Bonus Features 1

Driving Simulator

Features **2**

Feature Solutions **3**

Easy - Obstacle Pyramids 3

Medium - Vehicle Spawner 9

Hard - Camera Switcher 15

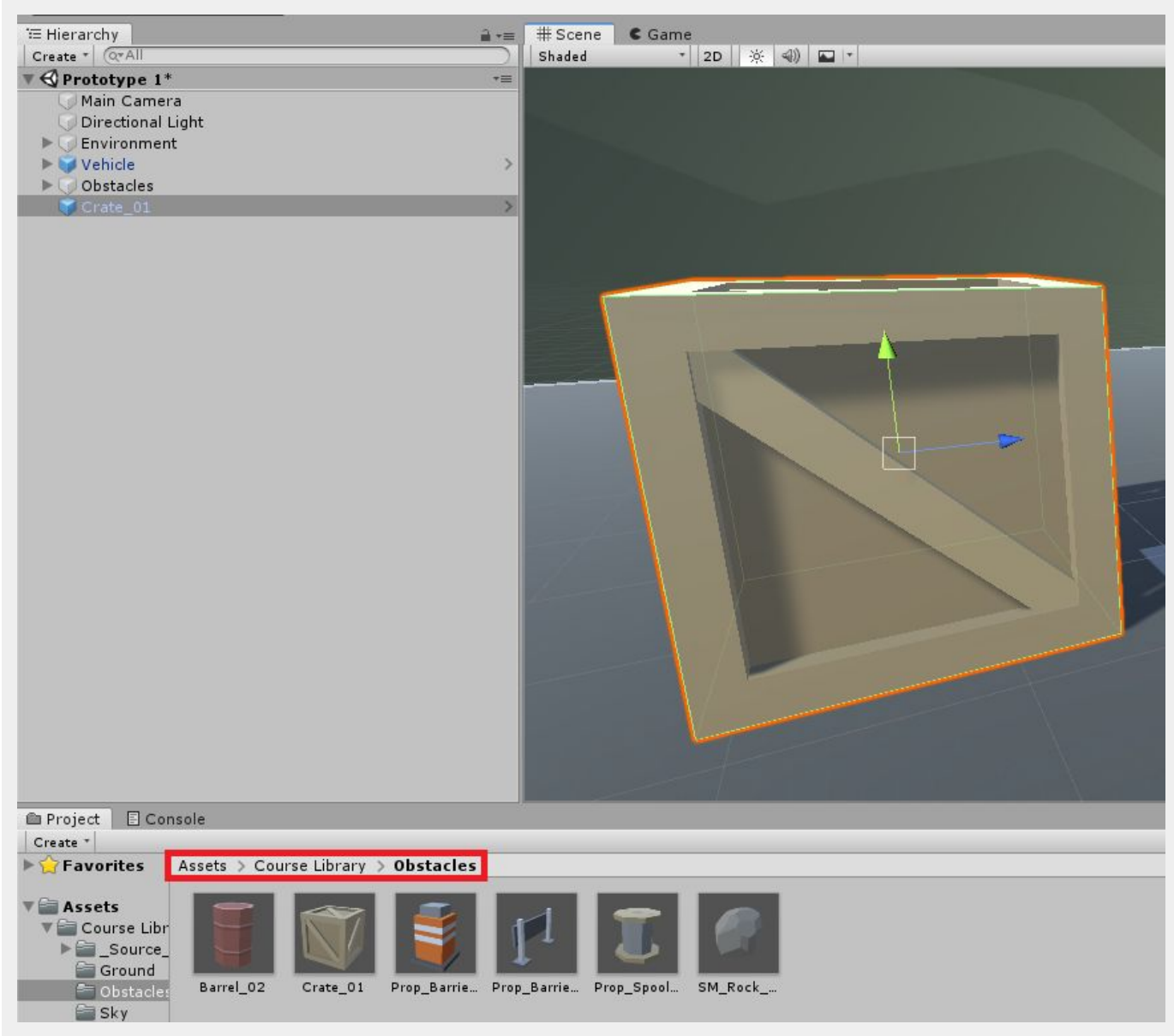
Expert - Local Multiplayer 20

	Features	Hints
Easy Obstacle Pyramids	Create stacks/piles/pyramids of obstacles for the vehicle to drive through.	Remember to add a Rigidbody!
Medium Vehicle Spawner	Add a couple of other cars that are automatically driving down the road, which the player also has to avoid.	Try using transform.Translate to move the car.
Hard Camera Switcher	Allow the player to press a key on the keyboard to switch camera views.	Hint description.
Expert Local Multiplayer	Transform this into a “local multiplayer” split-screen game with two cars where one car is controlled by WASD and the other is controlled by the arrow keys.	Hint description.

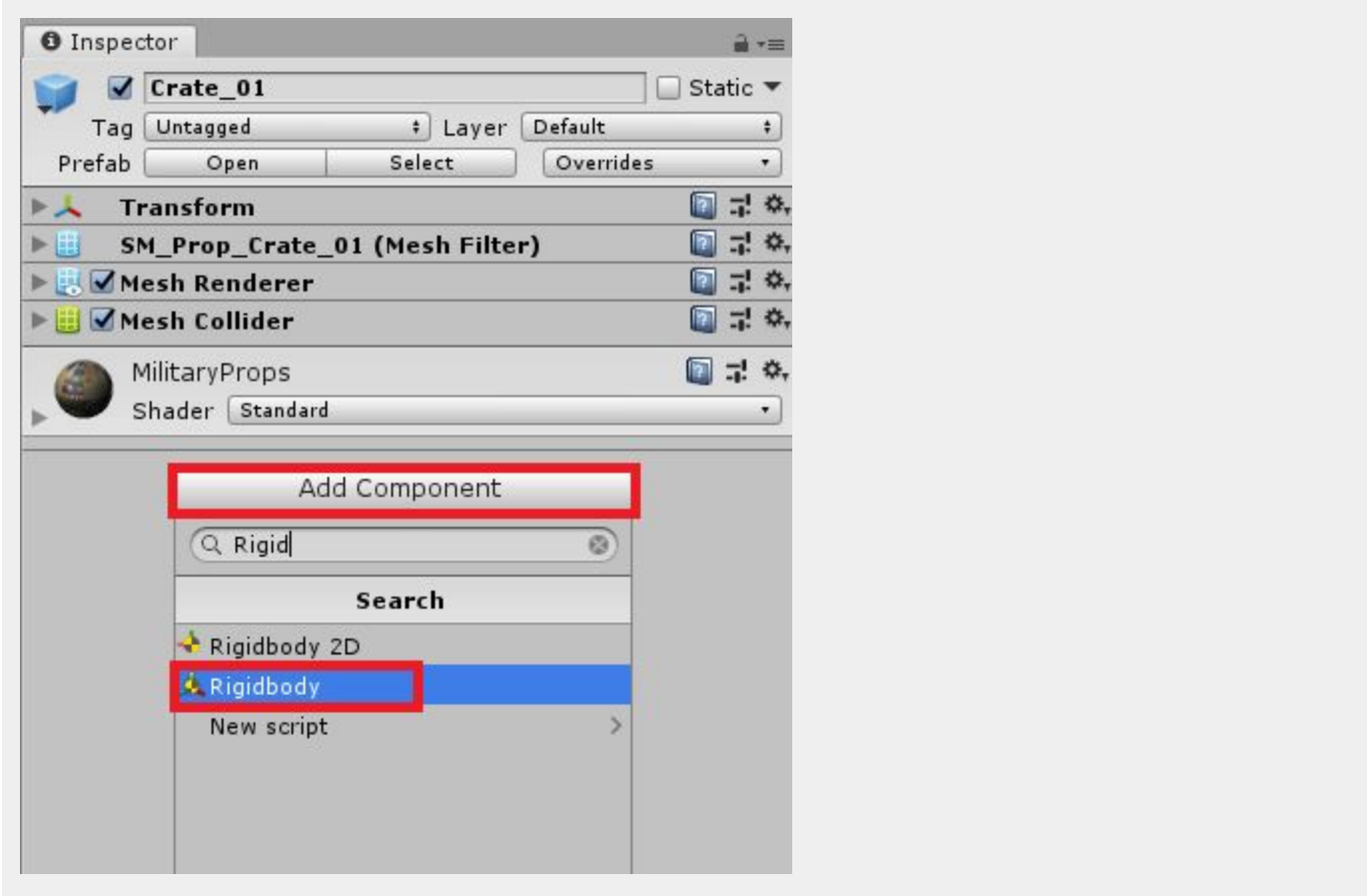
Feature Solutions

Easy - Obstacle Pyramids

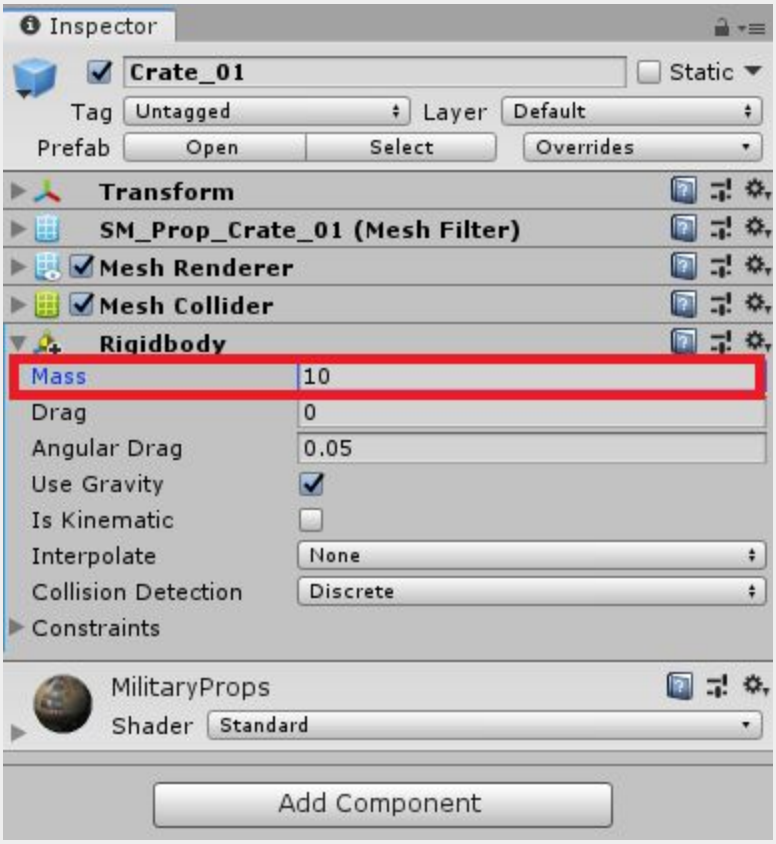
1. Navigate to **Assets > Course Library > Obstacles**. Drag and drop any obstacle you like from the Project Window into the Scene View/Hierarchy



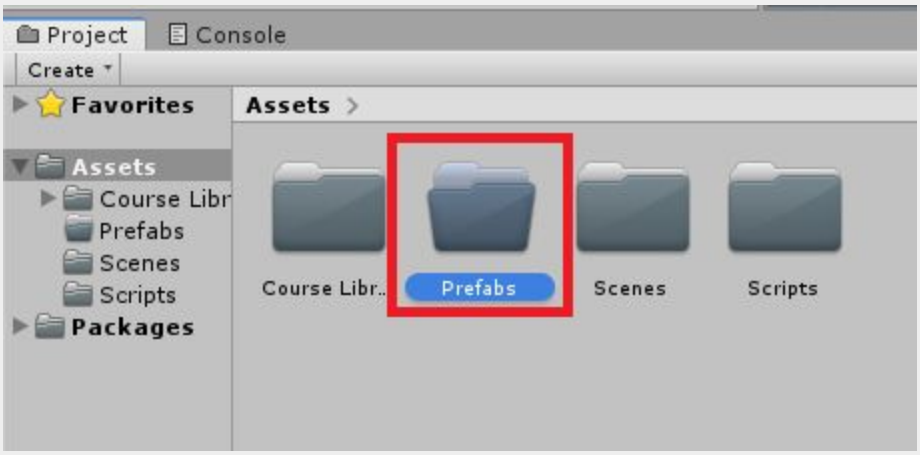
2. With the object select, navigate to the Inspector Window and click the **Add Component** button. Search for a **Rigidbody** and add that to your obstacle Game Object (Make sure it's not the Rigidbody2D component!)



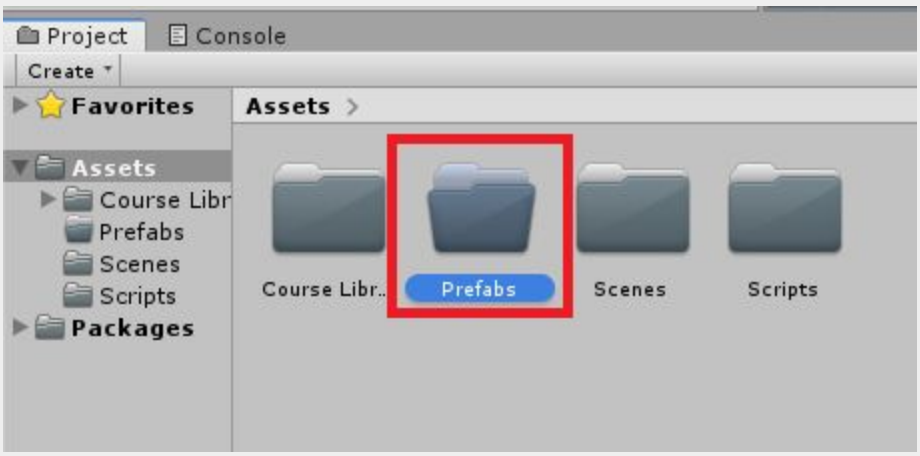
3. Edit the **Mass** value in the **Rigidbody** component to be more realistic! You can enter Play Mode to test your changes and see how heavy your obstacle is by driving into it. Just remember to exit Play Mode before making changes or they will not be saved!



