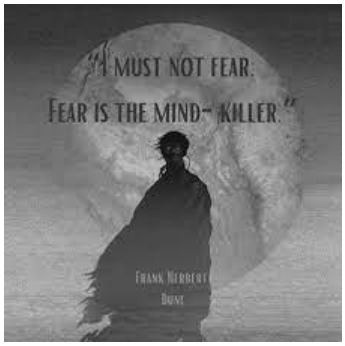
Homework 1

Start Assignment

Due No Due Date **Points** 10 **Submitting** a file upload **Available** after Sep 22 at 6pm



Prework

** Make sure you've completed **ICL 1** before doing this assignment. **

Note: HW 1 must be completed individually.

Requirements

In HW 1, you will make one or more extensions to the `Greeter` module we developed in **ICL 1**. For assignment submission, you will include quotes from the book *Dune* in the greeting by completing the following:

1. Start by creating your own project and copying in the **dune_quotes.txt (https://github.com/stephen-riley/cpsc-5011-fall-2021-projects/blob/main/week_01/homework/dune_quotes.txt)** file from the class repo.
2. Extend the template language so the variable `$quote` will insert a random Dune quote into the greeting.
3. You have noticed that Dune quotes can make the greeting pretty long; extend the template language so the template variable `$newline` will insert a newline character into the greeting. Place a `$newline` before the quote is displayed.
4. Demonstrate all newly implemented functionality in the driver.
5. Documentation is not required for this homework; however, please review the **(Java) Documentation & Style Standards** for subsequent assignments.

Sample Output

```
Good night Sriley - that's a nice blue shirt.
 Dune quote: "The sleeper must awaken. - Duke Leto Atreides"
```

Extra Credit

> ⓘ Extra Credit assignments will not be graded unless you receive at least an 85% on the regular assignment.

1. The Dune module can optionally emit the quotes in the order in which they appear in the file. You should make the greeter generator configurable so it will can generate either random or sequential Dune quotes. Come up with some way that the driver program can communicate to the greeter generator whether it wants random or sequential Dune quotes.
2. Our implementation of the template language has problems with certain edge cases.
   - For example,
     - `"Hello $name."` may not work because of the period at the end of the template variable.
     - `"I owe you $1.25"` will act as though the dollar amount is a variable.
   - Both edge case examples above will throw an exception.
   - Define the template language precisely as it relates to how variables are handled, and fix the code in `Template.java` accordingly.

3. In class we hardcoded the person's name into a variable `$name` that was passed to the greeter generator. Extend this functionality as follows:
   - If the greeter is asked to instantiate a name and the name appears in its input dictionary, it uses that name. Otherwise, it finds the user's (system) username, and uses that instead, capitalizing the first letter in the username. Hint: you may want to use `System.getProperty("user.name");`.
   - For example,
     - My username is "sriley" so a greeting of `"Hello $name"` on my system without supplying a name explicitly should be "Hello SRiley" (note the capitalization!).
     - If the input hash does not contain a name and the system cannot provide a username, the variable should be instantiated as "Unnamed Person".

Sample Output (for EC #3)

```
Good night SRiley - that's a nice green shirt.
  Dune quote: Am I in GRADUATE SCHOOL yet?
```

Notes:

- Demonstrate all implemented extra credit functionality in the driver.
- You may receive at most 5 extra credit points.

Submission

- Use the IntelliJ export feature (as shown in class) to create the `[su-username]_hw1.zip` file.
- All Java classes should reside inside the package `[su_username]_hw1`. For example, my username is "sriley", so my package name is `sriley_hw1` (no spaces, no quotation marks). Use your own SU username!
- Click on the "Submit Assignment" button for and submit a single ZIP file. You may submit multiple times. Only your latest submission will be graded.
- Include a README.md file with informal comments answering the following questions:
  - Is your solution fully working or not? How does your driver program demonstrate this?
  - What extra credit problem(s) did you work on (#1, 2, 3, all)? Briefly describe how do you demonstrate the functionality?
  - How much time did you spend on the assignment?
  - Any feedback on the assignment?