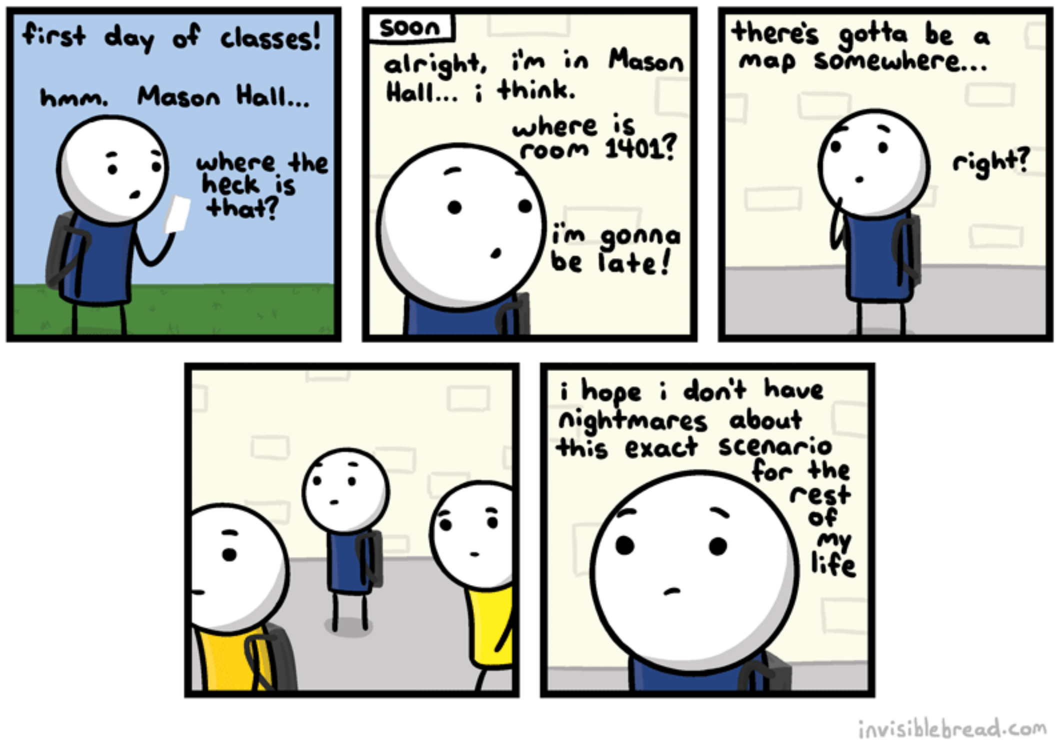
Python, PyCharm, and Gradescope


first day of classes!
hmm. Mason Hall...
where the heck is that?
soon
alright, i'm in Mason Hall... i think.
where is room 1401?
i'm gonna be late!
there's gotta be a map somewhere...
right?
i hope i don't have nightmares about this exact scenario for the rest of my life
invisiblebread.com

`http://joyreactor.com/post/954711`

This document will walk you through the basics of installing Python and PyCharm on your personal laptop and navigating Gradescope for assignment submission.

1 Installing Python and PyCharm

In this class, we will learn the Python programming language and work on challenging (but fun!) projects. In order to do so, we need to install Python and PyCharm (one of the representative Python integrated development environments (IDEs)), and use them for coding.

The latest Python interpreter can be downloaded from `https://www.python.org/downloads` (with packages for both Windows and Mac OS X operating systems). You should select the latest distribution of Python 3.11.1, and install it after it is downloaded.

PyCharm can be downloaded by going to `https://www.jetbrains.com/pycharm/` and selecting your platform (Windows or macOS). **Make sure you download the (free, open-source) Community Edition.**

It might be best if you attempt to do this before coming to the first lab next week, so that we can help you more quickly to resolve any problems.

