Class 9

[w200] MIDS Python Bridge Course Summer 2018

MISSION COMPLETED! | Programming Stream

Unit 1 | Introduction, the Command Line, Source Control

Unit 2 | Starting Out with Python

Unit 3 | Sequence Types and Dictionaries

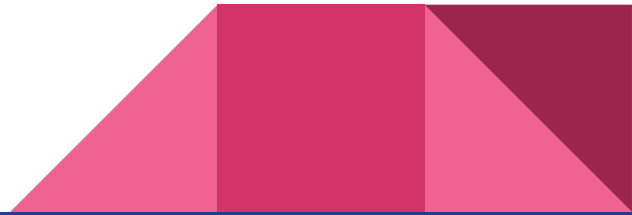
Unit 4 | More About Control and Algorithms

Unit 5 | Functions

Unit 6 | Modules and Packages

Unit 7 | Classes

Unit 8 | Object-Oriented Programming



Up next | numpy, strings, Data analysis

Project 1 | build your own object oriented project

- Code at home and collaborate in class

Unit 9 | Working With Text and Binary Data

Unit 10 | NumPy

Unit 11 | Data Analysis With Pandas

Unit 12 | More Analysis With Pandas

Unit 13 | Testing

Reminders |

Course Schedule

https://docs.google.com/spreadsheets/d/11DxadnNwyFaJIPYLUJSPUlNGCtTenBCR4yaR1CbFBKg

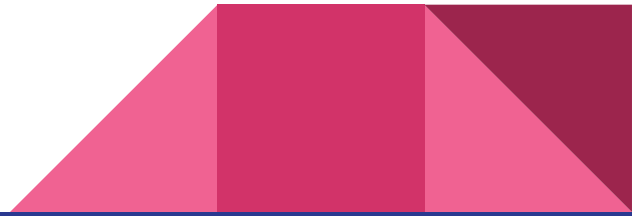
Agenda

Project Due Date

Midterm - Discussion

Project - Class Discussion

Project - Breakout Discussion and Troubleshooting

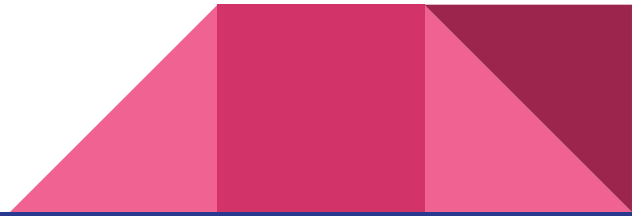


The Project | Your Mission

Create a small, object-oriented program of your choosing:

Examples:

- An ATM
- A flower shop
- An adventure game
- Something relating to your everyday work



The Project | Code

Python 3 code, 300-500 lines (750 max)

All code should be well commented!

Must use Object Oriented design and classes

Demonstrate various flow controls and data types

Robust to common user errors and exceptions

The Project | Your Mission

The user will interact with your program via Terminal/Shell

Three documents due before your class next week:

1. Proposal (10%)
2. Code(s) (80%)
3. Reflective Summary (10%)

You will demo your progress in a breakout room (today)

You may only use Python libraries that come installed with Anaconda

The Project | Proposal

Describe your project concept

Pseudocode your major classes and functions

1. Briefly describe the purpose of each class
2. List expected functions belong to each class
3. List inputs and outputs for each function

Instructors will “approve” your draft proposal

Coding is **iterative**. Your final code may not match the proposal exactly

The Project | Reflection

Submit a 1-page reflection with your code

Instructors will read your reflection before grading your project

Tell us how to use your project!

Discuss challenges you faced and how you overcame them

Agenda

Project Due Date

Midterm - Discussion

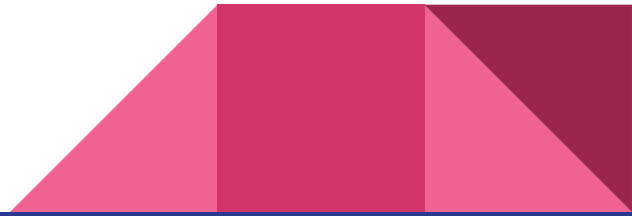
Project - Class Discussion

Project - Breakout Discussion and Troubleshooting

Midterm | Discussion

Please open up your midterm notebook.

