Creating Virtual Reality Experiences with Unity

Immersive Technologies Summer School

[VR Part One](#_jybi5xh7s349)

[Installing Software](#_oe530ayn30tp)

[The Interface](#_wwiw1ks050r0)

[Creating an Environment](#_4lz5cugcbt48)

[Preparing the scene for VR](#_yu6ko23dtuz4)

[Adding Basic Movement](#_ndpwa3bpyqmj)

[Object Interactions](#_y9w4jmt20ugh)

[Documentation](#_wjsb7agnk0uj)

[Free Assets](#_es1105cfrlsu)

# VR Part One

## Installing Software

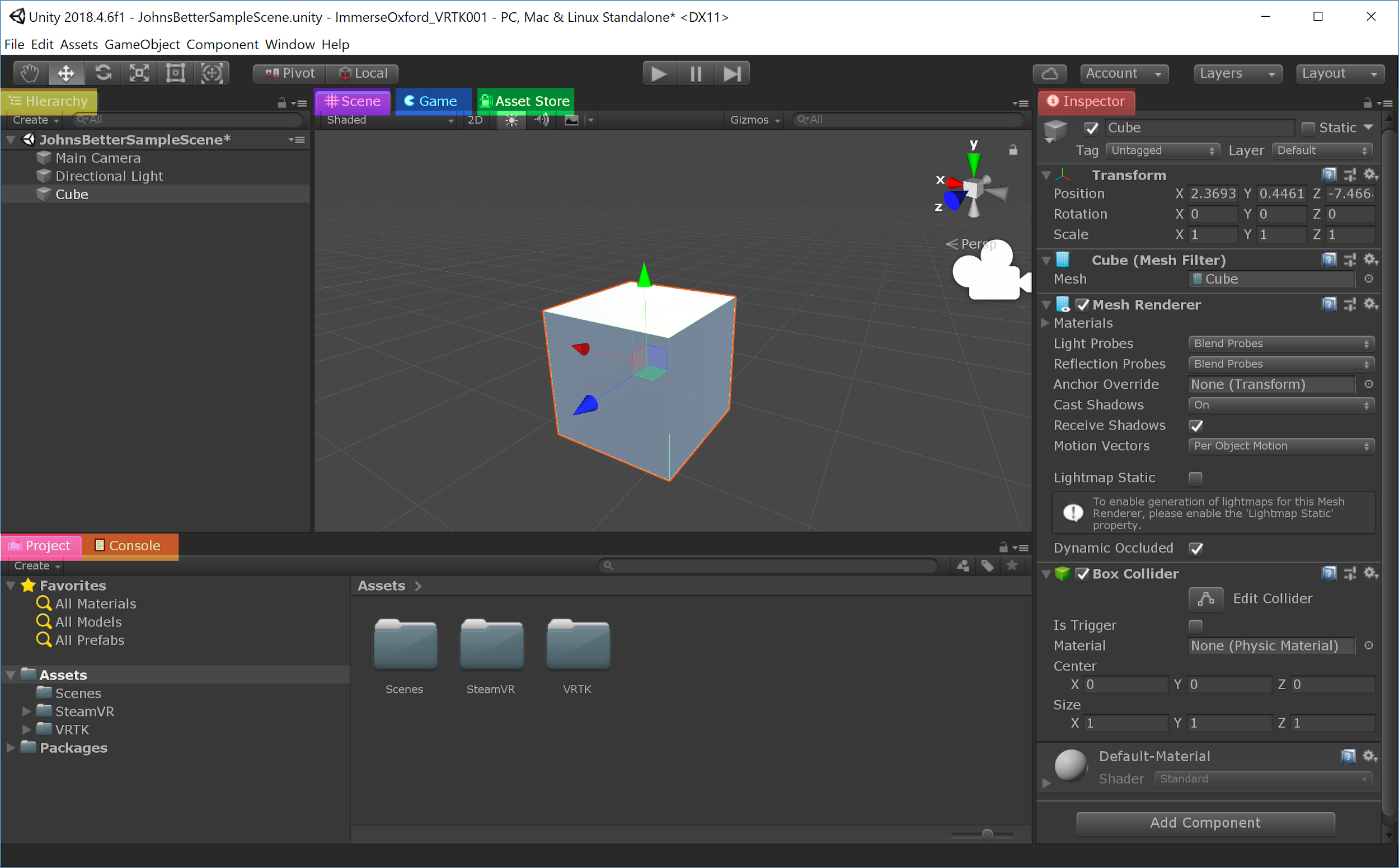
For this section of the course we will need

* [Unity 2018.4.6f1](https://unity3d.com/unity/whats-new/2018.4.6)
* [Github Desktop](https://desktop.github.com/)

[Example Project](https://drive.google.com/open?id=1BcTVvDpIT6hrytYEVEz5zenUAagrqZKG) (Following already installed)

* VRTK Version 3.3.0
* SteamVR Plugin Version 1.2.3

## The Interface



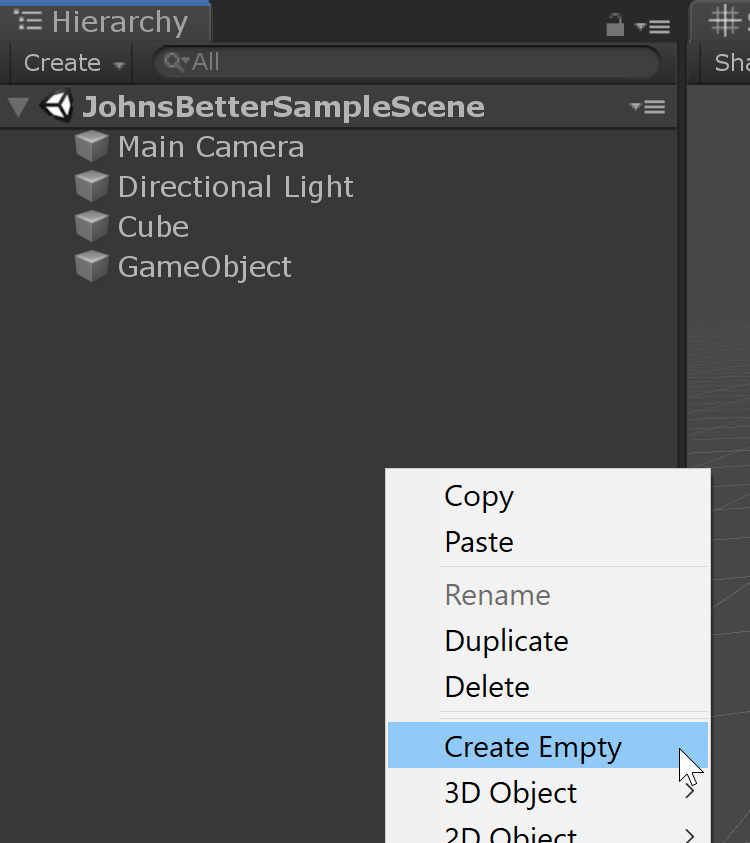
|  |  |
| --- | --- |
| **GameObject** | **Everything in the Scene is called a GameObject.**  In the screenshot above there are 3 GameObjects, one of which is the cube. |
| **Hierarchy** | **A list of every GameObject in the Scene.** GameObjects can be nested under each other by dragging one onto another in the Hierarchy.  New GameObjects can be created by right clicking in the Hierarchy. |
| **Scene** | **A first person view of the Scene.**  Selected GameObjects will be outlined in Orange.  *Basic Controls:* Left Click = Select GameObject Right Click = Look around  W, A, S, D (hold right click) = Forward, Left, Back, Right (Like arrow keys)  *Moving GameObjects:*  W = Move along XYZ  E = Rotate around XYZ  R = Scale XYZ  *Pro Hacker Controls:*  Q, E (hold right click) = Up, down  F = Center on selected GameObject  Shift = Speed up Camera |
| **Game** | **What the player will see.**   In the Scene View our camera is unconstrained, and not part of the Unity Scene.  The game view is what is rendered from the cameras in your scene. |
| **Asset Store** | **Unity’s online store for downloading assets.**   Assets include complete 3D environments, pre-made gameplay scripts, custom tools for the editor, sound files, etc. I’ve listed a few good free ones later in the document. |
| **Inspector** | **Information about the selected GameObject.**  Each titled section is called a “Component”  Components define the properties and behaviors of the GameObject. Unity has a library of built-in components, but you can script your own. In the above screenshot the “Transform” component defines the location, rotation, & scale. |
| **Project** | **A view of the files in your Unity Project.**  The “Asset” folder should be used for everything added to the game.  New assets can be created by right clicking in the Project view.  Existing assets can be added by dragging the file from Windows into the Project view. |
| **Console** | **A log of all errors, warnings and messages from the Engine.**  When a script isn’t working, an error will appear in the Console. Errors marked in red must be fixed before Unity can run the game. |