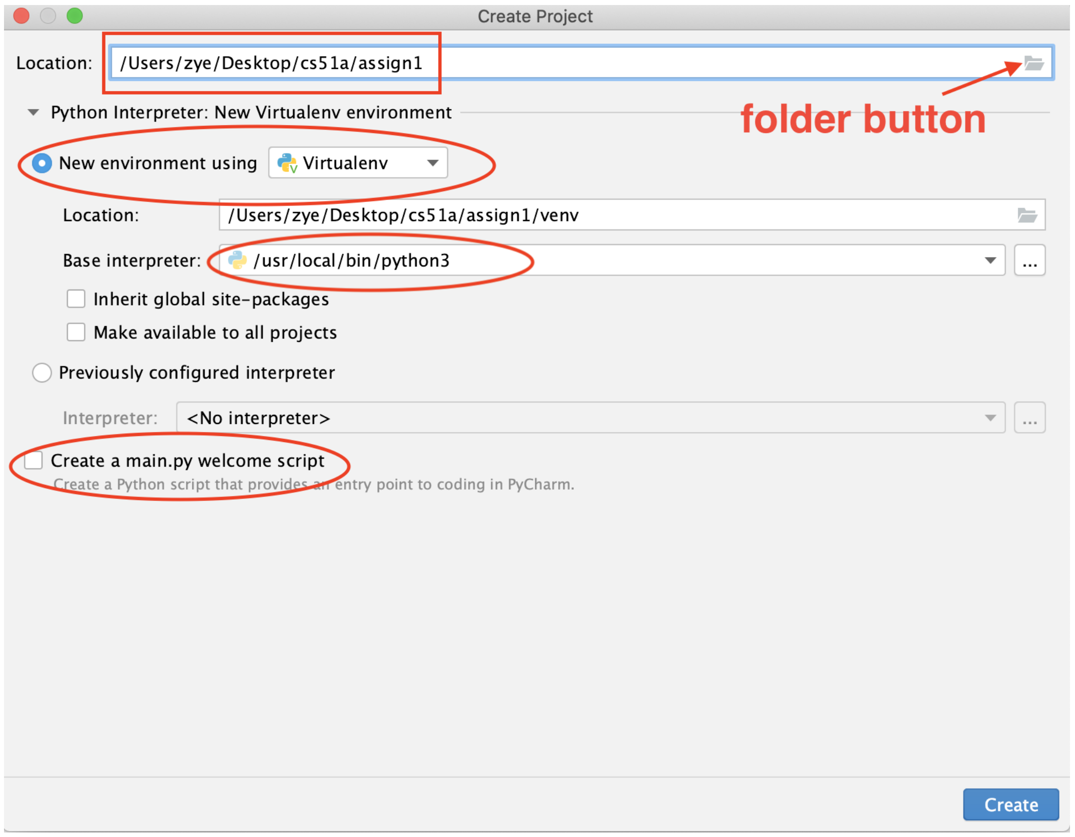
2 Starting PyCharm

PyCharm will, by default, try to maintain all of your projects under a single folder. Before starting PyCharm for the first time you should create a new workspace folder. We suggest that you create a new folder on your `Desktop` named `pays2023`.

- On a Mac, you can start PyCharm by clicking the `PC` icon (as shown below) on your menu-bar, or you can click on the ‘Launchpad’ or go to your ‘Application’ folder to find it to open.
- On a Windows system, you can start PyCharm by clicking on the ‘Windows’ button on your keyboard or the `Start` button on the bottom left of your screen, and then search the name of PyCharm to open it. For an easy access in the future, you can also drag it to generate a shortcut on your desktop or pin it to the start menu.

3 Creating a new project

After starting PyCharm you can accept the defaults and get to a screen that invites you to select `Create New Project`. Go ahead and create the new project, making sure that the location of the project is a new sub-folder (e.g. `assign1`) in the `pays2023` folder you just created on your `Desktop`. In order to achieve this, you need to click on the `folder` button as shown below to navigate to your `Desktop`, and then select the `pays2023` to open. After that, in the `Location` fillable space, you need to manually type in the assignment subfolder name `/assign1` to create this new subfolder.



Create Project

Location: /Users/zye/Desktop/cs51a/assign1

Python Interpreter: New Virtualenv environment

● New environment using Virtualenv

Location: /Users/zye/Desktop/cs51a/assign1/venv

Base interpreter: /usr/local/bin/python3

☐ Inherit global site-packages

☐ Make available to all projects

○ Previously configured interpreter

Interpreter: <No interpreter>

☐ Create a main.py welcome script

Create a Python script that provides an entry point to coding in PyCharm.