Let’s chat through the problems...



Agenda

Project Due Date

Midterm - Discussion

Project - Class Discussion

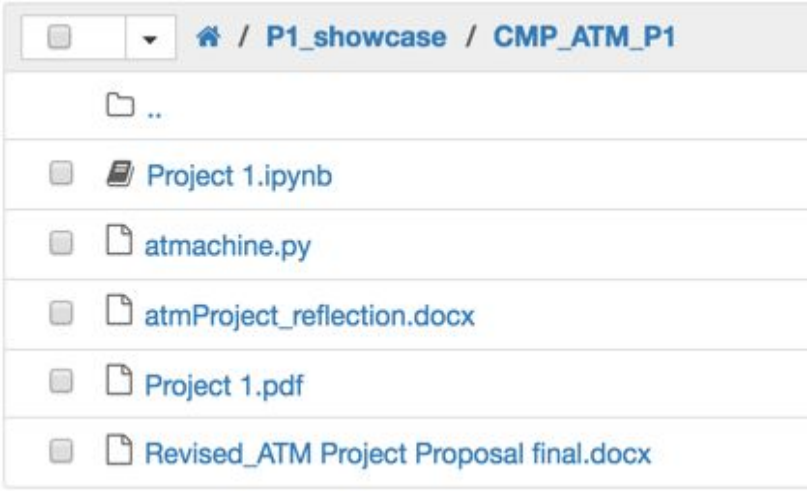
Project - Breakout Discussion and Troubleshooting

The project 1 example set

https://github.com/MIDS-INFO-W18/assignments_upstream_summer18/tree/master/project_1/past_examples

Example project | Atm machine (ATM_CMP_P1)

- Simple project
- Clean compact code
- Single reflection doc
- single .py file



Example project | Atm machine (ATM_CMP_P1)

- Run by a single function
- Self explanatory prompts
- Authentication

```
if __name__ == "__main__":
  # Initialization
  bank = Bank()
  atm = Atm(bank)

  # Main Page -> Account page
  print("Welcome to the ATM machine!")
  while True:
    cmd = input("Please enter a command([l]ogin - Lo
application) or 'h' for help\n")
    if cmd == "h":
      print("[l]ogin - Log in to your account")
      print("[c]reate - Create a new account")
      print("[q]uit - Close the application")
```

Example project | Atm machine (ATM_CMP_P1)

- 4 classes:
  - Bank
  - ATM
  - Account
  - Central window
- The main function initializes
- Central window also manages
- Errors are implemented as classes

```
#These classes handle different types of ex
class AuthenticationError(Exception):
  pass;

class DuplicateAccountError(Exception):
  pass;

class NotLoggedInError(Exception):
  pass;

class InsuffficientFundsError(Exception): #
function
  pass;
```