## Creating an Environment

To create a prefab, simply select an Object in the scene and drag it to the project folder.

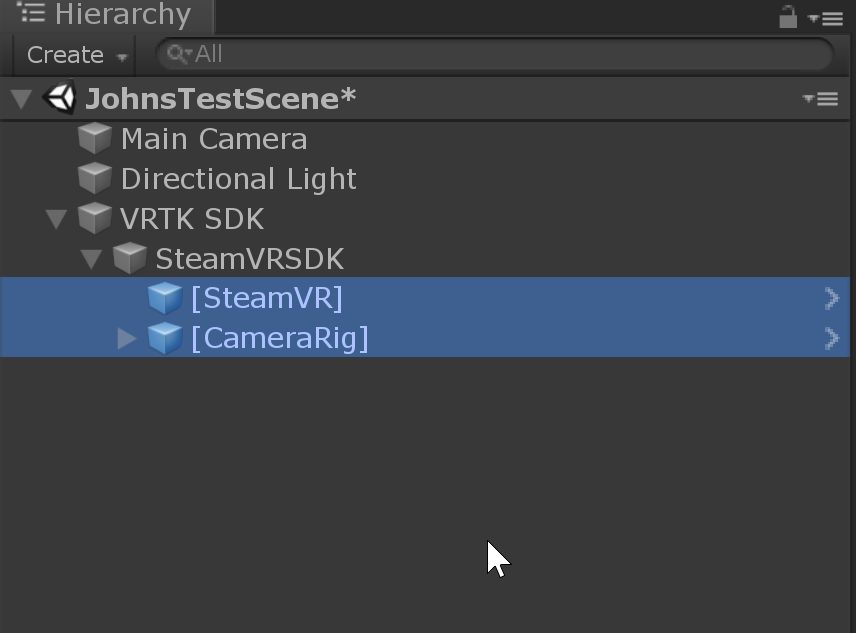
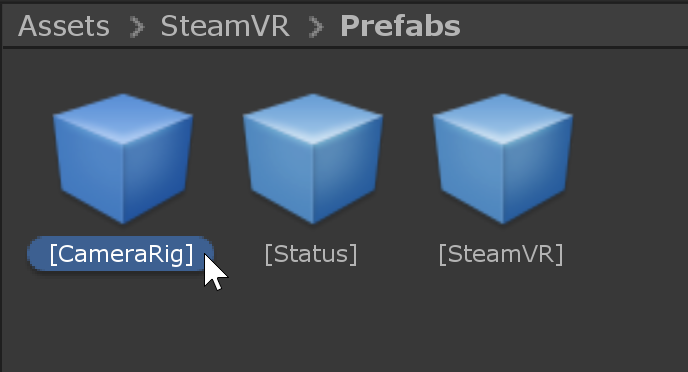
Probuilder is a simplified 3D modelling package for Unity. We’re going to install it and model a simple 3D environment.

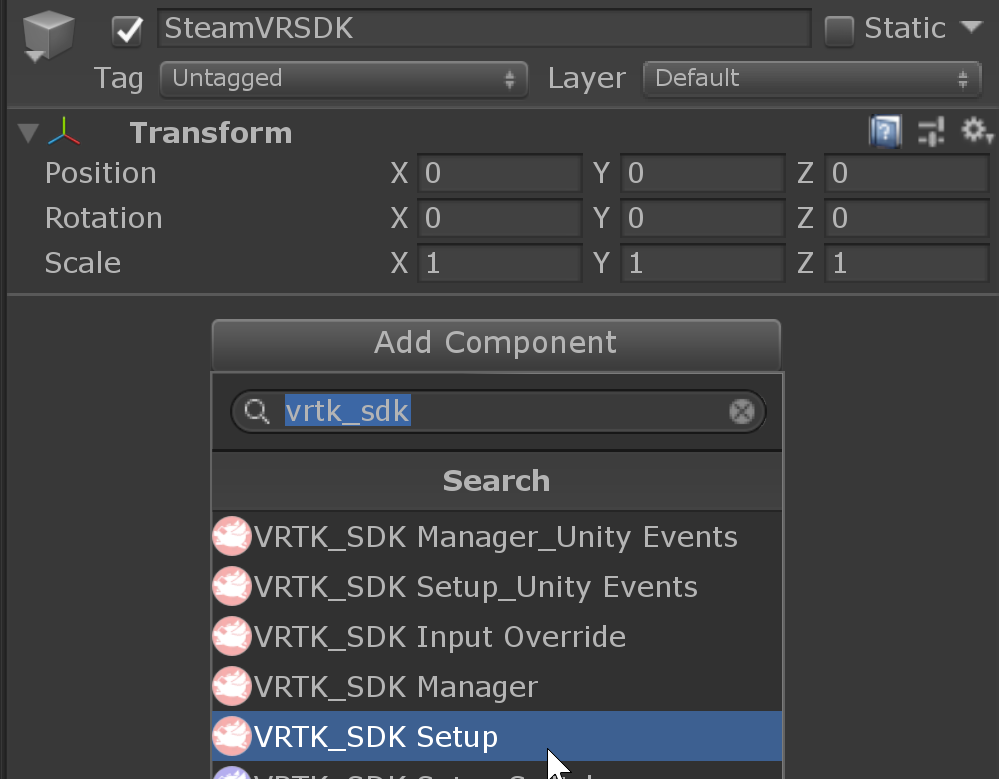
## Preparing the scene for VR

Now that we have a scene set up, it’s time to add the necessary Prefabs to the scene.   
  
