Mod the Cube

Summary : In this exercise, you'll be challenged to modify a provided script to change the behavior of an object it's applied to.

Type:Tutorial

Level:Beginner

Xp: 60
Estimated time to complete: 30 Mins

Materials: ModTheCubeChallengeAssets.zip

1.Overview

In this mission you learned how to open, read, and modify scripts. You practiced fundamental programming challenges, and hopefully you're starting to feel comfortable with basic scripting. In this exercise, you'll be tasked with modifying the behavior of an already existing script. We're leaving this challenge open-ended, so now's the time to get creative!

2.Download the provided assets and import them into a new Unity Project

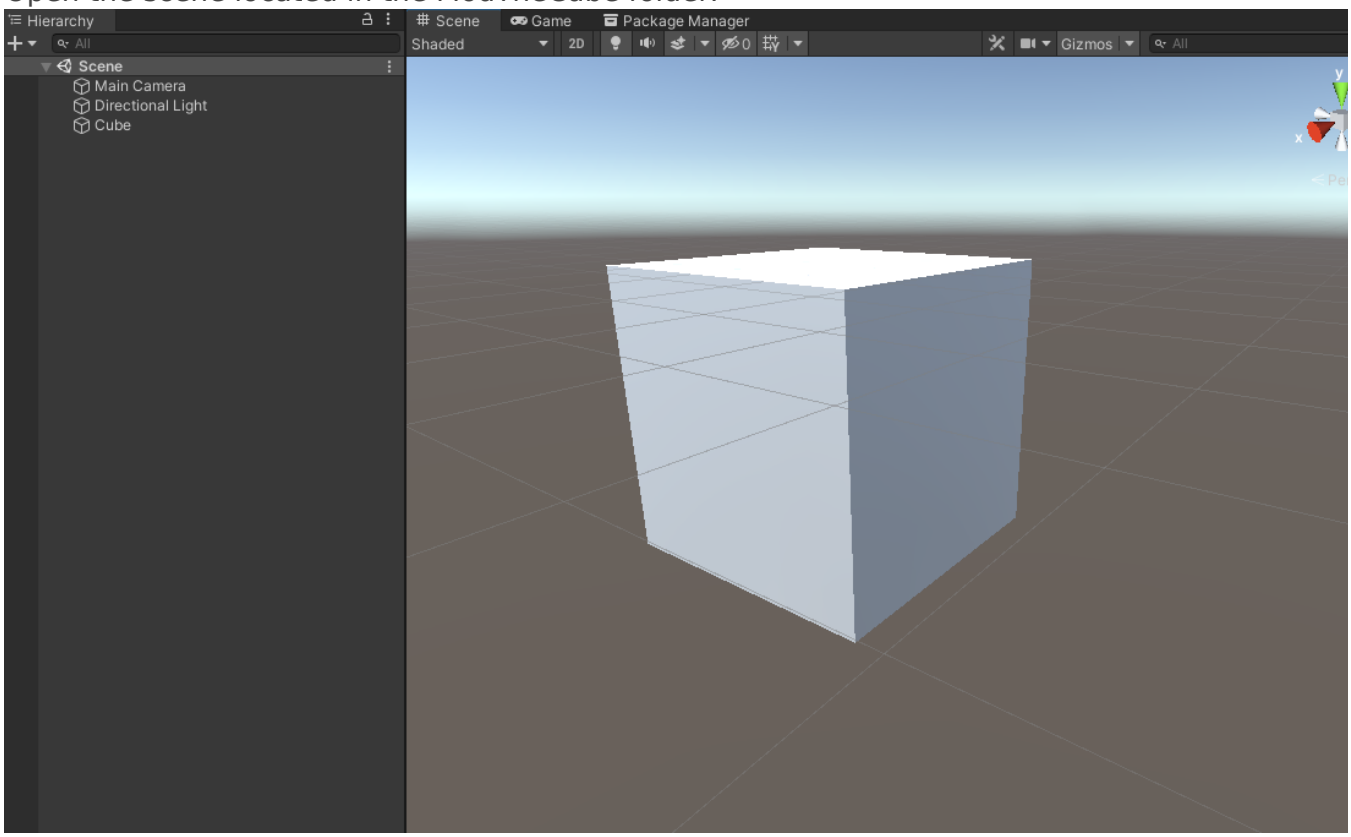
1. Download ModTheCubeChallengeAssets.zip.
2. To unzip the contents of the folder, right-click the folder and select **Extract All** (macOS: double-click the zipped folder). Inside you will find ModTheCube.unitypackage.

