COSC 1P02 Assignment 7
Stand up and be Counted

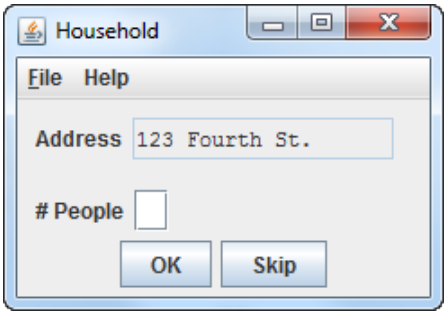
***Due: Dec. 2, 2016 @ 4:00 pm** (late date Dec. 5 @ 4:00 pm)*

The emphasis for this assignment is employing a GUI in a program and writing data files. In preparation for this assignment, create a folder called `Assign_7` for the DrJava project for the assignment.

Census Data Collection

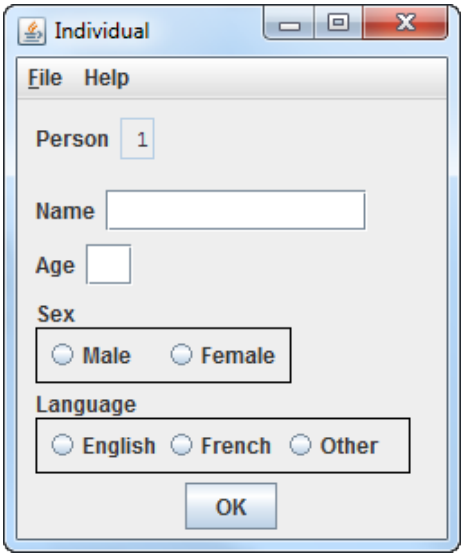
Every 5 years Statistics Canada completes a population census of the country. There are plans to have census takers go door-to-door to collect the census data. To facilitate this, the census takers will be supplied with a computer tablet to assist them in collecting and recording the data. You are to write the application program that will run on the tablet.

The program will be form based. The user (census taker) will launch the program and select a file of addresses that they will visit. The program will then present a form for each address in turn as the census taker walks from house to house. The `Household` form will look like:



The `Address` field is the address that the census taker should visit next. When she is at that address, if no one is home, she presses the `Skip` button and will be presented with the next address. If someone is home, she fills in the number of people that reside in the home and presses `OK`.

When `OK` has been pressed, the program will present an `Individual` form once for each of the people residing at the household such as:



The `Person` field is the person number of a person residing at the household (increasing by one for each form presented at the household). The census taker fills in the name and age and selects the sex and language (first learned). Upon pressing `OK` the data is recorded and the form for the next person residing in the household is presented. When there are no more members in the household, the next `Household` form is presented. When there are no more addresses in the address file, the program will terminate.

The program will produce a file (`ASCIIOutputFile`) of census data. This file will include one record (line) per household where someone was home, summarizing the individual data for the household. The record will include: address (`String`), number of people in the household (`int`), number of male members of the household (`int`), number of female members of the household (`int`) and number of English speakers (`int`), French speakers (`int`) and speakers of other languages (`int`). The file might look like:

```
123 Fourth St.  2       1       1       1       1       0
125 Fourth St.  3       1       2       2       0       1
128 Fourth St.  1       1       0       0       0       1
126 Fourth St.  5       2       3       0       5       0
124 Fourth St.  2       1       1       2       0       0
```

Hints:

- the program should process the address file to EOF.
- the `Individual` form is presented for as many people as reside in the household.
- a form can be visible or invisible on the screen. The method `hide` makes the form invisible. The method `accept` makes the form visible and the user can then interact with the form. The method `clear` clears out the data in a widget. The method `clearAll` clears all widgets on the form.
- you should include a method to handle the processing of a household.
- you should include a method to handle the processing of an individual.
- since the method handling an individual will have to update the counts (of number of males, females, etc.) and the method handling the household will have to write the counts, the counts will have to be declared as instance variables so both methods can access them.
- you should include other methods as appropriate

Submission:

Details regarding preparation and submission of assignments in COSC 1P02 are found on the COSC 1P02 Sakai Site as `Assignment Guidelines` under `Course Documents`. This document includes a discussion of assignment preparation, programming standards, evaluation criteria and academic conduct (including styles for citation) in addition to the detailed assignment submission process copied below.

To prepare and submit the assignment electronically, follow the procedure below:

1. Ensure your submission folder (`Assign_7`) contains the DrJava project for the assignment.
2. Using DrJava, print (to `CutePDF Writer`) the `.java` file for your assignment using the name *`ClassName`*`.pdf` where *`ClassName`* is the class name (i.e. same name as the `.java` file) and save the `.pdf` file at the **top level** of the submission folder (i.e. directly within `Assign_7`).

3. Run the program using the file `addresses.txt` as input. When the program presents the `Save As` dialog for the `ASCIIOutputFile` (census data) save the file as `censusData.txt` at the **top level** of the submission folder (i.e. directly within `Assign_7`). Follow the instructions in `Script,pdf` for data entry into the forms presented by the program.
4. Create a `.zip` file of your submission by right-clicking on the top level folder (i.e. `Assign_7`) and selecting `Send to/Compressed (zipped) folder`. A zipped version of the folder will be created. Use the default name (`Assign_7.zip`).
5. Log on to Sakai and select the COSC 1P02 site.
6. On the `Assignments` page select `Assignment 7`. Attach your `.zip` file (e.g. `Assign_7.zip`) to the assignment submission (use the `Add/Remove Attachments` button and select `Browse`). Navigate to where you stored your assignment and select the `.zip` file (`Assign_7.zip`). The file will be added to your submission. Be sure to check the `Honor Pledge` checkbox. Press `Submit` to submit the assignment. You should receive a confirmation email.

DrJava

The `.zip` folder you submit should contain the project folder including all files relevant to the project—the `.java` and `.class` files for the assignment—and the `.pdf` files for program listings and output at the top level.

Other Platforms

If you are using an IDE other than DrJava to prepare your assignment, you must include the `.java` source files and the `.pdf` files described above for each part as well as an executable file (likely `.class` or `.jar`) that will execute on the lab machines.