~~You must demonstrate your program during class time.~~ (It is an individual assignment)

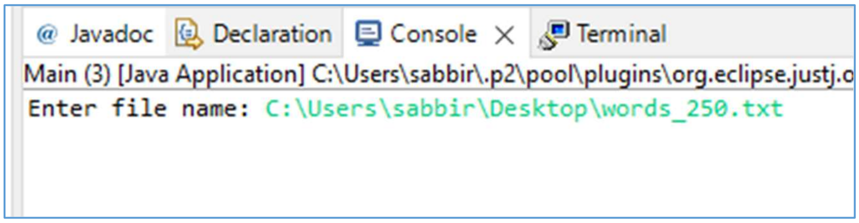
Assignment due (Submitting Softcopy to eConestoga):

Section	*Date*	*Time*
All sections	16-Dec-2022 (Unlike other assignments, this is due on Friday)	11:59PM

~~Demo is due on the first class after due time. Hardcopy must be submitted during demo.~~

Problem Specification: WordCounter app

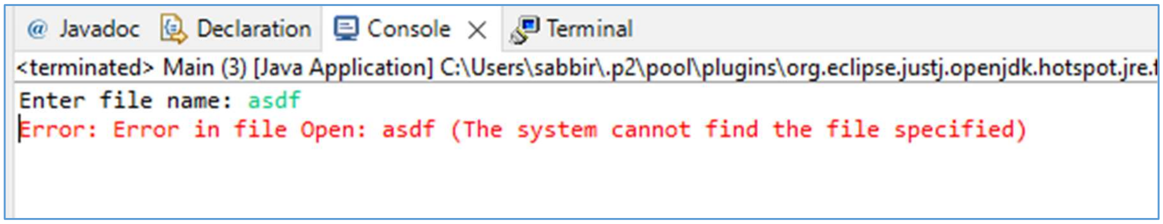
1. Read this full document and check the marking sheet (last page of this document) before your start working on the project.
2. This project must be completed in Java. Create a new Console based project in Java by using Eclipse.
3. Name the project as follows:
   *FirstNameInitialLastName*Assignment4
   Replace *FirstNameInitial* and *LastName* based on your name.
   Example: if a student's name is Jason Bourne, for Assignment 4, the name of the project will be JBourneAssignment4.
4. There is a sample input text file called `words_250.txt` that contains 250 "words" with one word for each line of the file. The file is also uploaded.
   - These are not dictionary words, rather, created from random series of characters.
   - Some might contain digits and special characters as well.
5. Write a program that do the following:
   - The program will ask the user to input a file path. (Absolute path that will start with c:\)



```
Main (3) [Java Application] C:\Users\sabbir\.p2\pool\plugins\org.eclipse.justj.o
Enter file name: C:\Users\sabbir\Desktop\words_250.txt
```

Figure 1: Program asks for input file containing works

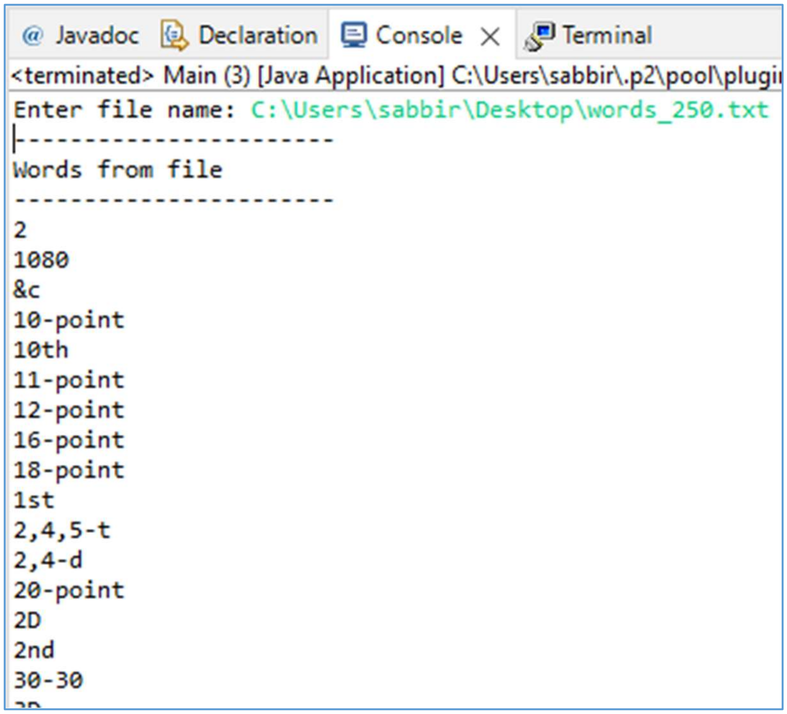
- Program shows error message on the console if invalid file path is given.



```
<terminated> Main (3) [Java Application] C:\Users\sabbir\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.f
Enter file name: asdf
Error: Error in file Open: asdf (The system cannot find the file specified)
```

