CECS 220
Summer 2019
Assignment 2
May 30, 2019

Due Date: Thursday, June 6, 2019, 23:59
Total Points: 20

1. **(12 points)** Solve the following problem
   (a) Write a class called *BabysittingJob* that represents the concept of a babysitting service. The class contains the following instance variables:
   - A job number that contains six digits. The first two digits represent the year, and the last four digits represent a sequential number. For example, the first job in 2019 has a job number of 190001.
   - A code representing the employee assigned to the job. Assume that the code will always be 1, 2, or 3.
   - A name based on the babysitting code. Assume the service currently has three babysitters: (1) Cindy, (2) Greg, and (3) Marcia.
   - The number of children to be watched. Assume that this number is always greater than zero.
   - The number of hours in the job. Assume that all hour values are whole numbers.
   - The fee for the job is based on the babysitter and the number of children are watched. Cindy is paid $7 per hour per child. Greg and Marcia are paid $9 an hour for the first child, and $4 per additional hour for each additional child. For example, if Greg watches three children for two hours, he makes $17 per hour, or $34 for the two hours.

   Create a constructor for the *BabysittingJob* class that accepts arguments for the job number, babysitter code, number of children, number of hours. The constructor determines the babysitter name and fee for the job. Also include a *toString* method that returns a nicely formatted string containing the details of a *babysittingJob* object.

   (b) Write a client class that prompts the user for entering the data for a babysitting job. The application should keep prompting the user for each of the following values until they are valid:
   - A four-digit year between 2019 and 2025 inclusive.
   - A job number for the year between 1 and 9999 inclusive.
   - A babysitter code of 1, 2, or 3.
   - A number of children for the job between 1 and 9 inclusive.
   - A number of hours between 1 and 12 inclusive.

   When all the data entries are valid, construct a job number from the last two digits of the year and a four-digit sequential number, which might require adding leading zeros. Accordingly, create a *BabysittingJob* object, and display its values.

2. **(8 points)** Solve the following JavaFX application.
   Write a JavaFX application that analyzes a word. The user would type the word in a text field, and the application provides three buttons for the following:

   - One button, when clicked, displays the length of the word.

- Another button, when clicked, displays the number of vowels in the word.
- Another button, when clicked, displays the number of uppercase letters in the word.

(Hints: Use the GridPane or HBox and VBox to organize the GUI controls. See the Listings 4.8, 4.9, and 5.12 for related examples.)

Report and Program Submission Guidelines

It is expected that your report to be well written and organized. The report reflects your understanding of the assignment and its solution. So, you must assume that your marks assigned to the report part is related to how is the report is written and presented. All reports must be submitted in PDF format. Each assignment should contain the following:

1. **Report**
   1.1. Title page with your name, assignment number and the day you are actually submitting this report (Not the assignment due date).
   1.2. A brief description of your solution of each problem of the assignment, you can also explain your solution using a pseudocode. *Number your descriptions according to the problem numbers.*
   1.3. A comprehensive set of snapshots showing the inputs submitted, outputs obtained in the case of a successful output or a failure, including required output formatting, prompts, and messages.

2. **Submission Procedure:**
   2.1. Java source files that contain your solutions. They must be a "*.java" files. Source programs should contain meaningful comments and variable names.
   2.2. Please zip both the PDF document with the source codes and submit one zipped file. Please name your zipped file as "HWx_firstname_lastname.zzz". Where, "firstname" and "lastname" refer to your first and last names, "x" refers to the homework number (e.g., 1, 2, etc), "zzz" refers to the file name extension for the software used for archiving.
   2.3. Submissions after the due date are accepted with a penalty of 25% per day (weekend days are counted as one-day delay).

Grading Table

Problem	Item		Points
Problem 1	Report (Description of solution)		2
	Output snapshots (input/output, formatting)		1
	Java Source Code Implementation	*BabysittingJob* class	5
		Client Class	4
Problem 2	Report (Description of solution)		1
	Output snapshots (input/output, formatting)		1
	Java Source Code Implementation	*Start* method	4
		Handler methods	2
	Total		20