Project 1 – Displaying Text – 100 points

Deadline

Final push no later than 12AM Sunday on 18 September 2023

Goals

To begin development of the term project: a simple, interactive "robot" simulator.

To reinforce your budding Git and Python skills.

Preparation

For this assignment, you must have already created your *{YourLastname}-Work* repository hosted under our class organization on GitHub.com. Open your local repository in the VSCode IDE and create a new subfolder named *Homework*.

Instructions

Create the first version of your project by implementing the required features listed below.

★ *Commit #1 – Create a* `README` *file for the project*   **10pts**
  ◦ This file must be named `README.md` and located in the *Homework* subfolder.
  ◦ Include the title of your project, your name, and a brief description of your project.

★ *Commit #2 – Initial version of* `main` *script*   **30pts**
  ◦ This file must be named `main.py` and located in the *Homework* subfolder.
  ◦ Include a `main` function in which the rest of your code will reside.
  ◦ At the beginning of this function print a title and introduction for your program. This test should indicate to the user what the program is about.
  ◦ Use an input statement to ask the user for the room dimensions (width and height).
  ◦ Next, prompt the user to press <Enter> to begin– e.g., "Press enter to begin." Nothing else should happen until the user presses the <Enter> key.
  ◦ After the user presses <Enter>, your program should display a closing message – e.g., "Thanks for using the Robot Simulator!" – followed by the credits. The credits should include your own name as well as the course number and instructor name.
  ◦ Finally, be sure to <u>call</u> your `main` function so that you can test the program.

★ *Commit #3 – Display an empty room.*   **30pts**
  ◦ Using the dimensions that the user enters, print multiple lines of output to draw an ASCII (i.e., text) rectangle representing the room within which the "robot" will operate. Use the hash tag "#" for drawing the boundaries of the room.
  ◦ Be sure to display this initial empty room <u>before</u> prompting the user to begin.

★ *Commit #4 – Place the robot in the room.*   **20pts**
  ◦ Use Python's built-in random number generation to select an initial start location for the robot. Do this by randomly selecting the *x*-coordinate (column number) and *y*-coordinate (row number).
  ◦ <u>After</u> prompting the user to begin, draw the room again but this time with the letter "R" representing the robot.

★ *Commit #5 – Prompt the user to quit.*   **10pts**
  ◦ After printing the room with the robot, prompt the user to press <Enter> again to quit the program.
  ◦ Be sure that the closing text and credits only print <u>after</u> the user quits.

Advice   Be sure to stage your changes a little at a time and commit frequently. Push any local work to your remote GitHub repository regularly. Don't forget to:

- Write short but meaningful messages for every commit. (<u>Tip</u>: Consider using the instructions themselves as your commit messages.)
- Look at the differences between successive versions of your code. Do this on GitHub as well as in Terminal/Git Bash using the `git diff` command.

Test, test, test… and test again. Then test some more. When you think you've tested enough, go back and test yet again. Then get someone else to test for you while you test theirs. Etc.

Submitting   You must <u>push</u> your changes to GitHub before the due date.

<u>Note</u>: Pushing regularly will reduce the risk of losing your work, so do not wait until after you have made all changes and commits before pushing.