Assignment 0: Python, PyCharm, and Gradescope

http://joyreactor.com/post/954711
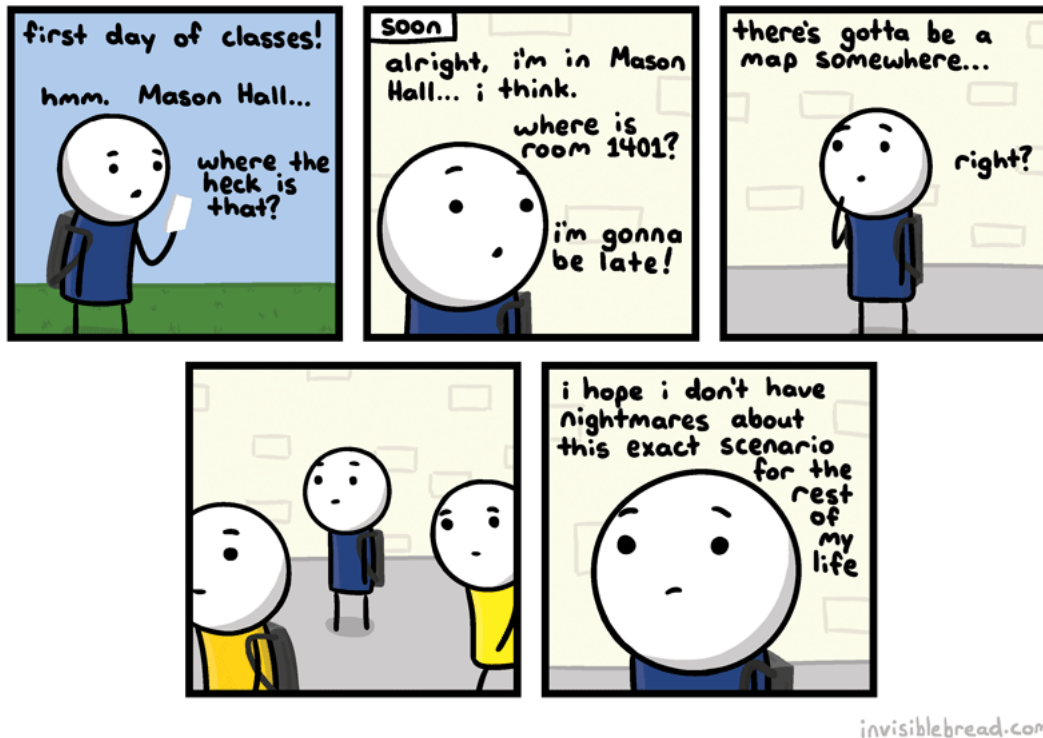
For our first assignment, we will walk through the basics of Python and PyCharm and navigating Gradescope for assignment submission.

1 Assignment overview

For this first assignment we'll be doing a few different things:

- Install Python and PyCharm

- Create a new project and .py file in PyCharm.

- Play a bit with writing and running code in PyCharm. Note, the code you write will *not* be graded, only that you tried out a few things.

- Read and summarize our first ethics paper.

- Familiarize yourself with Gradescope for submitting assignments.

2 Installing Python and PyCharm

Most of the assignments in this class will use Python, 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 3.13.7, 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).

3 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 `cs51a`.