Figure 2: Program shows error message for invalid file path

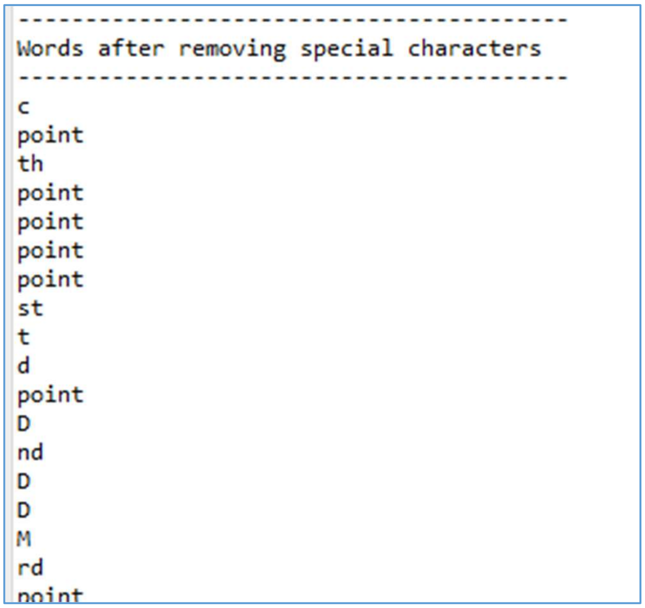
- Read all the words from the file and <u>print them to console</u>.



```
<terminated> Main (3) [Java Application] C:\Users\sabbir\.p2\pool\plugi
Enter file name: C:\Users\sabbir\Desktop\words_250.txt
-----------------------
Words from file
-----------------------
2
1080
&c
10-point
10th
11-point
12-point
16-point
18-point
1st
2,4,5-t
2,4-d
20-point
2D
2nd
30-30
```

Figure 3: Words in the file are printed on the Console (Only beginning part is shown here)

- For each word, remove all special characters and print to the console
  i. Special characters are: `0123456789-&,./'`
  ii. Please note: there are some words containing only special characters. So, they will turn into words with length 0 (they are not printed in the below example)

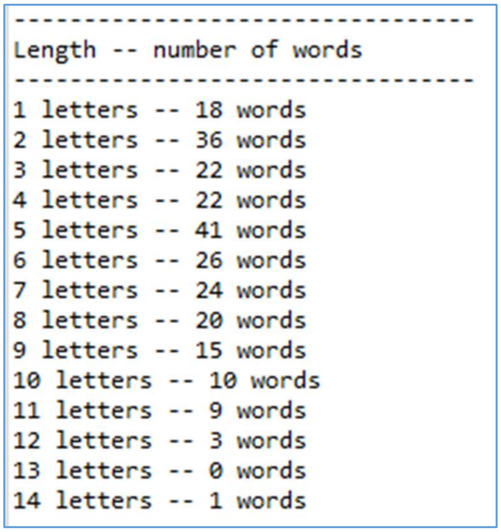


```
----------------------------------------
Words after removing special characters
----------------------------------------
c
point
th
point
point
point
point
st
t
d
point
D
nd
D
D
M
rd
point
```