Note that, on a Windows system, you may need to navigate to your `C:` directory, and go to `Users`, then go to the folder under your login account to find your `Desktop` folder. The `Location` might look like this after your setting, `C:\Users\YOURLOGINNAME\Desktop`

WARNINGS for new project creation

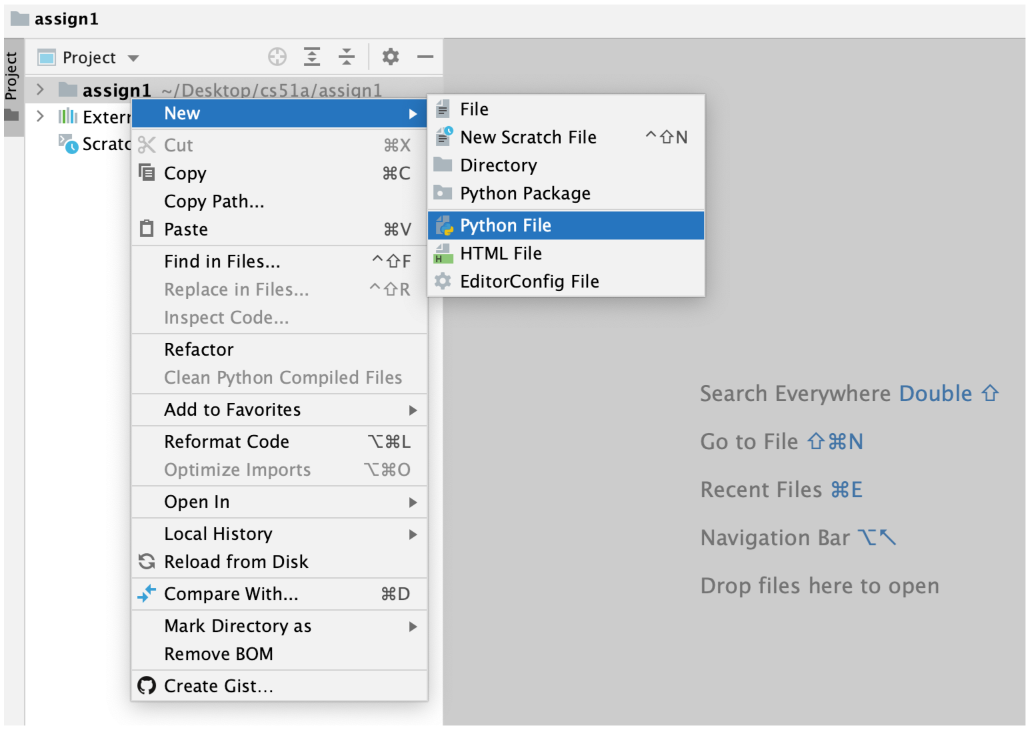
- make sure you put the new project in the `pays2023` folder you just created on your `Desktop`, rather than in the default location. It will be very difficult for you to find the default location and submit your assignment in the future.
- check the option `New environment using Virtualenv` to create a virtual environment, as highlighted in the above picture.
- make sure the base interpreter is python3, rather than python2 (which is installed by default in some computers).
- uncheck the option `Create a main.py welcome script`, as highlighted in the above picture.

After these configurations, you can click on the `Create` button on the bottom right to create a new project.

Important Note: In this class it's probably best to use one project per assignment. Make sure it's inside of your `pays2023` folder and then go ahead and call it `assign1`, `assign2`, `assign3`, ...