1. Create an empty game object by right clicking in the hierarchy, name it “VRTKSDK”   


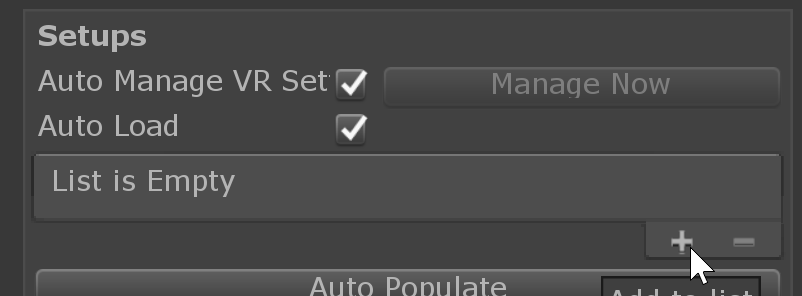
2. Add the VRTK\_SDK Manager script to the object

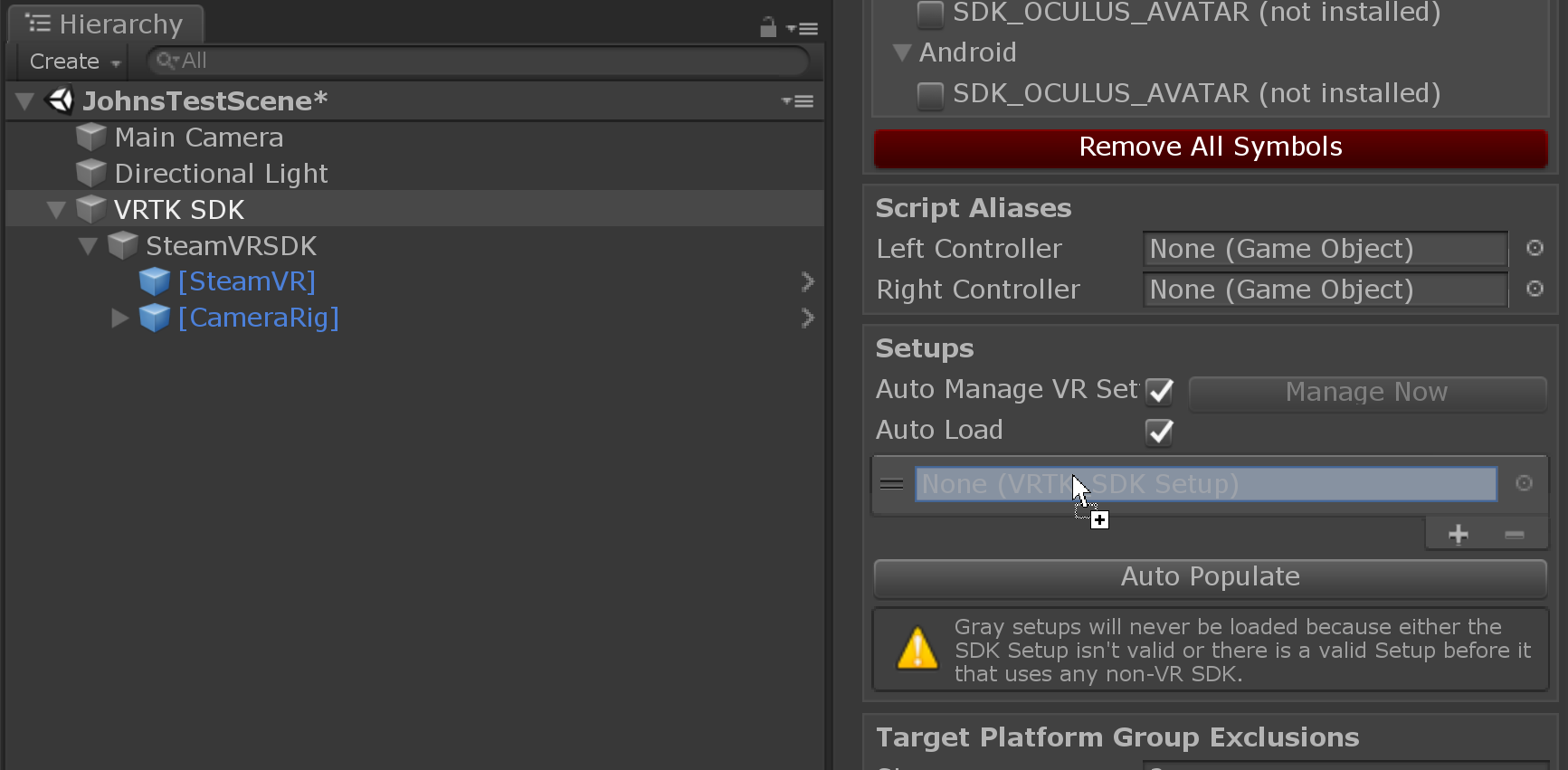
3. Create a second empty game object, and nest it under VRTKSDK. Name it “SteamVRSDK”

4. In the project folder, navigate to “SteamVR > Prefabs”   
select [CameraRig] and [SteamVR] prefabs   
drag them onto “SteamVRSDK” so that it is nested.   
Delete the existing “Main Camera” from the Scene.   


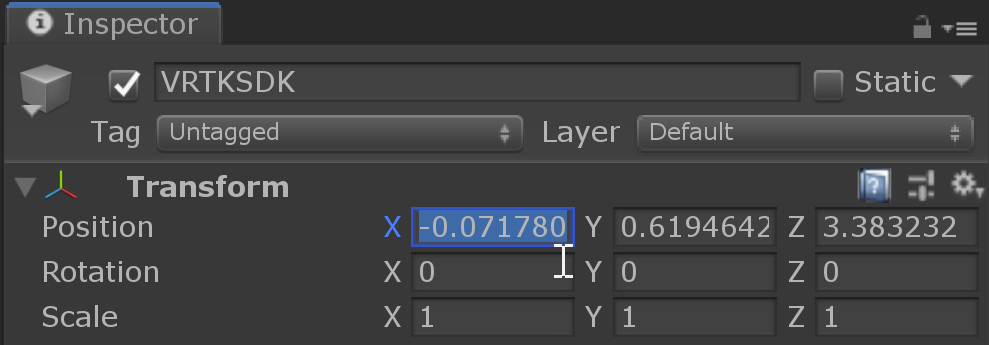
5. Select the SteamVRSDK Object and add the “VRTK\_SDK Setup” script  