Example project | TownTasks_MJT_P1

Single .py file

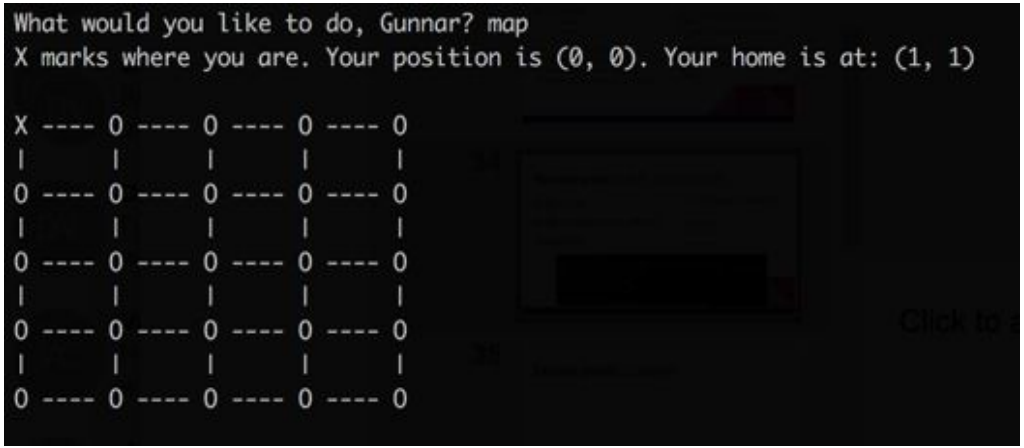
Single completion doc with tips

Detailed help

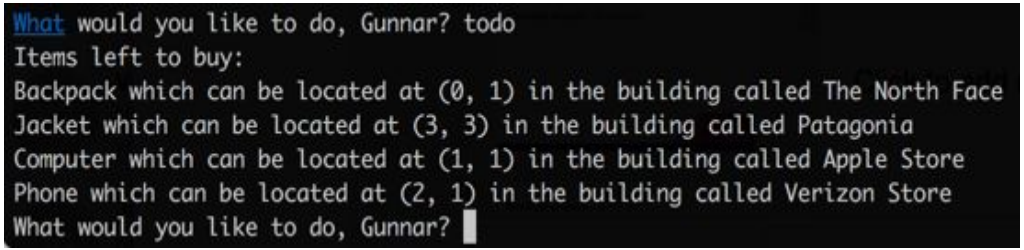

```
What's your name?: Gunnar
You woke up with a bunch of errands to do! Type 'todo' to see what you have to do.
IMPORTANT: Type the word help into the 'What would you like to do?' prompt to reveal the details of your
character and the list of available commands.
What would you like to do, Gunnar? help
Your name is Gunnar and you have $5000
You are currently in the position (0, 0)
Here is a list of the possible commands:
quit: quits the game
buildings: prints a list of building names that you use to enter a building
```

Example project | TownTasks_MJT_P1

- I get the map



- Informative report



Example project | TownTasks_MJT_P1

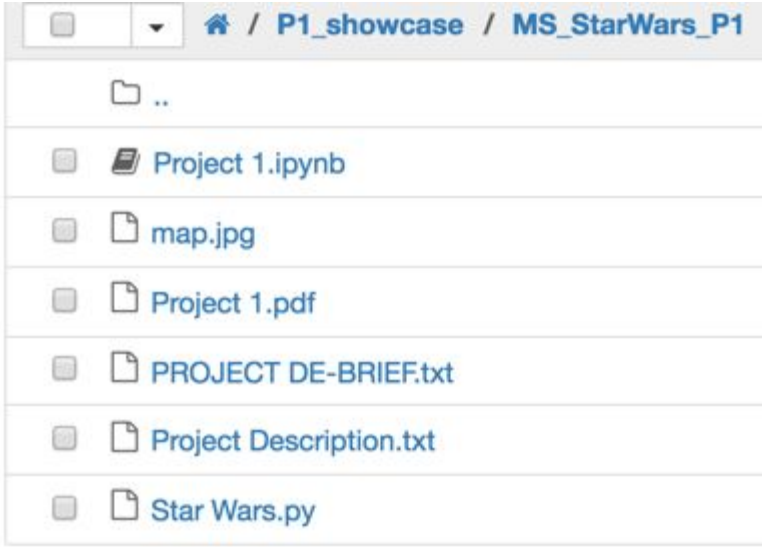
- Relies on 5 parts
  - Top script (menu and help)
  - City (initializes the human...)
  - Human (person attributes)
  - Point (stores map)
  - Building (enter and get items)
- Compact (less than 300 lines)

```
class City:
    """ Represents the city object. Has streets
    items = ['Backpack', 'Flashlight', 'Jacket'

    def __init__(self, person, x, y):
        self.person = Human(self, person)
        self.items = []
        self.x = x
        self.y = y
        self.locations = {}
        self.house = (1,1)
        self.points = self.generatePoints()
        self.person.setPos(0, 0)
```

