
Due Date:	By 11:59pm Wednesday September 29, 2017
Evaluation:	2% of final mark (see marking rubric at the end of handout)
Late Submission:	none accepted
Purpose:	The purpose of this assignment is to help you learn the Java basics: read input from the keyboard, display text on the console, manipulate variables of different types such as String, double, etc.
CEAB/CIPS Attributes:	Design/Problem analysis/Communication Skills

General Guidelines When Writing Programs:

- Include the following comments at the top of your source codes

```
// -----  
// Assignment (include number)  
// Written by: (include your name and student id)  
// For COMP 248 Section (your section) - Fall 2017  
// -----
```
- In a comment, give a general explanation of what your program does. As the programming questions get more complex, the explanations will get lengthier.
- Include comments in your program describing the main steps in your program.
- Display a welcome message.
- Display clear prompts for users when you are expecting the user to enter data from the keyboard.
- All output should be displayed with clear messages and in an easy to read format.
- End your program with a closing message so that the user knows that the program has terminated.

Question 1 – Display a design (3 pts)

Write a complete Java program that prints the following output:

```
  \/  
 \\\/  
\\\\/  
///\\/  
//\\/  
/\\/  
/\n  
All done!
```

Question 2 - Why my assignment is late (3 pts)

Write a Java program that helps a student write the text to an email explaining to their professor why their assignment is late.

Your program should prompt the user to enter the following information.

Note: String inputs are to be one word and the number input is to be stored in a variable of type double.

- The first or last name of your instructor
- Your name
- A food
- A number between 37.0 and 40.0
- An adjective
- A color
- An animal


After the strings and double are input, they should be substituted into the story below and output as an email to your prof using a single `System.out.println()` statement to the console (See sample output in figure 1).

Dear professor [*Instructor name*],

I am sorry that I am unable to turn in my computer assignment at this time. First, I ate a rotten [*food*], which made me turn [*colour*] and extremely ill. I came down with a fever of [*fever*]. Next my [*adjective*] pet [*animal*] must have smelled the remains of the [*food*] on my USB key (which is where my program was stored) because he ate it.

I am currently rewriting my programs and hope you will accept my assignment late.

Sincerely,
[*student name*]

Here is a sample output screen. User input is marked by 



```
<terminated> A1Q2 [Java Application] C:\Program Files\Java\jre1.8.0_131\bin\javaw.exe (Aug 27, 2017, 8:42:14 PM)

-----
Welcome to Nancy's Mad Lib Program
-----

What is your teacher's first or last name? Nancy
What is your name? Laura
Enter a food: lasagna
Enter the amount of fever (between 37.0 and 40.0): 39.2
Enter an animal: flamingo
Enter an adjective that will describe your pet: curious
Enter a colour: orange
|

Dear professor Nancy,

I am sorry that I am unable to turn in my computer assignment at this time. First,
I ate a rotten lasagna, which made me turn orange and extremely ill. I came down
with a fever of 39.2. Next my curious pet flamingo must have smelled
the remains of the lasagna on my USB key (which is where my program was stored)
because he ate it.
I am currently rewriting my programs and hope you will accept my assignment late.

Sincerely,
Laura
```