4 Editing and running a Python program

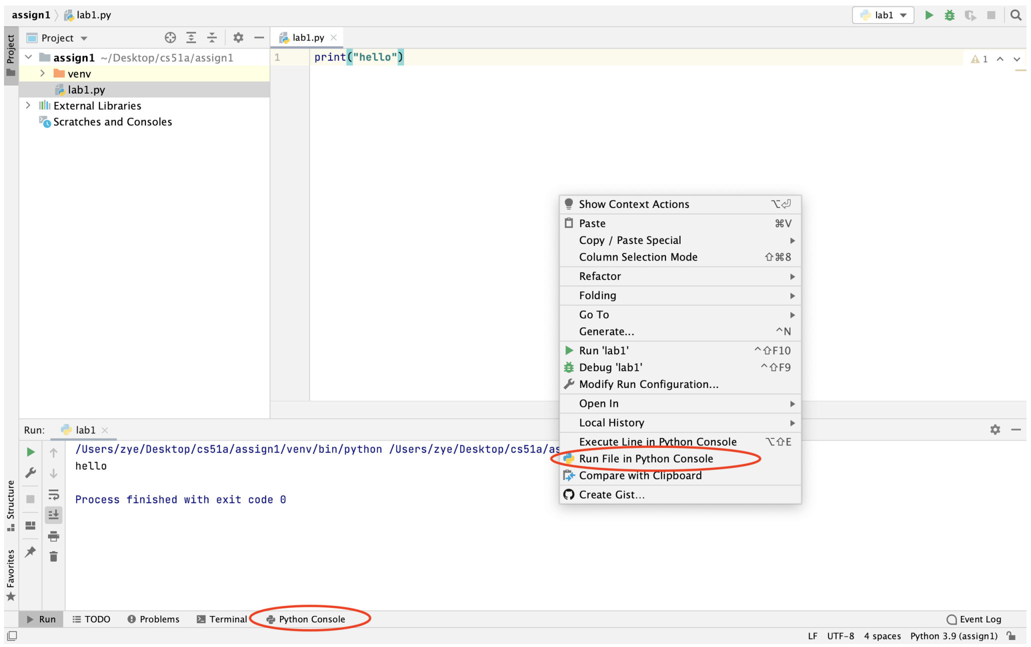
Once a new project is created, you can right click the project name (e.g., `assign1`) on the left panel, choose `New` and navigate to `Python File` to create a new python file in this project. An example is shown below.



You can give the name to the python file, e.g., `assign1`, and then enter. After that, you should be looking at the main screen with the following components:

- a menu bar at the top
- a (nearly empty at this point) list of projects and files on the left
- a main window on the right
- a list of screen selections on the bottom (note that a screen that displays the program output will be shown after you run a program)

You can edit your python program in the main window. For example, you can add a print statement in the file. Then, you can save the file by selecting `File/Save all` from the menu, or by typing `Command + s` on Mac or `Ctrl + s` on Windows.



```
print("hello")
```

```
/Users/zye/Desktop/cs51a/assign1/venv/bin/python /Users/zye/Desktop/cs51a/as
hello

Process finished with exit code 0
```

The easiest way to Run a program is to place the cursor in the empty place in the file editing window and right-click. As shown in the above figure, one of the options will be `Run File in Python Console`. If you select this option, the Python interpreter will execute the selected program and display the output in a `Python Console` window at the bottom of the screen.

Another way to run your python program is to right-click the empty place in the file editing window, and select the option of `Run` to run your current file. A third option to run your code is to use the green triangle button on the top right of the window. You can choose which python program you want to run, e.g., the `Current File`.

Important Note: at the bottom right of the screen, you will be able to see the python interpreter that you are using for this project. Make sure that it is `Python 3.x`, e.g., `Python 3.11`. If you see it is `Python 2.x`, then you are **not** using the correct interpreter, as the coding syntax will be different between python3 and python2.

After running your first python program, you can perform another practice by adding a few more print statements. Add the following print statements to your program and run it again. You should now see some results printed out.

```
print(10)  # printing a number
print(22/7) # printing another number
pi = 22/7  # assign 22/7 to pi
print(pi) # print out what is stored in pi
print("Hello computer user") # printing a string
```

5 Playing with the Python Console

In PyCharm, we will mostly use the file editing window for writing codes. Besides that, we can use an interactive shell, which is the `Python Console`. It is located at the bottom of the PyCharm window, as highlighted in the figure above. You can click it to open the Python Console, or you can go to `Tools` menu, pick `Python Console...` or hit Command, Shift, and A at the same time and type "Python Console" and hit return.

```
>>>
```

is the prompt, meaning Python is ready for a command.

```
...
```

means it's a line continuation (i.e. Python is waiting for you to finish the statement) and a line without anything in front of it is generally the response from the interpreter.

- Try a few simple mathematical equations (e.g. "1+1", "2**3", "(100/20)+45*7"). Notice that Python makes for a pretty easy to use calculator.
- 22/7 is an approximation for Pi. Type this in.
- Remember that we can use variables to store intermediary values. Assign 22/7 to a variable called `pi`. Use that variable to calculate the area of a circle with radius 15 (remember, the area of a circle is π times the radius squared).
- In 2011, Pomona built a new parking structure. For a variety of reasons, the college decided to put a field on top of the parking structure (pretty cool, right?!).