6. Set the Quick Select mode to “Steam VR (Standalone)”

7. Select “VRTKSDK” and navigate to the Setup section of the VRTK\_SDK Manager component. Click the Plus Symbol:   
****

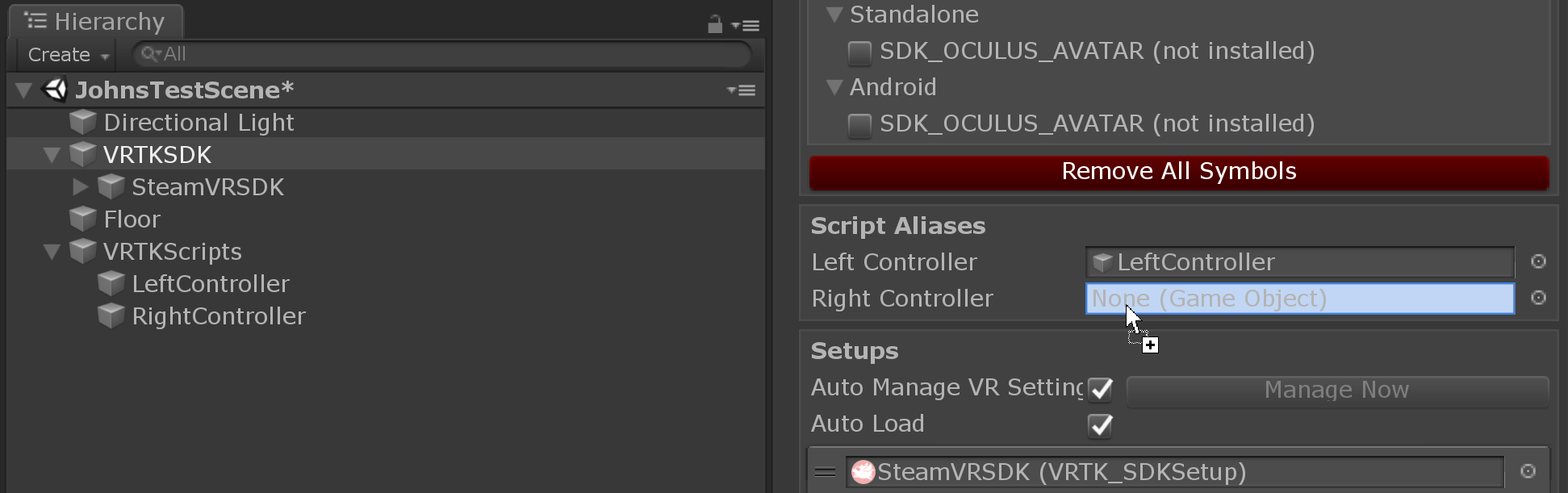
8. Drag the SteamVRSDK object from your hierarchy to the newly created slot   
  
  
Your scene should now enable you to stand and look around!   
If you see “no cameras rendering” after pressing play, clicking “populate now” on the SteamVRSDK Object may fix this issue.

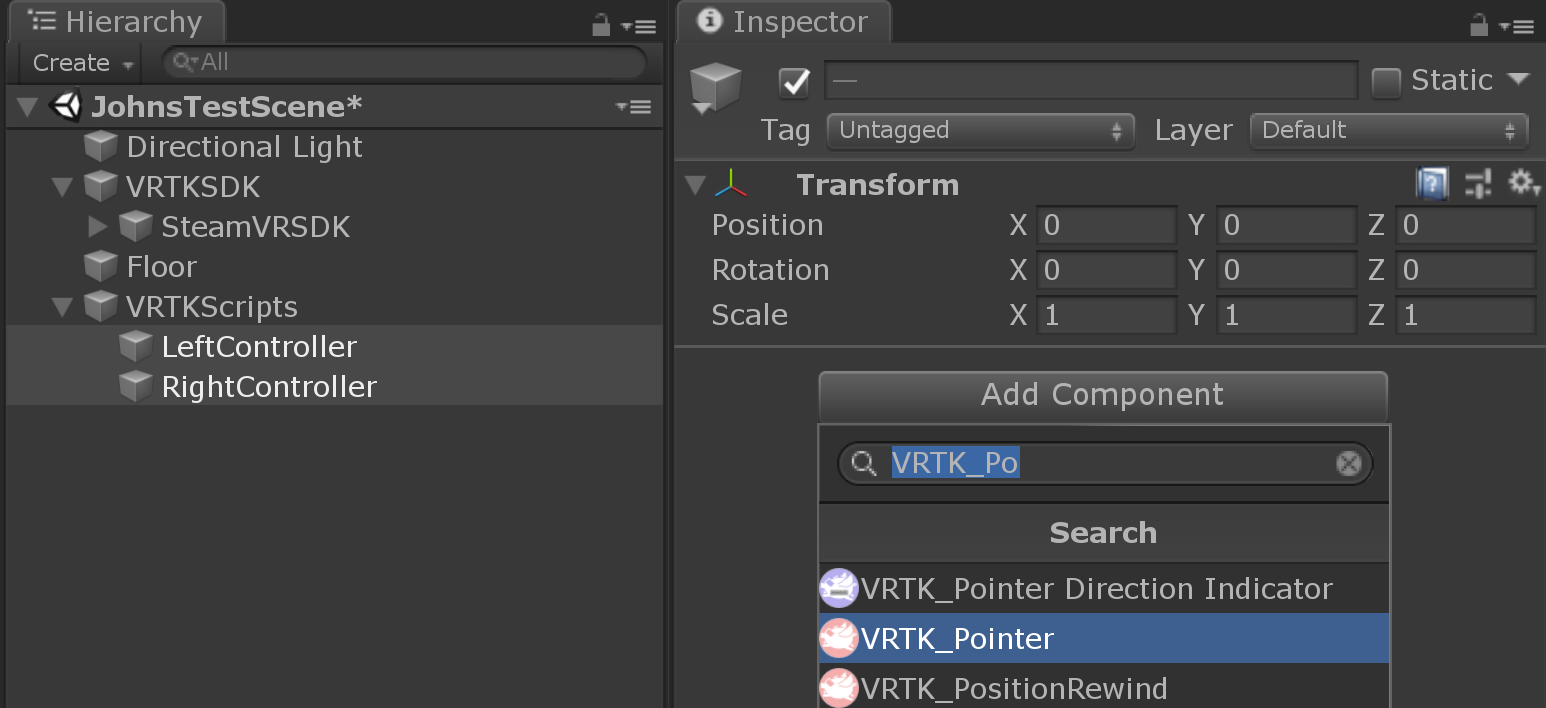
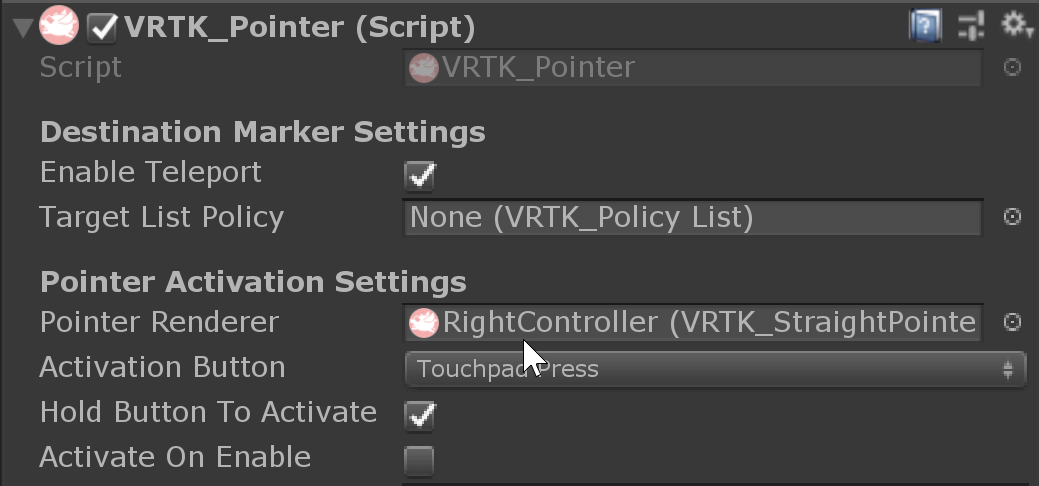
## Adding Basic Movement

1. First we’re going to ensure that all the objects we have added to our scene are set to 0,0,0 for their position   
  
Additionally, ensure that your SteamVRSDK Object is resting above the floor. Adjust your Y Axis if it is clipping with your terrain.