Figure 1. Sample output screen for Question 2

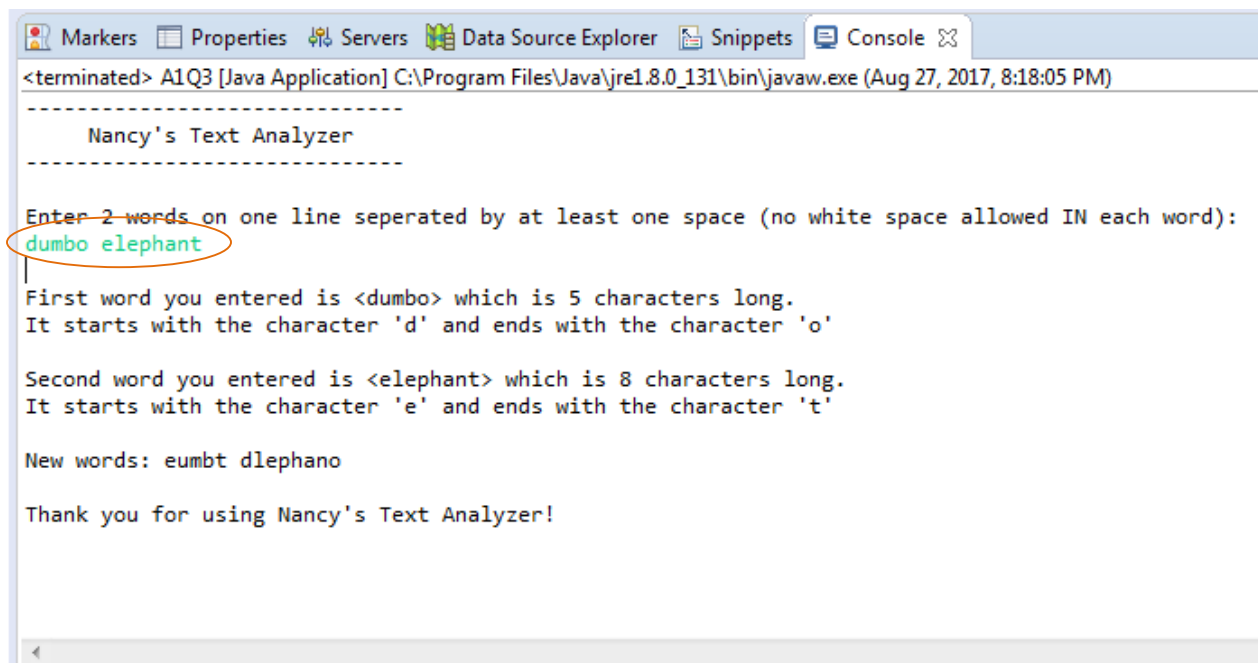
Question 3 - Manipulating variables of type String (6 pts)

Write a program that

1. Displays a welcome message of your choice.
2. Prompts the user for two words (can contain upper and lower case letters, digits and special characters excluding whitespace characters) then displays the number of characters as well as the first and last characters of each word.
3. Next, create 2 new words such that the first word starts and ends with the 1st and last characters of the second word and the second word starts and ends with the 1st and last characters of the first word. Display the new words.
4. Displays a closing message.

You can assume that the user enters 2 words with at least 2 characters each.

