

Imaginarium assignment 1:

A Medieval Fantasy Shop Generator

Introduction

In this assignment, you'll work with a generator for shops (businesses) in a D&D-like medieval fantasy setting. It defines a number of different kinds of shops and shopkeepers.

- **Shops:** magical supply store, apothecary, materia shop, healer, ritual objects shop, magical herb shop, scary animal parts shop, bakery, armorer, jeweler, butcher, brewer and fishmonger.
- **Shopkeepers:** human, child, dog, cat, [kitsune](#), bear, owl, and spider.

Each of these has a set of properties that can apply to them. Shops have alignments (shopkeepers are assumed to have the same alignment) and levels of expensiveness. Shopkeepers have attributes related to their appearance, intelligence, wisdom, greed, and so on.

Getting started

To begin with, drag the Fantasy Shops folder into the Generators folder on your machine (Generators is inside the Imaginarium folder inside your Documents folder). Then, open the Fantasy Shops folder you just put inside the Generators folder, and open the file `shop.gen` in Visual Studio Code.

Finally, start Imaginarium, and select the Fantasy Shops generator you just added to the Generators folder.¹ You'll see two buttons: **Shops** and **Test**.

Press Shops a few times to see what kinds of things are getting generated. Now look at `shop.gen` to get a sense of how the system is generating those things.

Part One: Adding constraints

You'll probably notice that the system is generating some questionable combinations, like tall spiders. We've peppered the code with `should not exist` statements. These are tests that are automatically run when you press the **Test** button (or type the command `test`). Your first job is to add statements to the file that will outlaw the problematic combinations that shouldn't exist, so that when you press **Test**, it tells you that all the tests passed:

- Find the first test. It says "non-evil scary animal parts shops should not exist". If you hit the **Test** button, you'll see that that test fails and prints in red.
- What is a line that you can add that will ensure that when the system produces scary animal parts shows, they're always evil? Add that line to the file, save it.

¹ To select the generator, press ESC to get to the **Main menu**, then click **Select generator**, and then click **Fantasy Shops**.

- Go back to Imaginarium and press **ESC twice**. The first ESC will move you to the Main menu, but the second ESC will bring you back, reloading the file in the process. Now you're ready to try your new version.
 - If Imaginarium displays some kind of an error message, that almost certainly means that the line you added has a problem. Read the message and try to fix the line, the resave and reload (ESC ESC) again.
 - If you can't figure out how to fix it, come to office hours or write on #imaginarium-questions on Discord.
- Press **Test** again.
 - If that first test is now passing, congratulations! Just move on to the next test and repeat the process.
 - If it's not passing, it will display an example of a shop that contradicts the test. Ask yourself why it is that the line you wrote doesn't rule this example out, then try to formulate a new version that does rule it out.

When you have all the tests passing, congratulations!

Part Two: Fun (honor system)

Now extend the generator in some way that you find fun and/or interesting. You can add whatever you like, but you should at a minimum add:

- 4 new kinds of shops
- 6 new adjectives for shops
- 3 new kinds of shopkeepers
- 6 new adjectives for shopkeepers
- 5 new constraints limiting the application of adjectives (such as you did in part one).

Turning it in

To be paranoid, we recommend you:

- Save your file
- Quit imaginarium
- Start imaginarium again
- Press the Test button and make sure all the tests still pass

Assuming all the tests pass, just upload your shop.gen file to Canvas. For this assignment, your score on this assignment will just be the proportion of tests that pass when we run them, so if all your tests pass, you are guaranteed to get a perfect score!