Mocking – CS Term

Architecture of our game (Model-view-controller: MVC architecture)

A CLI (Command Line Interface) for the user to type commands in. Those commands should then be analyzed by the game engine. This game engine should save state to yaml if need be (1. Restarting game later 2. For testing purpose 3. IT IS REQUIRED TO BE IN YAML (Design constraint))

Our State.yaml file can be read by our game engine.

The game engine should still run even if the user is not using the command line (like a web page).

A design heavily advised for us is a repl. (REPL: Read-Eval-Print-Loop) : Read input, evaluate something on that input, print out that action, and go back. Infinite loop, and only exits by a certain condition you put into the repl.

Game Engine: A map system that is cardinal-based. You must be able to move along all the cardinal directions in the room. CLI should be able to take in commands that allow you to move in between rooms.

There should be items in the room that the user can interact with.

NPCs should be treated as a player. (not in scope for this sprint)

State(What room I am in, All my objects)

Yaml is used for configurations or a saved file. It is a configuration to run a program. It is a description. It is a markup language (ex: HTML is a markup language that describes stuff, but the web browser generates it)

Test

Write a test that if you pass none, you should get back none or empty string or exception or etc.

A test that previously and does not work anymore is also called “broke tests” and “regression”

is and == are different in python