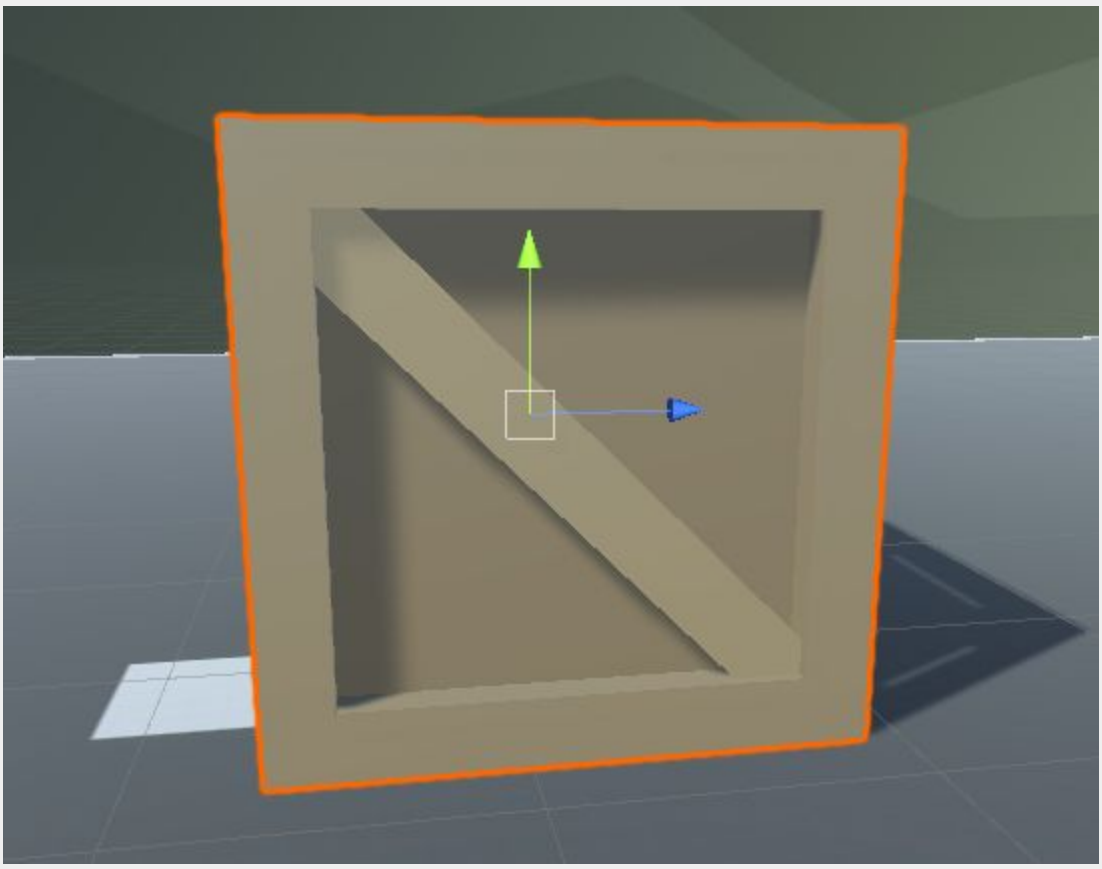
4. In the Project Window, navigate to **Assets**. Right click on an empty space within the Project Window and Choose **Create > Folder**. Name this folder "*Prefabs*".



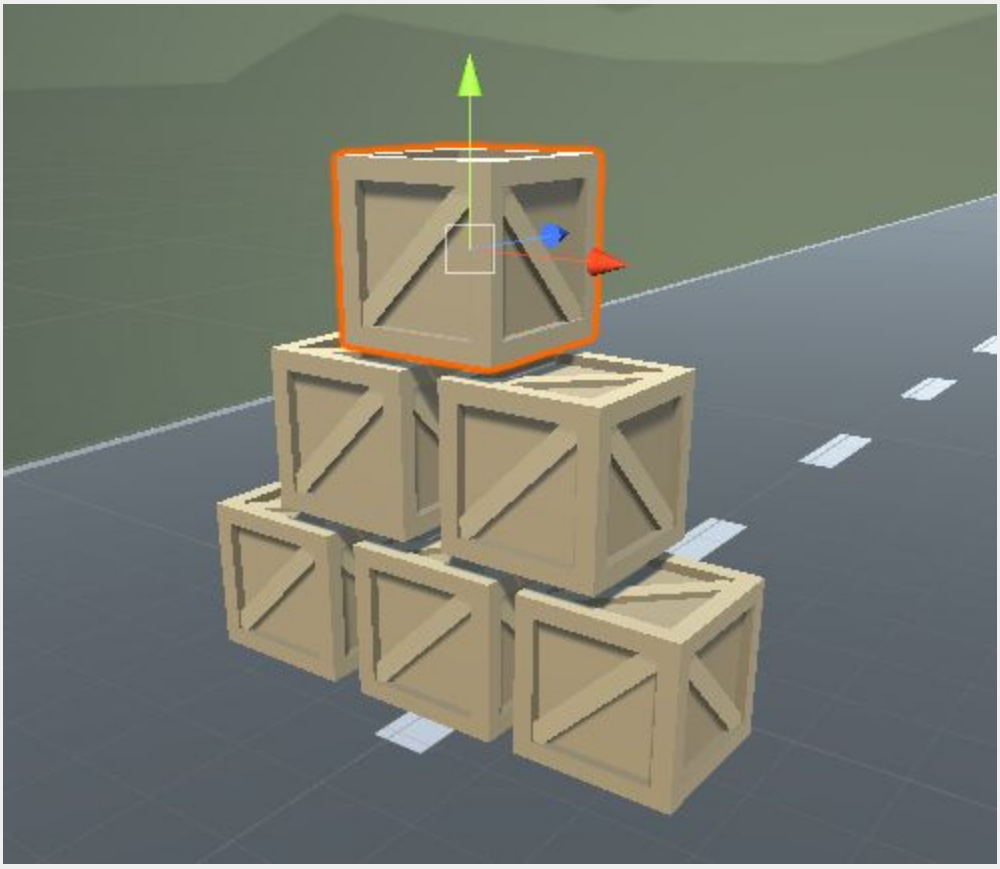
5. From the Hierarchy drag and drop your chosen obstacle to the newly created Prefabs folder. This will enable you to make changes to all existing instances of your prefab within the scene view much quicker. If a pop-up appears, click **Original Prefab**.



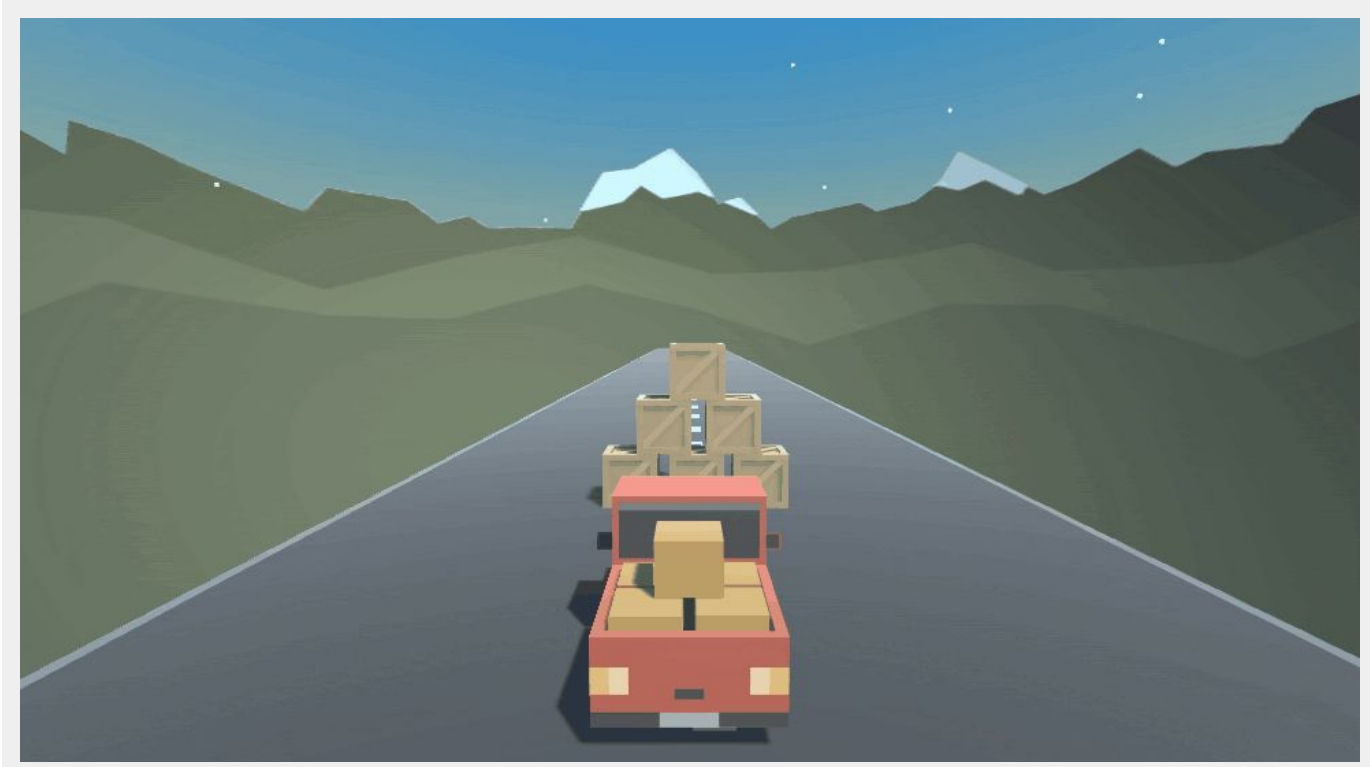
6. Select the newly placed object(s) and reposition them using the Transform Tool
Tip: You can select multiple objects at once by holding down ctrl/cmd whilst selecting. This will allow you to move multiple objects at once!)



7. With your object selected in the Scene View, you can press ctrl/cmd + D to duplicate an individual object or a selection of objects. Create a pyramid out of the new objects.

