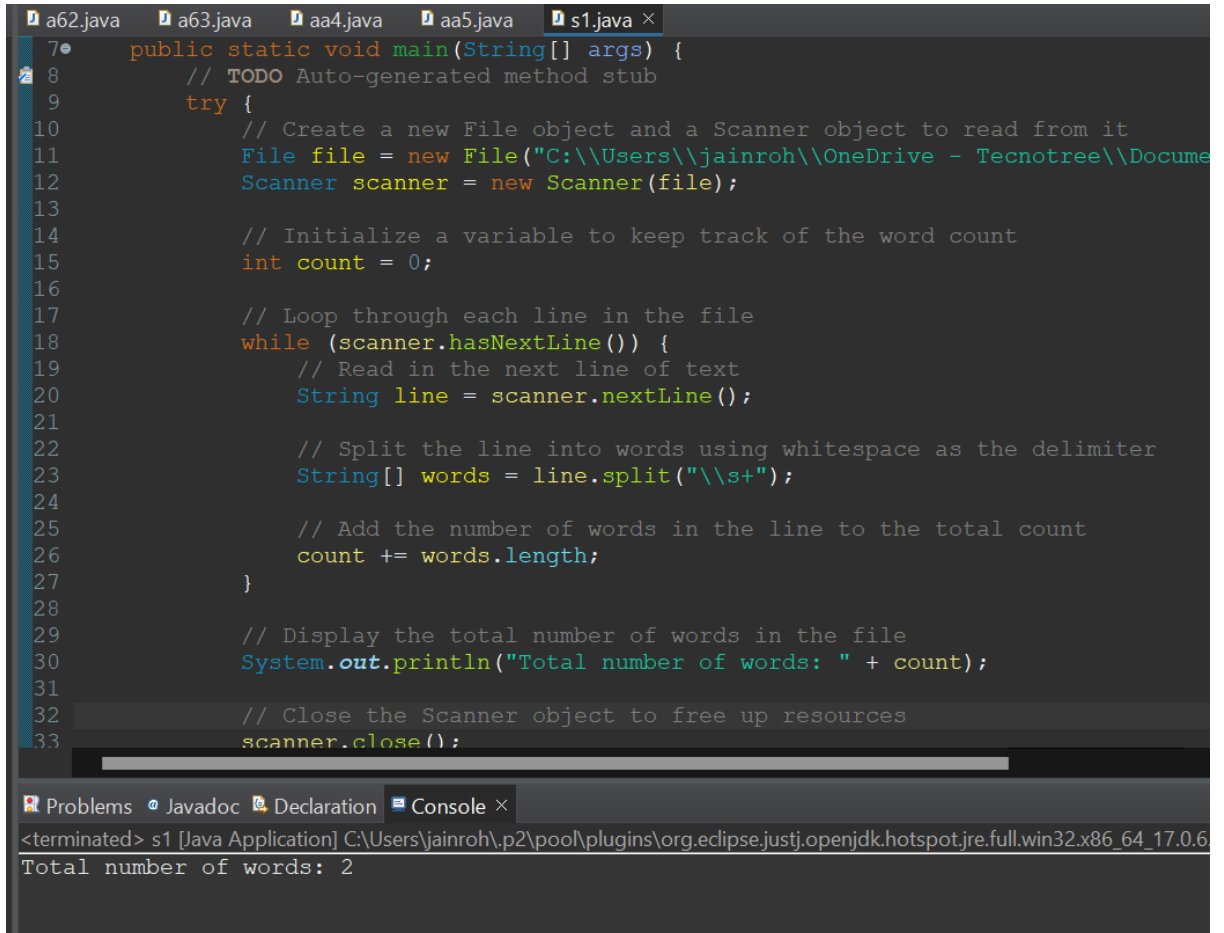# Assignment 7:

