Raycasting Homework (Due 2/20)

In this assignment, you are going to get some extra practice with raycasting, arrays and loops.

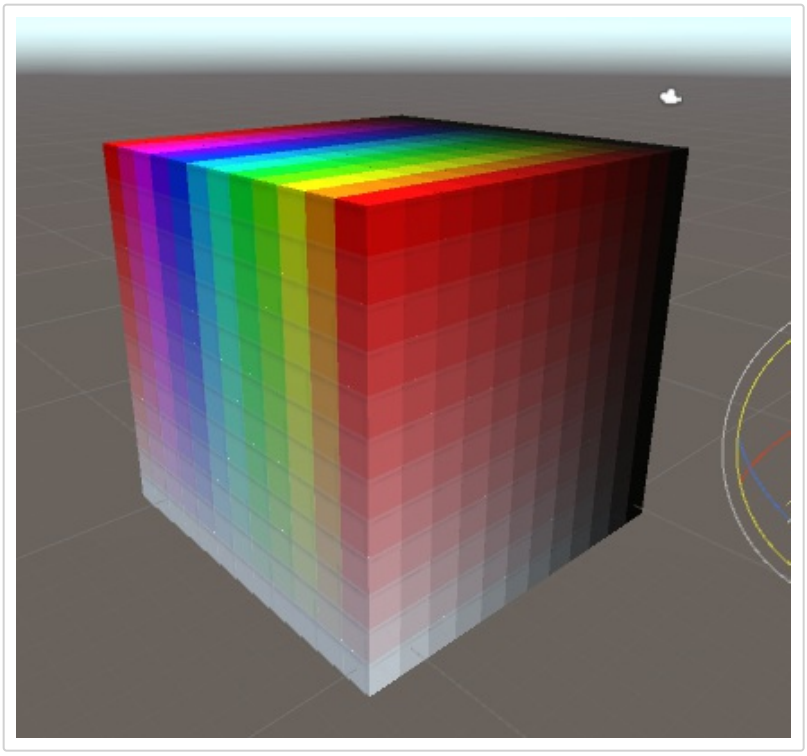
Project Setup

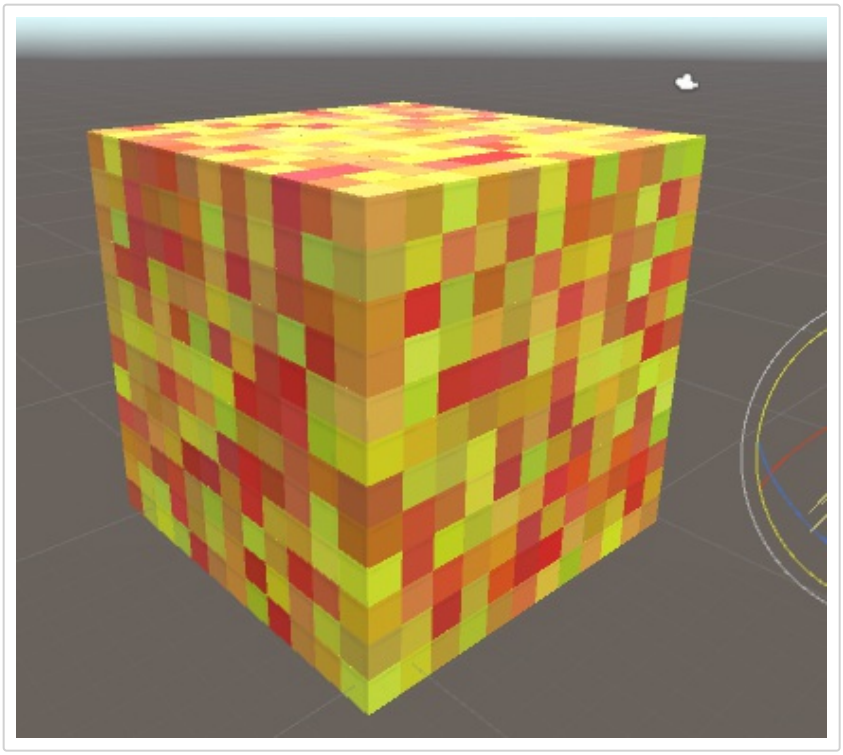
Create a new project (put your name in the title) and then:

1. Import the "Characters" Unity package. (`Assets -> Import Package -> Characters`)

Part 1: Creating a Cube of Cubes

1. Create an empty game object.
2. Attach a new script (GenerateGrid.cs) to it.
3. Use the script to spawn a 3D grid of cubes. Hints:
    - You'll need a prefab for this.
    - Remember that we could create a 2D grid by nesting one loop (for the x position) inside of another (for the y position)? You'll need three loops here - one for x, one for y and one for z.
    - If you are stuck, start by creating a row. Then try turning that in a 2D grid. Then finally into 3D grid.
4. Randomize the color of the cubes, or use your loop variables to control the color of the cubes. It could look something like either of the following:





Part 2: Raycast Digging

1. Bring in the FPSController (not the RigidBodyFPSController) for this exercise. Position your character on top of the cube grid you are generating.
2. We want this FPSController to be able to fit inside of 1 meter gaps, so change the character controller component's radius to 0.3.
3. Create a new script (RaycastDig.cs) and attach it to the player.
4. Write the script so that when the player left clicks, a ray is cast from the player. If the ray hits a cube within a few meters, destroy it.
   - To destroy a cube, you will need to use `Destroy(...)`. You need to pass it a `GameObject` instance, which you can get from a `RaycastHit` object.
5. Turn down the lights and remove the skybox. When the player right clicks, place a "torch" a couple meters in front of the player *only* if there is nothing in front of the player.
   - Make your torch like the "light bulbs" from the popping lights exercise.
   - For placing the light, you can get a point along a ray using `Ray.GetPoint(...)`.
6. (Bonus) Dealers choice: explosions or laser beams. If the player presses the "e" key:
   - Explosion - destroy everything within a short range of the player using `Physics.OverlapSphere`.
   - Laser beam: destroy everything within a beam area using `Physics.SphereCastAll`. This is like raycasting, except that instead of sending out a line, you send out a beam with a thickness.
   - Be careful not to destroy the player. Hint: remember tags?

In the end, it will look something like:

- Digging and lights video
- Explosion video
- Beam video

Submitting the Assignment

Before the start of class on 2/20, direct message me on Slack:

1. A zip of your Unity project folder. Note: the project folder is the one that contains `Assets`, `Project Settings`, etc. If you share a zip of the `Assets` folder itself, I won't be able to see your project - it needs to be the folder that *contains* the `Assets` folder.