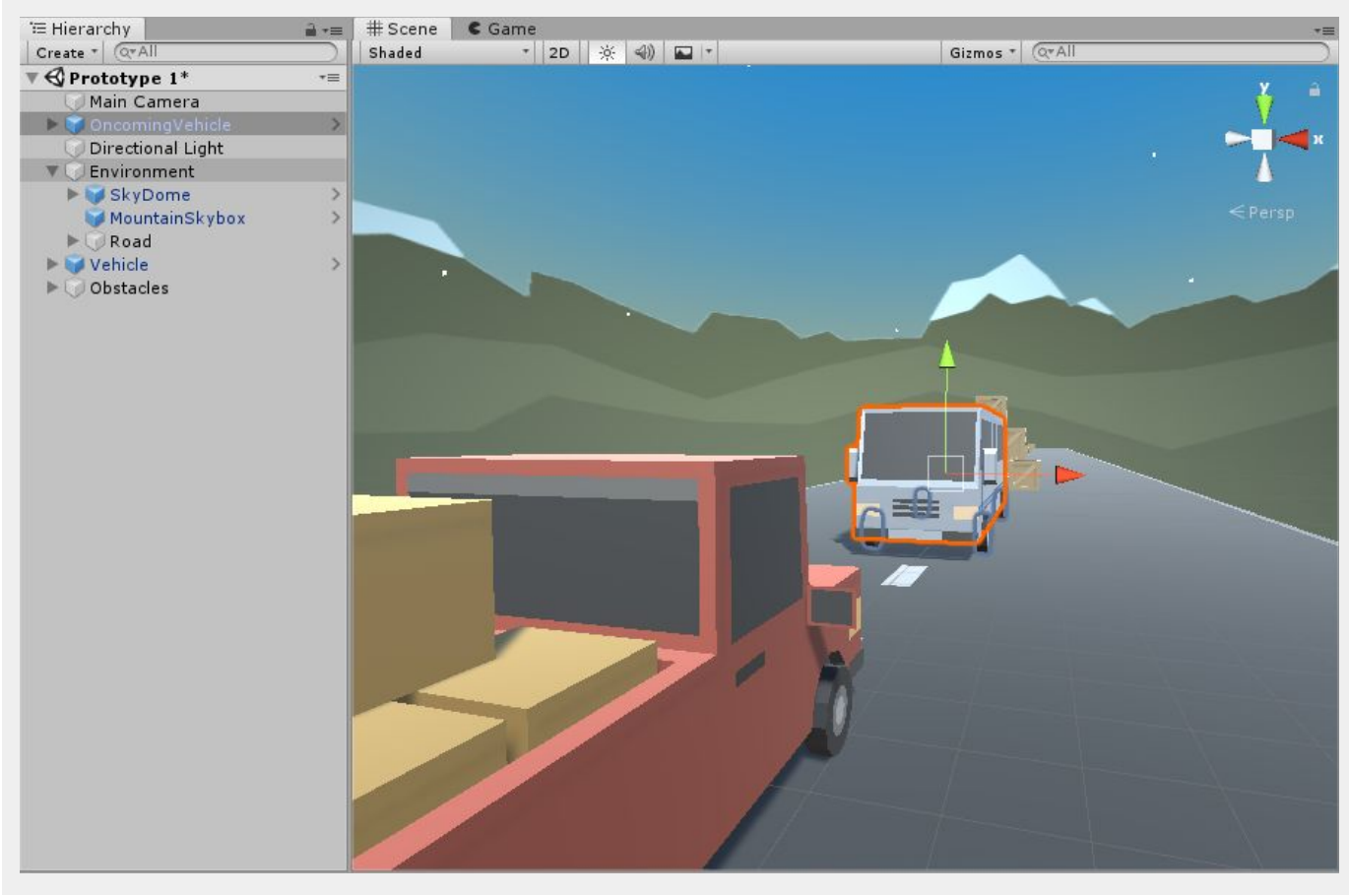


8. Save the scene. Press play and drive into the pyramid.

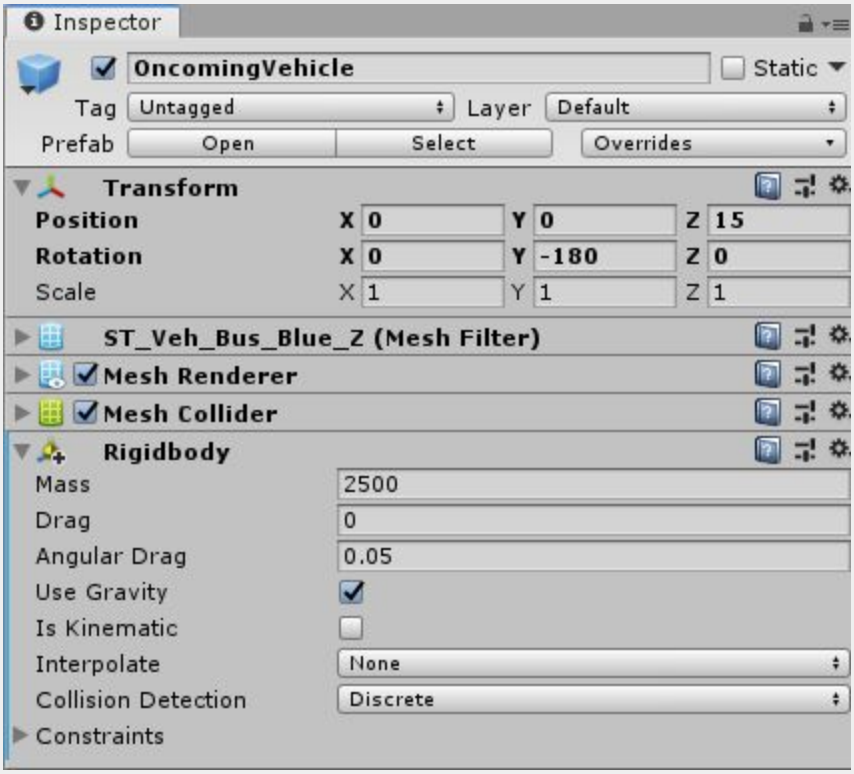


Medium - Vehicle Spawner

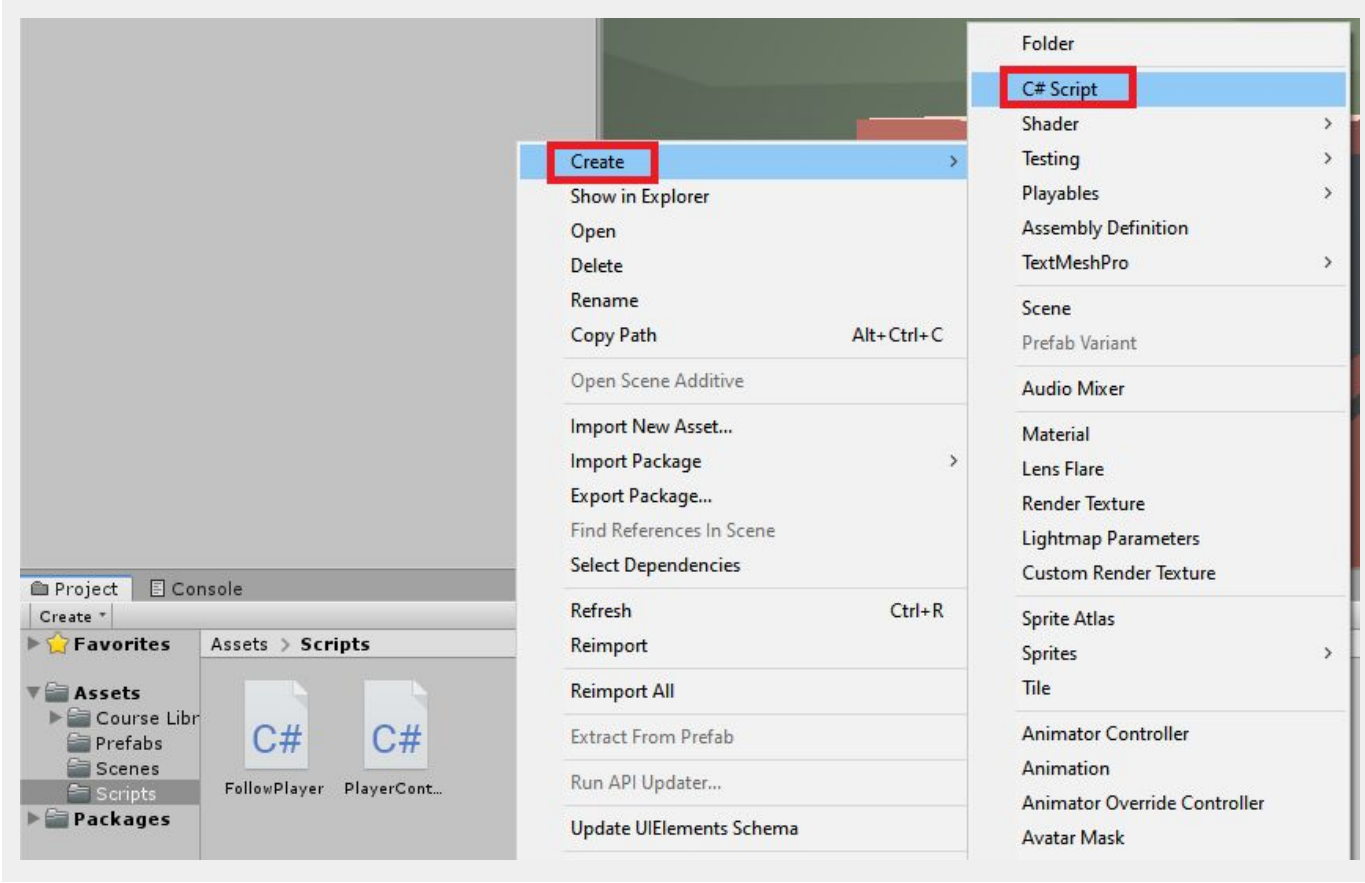
1. Navigate to **Assets > Course Library > Vehicles.** Drag and drop a vehicle you like into the Scene View. Make sure to rotate it around 180 degrees so it's facing towards the player vehicle. Change its name to *OncomingVehicle*



2. In the Inspector for the Oncoming Vehicle, click Add **Component** and search for **Rigidbody**. Adjust the **Mass** value to be appropriate for the new vehicle.



3. Right click within the Scripts folder in the Project Window and choose **Creare > C# Script**. Name this script **MoveForward** and open it.



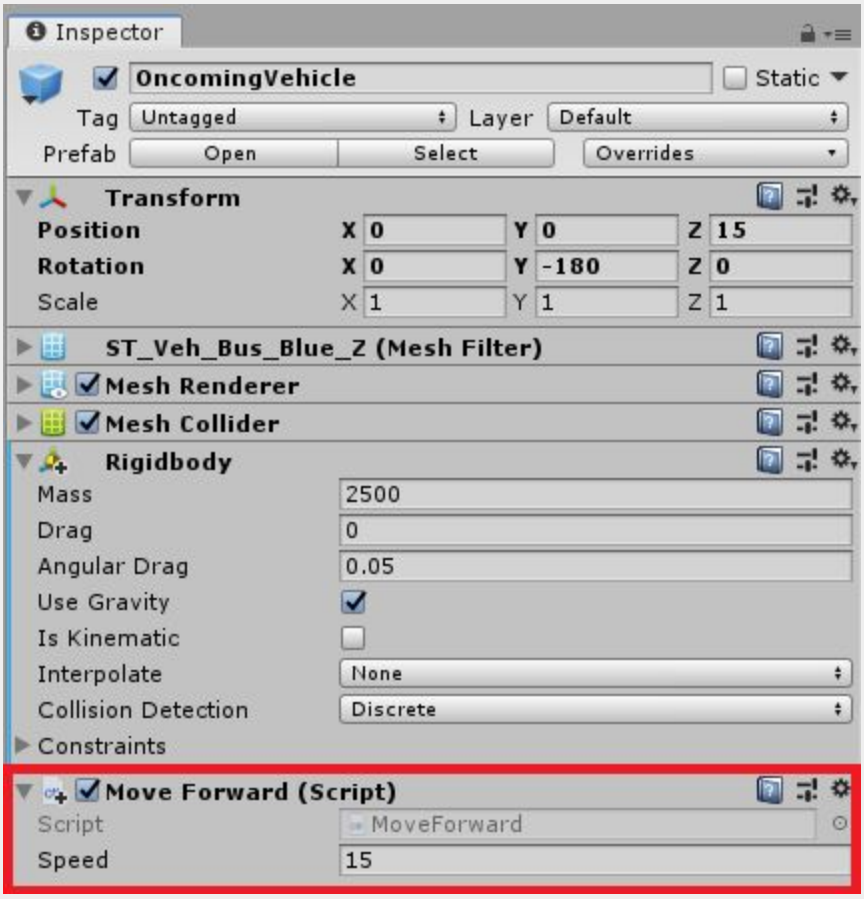
4. Before the Start method, add a new variable:

```
public int speed;
```

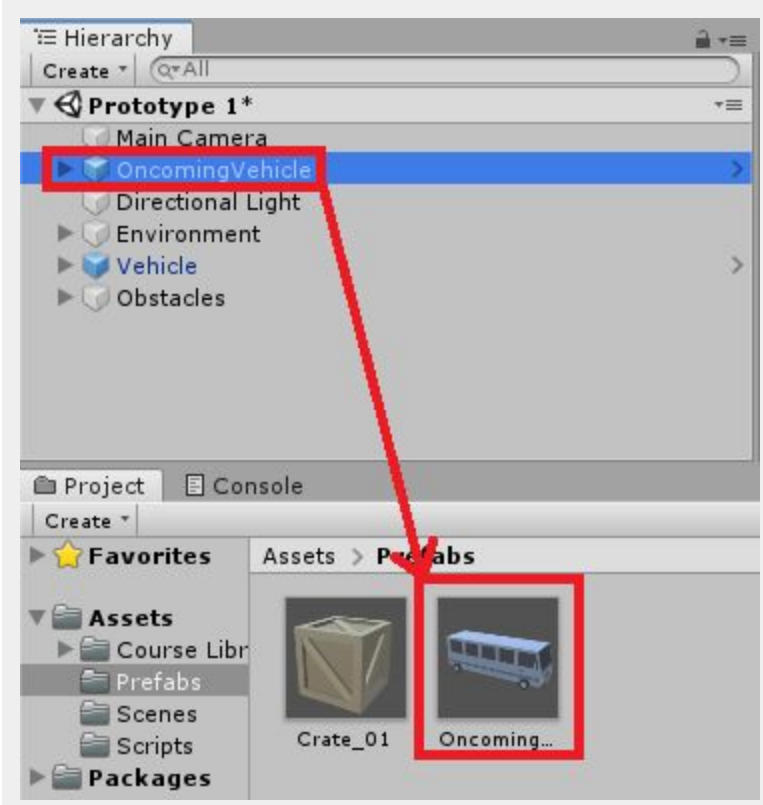
Next, within the Update method have the Vehicle automatically move forward multiplied by the speed and Time.deltaTime:

```
void Update()
{
    transform.Translate(Vector3.forward * speed * Time.deltaTime);
}
```