## Name : Rohan Jain T B

**1)** Create a program that reads in a text file and counts the number of words in the file. The program should display the total number of words at the end.
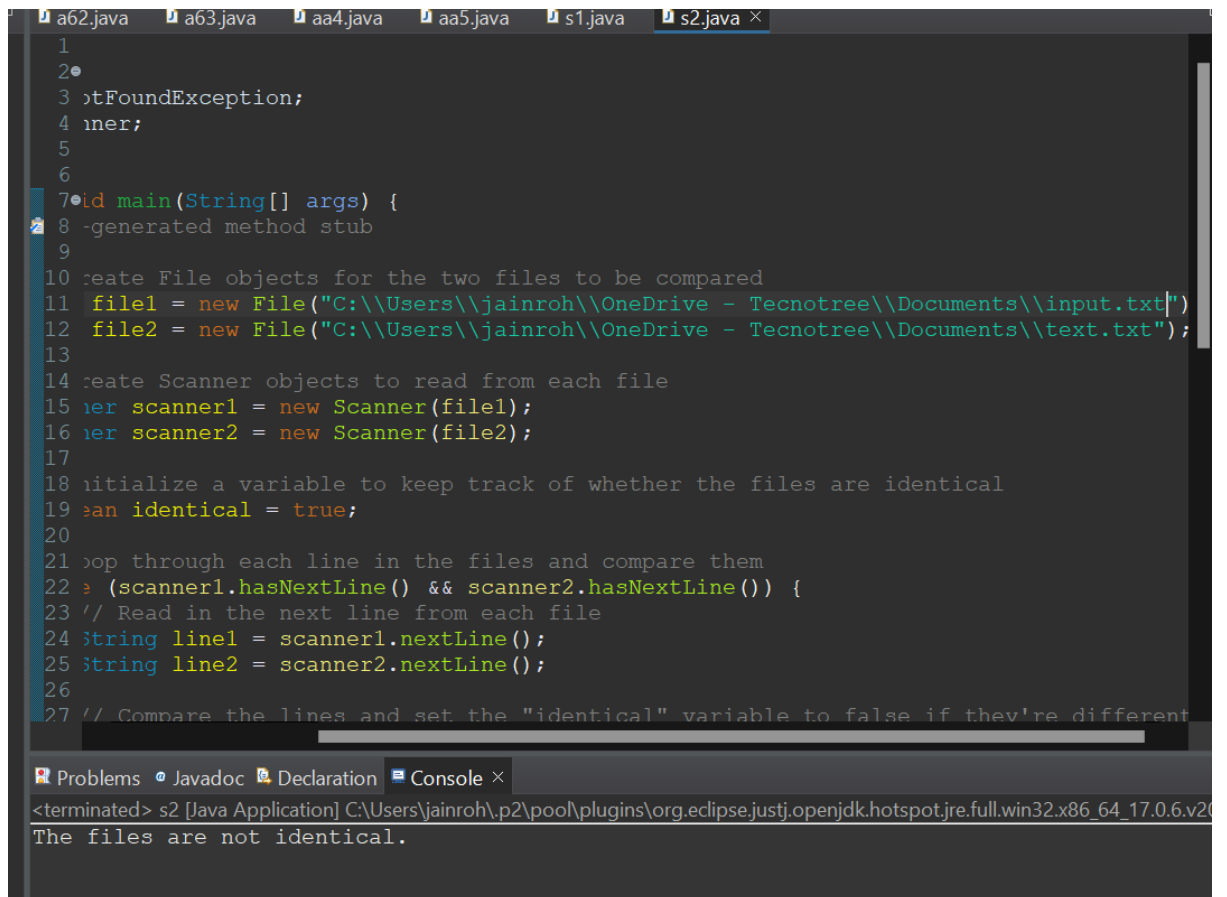
https://codeshare.io/JbMNr6

```java
public static void main(String[] args) {
    // TODO Auto-generated method stub
    try {
        // Create a new File object and a Scanner object to read from it
        File file = new File("C:\\Users\\jainroh\\OneDrive - Tecnotree\\Docume
        Scanner scanner = new Scanner(file);

        // Initialize a variable to keep track of the word count
        int count = 0;

        // Loop through each line in the file
        while (scanner.hasNextLine()) {
            // Read in the next line of text
            String line = scanner.nextLine();

            // Split the line into words using whitespace as the delimiter
            String[] words = line.split("\\s+");

            // Add the number of words in the line to the total count
            count += words.length;
        }

        // Display the total number of words in the file
        System.out.println("Total number of words: " + count);

        // Close the Scanner object to free up resources
        scanner.close();
```

Problems  Javadoc  Declaration  Console ×
\<terminated> s1 [Java Application] C:\Users\jainroh\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.6.
Total number of words: 2

**2)** Create a program that reads in two text files and compares them to see if they are identical. The program should display a message indicating whether the files are identical or not.