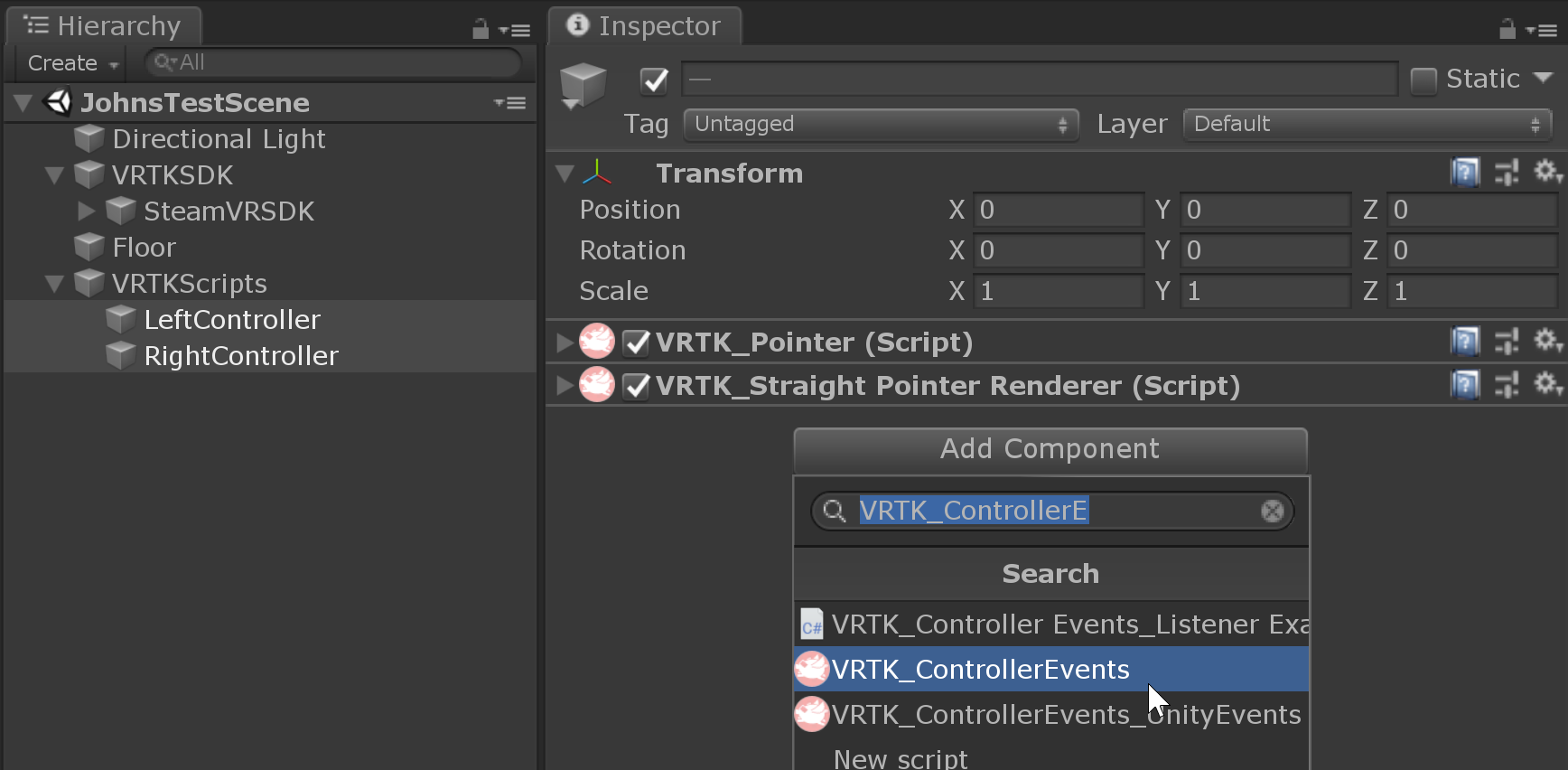
2. Create a new empty GameObject in your scene, and name it VRTKScripts   
Create two new game objects nested under VRTKScripts, name one “LeftController” and the other “RightController”

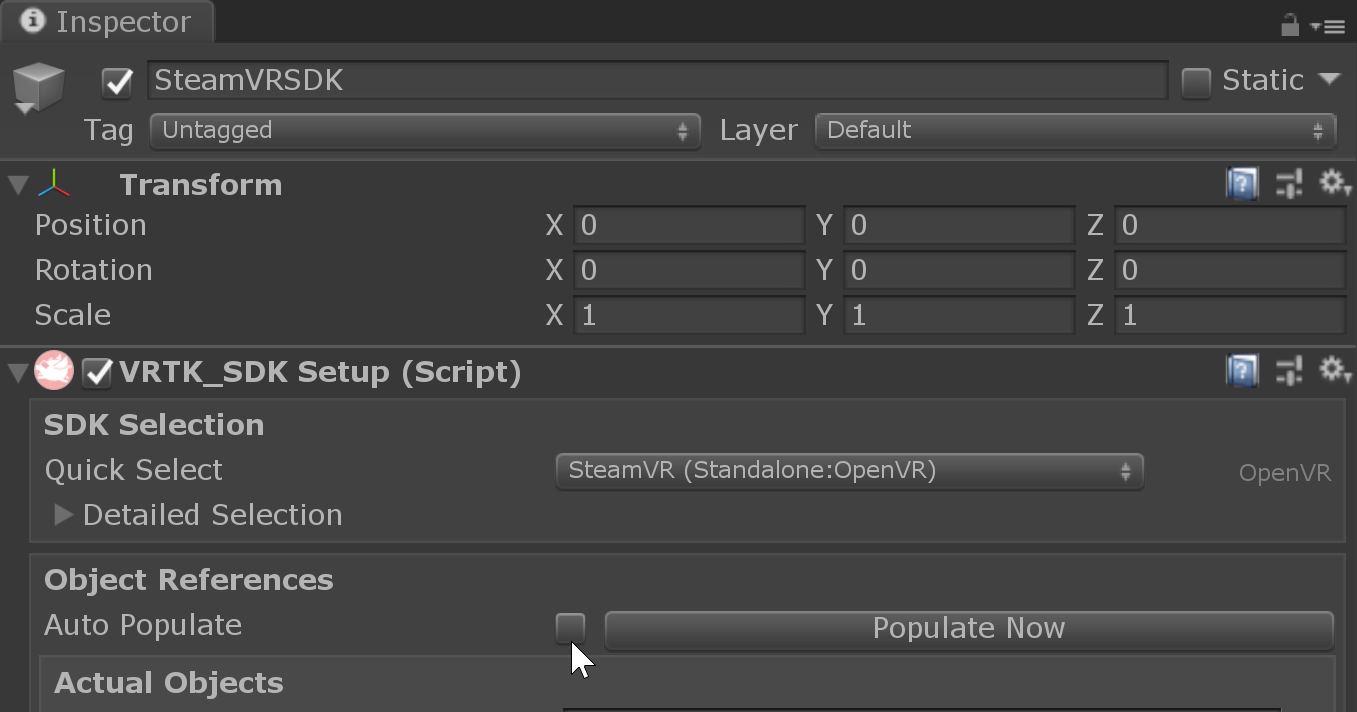
3. Select the VRTKSDK Object and drag the new controller objects into their respective slots