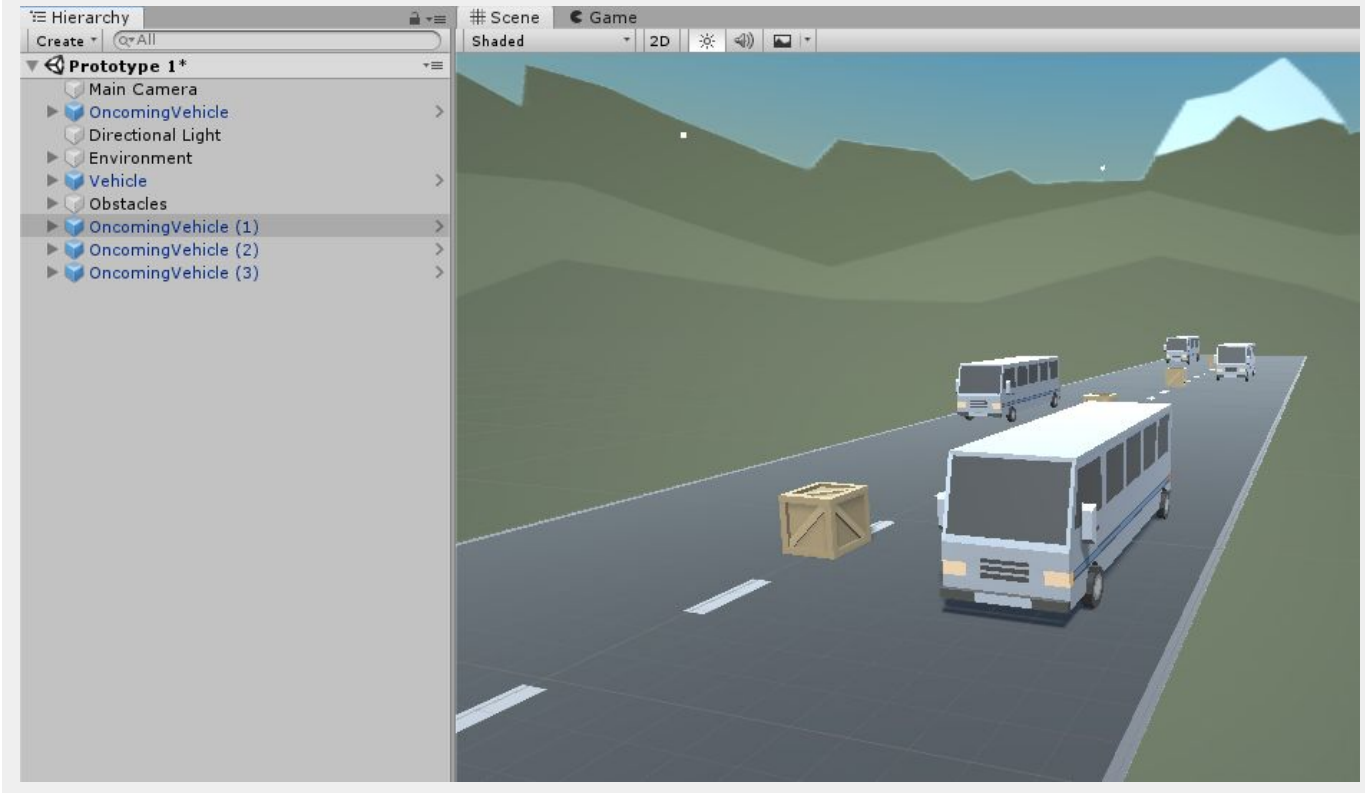
Save the script, go back to the Unity Editor, add the Move Forward script to the Oncoming Vehicle and assign an appropriate speed value in the Inspector:



6. Drag the Oncoming Vehicle from the Hierarchy into the Prefabs folder. If a window appears, click **Original Prefab**.



7. Duplicate the OnComing Vehicle in the Hierarchy and place them throughout the road.



8. Save the scene and press play. The vehicles will draw towards you and try to push you out the way.

Hard - Camera Switcher

1. Navigate to the Scripts folder and open up the **Player Controller** script. Before the **Start** method add the following variables:

```
public Camera mainCamera;
public Camera hoodCamera;
public KeyCode switchKey;
```

2. Next, in the **Update** method, add the following after the **transform.Rotate** line.

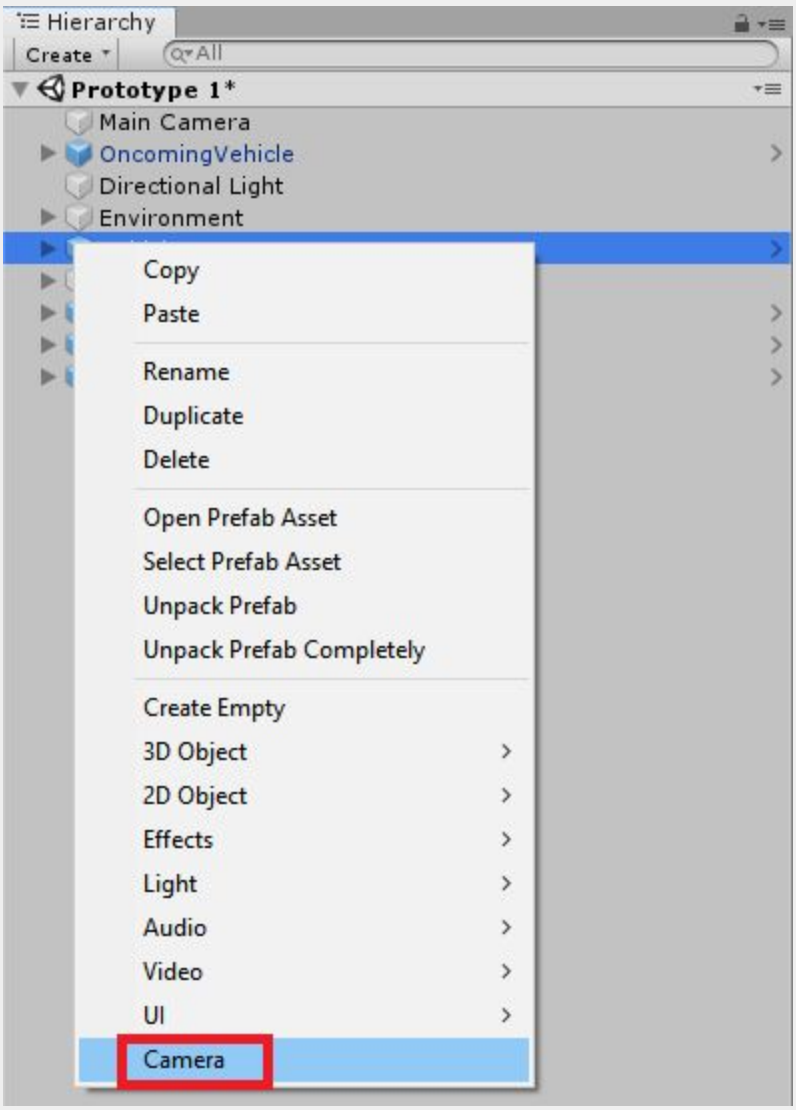
```
if(Input.GetKeyDown(switchKey))
{
    mainCamera.enabled = !mainCamera.enabled;
    hoodCamera.enabled = !hoodCamera.enabled;
}
```

This code will toggle which camera is enabled when the F key is pressed. The updated Update method should look like this:

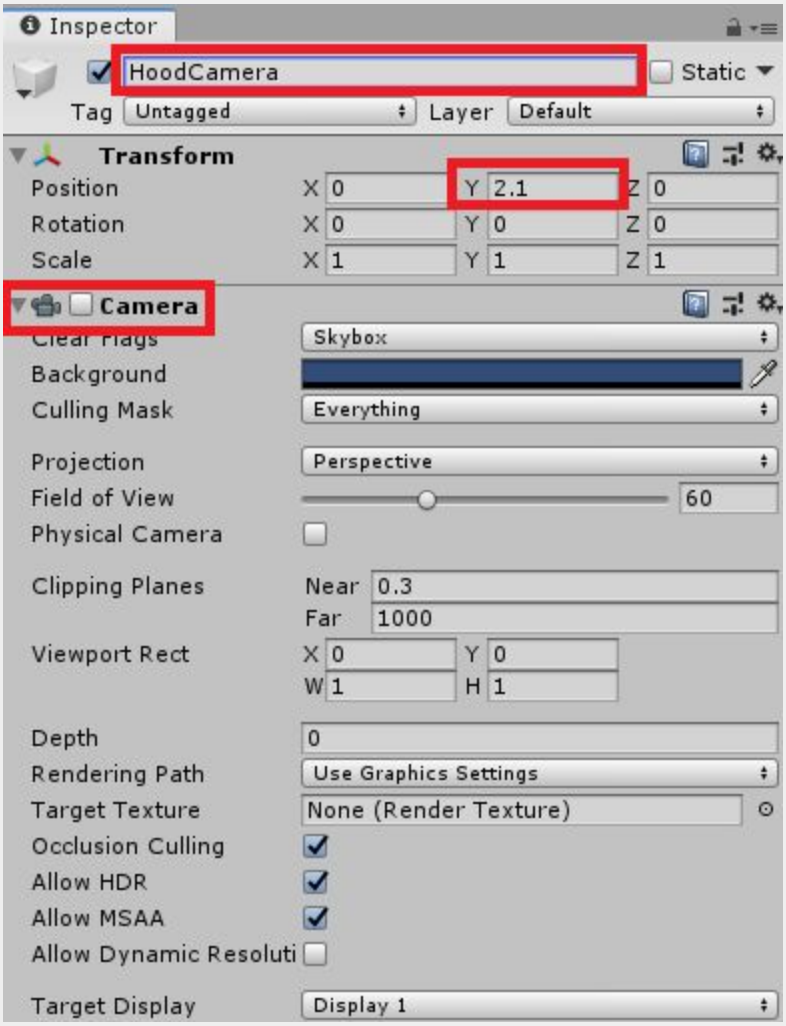
```
void Update()
{
    // Axis setup
    horizontalInput = Input.GetAxis("Horizontal");
    forwardInput = Input.GetAxis("Vertical");

    // Move the vehicle forward
    transform.Translate(Vector3.forward * Time.deltaTime * speed * forwardInput);

    // Rotate the vehicle left and right
    transform.Rotate(Vector3.up, turnSpeed * horizontalInput * Time.deltaTime);

    if(Input.GetKeyDown(switchKey))
    {
        mainCamera.enabled = !mainCamera.enabled;
        hoodCamera.enabled = !hoodCamera.enabled;
    }
}
```

Save the script and head back to Unity.

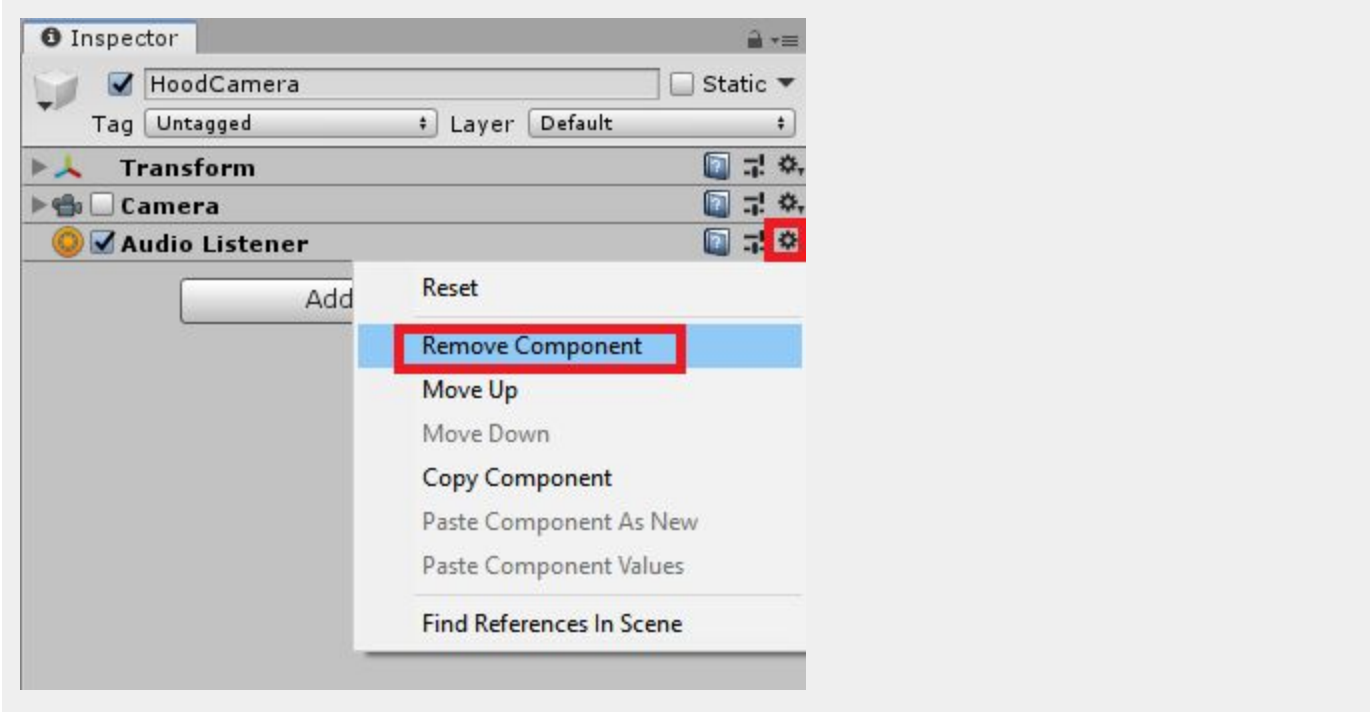
3. In the Hierarchy, right-click on the Vehicle GameObject and select **Camera**.



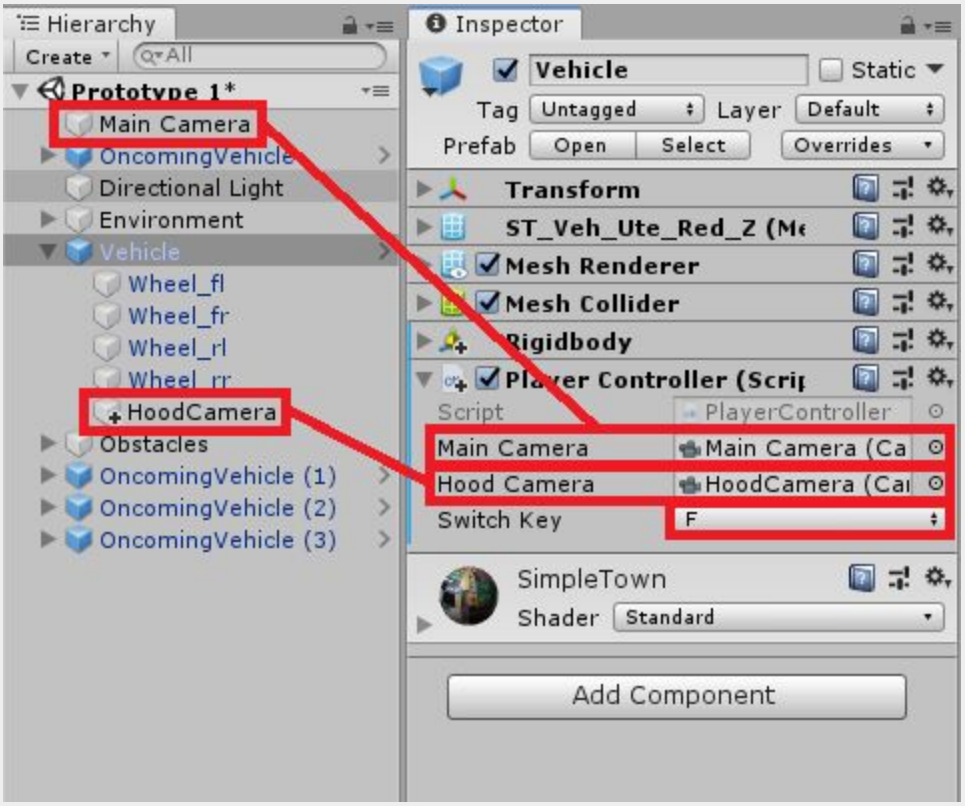
4. Rename the new Camera to be *HoodCamera*. Change the **y** position of the camera to **2.1**. Disable the Camera component by clicking the checkbox to the right of the component name.