Since we're in southern California, we don't need any covering for this field. However, let's pretend that we anticipate bad weather more regularly and we decide we need to cover it with a dome (global warming, maybe?). If we make the approximating assumption that the field is a circle of radius 100 feet, what is the surface area of the dome that would cover it?

Hint: the surface area of a *sphere* is $4\pi r^2$.

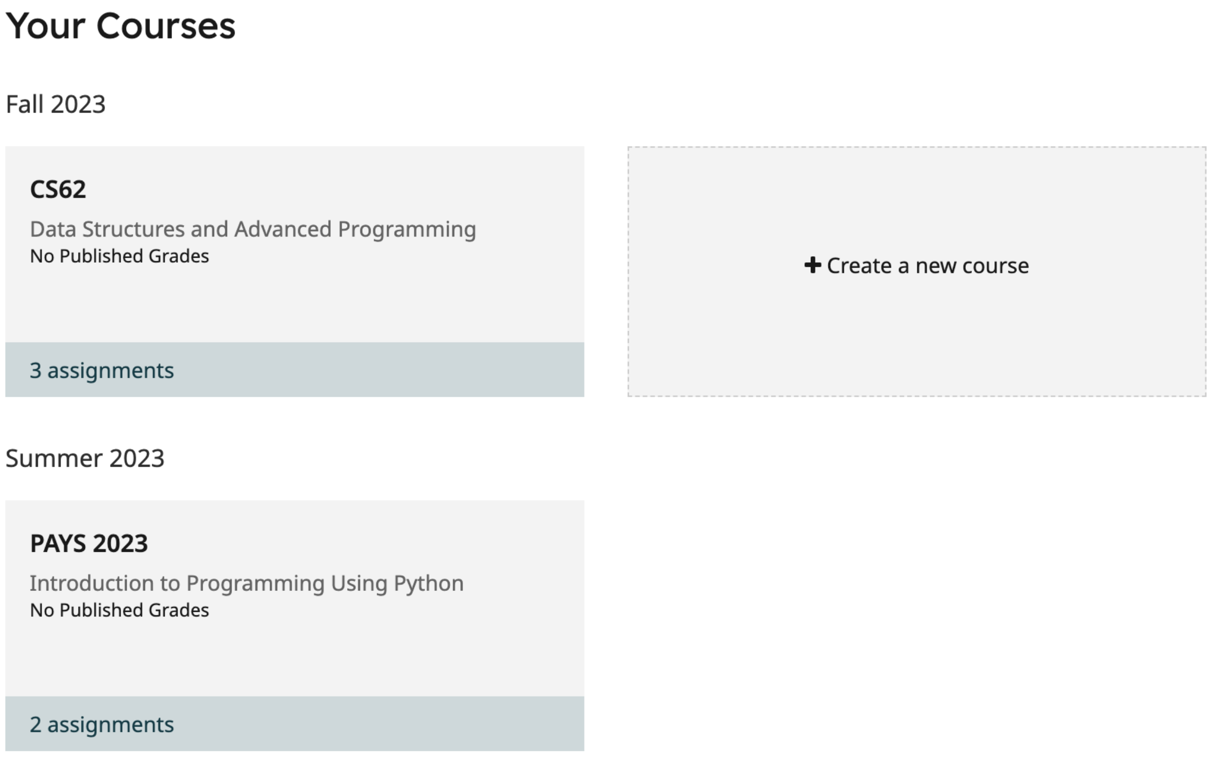
In the Python shell, you can use the up and down arrow keys to revisit commands you typed previously. Use the up arrow key to get your previous statement and then edit it to get the surface area of the dome assuming that the radius is 200 feet. Discuss your answers with your neighbor and make sure you agree.

6 Using Gradescope for Assignments

The weekly lab assignments will need to be submitted through the Gradescope web application. The lab assignments' grades and feedback reports will be returned through gradescope as well.

6.1 Accessing Gradescope

When you log into Gradescope, you should see PAYS 2023 on your dashboard.