Example project | Star_Wars_Manish_Singh

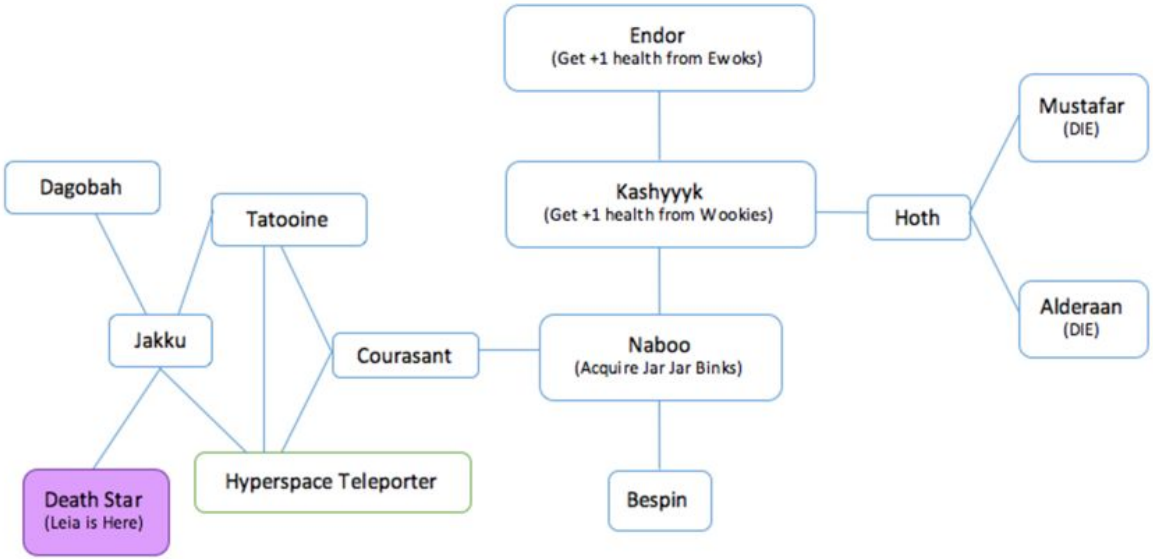
- This one has more parts
- There is a static map
- There are 2 project descriptions



- Adventure games have been the most involved

Example project | Star_Wars_Manish_Singh

- He gives us the underlying structure of his game !



Example project | Star_Wars_Manish_Singh

- Much more detailed code
- 5 classes multiple functions
  - item
  - Player
  - character
  - Planet
  - Room

```
class player:
    def __init__(self, character):
        if character == "Luke":
            self.name = "Luke"
            self.last_name = "Skywalker"
            self.weapon = "Lightsaber"
            self.weapon_damage = 8
            self.partner = "R2-D2"
            self.health = 10
        elif character == "Han":
            self.name = "Han"
            self.last_name = "Solo"
            self.weapon = "Blaster"
            self.weapon_damage = 5
            self.partner = "Chewbacca"
            self.health = 13
        self.fanny_pack = ["bread", "bread"]
```

Discussion | Complicated “IF” Statements

How many students have complicated “if” statements in their code?

What are some solutions to help deal with it?

Discussion | Dictionaries with Objects

Is anybody using a dictionary of objects in their code?

Discussion: How is this useful?

Discussion | Dictionaries with Objects

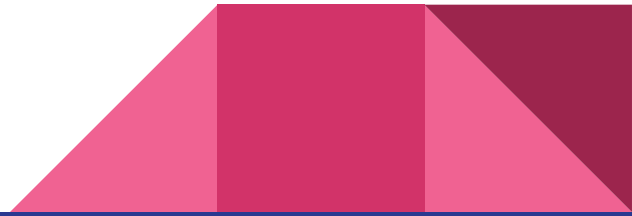
```
dict['str1'] = func1
dict['str2'] = func2
and so on..

def main_func(str):
    return dict[str]
```

```
def main_func(str):
    if 'str1':
      return func1
    elif 'str2':
      return func2
```

http://stackoverflow.com/questions/11445226/better-optimization-technique-using-if-else-or-dictionary

Example project | Questions



Agenda

Project Due Date
Midterm - Discussion
Project - Class Discussion
Project - Breakout Discussion and Troubleshooting

Project | Breakouts

We will put you into breakouts. Please discuss your projects and use your teammates to help think through any roadblocks.

If you would like to talk with me, send a message and I'll stop by your room when I can.