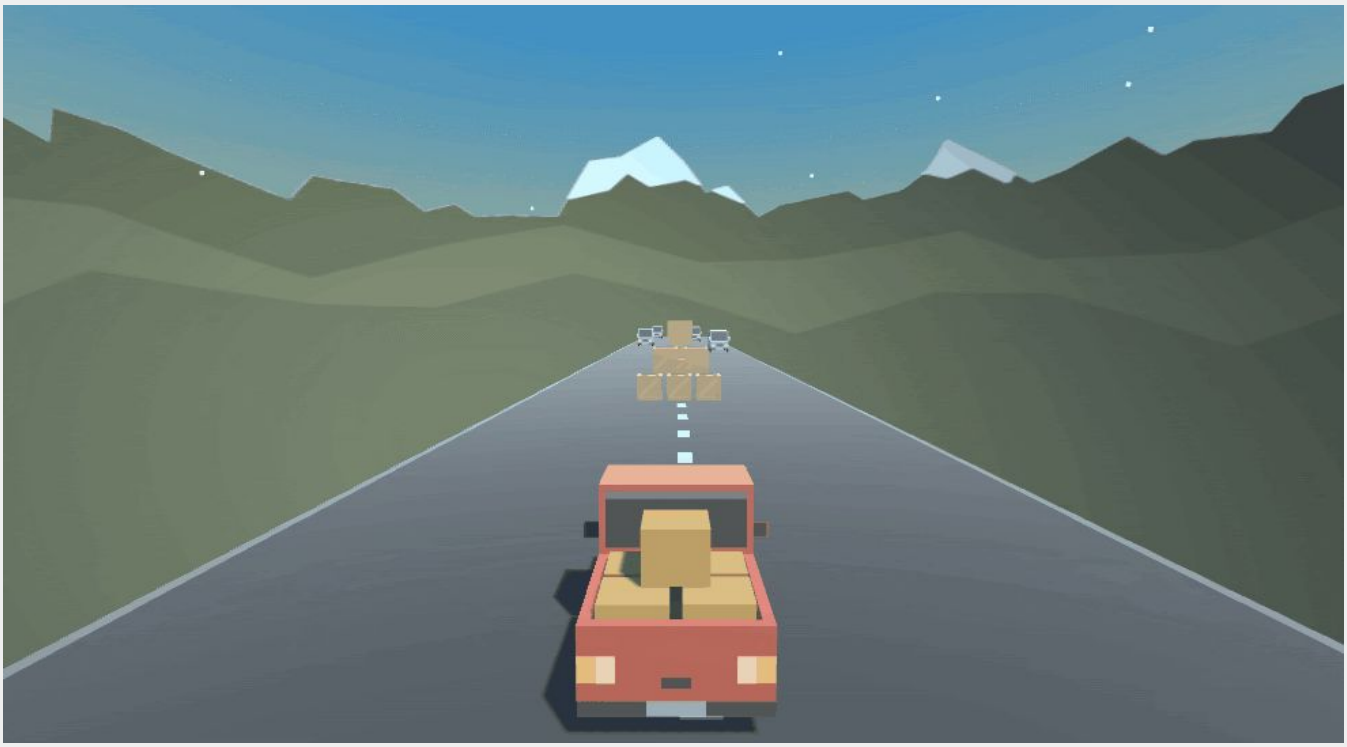
5. Remove the Audio Listener that is on the HoodCamera by clicking on the Gear and selecting **Remove Component**.



6. In the Hierarchy, select the Vehicle GameObject. Drag the *Main Camera* from the Hierarchy into the **Main Camera** field on the **Player Controller** component. Then, drag the *HoodCamera* from the Hierarchy into the **Hood Camera** field on the **Player Controller** component. Ensure that the Switch Key parameter is set to **F**.



7. Save the scene and press play. Try press F to switch between the cameras.



Expert - Local Multiplayer

1. Navigate to the Scripts folder and open up the **Player Controller** script. Before the **Start** method, add a new variable. We will use this variable to determine which player is using the script.

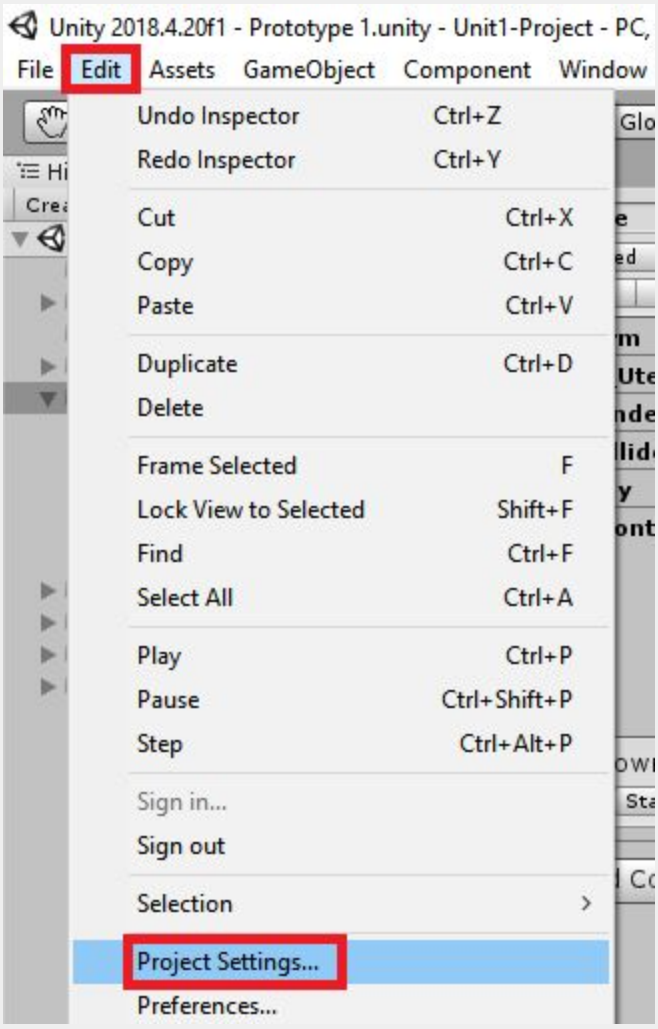
```
public string inputID;
```

2. Inside the **Update** method, update the way the horizontal and forward inputs are set.

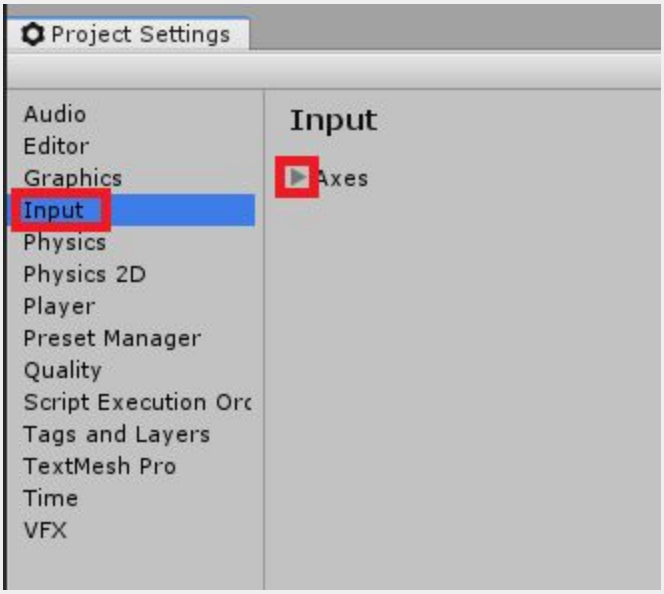
```
horizontalInput = Input.GetAxis("Horizontal" + inputID);
forwardInput = Input.GetAxis("Vertical" + inputID);
```

Save the script and head back to Unity.

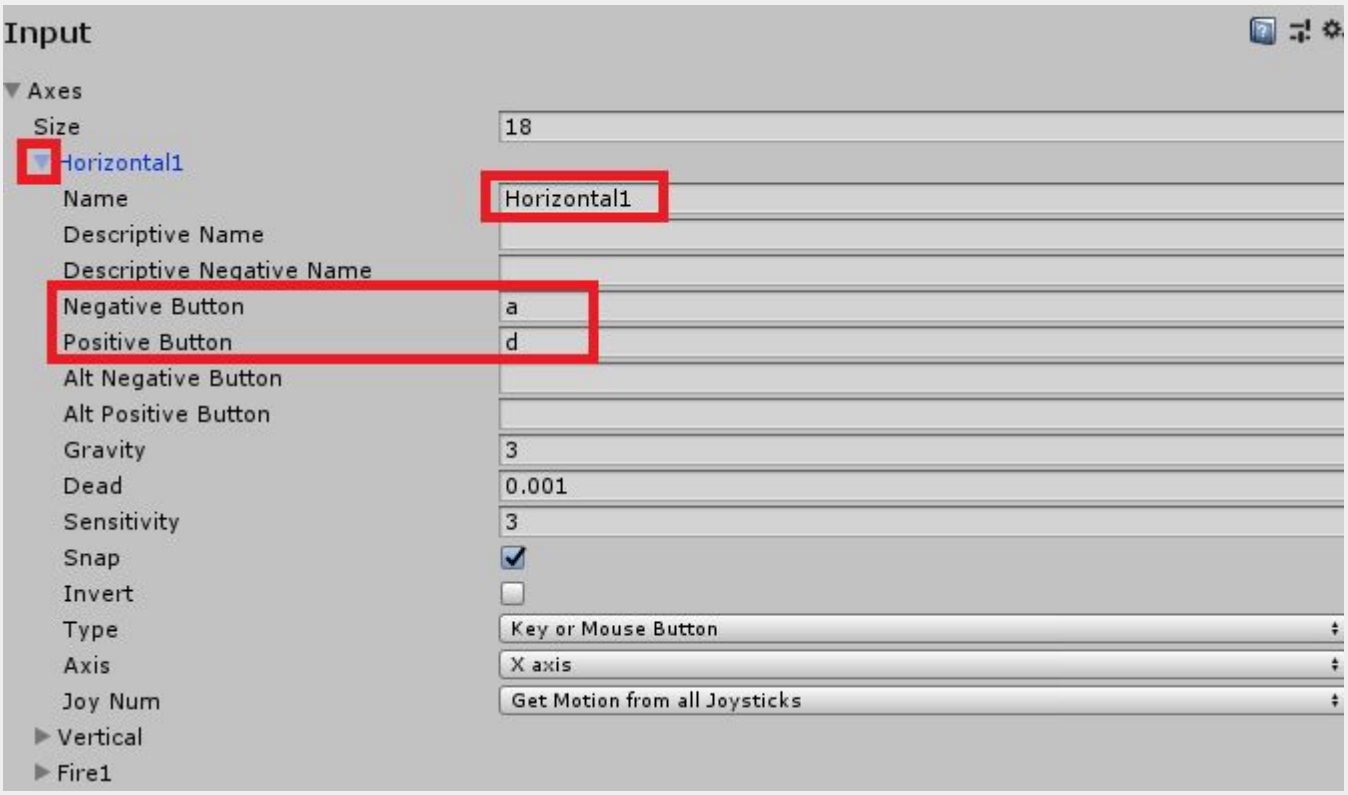
3. Now let's set up the Input Manager to have the two different inputs. Go to **Edit** > **Project Settings**



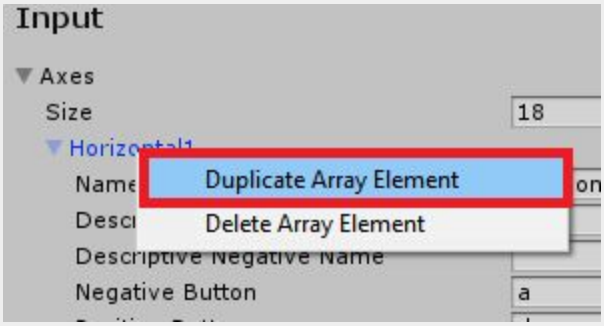
4. In the new window that opens, select **Input** from the left hand side. Then click the arrow to the right of the Axes heading.