Your Courses

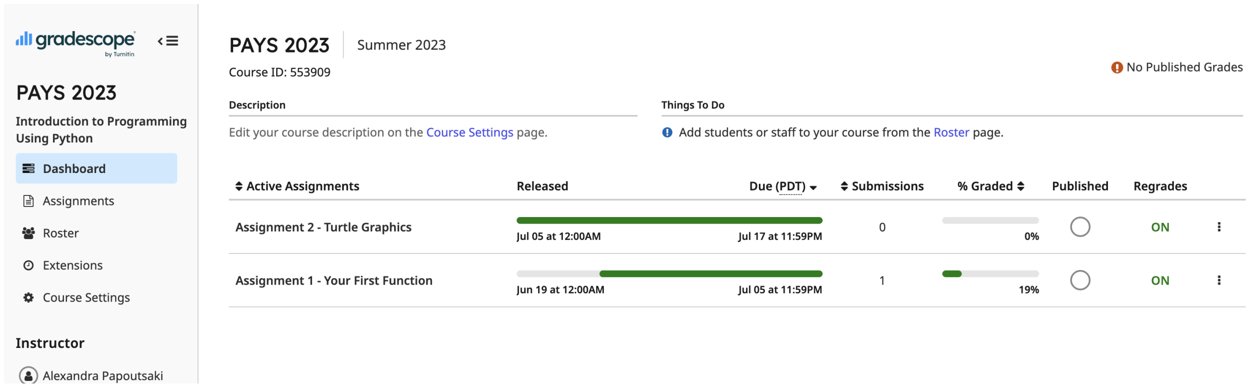
Fall 2023

CS62
Data Structures and Advanced Programming
No Published Grades
3 assignments

Summer 2023

PAYS 2023
Introduction to Programming Using Python
No Published Grades
2 assignments

By entering this course (clicking) you will be able to access a list of lab assignments that you have submitted and work that is due in the near term. An example of this facility is shown in the following image.



gradescope

PAYS 2023

Introduction to Programming Using Python

Instructor

Alexandra Papoutsaki

PAYS 2023 | Summer 2023

Course ID: 553909

No Published Grades

Description

Edit your course description on the Course Settings page.

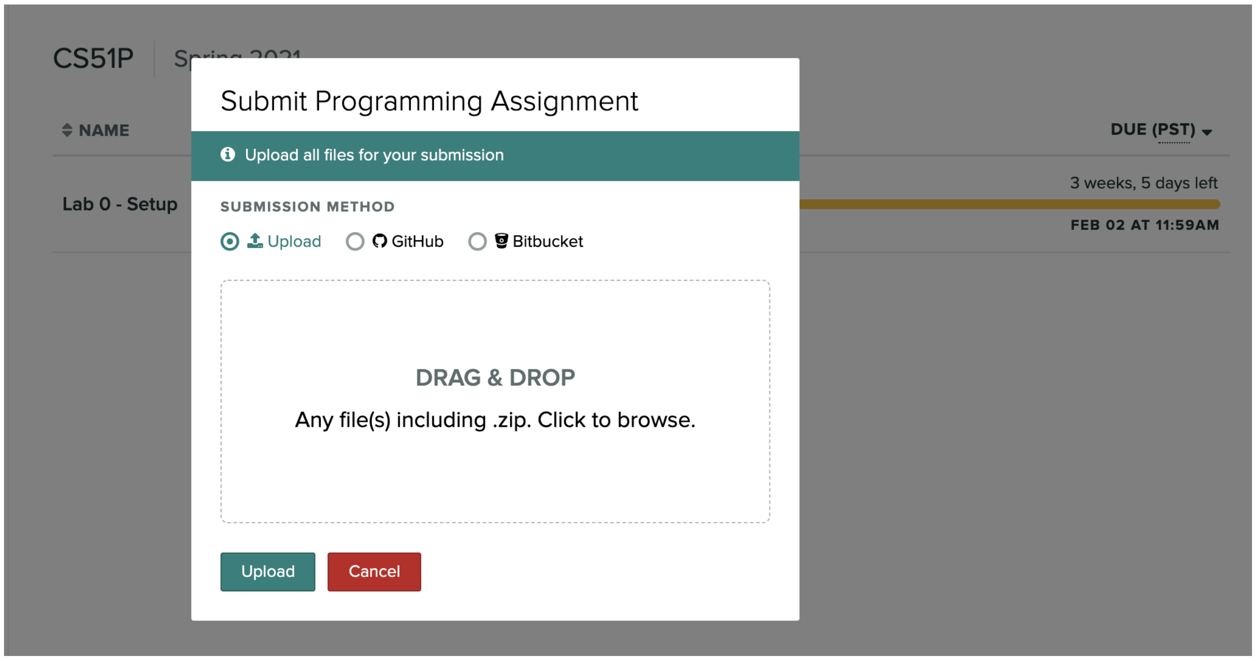
Things To Do

Add students or staff to your course from the Roster page.