Mac

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.
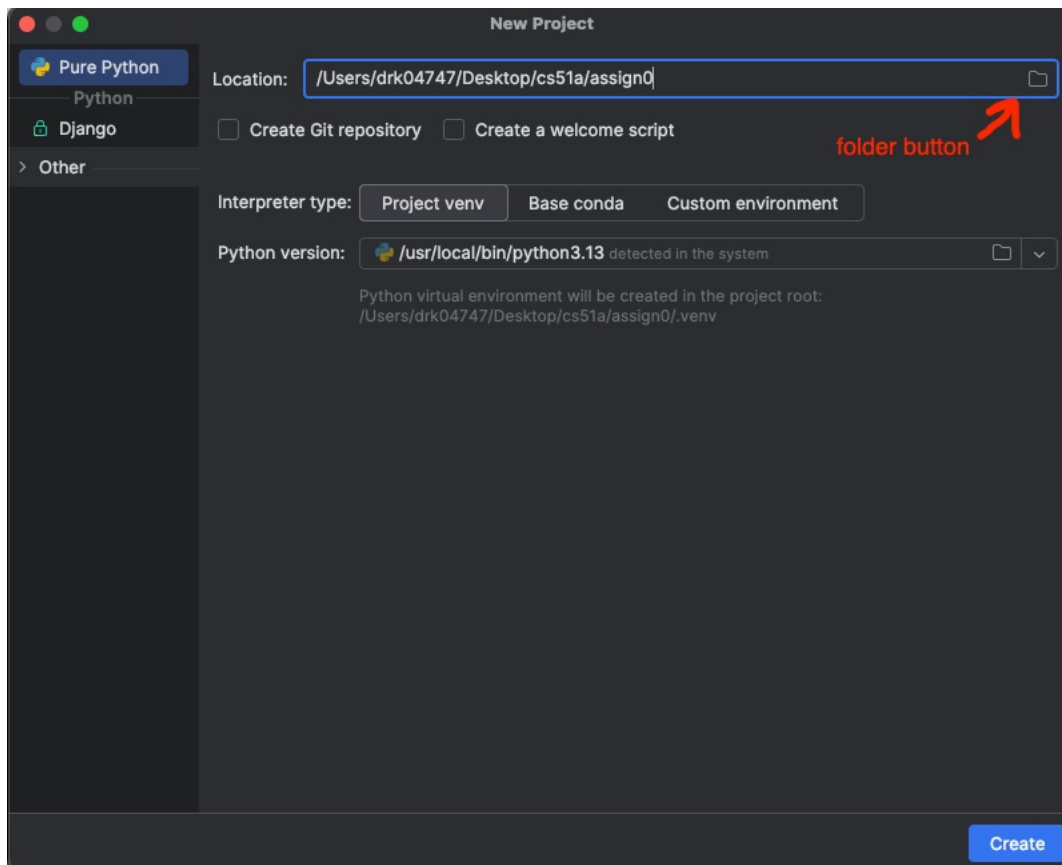

Windows

On a Windows computer, 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.

4 Creating a new project

After starting PyCharm, accept the defaults and you'll get to a screen that says "Welcome to PyCharm." Click the "New Project" button to create a new project (you'll do this for every assignment). After clicking the button, you'll see the details for the new project (see example below). Make sure that the location of the project is a new sub-folder (e.g. `assign0`) in the `cs51a` folder you just created on your `Desktop`. You'll need to click on the `folder` button as shown below

to navigate to your `Desktop`, and then select the `cs51a` to open. After that, in the `Location` fillable space, you need to manually type in the assignment subfolder name `/assign0` to create this new subfolder.

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 `cs51a` 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.

- make sure the python version is python3 (.something), rather than python2 (which is installed by default in some computers).

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 `cs51a` folder and then go ahead and call it `assign1`, `assign2`, `assign3`, ...

5 Editing and running a Python program

Once the new project is created, right-click (or ctrl+click) on the project name (e.g., `assign0`) 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.
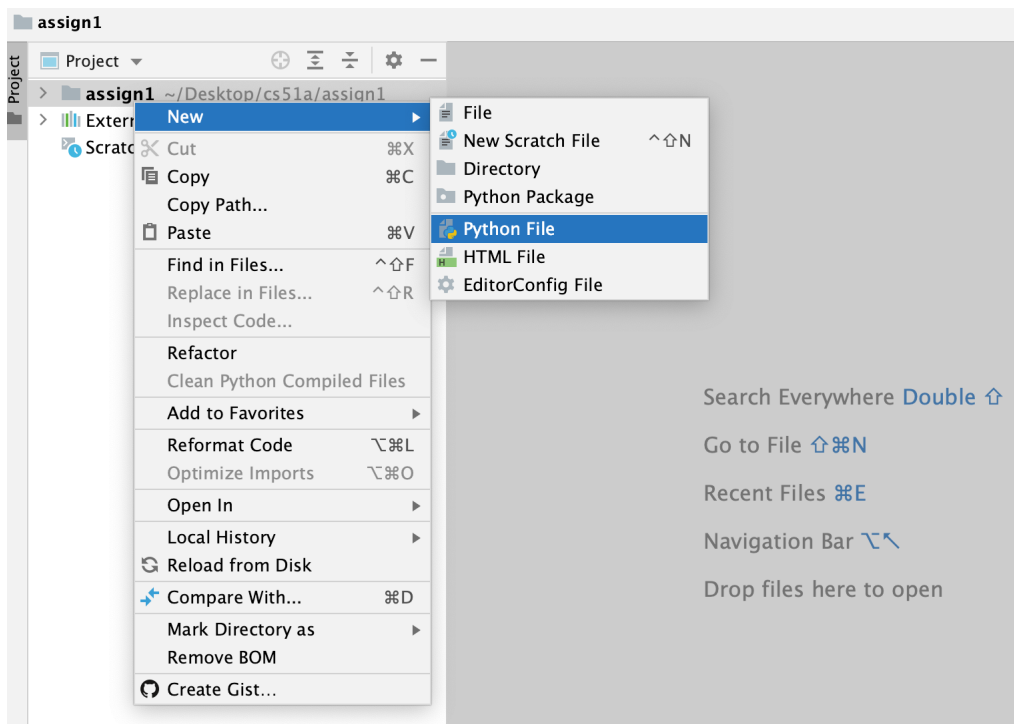
Name the file `assign0.py` and then hit enter. After that, you should be looking at the main screen with the following components:

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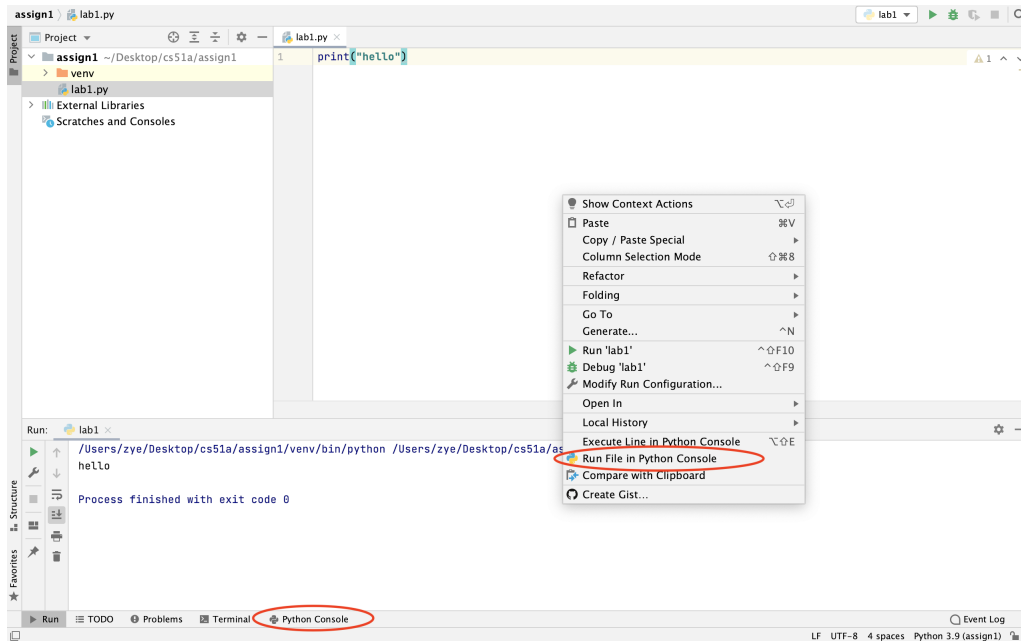
The easiest way to `Run` a program is to place the cursor in the empty place in the file editing window and right-click. As shown in the above figure, one of the options will be `Run File in Python Console`. If you select this option, the Python interpreter will execute the selected program and display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

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 3.13`. 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
```

6   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.

5