https://codeshare.io/lonyp8

```
1
2
3  otFoundException;
4  iner;
5
6
7  id main(String[] args) {
8  -generated method stub
9
10 :eate File objects for the two files to be compared
11  file1 = new File("C:\\Users\\jainroh\\OneDrive - Tecnotree\\Documents\\input.txt")
12  file2 = new File("C:\\Users\\jainroh\\OneDrive - Tecnotree\\Documents\\text.txt");
13
14 :eate Scanner objects to read from each file
15 ier scanner1 = new Scanner(file1);
16 ier scanner2 = new Scanner(file2);
17
18 iitialize a variable to keep track of whether the files are identical
19 ean identical = true;
20
21 )op through each line in the files and compare them
22 e (scanner1.hasNextLine() && scanner2.hasNextLine()) {
23 '/ Read in the next line from each file
24 String line1 = scanner1.nextLine();
25 String line2 = scanner2.nextLine();
26
27 // Compare the lines and set the "identical" variable to false if they're different
```

Problems  Javadoc  Declaration  Console ×

`<terminated> s2 [Java Application] C:\Users\jainroh\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.6.v20`

```
The files are not identical.
```

**3)** Create a program that reads in a text file and creates a new file that contains the same text, but with all the vowels removed.

https://codeshare.io/EBE1BP

```
6  public class S3 {
7
8      public static void main(String[] args) {
9          // TODO Auto-generated method stub
10         try {
11             // Create File objects for the input and output files
12             File inputFile = new File("C:\\Users\\jainroh\\OneDrive - Tecnotree\\D
13             File outputFile = new File("C:\\Users\\jainroh\\OneDrive - Tecnotree\\
14
15             // Create Scanner and PrintWriter objects to read from and write to th
16             Scanner scanner = new Scanner(inputFile);
17             PrintWriter writer = new PrintWriter(outputFile);
18
19             // Loop through each line in the input file and remove vowels from eac
20             while (scanner.hasNextLine()) {
21                 // Read in the next line of text
22                 String line = scanner.nextLine();
23
24                 // Remove vowels from the line and write the result to the output
25                 String lineWithoutVowels = line.replaceAll("[aeiouAEIOU]", "");
26                 writer.println(lineWithoutVowels);
27             }
28
29             // Close the Scanner and PrintWriter objects to free up resources
30             scanner.close();
31             writer.close();
32
```

Problems  Javadoc  Declaration  Console ×

`<terminated> s3 [Java Application] C:\Users\jainroh\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.6.v2`

**4)Create a program that reads in a CSV file containing student grades, and calculates the average grade for each student. The program should then write the results to a new CSV file.**

https://codeshare.io/mpb7pw

```
a62.java    a63.java    aa4.java    aa5.java    s1.java    s2.java    s3.java    s4.java ×
31              String[] fields = line.split(",");
32
33                  // Extract the student name and grade from the fields
34                  String student = fields[0];
35                  int grade = Integer.parseInt(fields[1]);
36
37                  // Update the total grade and number of grades for the student
38                  if (totalGrades.containsKey(student)) {
39                      totalGrades.put(student, totalGrades.get(student) + grade)
40                      numGrades.put(student, numGrades.get(student) + 1);
41                  } else {
42                      totalGrades.put(student, grade);
43                      numGrades.put(student, 1);
44                  }
45              }
46
47              // Loop through each student in the Map and calculate their averag
48              for (String student : totalGrades.keySet()) {
49                  int total = totalGrades.get(student);
50                  int num = numGrades.get(student);
51                  double average = (double) total / num;
52
53                  // Write the student name and average grade to the output file
54                  writer.println(student + "," + average);
55              }
56
57              // Close the Scanner and PrintWriter objects to free up resources
```

**5)** Create a program that reads in a binary file containing image data, and displays the image on the screen.

https://codeshare.io/yo0Aoe

```
a62.java    a63.java    aa4.java    aa5.java    s1.java    s2.java    s3.java    s4.java    *s5.java ×
19              JFrame frame = new JFrame("Image Viewer");
20              frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
21
22              // Create a JPanel to hold the image and add it to the JFrame
23              JPanel panel = new JPanel() {
24                  @Override
25                  public void paintComponent(Graphics g) {
26                      super.paintComponent(g);
27                      g.drawImage(image, 0, 0, null);
28                  }
29              };
30              panel.setPreferredSize(new Dimension(image.getWidth(), image.getHeight
31              frame.getContentPane().add(panel);
32
33              // Set the size of the JFrame based on the size of the image and show
34              frame.pack();
35              frame.setVisible(true);
36
37          } catch (IOException e) {
38              // Handle the case where the file is not found or cannot be read
39              System.out.println("Error reading file: " + e.getMessage());
40          }
41
42      }
43
44 }
45
```