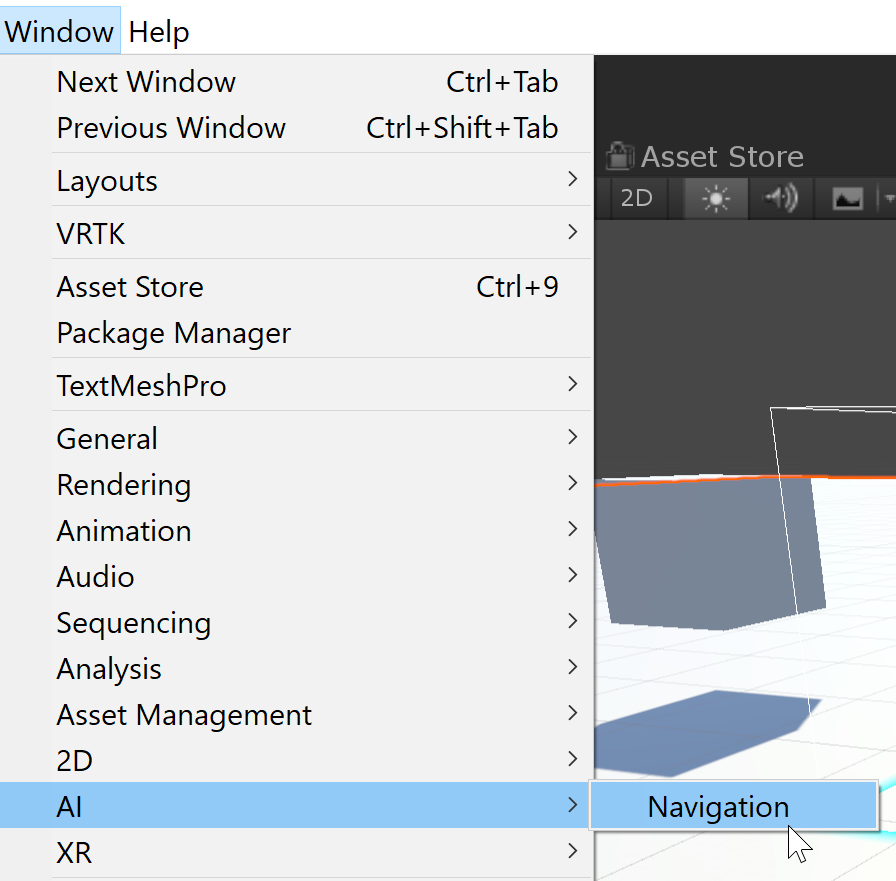
4. Select both of the Controller Objects in your Hierarchy and add the “VRTK\_Pointer” and “VRTK\_Straight Pointer Render” scripts.   
  
  
5. Select your RightController in the Hierarchy and drag the “VRTK\_Straight Pointer Render” component from below and drag it up into the “Pointer Render” section of the “VRTK\_Pointer” Script.   
  
Repeat this for the Left Controller.

