**FoxDenier** by Bablab3D

For Unity Learn Junior Programmer Pathway: Programming theory in action.

**Player control**

The player will have a top-down view and be able to click and drag on the ground to create temporary obstacles for the animals.

**Basic Gameplay**

In this game, chickens, foxes, and moose will all exist in a small ecosystem. Chickens will breed, foxes will try to eat chickens, and moose will protect chickens if they are being hunted. The player’s goal will be to protect the chickens by placing obstacles in front of the foxes.

**Sounds and Effects**

Chickens will constantly cluck and scream when chased, moose will moo angrily when chasing foxes, and foxes will be silent.

When chickens are eaten, they will disappear into a cloud of dust. When chickens mate, the new chicken will appear with a sparkle.

**Gameplay Mechanics**

Over time, new foxes will appear, making it harder to keep them away from chickens.

**User Interface**

At the start of the game, some instructions will appear showing that chickens mate, foxes eat chickens, and moose chase foxes.

**Class Architecture**

* Game manager static class singleton
* Animal class (loiter, pursue, caught, rest, has nav mesh agent)
  + Chicken child class (pursue chickens to mate, run away from fox when being chased, destroyed when caught by fox)
  + Fox child class (pursue chickens, run away from moose when caught)
  + Moose child class (pursue foxes when foxes pursuing chickens)
* Player controller class (create obstacle where clicked)

Components:

Has nav mesh agent

Variables:

* Boolean if resting
* Boolean if pursuing
* String target tag
* Float

New Methods:

* Loiter
  + Move in a random direction for 2 seconds then move in a new random direction
  + If hit a wall or obstacle, turn around and move away from the obstacle for 2 seconds
* Pursue
  + Move towards target at a faster pace
  + If very close to target, make target caught
* Caught
  + Will depend on animal, but all animals will do something when caught
* Rest
  + Stand still for