configured, use https instead

git clone <https://github.com/omarfaruge/CSE2142SDLEExam2019.git>

or if you don't have internet connection, please write the following code.

```
//File Name StudentList.java
import java.io.*;
import java.text.*;
import java.util.*;

public class StudentList {
public static void main(String[] args) {
//      Check arguments
    if(args[0].equals("a")) {
        System.out.println("Loading data ...");
        try {
            BufferedReader s = new BufferedReader(
                new InputStreamReader(
                    new FileInputStream("students.txt")));
            String r = s.readLine(); String i[] = r.split(",");

            for(String j : i) { System.out.println(j); }
        } catch (Exception e){}
        System.out.println("Data Loaded.");
    }
    else if(args[0].equals("r"))
    {
        System.out.println("Loading data ...");
        try {
            BufferedReader s = new BufferedReader(
                new InputStreamReader(
                    new
FileInputStream("students.txt")));
            String r = s.readLine(); System.out.println(r);
            String i[] = r.split(",");
```

```

Random x = new Random();
    int y = x.nextInt();
    System.out.println(i[y]);
} catch (Exception e){}
System.out.println("Data Loaded.");
}

else if(args[0].contains("+")){
    System.out.println("Loading data ...");
    try {
        BufferedWriter s = new BufferedWriter(
            new FileWriter("students.txt", true));
        String t = args[0].substring(1);
        Date d = new Date();
        String df = "dd/mm/yyyy-hh:mm:ss a";
        DateFormat dateFormat = new SimpleDateFormat(df);
        String fd= dateFormat.format(d);
        s.write(", "+t+"\nList last updated on "+fd);
        s.close();
    } catch (Exception e){}

    System.out.println("Data Loaded.");
}

else if(args[0].contains("?"))
{
    System.out.println("Loading data ...");
    try {
        BufferedReader s = new BufferedReader(
            new
FileInputStream("students.txt"));
        String r = s.readLine();
        String i[] = r.split(",");
        boolean done = false;
        String t = args[0].substring(1);
        for(int idx = 0; idx<i.length && !done; idx++) {
            if(i[idx].equals(t)) {
                System.out.println("We found it!");
                done=true;
            }
        }
    }
}

```

```

    }
    } catch (Exception e){}
    System.out.println("Data Loaded.");
}
else if(args[0].contains("c"))
{
    System.out.println("Loading data ...");
    try {
        BufferedReader s = new BufferedReader(
            new InputStreamReader(
                new
FileInputStream("students.txt")));
        String D = s.readLine();
        char a[] = D.toCharArray();
        boolean in_word = false;
        int count=0;
        for(char c:a) {
            if(c == ' ')
            {
                if (!in_word) { count++; in_word =true; }
                else { in_word=false;}
            }
        }
        System.out.println(count + " word(s) found " +
a.length);
    } catch (Exception e){}
    System.out.println("Data Loaded.");
}
}
}

```

## Data File name

students.txt

## File Contents (Initial Stage)

Student1, Student2, Student3, Student4

Data file and java file should be in same location.

**Steps #0** (Initial Stage as in question. Check the output whether it matches with the output given here. If your output doesn't match, please correct your program so that it produce exactly the same output given here.)

**Run #1** `$java StudentList a`

Loading data ...

Student1

Student2

Student3

Student4

Data Loaded.

**Run #2** `$java StudentList r`

Loading data ...

Student3

Data Loaded.

**Run #3** `$java StudentList r`

Loading data ...

Student1

Data Loaded.

**Run #4** `$java StudentList c`

Loading data ...

2 word(s) found

Data Loaded.

**Run #5** `$java StudentList ?Student1`

Loading data ...

We found it!

Data Loaded.

**Run #6** `$java StudentList +Another`

Loading data ...

Data Loaded.

## File Contents (After running)

Student1, Student2, Student3,

Student4 ,Another

List last updated on 2019-07-25

2:10:58 PM

In case you write the code by yourself instead of downloading, you need to initialize a git repository on your project. You should create git branch for every step with the name of the step, work on that particular step, commit the code with appropriate commit message and finally merge your branch to the master. For better understanding, write the commit message same as the task on each steps. And check every time you change something, you didn't break anything by going through Run #1 to Run #6 and it matches the output.

STEP #1. Update code style for better consistency.

STEP #2. Application now terminates early if the number of arguments passed into it is wrong, fix it.

STEP #3. Makes improvements to variable names

STEP #4. Refactors duplicate file read and write logic into methods

STEP #5. Replaces string literals with constants, storing those constants in a new class called Constants.java

STEP #6. Remove Temporary variables

STEP #7. Eliminates the 'done' control-flow variable. Adds better response for search operation.

STEP #8. Simplifies the logic behind the count operation

STEP #9. Adds handling for case when user enters invalid arguments

STEP #10. Add more comments and makes more naming improvements.

### Grading Rubric

SL#	Criterion (Score 0 if element is absent)	Below Expectations (1)	Meets Expectations (2)	Exceeds Expectations (3)	Score
1.	Able to create java project in any modern IDE like eclipse, IntelliJ, etc.	Don't know how to create java project	Java project created and run hello world program	Rearrange project structure and created appropriate packages	
2.	Initialize git repository and add, commit, etc.	Don't know how to initialize git repo	Added all files to git repo and commit them with appropriate commit message	Use terminal to do the task	
3.	Step #0	Program is not running as instruction	Program is running and producing same output as in question	Corrected some errors and mistakes in Run #6, if you run it multiple time	
4.	Step #1	Don't know how to create git branch	Created git branch and update program as instructed	Merged branch to master without errors or conflicts	
5.	Step #2	Don't know how to create git branch	Created git branch and update program as instructed	Merged branch to master without errors or conflicts	
6.	Step #3	Don't know how to create git branch	Created git branch and update program as instructed	Merged branch to master without errors or conflicts	
7.	Step #4	Don't know how to create git branch	Created git branch and update program as instructed	Merged branch to master without errors or conflicts	
8.	Step #5	Don't know how to create git branch	Created git branch and update program as instructed	Merged branch to master without errors or conflicts	
9.	Step #6	Don't know how to create git branch	Created git branch and update program as instructed	Merged branch to master without errors or conflicts	
10.	Step #7	Don't know how to create git branch	Created git branch and update program as instructed	Merged branch to master without errors or conflicts	
11.	Step #8	Don't know how to create git branch	Created git branch and update program as instructed	Merged branch to master without errors or conflicts	
12.	Step #9	Don't know how to create git branch	Created git branch and update program as instructed	Merged branch to master without errors or conflicts	

13.	Step #10	Don't know how to create git branch	Created git branch and update program as instructed	Merged branch to master without errors or conflicts	
14.	Visualize git log	Don't know how to do that	Visualize on paper with branch, commit, commit message, merge, workflow	Use third party app like smartgit, gitkraken, etc.	
<b>TOTAL</b>					