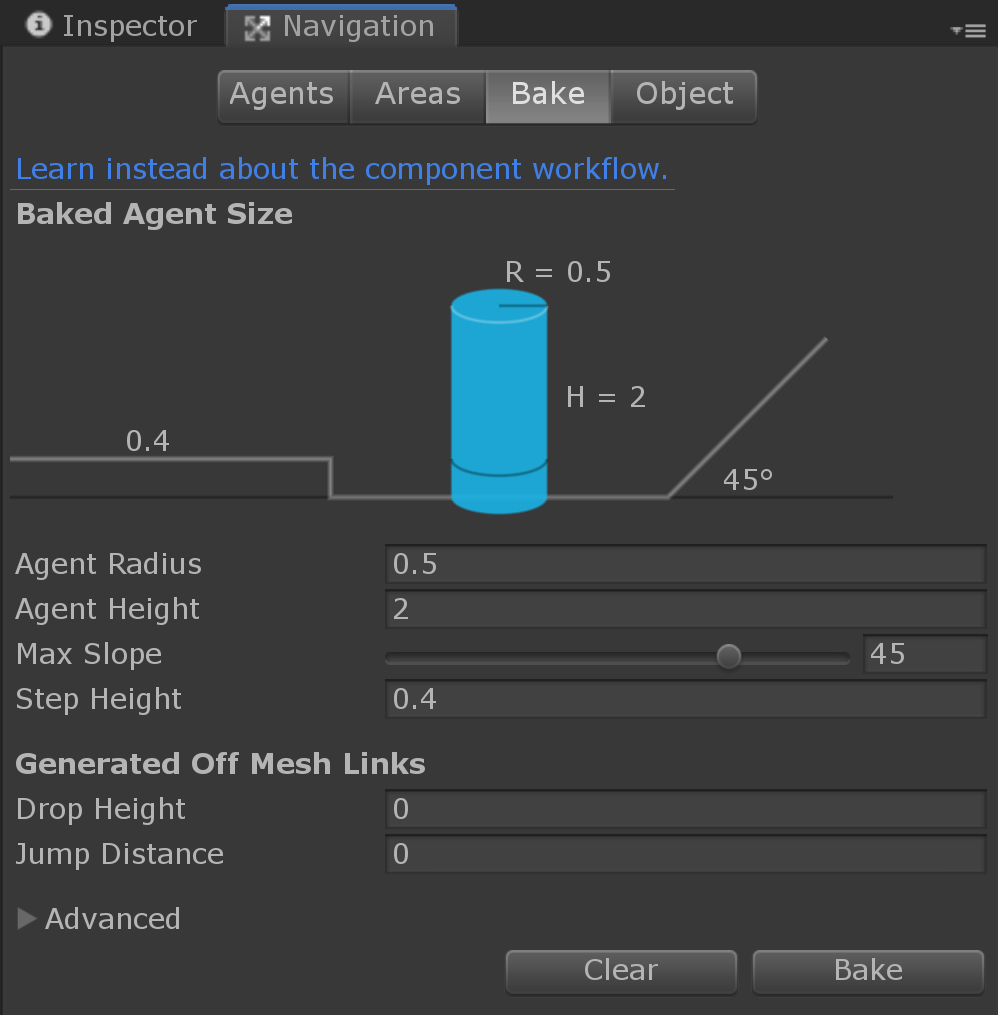
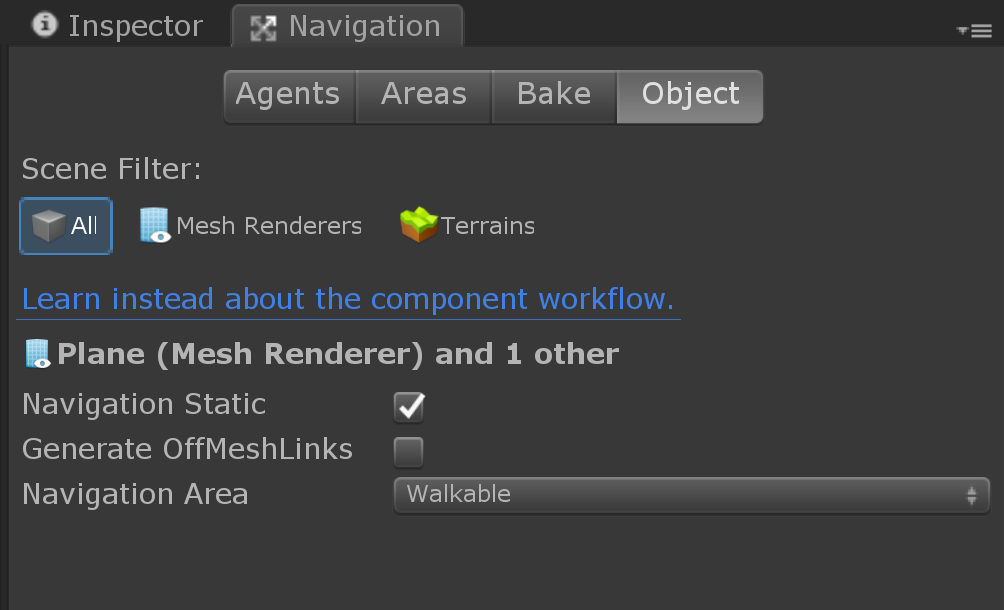
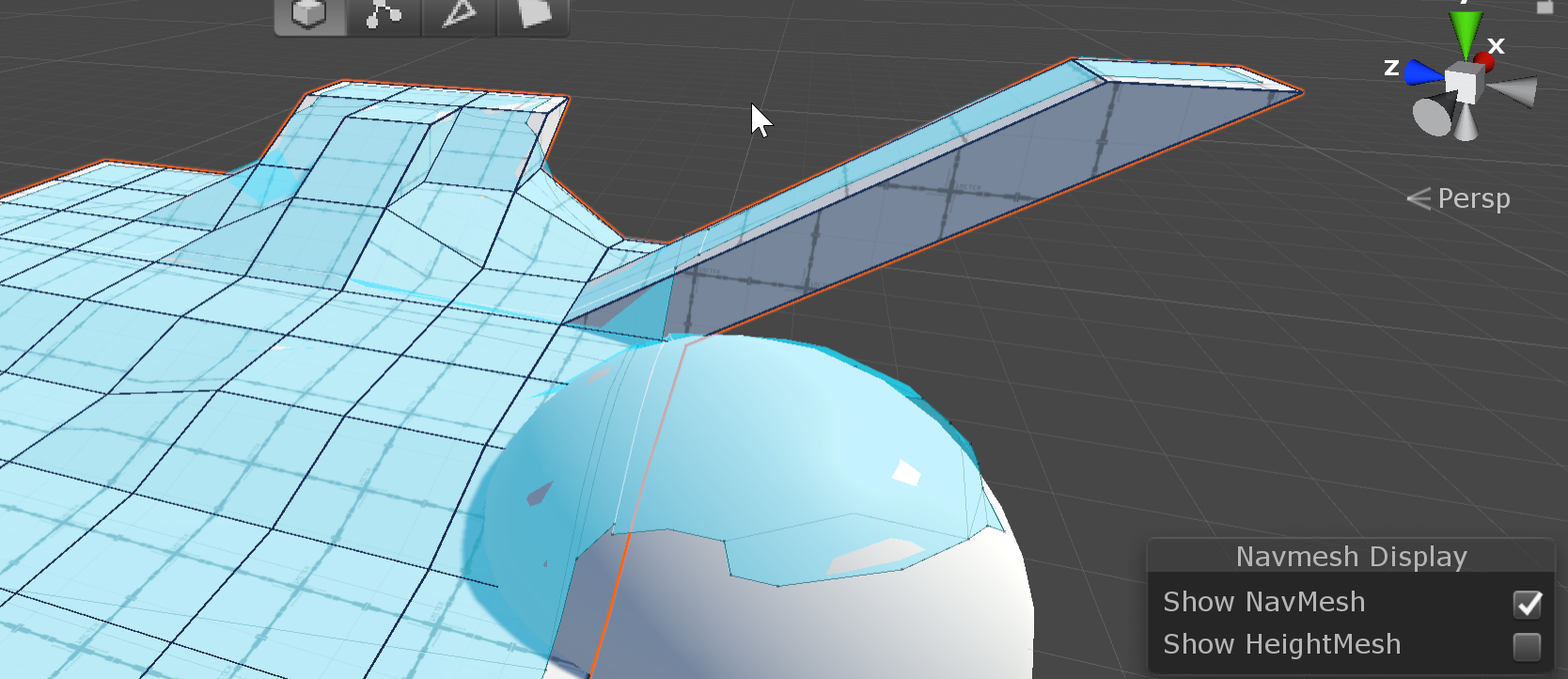
Entering the scene now will render our controllers, but will give us no way to navigate. This is because the controllers have no way to listen to the button presses.

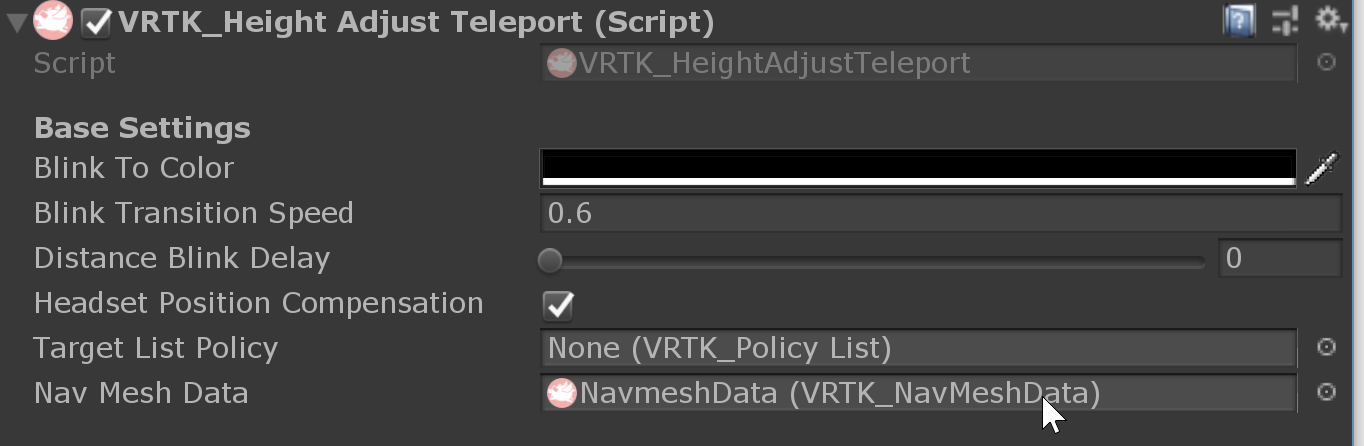
6. With both controllers selected, add the “VRTK\_Controller Events” Script   


