

Mod 1 Lab - Basic Python

Part 1 – Solo – hello.py

1. Click the **Mod 1 Lab** link in HuskyCT (**Learning Modules > Module 1 – Basic Python > Mod 1 Lab**). This will open Lab 1 in Mimir.
2. Download the starter code from Mimir.
3. Unzip the starter code somewhere convenient (for instance, a folder named **lab1**).
4. Open that folder in VS Code (**File > Open Folder**). In Part 2, you will be inviting another student to work with you in this folder -> make sure it only contains files you are okay with sharing.

You should see this pdf, the .md file used to generate this pdf, and two .py files in your workspace. Add the following to hello.py :

```
print("Hello World!")
```

Now we are ready for our first submission to Mimir.

5. Go to **Mod 1 Lab** in Mimir and click **Submit**.
6. Select and submit all the .py files for this lab (hello.py and stats.py)
7. You should pass the test case associated with hello.py, but not the test cases associated with stats.py. Congrats!

Click on the test cases to see the debugging information we have decided to provide. An important part of this class is learning to debug yourself, so we will sometimes provide less information than others. Take advantage of the debugging information we share to learn what good tests look like!

This concludes the solo portion of **Mod 1 Lab**. You should be able to:

- Access Mimir assignments through links on HuskyCT
- Download labs from Mimir
- Open and edit starter code in VS Code
- Submit code to Mimir

Next, you will start a collaborative session with a classmate to work on stats.py.

Part 2 – Group – stats.py

Setting up a collaborative environment

Begin this session once we have created breakout rooms in Collaborate Ultra. Move yourself to the group noted on the google sheet and decide who will host the VS Code LiveShare session. That partner should:

1. Click the **Live Share** icon on the left-hand side bar
2. Click **Start collaboration session...** in the next window
3. Paste the link created in your breakout room

The other partner(s) should use the provided link to join the Live Share session. You must log into a Microsoft or Github account to be able to edit code in a VS Code Live Share session. You can log into your UConn Microsoft account using your UConn email + password when prompted, or use a personal Github account.

Once everyone is in the LiveShare session, begin working on the assignment. You can decide how to collaborate as a group, but one successful way to do this is pair programming:

- One student (the driver) writes code
- Other students (the navigators) observe, ask questions, and make design decisions
- Swap roles at ~15-minute intervals

Regardless of how you choose to collaborate, this is an active activity: there should be conversation flowing continuously.

stats.py

`stats.py` provides code to read user input from a terminal and store it in a collection. You must:

- implement three `compute_*` functions to return the mean, median, and mode of a series of numbers
- modify the input loop to filter out non-numeric types. A `try/except` block is helpful here ([docs](#))

Some edge cases to watch out for:

- If there are an even number of items, return the lower of the two middle values for the median
- If there are multiple modes, return the lowest
- For testing, it may be helpful to modify the input loop to terminate when a user types a certain phrase (e.g. when a user types "done")

External Modules

Do not use any imported modules (`math`, `collections`, ...) when implementing functionality. It is okay to use imported modules for testing.

It is okay to import modules you write yourself; e.g. any data structures you write yourself.

Submitting

At a minimum, submit the following files:

- `hello.py`
- `stats.py`

Students must submit to Mimir **individually** by the due date (typically, two days after lab at 11:59 pm EST) to receive credit.

Grading

- 10 - `hello.py`
- 90 - `stats.py`
 - 15 - sequential ints
 - 15 - mix of words and ints
 - 20 - ints on multiple lines
 - 20 - mix of words and ints on multiple lines (test info hidden)
 - 20 - mix of words and ints on multiple lines (test info hidden)

Feedback

If you have any feedback on this assignment, please leave it [here](#).

We check this feedback regularly. It has resulted in:

- A simplified, clear **Submitting** section on all assignments
- A simplified, clear **Grading** section on all assignments
- Clearer instructions on several assignments (particularly in the recursion module)