Homework and Projects

### [M3T1](https://faytechcc.blackboard.com/webapps/assignment/uploadAssignment?content_id=_6067859_1&course_id=_35059_1&group_id=&mode=view)

**M3T1 - Room Class**

In this assignment, we're developing a **Room** class, and putting it through its paces.

Examples of how to get started can be found in the References section of this module (showing sample code from in class)

Our goal is to develop a class that will hold Room data, based on the kind of text adventure game we've been looking at.

At minimum, it should hold a name, and a description.

We'll need to figure out how to link rooms together -- but for now, simply making a list of them, and iterating through them, is sufficient.

You'll need to come up with five (5) rooms with their own names and descriptions for this version.

### [M3T2](https://faytechcc.blackboard.com/webapps/assignment/uploadAssignment?content_id=_6067866_1&course_id=_35059_1&group_id=&mode=view)

**M3T2 - Doctest**

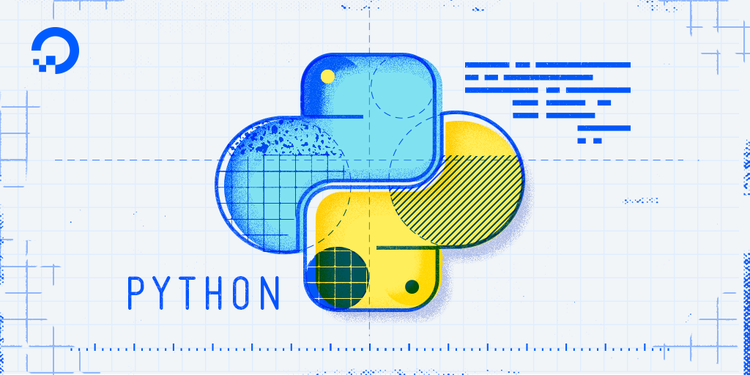
**This will be our introduction to automated testing -- which can be very powerful, but also a bit of a rabbit hole.**To keep it simple, our first **doctest** is for some simple math functions (addition and multiplication)

Below are some references for the **doctest** module.

[doctest â Test interactive Python examples — Python 3.9.7 documentation](https://docs.python.org/3/library/doctest.html)

(<https://docs.python.org/3/library/doctest.html>

)

[](https://www.digitalocean.com/community/tutorials/how-to-write-doctests-in-python)

<https://www.digitalocean.com/community/tutorials/how-to-write-doctests-in-python>

[How To Document and Test Your Code with Doctest in Python | DigitalOceanPython’s standard library comes equipped with a test framework module called doctest. The doctest module programmatically searches Python code for pieces of text within comments that look like interactive Python sessions. Then, the module executes thoDigitalOcean](https://www.digitalocean.com/community/tutorials/how-to-write-doctests-in-python)

[doctest – Testing through documentation - Python Module of the Week](https://pymotw.com/2/doctest/)(<https://pymotw.com/2/doctest/>

)

For this example, we'll create a simple **Calculator** class that has **add**and **multiply** functions. Using the references above as a guide, first implement the functions, then create doctests for them.

Finally, you'll need to run the doctests from the command line.

### [M3T3 - Linked Lists](https://faytechcc.blackboard.com/webapps/assignment/uploadAssignment?content_id=_6072952_1&course_id=_35059_1&group_id=&mode=view)

We'll build an example data structure (using "Room-like" data) using a linked list.

Examples included. Take the existing code, complete it, and confirm that you can walk from Room 1 to Room 5, and then back from Room 5 to Room 1.

 [m3t3\_partial\_completed.py](https://faytechcc.blackboard.com/bbcswebdav/pid-6072952-dt-content-rid-55238359_1/xid-55238359_1)

### [M3LAB1](https://faytechcc.blackboard.com/webapps/assignment/uploadAssignment?content_id=_6079382_1&course_id=_35059_1&group_id=&mode=view)

M3LAB1 - TextAdv prototype

As a reminder, here's the latest prototype code:  
 [textadv21v4.zip](https://faytechcc.blackboard.com/bbcswebdav/pid-6079382-dt-content-rid-55374695_1/xid-55374695_1)

For this version, replace the game setup code to use your five rooms and room layout.

Then test to confirm that using commands in the "go north", etc., format, you can reach all five rooms.

(I'll also ask you not to adopt my fast-coding bad habits, such as some pretty weak error checking. This stuff will get improved over time but you may find it easier to go ahead and make changes.)

### M3Pro Objects ( assignment Instructions)

[M3Pro Objects ( assignment Instructions)](https://faytechcc.blackboard.com/webapps/blackboard/content/listContent.jsp?course_id=_35059_1&content_id=_5950111_1)

Open the link below to view assignment details

[M3Pro StudentClass \_Assigment Instructions.docx](https://faytechcc.blackboard.com/bbcswebdav/pid-5950114-dt-content-rid-54266709_1/xid-54266709_1) [M3Pro StudentClass \_Assigment Instructions.docx - Alternative Formats](https://faytechcc.blackboard.com/webapps/blackboard/content/listContent.jsp?course_id=_35059_1&content_id=_5950111_1)  
 (opens in a new window)

### [M3Pro-Objects](https://faytechcc.blackboard.com/webapps/assignment/uploadAssignment?content_id=_5950116_1&course_id=_35059_1&group_id=&mode=view)

Download instructions from post above ( **M3Pro Objects** (**Assignment Instructions)**)

**Submit** your finished code solution file(s) through this assignment link by the posted deadline

### [M3LAB2](https://faytechcc.blackboard.com/webapps/assignment/uploadAssignment?content_id=_6135796_1&course_id=_35059_1&group_id=&mode=view)

M3LAB2 - Technology Assessment

For this assignment, you'll break into small groups and assess a specific library or technology for possible use with the Text Adventure later.

Some options:

- Text based console (plain print/input, or possibly another technique)

- Flask (Webapp) similar to how 9:05 is played online

- PyGame (tile based? vector based?) with text windows and possibly extra graphics

- Discord bot (players play through messaging the bot)

This can be a page or two of info, and a simple work in progress / prototype (this can be almost anything simple)

**Resources:**

Pygame book: <http://inventwithpython.com/makinggames.pdf>

PyGame - Snake game: <https://youtu.be/9bBgyOkoBQ0>  
<https://github.com/kiteco/python-youtube-code/tree/master/snake>

PyGame Performance (good background): <https://www.youtube.com/watch?v=hnKocNdF9-U>

### [M3HW1](https://faytechcc.blackboard.com/webapps/assignment/uploadAssignment?content_id=_6137411_1&course_id=_35059_1&group_id=&mode=view)

**M3HW1 -**Text Adventure Prototype 2

(due 11/1)

For this assignment, do the following:

- Set up a project board on your GitHub repository for TextAdv if you have not already done so

- Add the relevant user stories ("As a player, I want to [feature] so that I can [reason]") for rooms and items, at first. (You'll want to add more as we go on.)

- Move the user stories you're working on, or are completed, over to the relevant column.

- Deliver a working prototype of the game, with the current features in place. At minimum you should be able to walk around rooms and fiddle with objects (get, drop, etc.)