7. Navigate to SteamVRSDK and Untick Auto Populate   


8. Create a new empty GameObject under VRTKScripts and add the “VRTK\_Height Adjust Teleport” script.

9. Open the Navigation window via Window > Ai >Navigation  


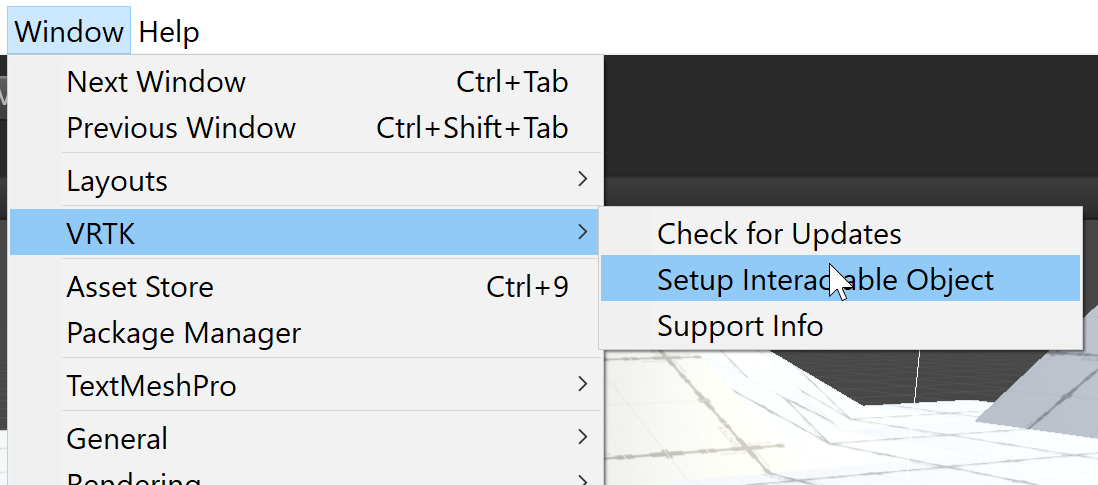
10. Select all the objects you wish to allow the player to walk on, and tick the “Navigation Static” box in the Navigation window. Switch to the “Bake” view and click the “Bake” button.  
  
Your terrain should now have a blue overlay indicating that the player can navigate these areas.   


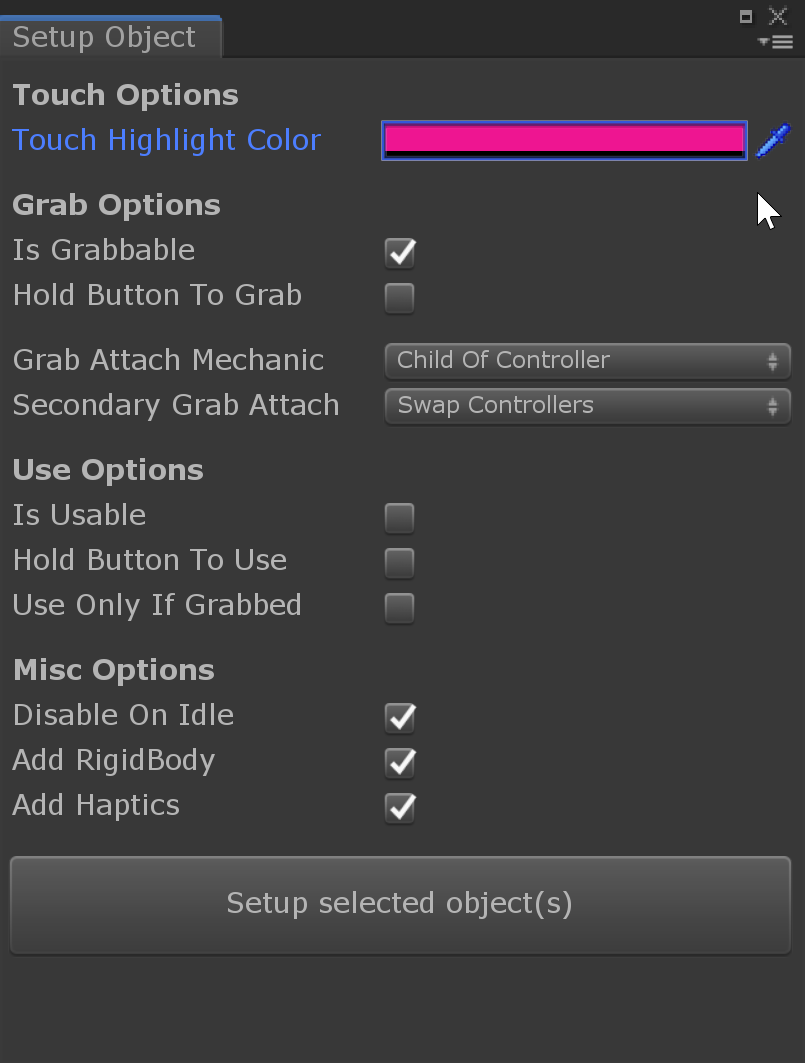
11. Create a new empty GameObject under VRTK Scripts, name it “NavMeshData” and add the “VRTK\_Nav Mesh Data” Script.   
  
Select your “VRTK\_Height Adjust Teleport” script and drag the “NavMeshData” object onto the Nav Mesh Data box.   
****The teleport script will now limit movement to your baked NavMesh.

## Object Interactions

1. Add objects to the scene. Ensure they have coliders.

2. Select your controllers and add the “VRTK\_Interact Touch” and “VRTK\_Interact Grab” scripts.

3. Select your objects, then navigate to Window > VRTK and click “Setup Interactable Object”  


4. The following window will appear, allowing you to adjust in what way the object is intractable.   


5. Change the Touch Highlight colour to something other than black, and click Setup selected object(s)

Your objects will now be interactable, as well as simulate gravity.

# Documentation

* [Unity 2018.4 Documentation](https://docs.unity3d.com/2018.4/Documentation/Manual/index.html)
* [VRTK Documentation](https://vrtoolkit.readme.io/docs)
* [SteamVR Unity Plugin Articles](https://valvesoftware.github.io/steamvr_unity_plugin/articles/intro.html)
* [GitHub Desktop Documentation](https://help.github.com/en/desktop)

# Free Assets

* [Modular Environment - Sci-fi / Industrial](https://assetstore.unity.com/packages/templates/packs/snaps-prototype-sci-fi-industrial-136759)
* [Modular Environment - Office](https://assetstore.unity.com/packages/templates/packs/snaps-prototype-office-137490)
* [Particle Pack (Fire, water, etc)](https://assetstore.unity.com/packages/essentials/tutorial-projects/unity-particle-pack-127325)
* [Adam Character Pack (3 Sci-fi characters)](https://assetstore.unity.com/packages/essentials/tutorial-projects/adam-character-pack-adam-guard-lu-74842)
* [Standard Assets (Jet, Car, Character)](https://assetstore.unity.com/packages/essentials/asset-packs/standard-assets-32351)
* [Motion Capture data for Bipedal Characters](https://assetstore.unity.com/packages/3d/animations/raw-mocap-data-for-mecanim-5330)
* [Realistic Forest Environment (Ask John before using this one)](https://assetstore.unity.com/packages/essentials/tutorial-projects/book-of-the-dead-environment-121175)
* [GitHub Student Pack](https://education.github.com/pack)

<https://education.github.com/pack>