```
>>>
```
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.

7 Ethics

In addition to a coding part, many of the assignments will also have a short ethics reading. For our first reading, we will look into a relatively old (2016) article from ProPublica which suggests caution about incorporating AI into the justice system. It can be found at `https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing`. The article by Julia Angwin, Jeff Larson, Surya Mattu and Lauren Kirchner was a finalist for the Pulitzer Prize for Explanatory Reporting.

In a text editor (pick whatever you're comfortable with), summarize the article in 1–2 paragraphs and discuss anything surprising you learned, any questions, and any ideas you might have for avoiding 'machine bias'. Export your document as a pdf, called `ethics0.pdf`.

8 Using Gradescope for Assignments

In CS51A, the weekly assignments need to be submitted through Gradescope. The grading and feedback will be returned through Gradescope as well. In addition, you will be able to submit regrade requests through Gradescope. At the beginning of the semester, all students who are enrolled in CS51A will be associated (by the instructors) with Gradescope. You should receive an email notification as soon as you are added. Please let us know ASAP if that is not the case.

8.1 Accessing Gradescope

When you log into Gradescope, you should see CS51A on your dashboard.
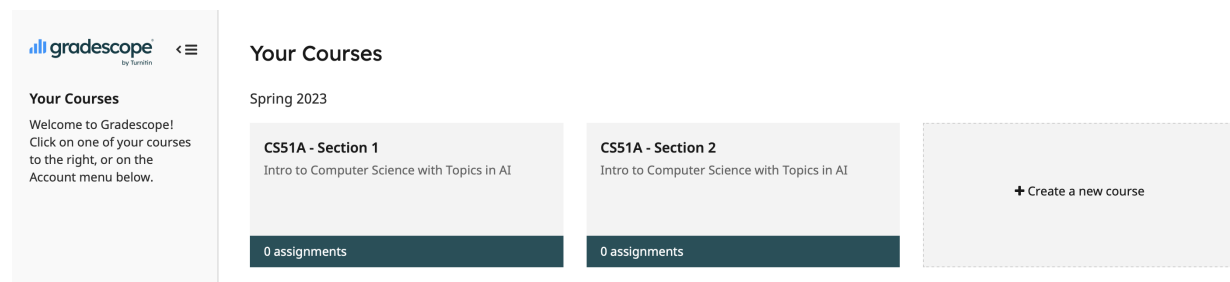
By entering this course (clicking) you will be able to access a list of assignments that you have submitted and work that is due in the near term. An example of this facility is shown in the following image.

8.2 Submitting Python Code for Lab Assignments

We will use Gradescope for code submission of your assignments. You can click on the 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.

Note: Some assignments may require you to upload multiple files (like this one!). 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.

For this assignment, submit:

- `assign0.py`

- `ethics0.pdf`

8.3 Autograder

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 assignments that have an autograder, you will be able to see the basic 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 did not pass. You may receive some feedback to the failed test cases to help you debug your program.

The purpose for basic checks are 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 basic test cases are not comprehensive. We will be using a more comprehensive set of test cases to grade your program after the due date. 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.

8.4 Adding Partners to Teamwork

Some of the lab assignments are conducted in the format of pair programming. Only one submission is needed per team. After one of the team members submit the work, you can click on the button of "Add group member" on the top right of the window to add the team mate to this submission.

A window will pop up for you to add your team member, through entering your team member's name in the blank space.
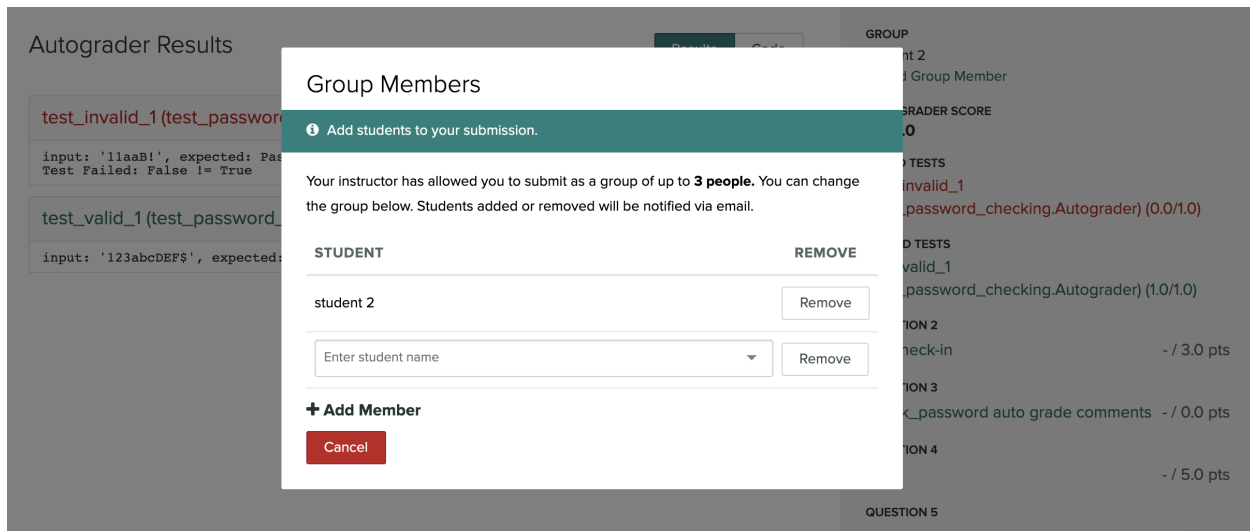
8.5   Accessing your Grades and Comments

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 (CS51P). 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.

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.

8.6 Regrading Requests

If you have specific questions about the grading of your assignment or you do not understand the comments provided, you may submit a regrade request.

To do so, you must first navigate to the question. There you will find the "Request Regrade" button at the bottom of the page. When you click this button, a window will pop up. Please provide a few sentences describing your question.

Once your regrade request has been handled, you will receive an email notification. By logging into

Gradescope and navigating to the question, you will see both the text of your original request as well the instructor's response

Please note that for all items, including assignments, quizzes and exams, the regrading request option will be open for 7 days after the grades are released. After that, the regrading option will be closed and you will no longer see the "Request Regrade" button on that item. We encourage you to review your grades as soon as the grades are released and use regrading request to clear all concerns you have, before the next assignment is due.