Overview

- We're writing a tool to help game masters manage the mechanics of a game
- Think of this as "the screen the DM uses to keep track of things"

Day 1

- Objects/Behaviors/Operations
- Working!
- Tested!

Characters

- Properties
  - attack hit weight
  - defense hit weight
  - attack damage weight
  - defense damage weight
  - health
  - Items
  - experience
- operations
  - get effective stats
  - heal

MOBs

- properties
  - attack hit weight
  - defense hit weight
  - attack damage weight
  - defense damage weight
  - health
  - Items
- operations
  - get effective stats
  - heal

Items

- attack bonus
- defense bonus

Character librarian

- CRUD
- persist characters

MOB librarian

- catalog of available mobs
- mobs non-persistent

Item librarian

Operations

- Combat!
    - individual turns
    - entire attack sequences
    - Bonus: item drops

Day 2: Microservices

- Add microservices for Librarian and Combat
- Retrieve characters and MOBs from Librarian
- Use Python FLASK (example provided in my example code) as your framework
- Combat should have the 2 combatants POSTed to it, should execute the combat, and return the results **AND** the combat sequence (who hit who for how much)
- Preserve the results with the Librarian