Here are a couple of sample output screen. User input is marked by 



```
<terminated> A1Q3 [Java Application] C:\Program Files\Java\jre1.8.0_131\bin\javaw.exe (Aug 27, 2017, 8:18:05 PM)

-----
Nancy's Text Analyzer
-----

Enter 2 words on one line separated by at least one space (no white space allowed IN each word):
dumbo elephant

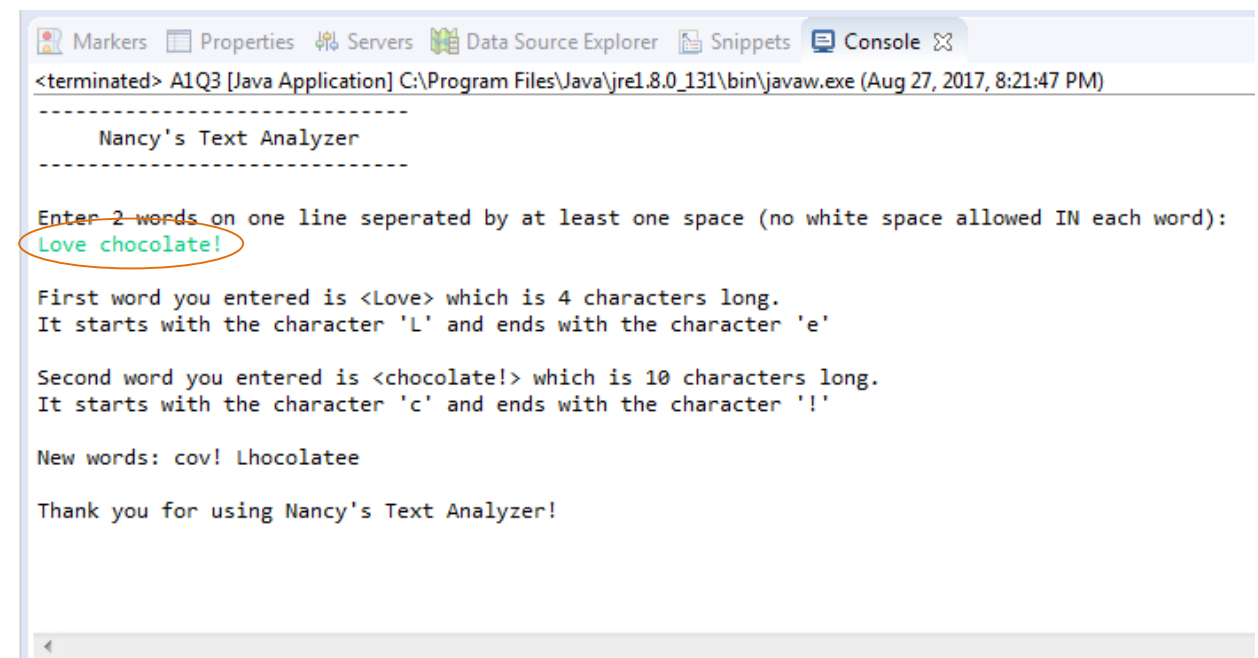
First word you entered is <dumbo> which is 5 characters long.
It starts with the character 'd' and ends with the character 'o'

Second word you entered is <elephant> which is 8 characters long.
It starts with the character 'e' and ends with the character 't'

New words: eumbt dlephano

Thank you for using Nancy's Text Analyzer!
```

Figure 2. Sample output screen for Question 3



```
<terminated> A1Q3 [Java Application] C:\Program Files\Java\jre1.8.0_131\bin\javaw.exe (Aug 27, 2017, 8:21:47 PM)

-----
Nancy's Text Analyzer
-----

Enter 2 words on one line separated by at least one space (no white space allowed IN each word):
Love chocolate!

First word you entered is <Love> which is 4 characters long.
It starts with the character 'L' and ends with the character 'e'

Second word you entered is <chocolate!> which is 10 characters long.
It starts with the character 'c' and ends with the character '!'

New words: cov! Lhocolatee

Thank you for using Nancy's Text Analyzer!
```

Figure 3. Sample output screen for Question 3

Submitting Assignment 1

- Zip the source codes (the .java files only please) of this assignment. Do not submit the actual project files.
- The zip file should be called *a#_studentID*, where # is the number of the assignment and *studentID* is your student ID number.

- For example, for the first assignment, student 123456 would submit a zip file named a1_123456.zip
- Refer to your section's Moodle page for instructions on where to submit your assignment.

Evaluation Criteria for Assignment 1 (20 points)

Source Code	
Comments for all 3 questions (5 pts.)	
Description of the program (authors, date, purpose)	2 pts.
Description of variables and constants	1 pt.
Description of the algorithm	2 pts.
Programming Style for all 3 questions (3 pts.)	
Use of significant names for identifiers	1 pt.
Indentation and readability	1 pt.
Welcome Banner/Closing message	1 pt.
Question 1 (3 pts.)	
Display requested shape	3 pts.
Question 2 (3 pts.)	
Read in string and double	1.5 pts.
Display email using a single System.out.println()	1.5 pts.
Question 3 (6 pts.)	
Prompting user/reading data	1 pt.
Display statistics on entered words	2 pts.
Create new words and display	3 pts.
TOTAL	20 pts.