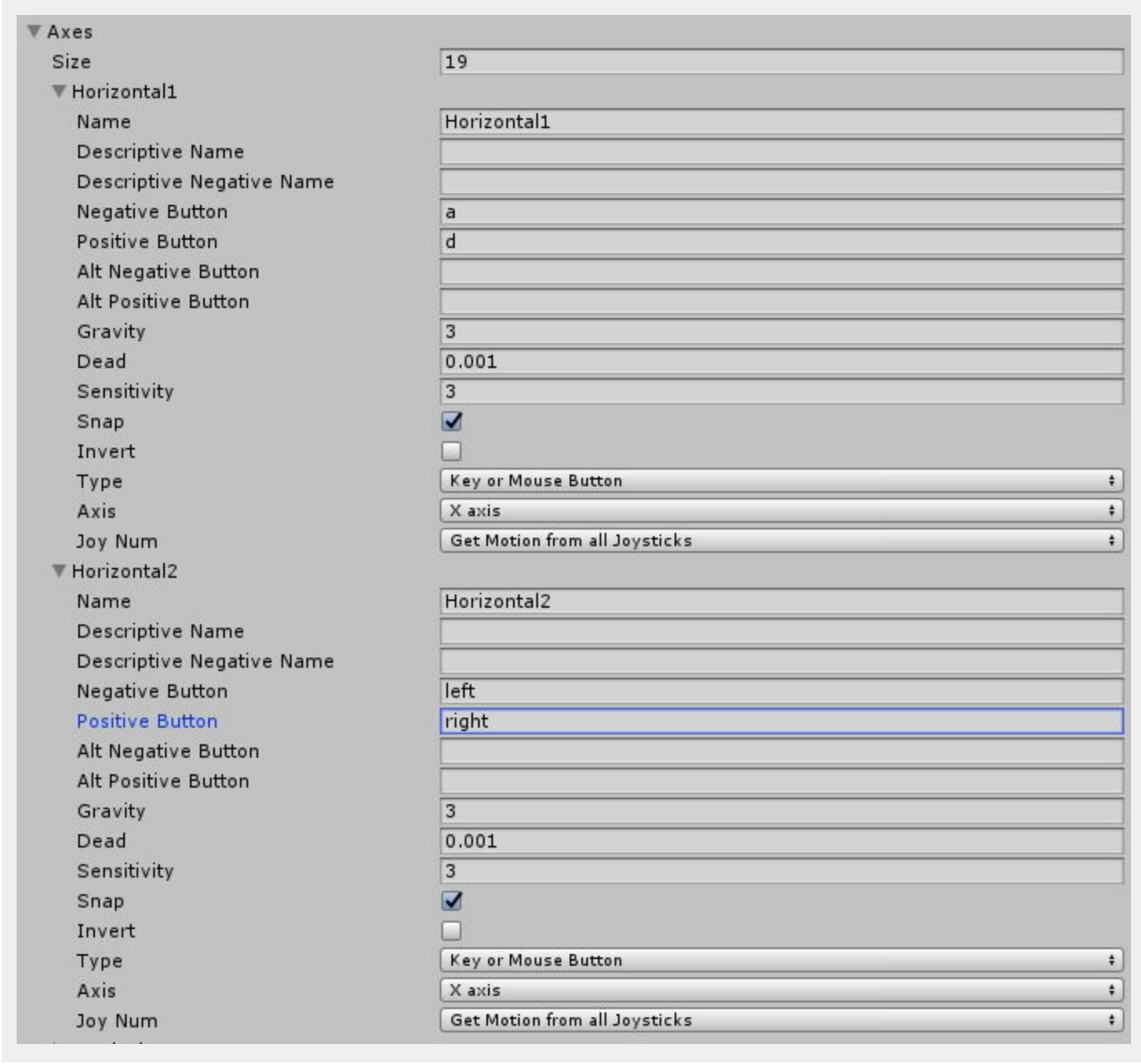
5. Rename the first option in the Axes to "*Horizontal1*". Change the **Negative Button** to **a** and the **Positive Button** to **d**. Ensure there is no Alt Negative or Alt Positive buttons set.



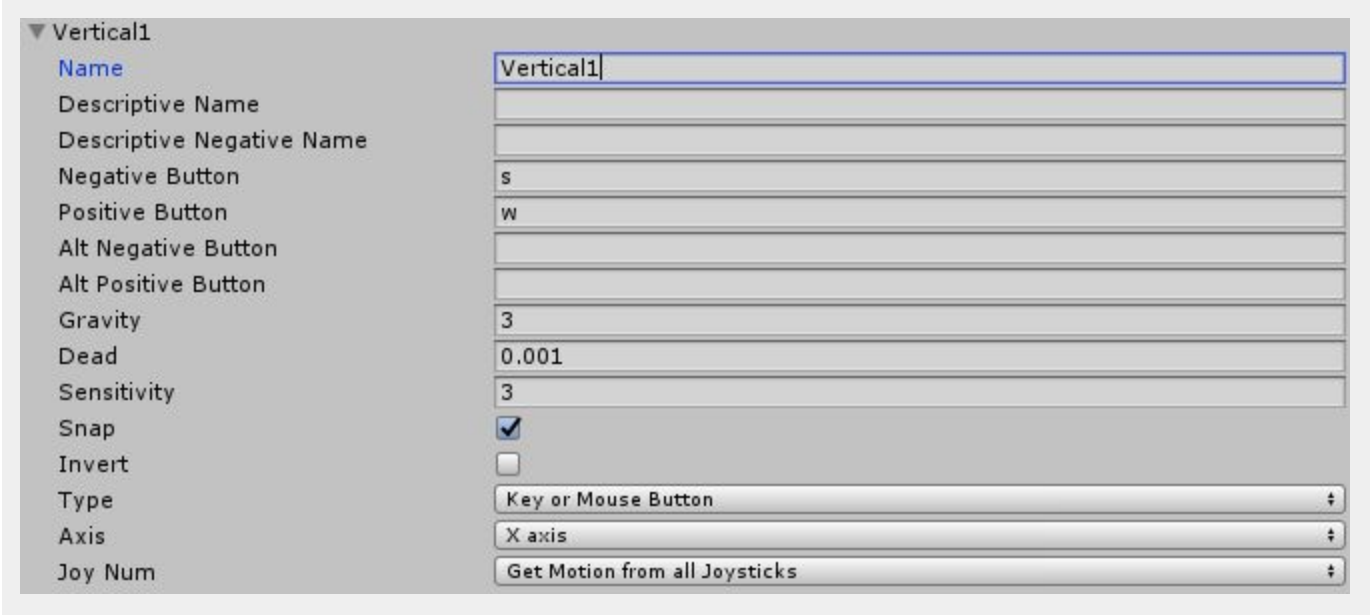
6. Right-click on the **Horizontal1** heading and select **Duplicate Array Element**.



7. Rename the new array element to *Horizontal2*. Adjust the **Negative Button** to **left** and the **Positive Button** to **right**. The completed Horizontal elements should look like this:

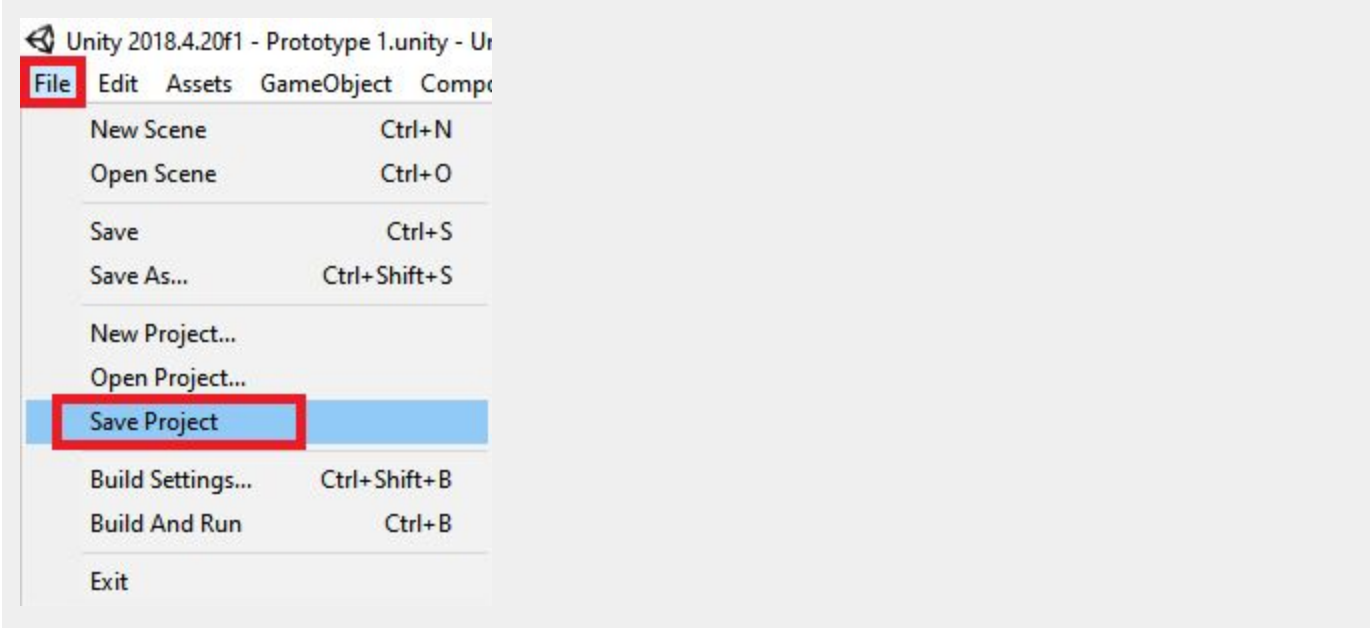


8. Collapse the two elements we just did and expand the Vertical array element and rename it to *Vertical1*. change the **Negative Button** to **s** and the **Positive Button** to **w**. Remove any parameters that are in the **Alt Negative Button** or **Alt Positive Button** fields.

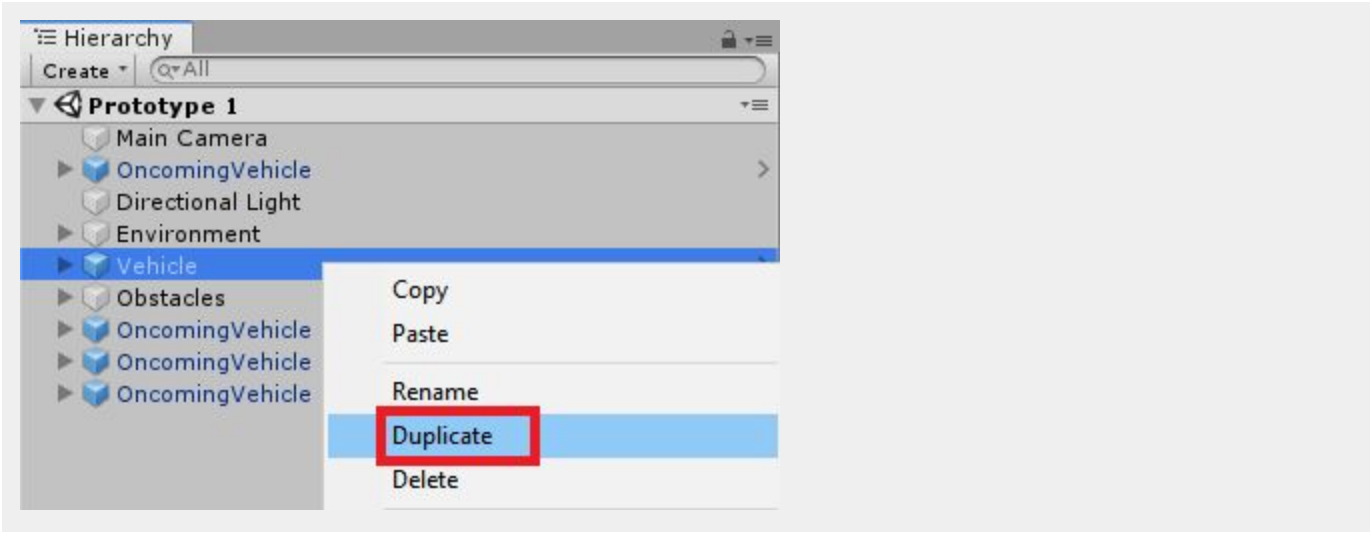


9. Duplicate the Vertical1 array element like we did with the Horizontal1 element. Rename the new Vertical element to *Vertical2*. Change the **Negative Button** to down and the **Positive Button** to **Up**. The completed Vertical elements should look like this:

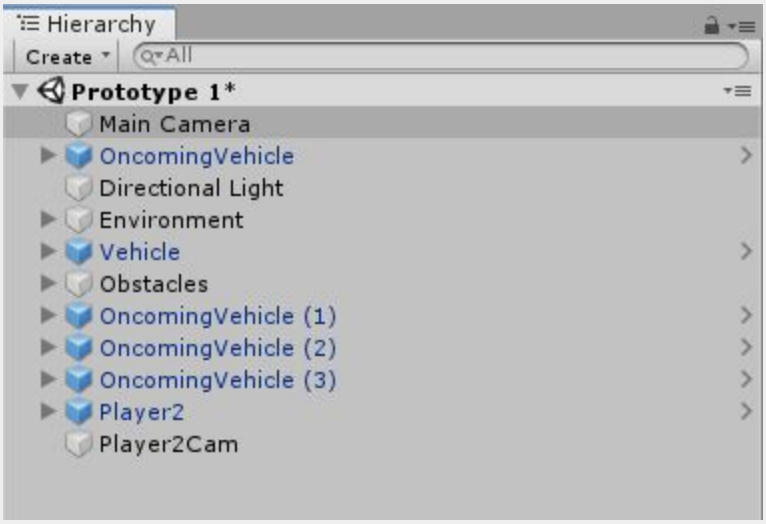
10. Close the Project Settings window and save the project by going to **File** > **Save Project**



