Task 3: Getting started with Replit

In this class, we will be teaching you Python programming from the ground up! The best way to learn Python is by practicing as often as you can. Rather than installing python on your computer, we are going to use Replit. Replit is an online platform that we can use to collaborate on code development. This means is you get stuck, I can always "pop in your repl" to see what is going on. Let's start by getting a free account.

Get a free Replit account

To get started, you need a free "Starter" account at Replit. This will give you limited access to compute time, but we don't need much given that you are just learning Python and the scripts we will write are not compute intensive.

Download the Replit app for your laptop

Code editors (or Integrated Development Environment, IDEs) are fantastic resources that make it easy to write and edit code on your laptop. Replit is an advanced code editor that it both available online, and via a Desktop app. You can use replit either way you'd like. I will show examples, when using the Desktop Editor. You can use this link to download the version you need for your laptop.

Connecting your GitHub account in Replit

Once you have installed Replit on your Desktop, you can connect it to your GitHub account. To do this, click on the icon for your user in the far right corner of the app. Next, navigate to "Connected Services" in the left navigation bar. Then click the "Connect" button next to GitHub. Login to your GitHub account when prompted.

Creating a Repl and Cloning a copy of your Class GitHub repository (be434-Spring2024) into it

Next, you will need to create a new "Repl" or compute environment, to copy the code from your GitHub class repository. You should have already "forked" a copy of the class repository in your own GitHub account (see setup2_github). To create a new Repl, go to the Replit icon in the far left corner, and select "Create Repl" from the menu. Once you have done this, you will be directed to the Replit website to create a Repl. Select the "Import from GitHub" tab and find the be434-Spring2024 that you forked. Select "Public" (your only option with the free account) and then click the "Import from GitHub" button.

You will be prompted to add a run command, copy and paste this command:

```
~/workspace/assignments/grade_hw.sh
```

Once you have added the run command click "create repl". You will now have access to your own code repository in Replit.

Installing Python modules for running tests on your code

Now that you have downloaded the class repository, you can install all of the Python modules we will use in the class to test you code (found in the be434-Spring2025/docs/requirements.txt file). Python modules (or code packages) are written by people in Python community and can be used by anyone to perform certain functions in Python. We are going to use several Python modules in this class to test your code and make sure it meets community standards. Open a shell in your be434-Spring2025 repl, which should open a unix terminal. Use the following command to import the required modules into your repl.

```
python3 -m pip install -r ~/workspace/docs/requirements.txt
```

To make sure you installed the modules correctly, try a few out by running these commands in the shell. You should get help messages from each program telling you how to run them.

```
flake8 --help
pylint --help
```

Overview of the commands:

```
black # this command will format your code properly
pytest # this command will test your code with default test discovery
mechanism.
mypy # this command will run type checks on your code with mypy.
flake8 # this command will check code linting (or how pretty it is!).
```

Author

Bonnie Hurwitz bhurwitz@arizona.edu