Figure 4: All words are printed after removing special characters (Only beginning part is shown here)

- Count the words based on their lengths and print the number of words of each word length to the console. Show the output according to the following format (You may skip showing information for number of words with length 0):

- A sample console output file called `sample_output_words_250.txt` is also uploaded so that you may verify your application. You DON'T need to save your program's output to a file. Just show everything on the console. (word counts with 0 length are not printed in the blow example)



```
--------------------------------
Length -- number of words
--------------------------------
1 letters -- 18 words
2 letters -- 36 words
3 letters -- 22 words
4 letters -- 22 words
5 letters -- 41 words
6 letters -- 26 words
7 letters -- 24 words
8 letters -- 20 words
9 letters -- 15 words
10 letters -- 10 words
11 letters -- 9 words
12 letters -- 3 words
13 letters -- 0 words
14 letters -- 1 words
```

Figure 5: Number of words for different length

6. Please note while marking your app, the instructor might use different files. This means, your app will not have prior knowledge about the maximum length of words in the file.
7. The project requires to print outputs at 3 different stages (check the marking sheet below).

Hints:

As always, these are just hints, there are various ways of solving this problem

- Useful classes:
    - `Scanner` available in `java.util` package.
        - https://docs.oracle.com/javase/8/docs/api/java/util/Scanner.html
    - `ArrayList<>` available in `java.util` package.
        - https://docs.oracle.com/javase/8/docs/api/java/util/ArrayList.html
    - `StringTokenizer` located in `java.util` package.
        - https://docs.oracle.com/javase/7/docs/api/java/util/StringTokenizer.html
    - Please check the links, useful examples are there.

- `Scanner` class is very powerful for both taking input from user and reading lines from a file
- `StringTokenizer` can be used to parse a string into multiple substrings based on some delimiters (in our case, the special characters).

  - Join the tokens to generate a new word (this way, all special characters will be removed) and add them to a list.
- You can always iterate through the list of words to find the length of the longest word – might be needed if you intend to use arrays.
- You may find the occurrence of words with same length by Iterating through the list containing all new words (without special characters).
  - You may use `ArrayList`, or a simple array to store <length, frequency> information (maximum length might help to declare an array)

Softcopy Submission Requirements

- Make a zip file following the project naming convention mentioned above. This zip file will contain:
  - The complete project in Java
- Log in to your eConestoga account.
- Select the course PROG2370 and locate the Assignment's dropbox
- Upload the zip file.

Make sure all Programming standards are followed. Please read Programming Standards-PROG2370.pdf file available under Resources module in eConestoga for detailed information.
Make sure all Assignment standards are followed.
Note: Repeated violation of the same standard is counted.

Assignment 4 Marking Sheet

Spec	Marks
Program takes input of a file name with absolute path	15
Program can read words from the file and prints all words to the console	20
Program shows appropriate error messages	5
Special characters are removed properly and prints all words without special characters	30
Count the words (special characters removed) based on their lengths and print the number of words of each word length to the console	30
Total	**____________/100**

Deduction:

Runtime errors	15 x ____________ =
Assignment Standard	5 x ____________ =________/20
Programming Standard	1 x ____________ =________/20
Late Submission	_______days =
Bugs	3-10 (Based on severity)
~~Failing to answer to questions during demo~~	~~5-10 (Based on severity)~~
~~No Demo~~	~~50~~
Total Deduction	

Total Marks	

Late Penalty (Softcopy submission)

Days Late	Penalty %
1	5
2	10
3	20
4	30
5	45
6	60
7	80
8	100

Please note: How "Days Late" is calculated: Your assignment is due on Friday 11:59pm. You are considered 1 day late if you submit on anytime on Saturday (until 11:59PM). If you submit anytime on Sunday (until 11:59PM). you are considered 2 days late.