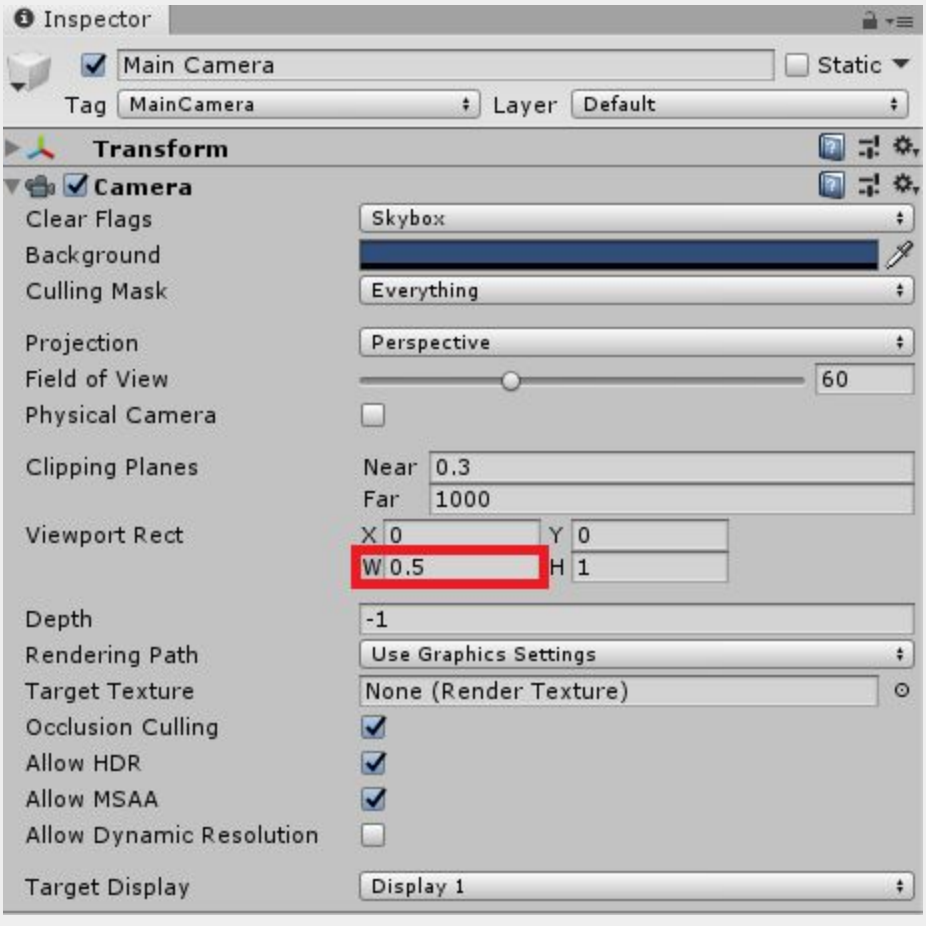
11. In the Hierarchy, right-click on the Vehicle GameObject and select **Duplicate**. Rename the duplicate to *Player2*



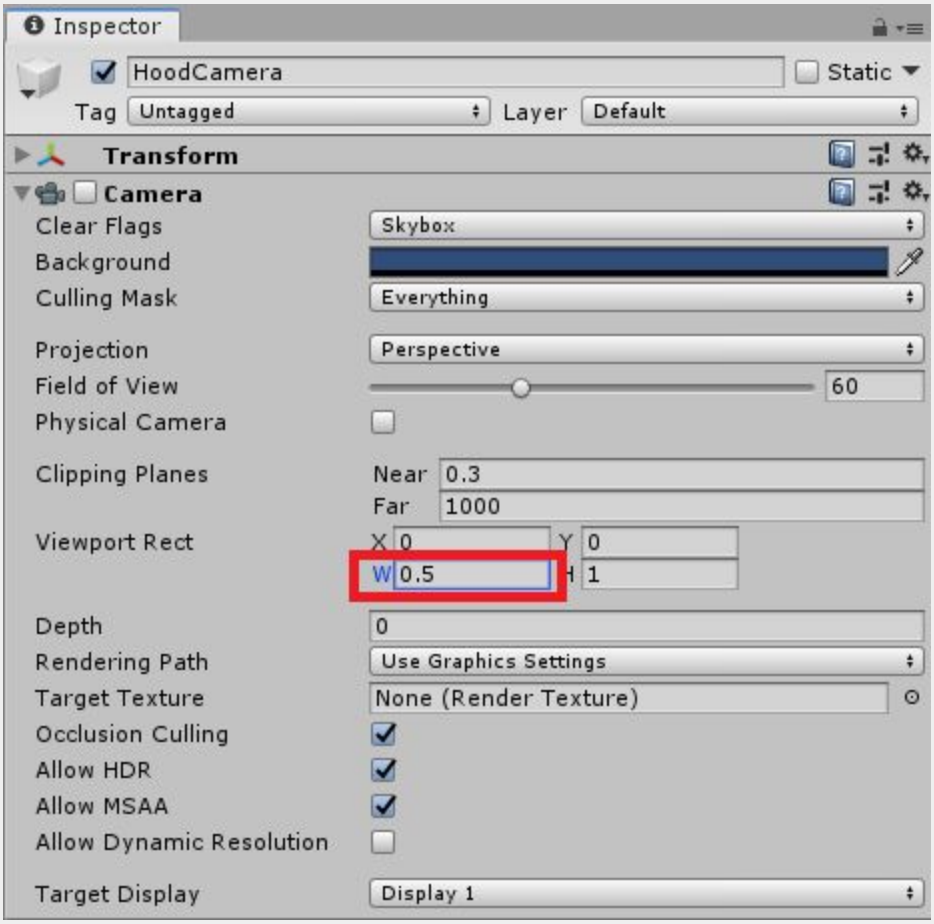
12. Do the same with the Main Camera GameObject and rename it to Player2Cam.



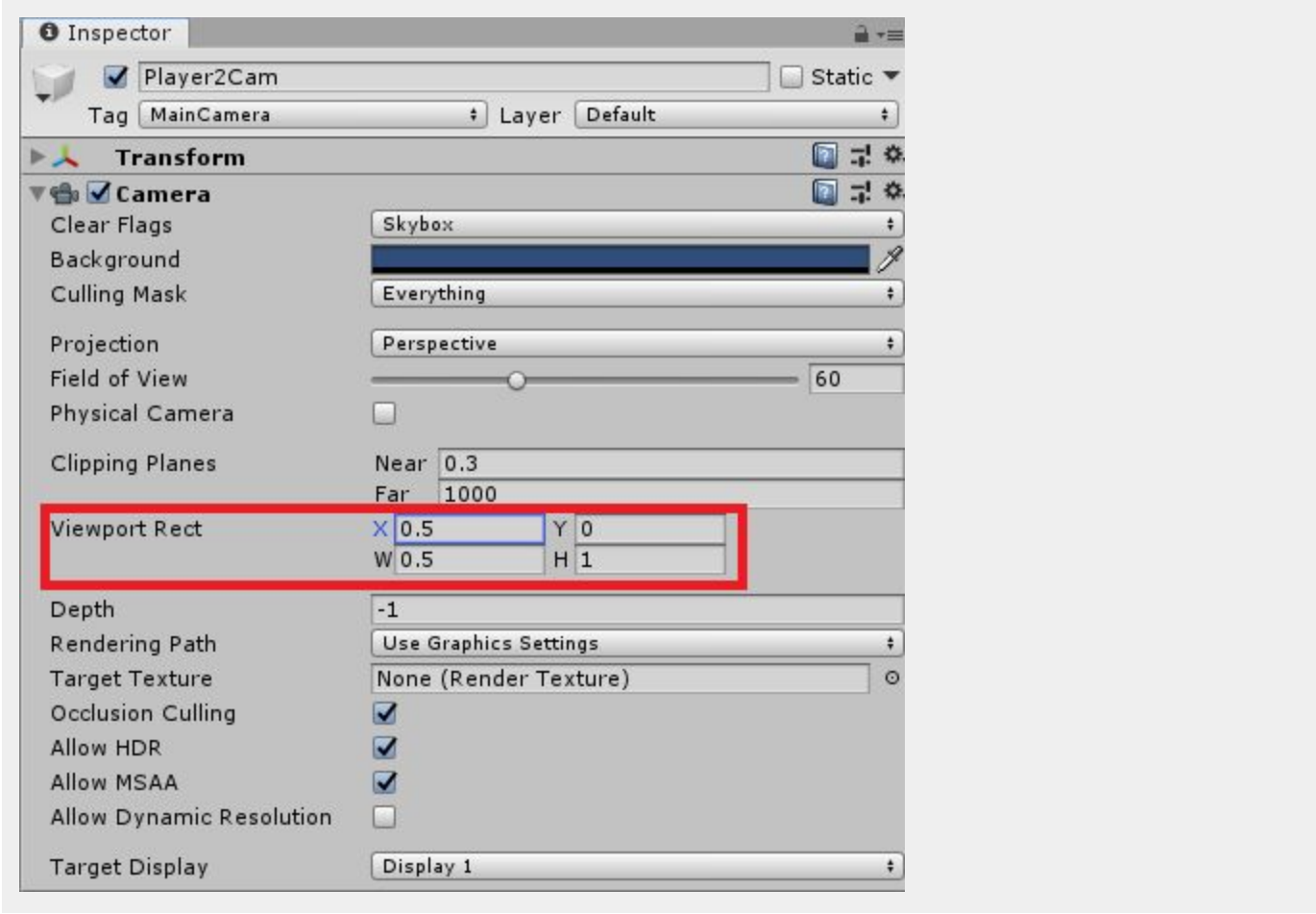
13. In the Hierarchy, select the Main Camera. In the Inspector, on the **Camera** component, change the **Viewport Rect**'s **W** value to 0.5.



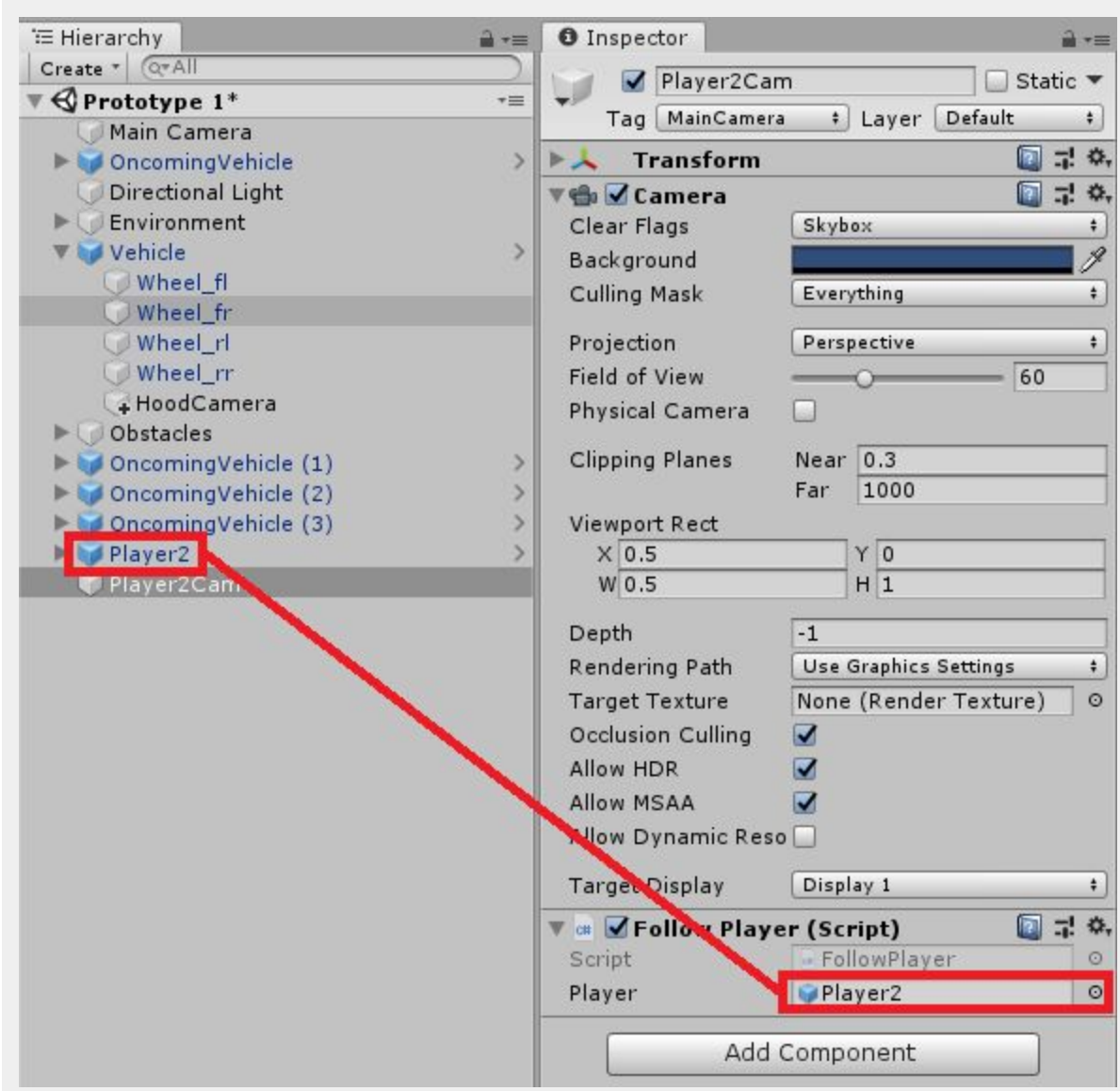
14. In the Hierarchy, navigate to the Vehicles Hood Camera. Change the **Camera** component's **Viewport Rect** to match the Main Camera.