Active Assignments	Released	Due (PDT)	Submissions	% Graded	Published	Regrades
Assignment 2 - Turtle Graphics	Jul 05 at 12:00AM	Jul 17 at 11:59PM	0	0%	○	ON
Assignment 1 - Your First Function	Jun 19 at 12:00AM	Jul 05 at 11:59PM	1	19%	○	ON

6.2 Submitting Python Code for Lab Assignments

We will use gradescope for code submission of your lab assignments. You can click on the lab assignment name (as highlighted in the above figure), and a window will pop up for you to submit the code, via "Drag and drop" or "Click to browse", as shown in the following image.



CS51P | Spring 2021

NAME | DUE (PST)

Lab 0 - Setup | 3 weeks, 5 days left | FEB 02 AT 11:59AM

Submit Programming Assignment

Upload all files for your submission

SUBMISSION METHOD

[✓ Upload] GitHub Bitbucket

DRAG & DROP

Any file(s) including .zip. Click to browse.

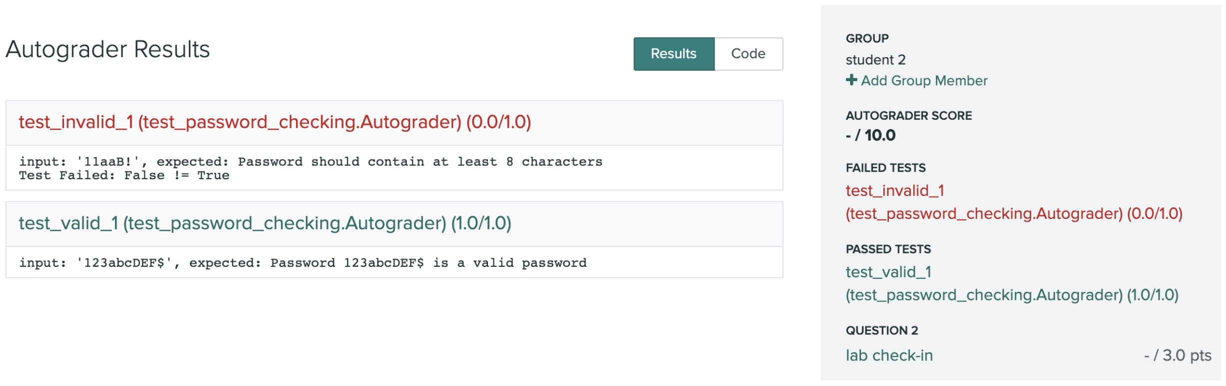
Note: Some assignments may require you to upload multiple files. In this case, you should submit all the required files together at once, rather than submitting them one by one. With "Drag and drop", you can select multiple files to perform the operation. With "Click to browse", you can select multiple files by pressing "Command" and clicking on multiple files to select them for uploading.

Note that you can resubmit your work many times as long as it is before the deadline. The very last submission will be considered for grading.

6.3 Autograder

As some parts of your program will be graded using an autograder, please make sure your python source files are using the correct names as required

For the lab assignments that have autograder, you will be able to see the sanity check results shortly after you submit your code. The test cases that are in the green color are passed, while the test cases with the red color are failed. You may receive some feedback to the failed test cases to help you debug your program.