   > **Note:** While it is possible to view the contents of a folder while it's still zipped, the data is still compressed. Attempting to import the contents of the ModThCubeChallengeAssets folder while it's still zipped will result in an error message in Unity. Ensure that you have unzipped the folder before proceeding to the next step.
3. Create a new Unity project using the 3D template - remember to use one of the supported versions of Unity (2018.4LTS, 2019.4LTS, or 2020.3LTS).
4. Import ModTheCube.unitypackage into your new 3D template project.

3.Observe the default script functionality

Before you begin editing the code, take a few minutes to understand what the script does by default. The imported .unitypackage contains three items: a C# script, a material, and a scene.

1. Open the scene located in the ModTheCube folder.



   The only thing in the scene aside from the Camera and Directional Light is a cube. Attached to the cube is the script you will be modifying.
2. Enter Play mode (select the play button on the toolbar) and observe the default behavior of the script assigned to the cube.
   When you enter Play mode, you will observe several of the cube's properties change. All of these changes are hardcoded into the cube.cs script attached to the cube in the scene. This means that the cube will be modified in the same way each time the scene is run.

4.Modify the cube.cs script

Now that you understand what the cube script is doing, you can begin to customize it. As previously mentioned, all of the changes the script is making to the cube are specified in the code. It would be easier to control the cube from the Unity Editor instead of editing the script. You can make this possible by changing variables to be public. Public variables appear in the Inspector window where you can modify their values.

1. Open the cube.cs script in your code editor and observe the code.
2. Identify at least two changes that you can make to the cube's behavior.

- Change the cube's location (transform).
- Change the cube's scale.
- Change the angle at which the cube rotates.
- Change the cube’s rotation speed.
- Change the cube’s material color.
- Change the cube’s material opacity.If you’re feeling confident, you can attempt the following as well:
- Modify any of the changes above so they change randomly each time the scene is played.
- Add extra functionality to the cube. For example, how might you change the color of the cube over time?

3. Implement your planned changes in the code.
4. Test the scene regularly as you work to ensure your code is performing as expected.

5.Create a WebGL project showing off your modded cube

Now that you have edited your script and thoroughly tested your changes, your remaining task is to submit your project here and show off your work! You will create a WebGL build of your project, upload it to Unity Play, and then complete this challenge by linking your submission below.

1. If you haven’t already done so, frame your camera in the scene so it’s easy to see the cube in the Play view.
2. Save your scene.
3. Ensure that your version of Unity has WebGL installed. Not sure if it does, or don’t know how to add it? Refer to step 2 of the Create and publish WebGL Builds tutorial.
4. Install the WebGL Publisher package. This package will allow you to easily publish your project on Unity Play. If you aren’t sure if you have the package or don’t know how to add it, refer to the tutorial linked in the step above.
5. Create a WebGL build of your project. This may take a few minutes, depending on your computer. If you need help creating the build, refer to step 4 of Create and publish WebGL builds.
6. Publish your WebGL build on Unity Play. If you need help publishing on Unity Play, refer to step 6 of Create and publish WebGL builds.
7. Copy your Unity Play link, and paste it below!

6.Criteria

You’ll know you’re successful if you’ve met these criteria:

- You’ve modified the cube script so that at least two changes are made to the cube’s behavior.
- You’ve created a WebGL build of your project.
- You’ve uploaded the WebGL build to Unity Play.
- You’ve copied your Unity Play link for submission.

7.Next steps

Congratulations on completing the first mission in Junior Programmer! You now have a strong foundational understanding of the structure of basic scripts and how to make simple modifications. As you continue through the pathway, consider taking time to think about how you might modify other scripts you encounter to customize them to your liking. Sometimes the best way to reinforce what you've learned is to apply it in different ways on your own!

8.Submission

Paste your Unity Play link below!

link to the original unity course

Revision #2
Created 2023-06-18 09:55:33 UTC by naruzkurai
Updated 2023-06-18 10:13:50 UTC by naruzkurai