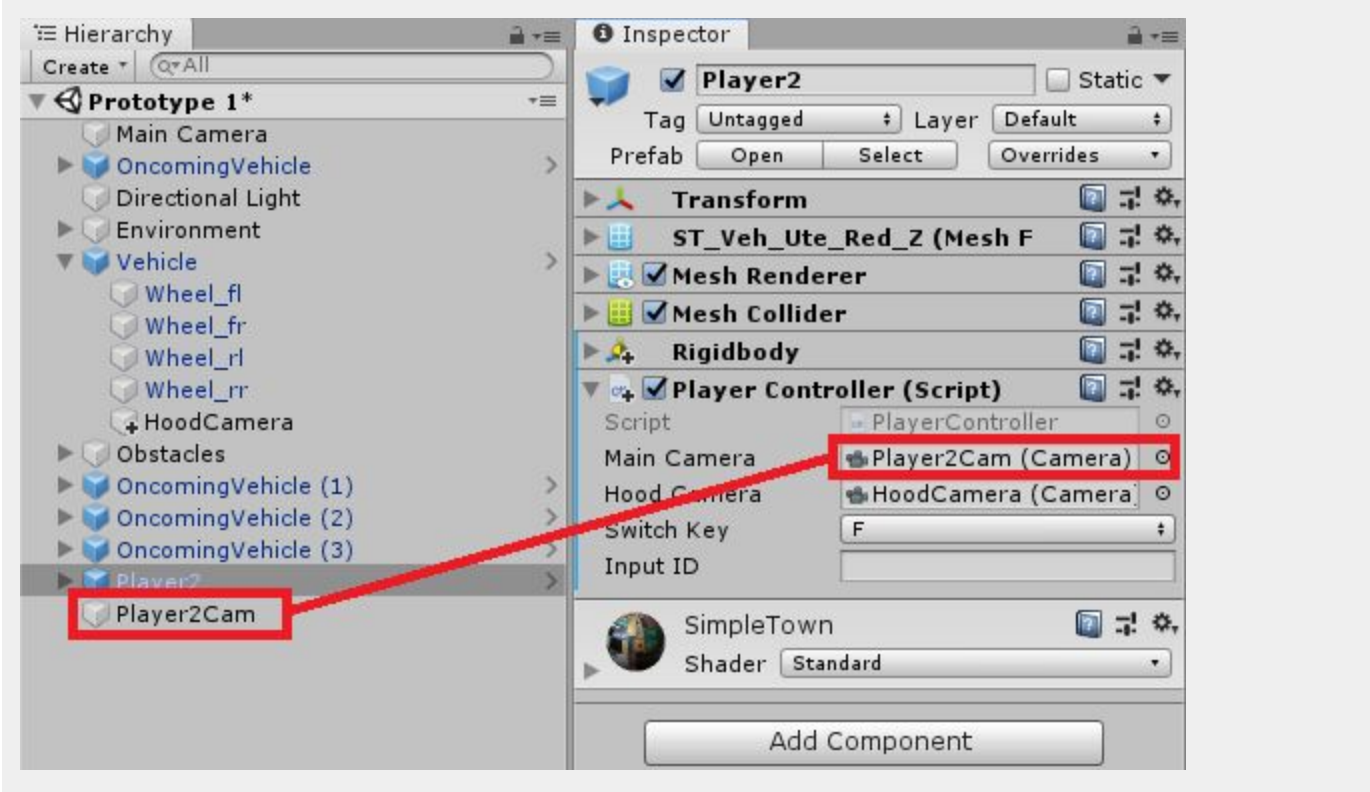
15. In the Hierarchy, select the Player2Cam GameObject. Adjust the **Camera** components Viewport Rect to be **0.5** in the **X** and **W** fields. Remove the **Audio Listener** component by clicking on the gear and selecting **Remove Component**.



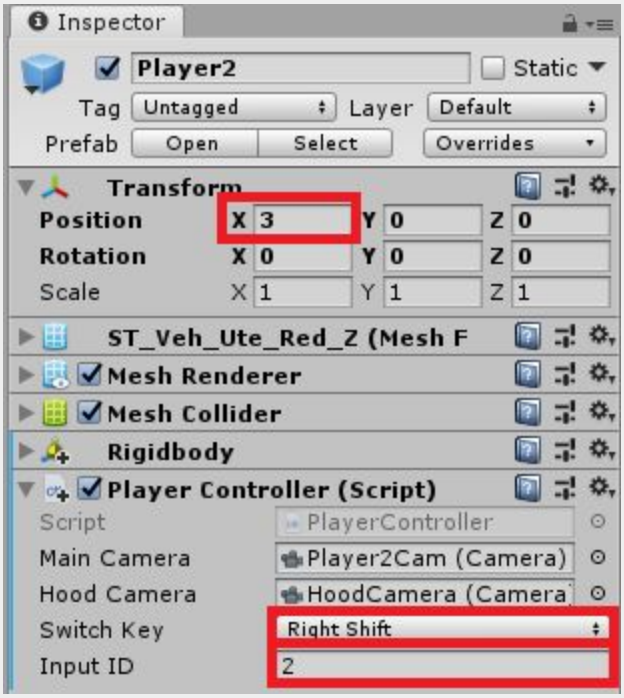
16. Drag the Player2 into the Player field of the **Follow Player** component of the **Player2Cam** GameObject.



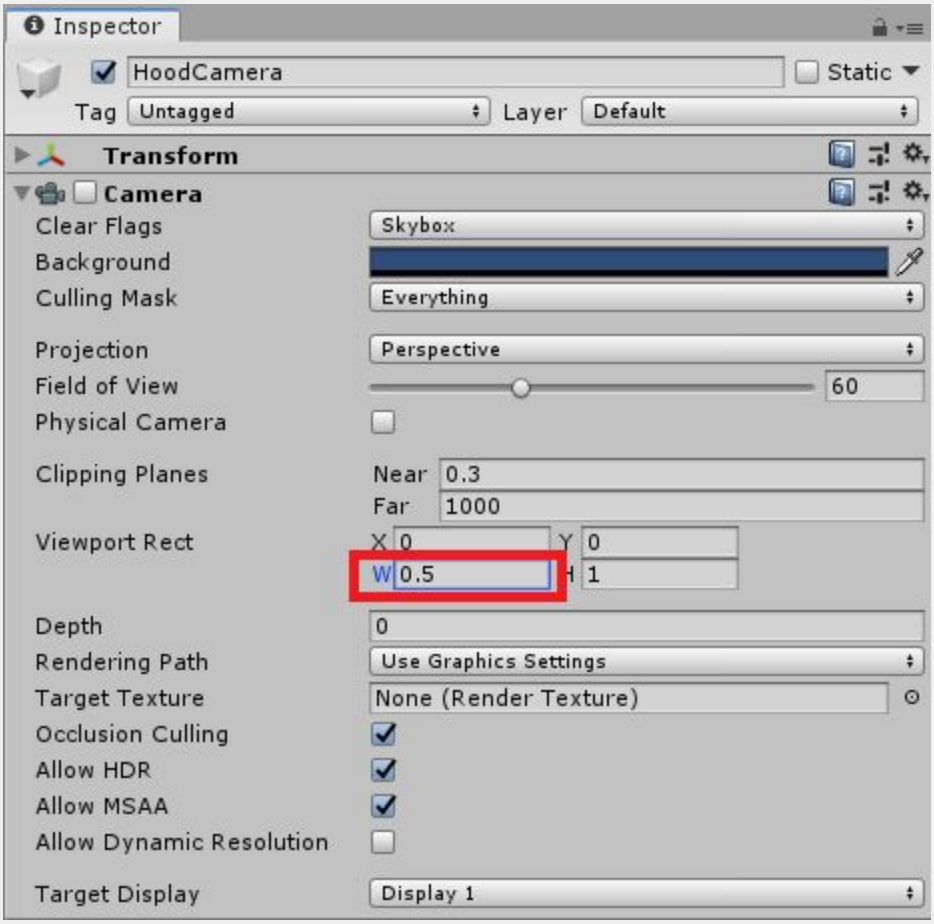
17. Select the Player2 GameObject in the Hierarchy. Drag the *Player2Cam* from the Hierarchy into the **Main Camera** field of the **Player Controller** component.



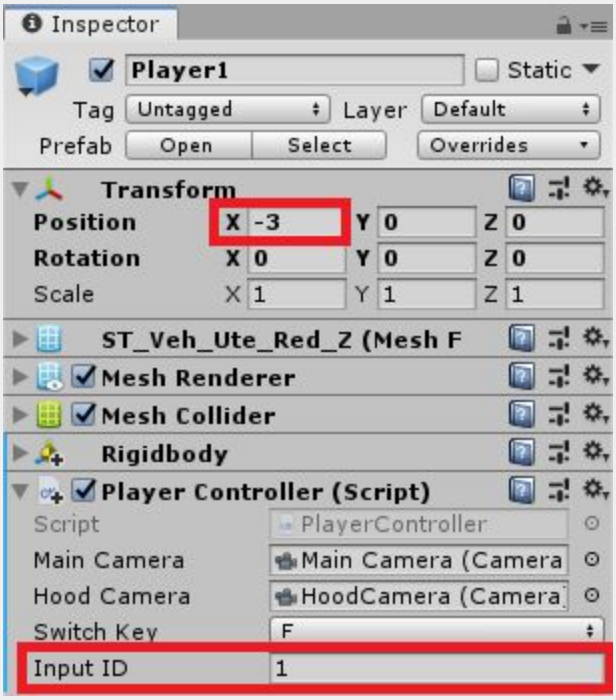
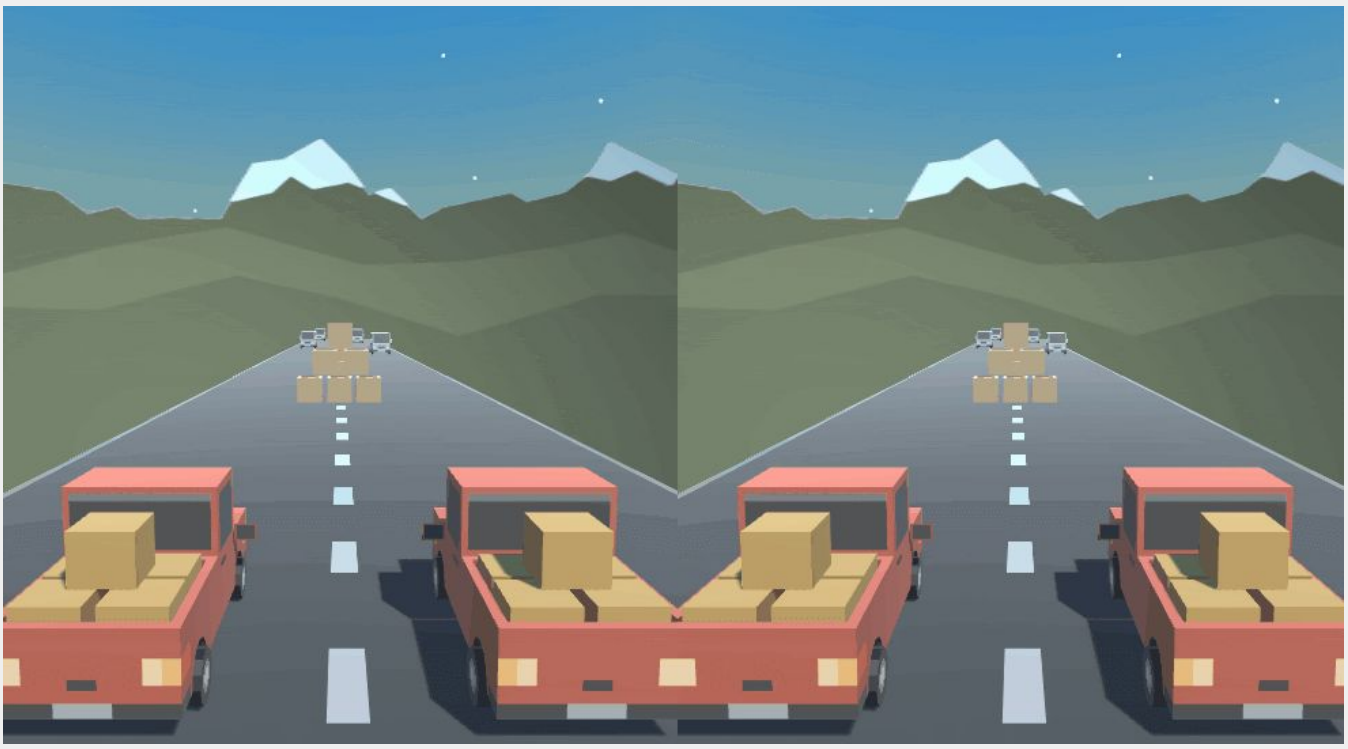
18. Change the **Switch Key** to **Right Shift** and set the **Input ID** to **2**. Change the **X Position** of Player2 to be **3**.



19. In the Hierarchy, expand the child GameObjects of Player2 and select the Hood Camera. On the Camera component, adjust the **Viewport Rect**'s **X** and **W** values to be **0.5**.



20. Select the Vehicle in the Hierarchy. Rename it to *Player1*. Change the **X Position** on the **Transform** component to **-3**. On the **Player Controller** component, set the **Input ID** value to **1.**



21. Save the scene and press play. Try playing as both Player 1 and Player 2, or get someone to help test with you. Try pressing the camera switching buttons (F and Right Shift) and notice how the cameras will keep to their side of the screen.