Autograder Results

test_invalid_1 (test_password_checking.Autograder) (0.0/1.0)

```
input: '11aaB!', expected: Password should contain at least 8 characters
Test Failed: False != True
```

test_valid_1 (test_password_checking.Autograder) (1.0/1.0)

```
input: '123abcDEF$', expected: Password 123abcDEF$ is a valid password
```

GROUP
student 2

AUTOGRADER SCORE
- / 10.0

FAILED TESTS
test_invalid_1 (test_password_checking.Autograder) (0.0/1.0)

PASSED TESTS
test_valid_1 (test_password_checking.Autograder) (1.0/1.0)

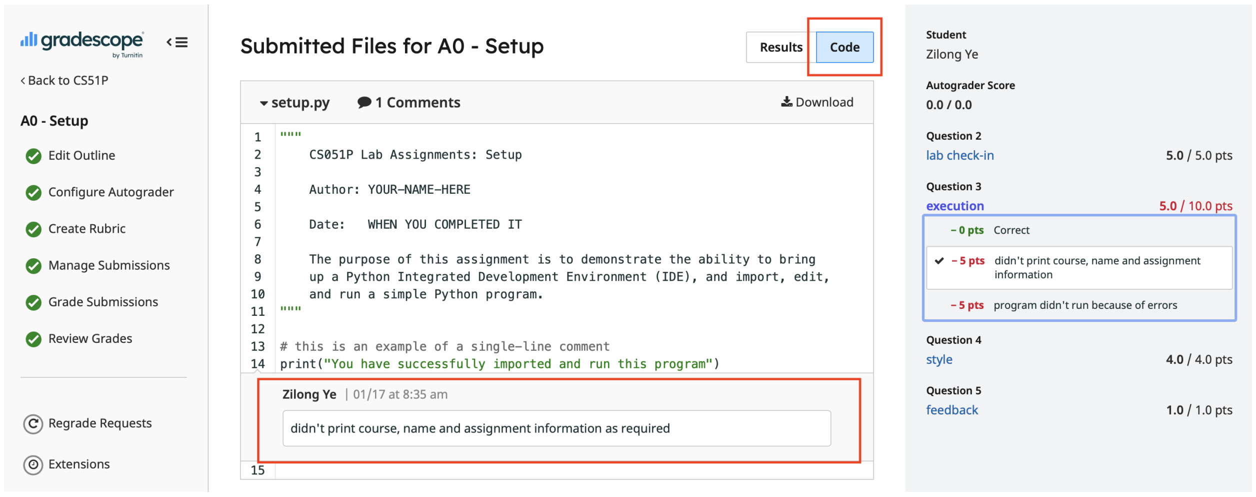
QUESTION 2
lab check-in - / 3.0 pts

The purpose for sanity check is to (1) provide a few basic test cases to make sure your program runs for some simple cases; (2) give you some feedback and a chance to improve your implementation; (3) give you a sense about how we will be testing your code and motivate you to generate additional test cases to examine your code thoroughly before you submit your final version. Note that you can submit as many times as you need, as long as it is before the deadline.

Note: the sanity check test cases are just basic ones, not comprehensive. We will be using a more comprehensive set of test cases to grade your program after the due date. So, please make sure that you test your program with all the possible test cases that you can think of, including regular user input cases and any possible corner cases.

6.4 Accessing your Grades and Comments

You will be able to access a detailed grading rubric and comments for your work as soon as the grades are released. For each lab assignment, you will have access to a pdf containing your work as well as our feedback. As an example of this facility, consider the following images in a previous semester for another class. Suppose that you did not get full marks for this lab assignment. Selecting this lab assignment from the dashboard leads to a detailed rubric including both the points for each question and the points that you received.



Submitted Files for A0 - Setup

```
"""
    CS051P Lab Assignments: Setup

    Author: YOUR-NAME-HERE

    Date:   WHEN YOU COMPLETED IT

    The purpose of this assignment is to demonstrate the ability to bring
    up a Python Integrated Development Environment (IDE), and import, edit,
    and run a simple Python program.
"""

# this is an example of a single-line comment
print("You have successfully imported and run this program")
```

Zilong Ye | 01/17 at 8:35 am

didn't print course, name and assignment information as required

Student
Zilong Ye

Autograder Score
0.0 / 0.0

Question 2
lab check-in — 5.0 / 5.0 pts

Question 3
execution — 5.0 / 10.0 pts

	− 0 pts	Correct
✓	− 5 pts	didn't print course, name and assignment information
	− 5 pts	program didn't run because of errors

Question 4
style — 4.0 / 4.0 pts

Question 5
feedback — 1.0 / 1.0 pts

Clicking on the specific question (e.g., "execution") where 5 points were deducted provides more details. Clicking on the "code" button can show the code you submitted and the instructor's grading feedback as comments in your code.