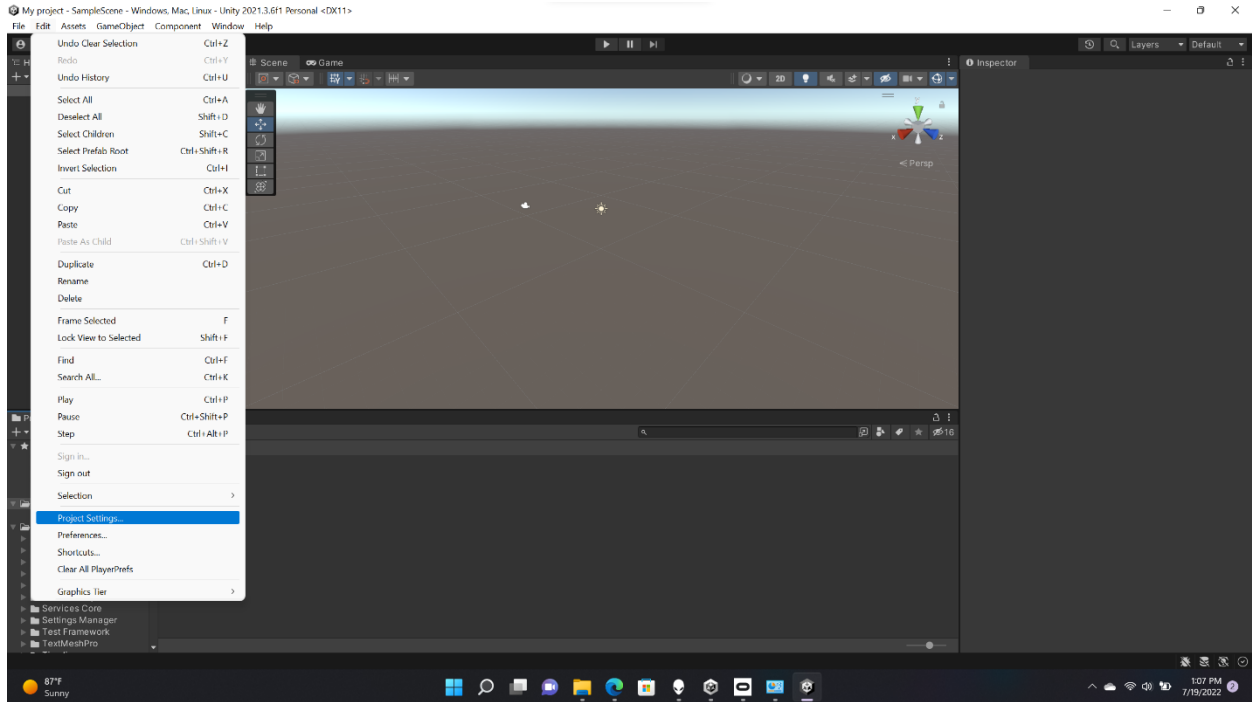


Set Up Unity project with Quest 2

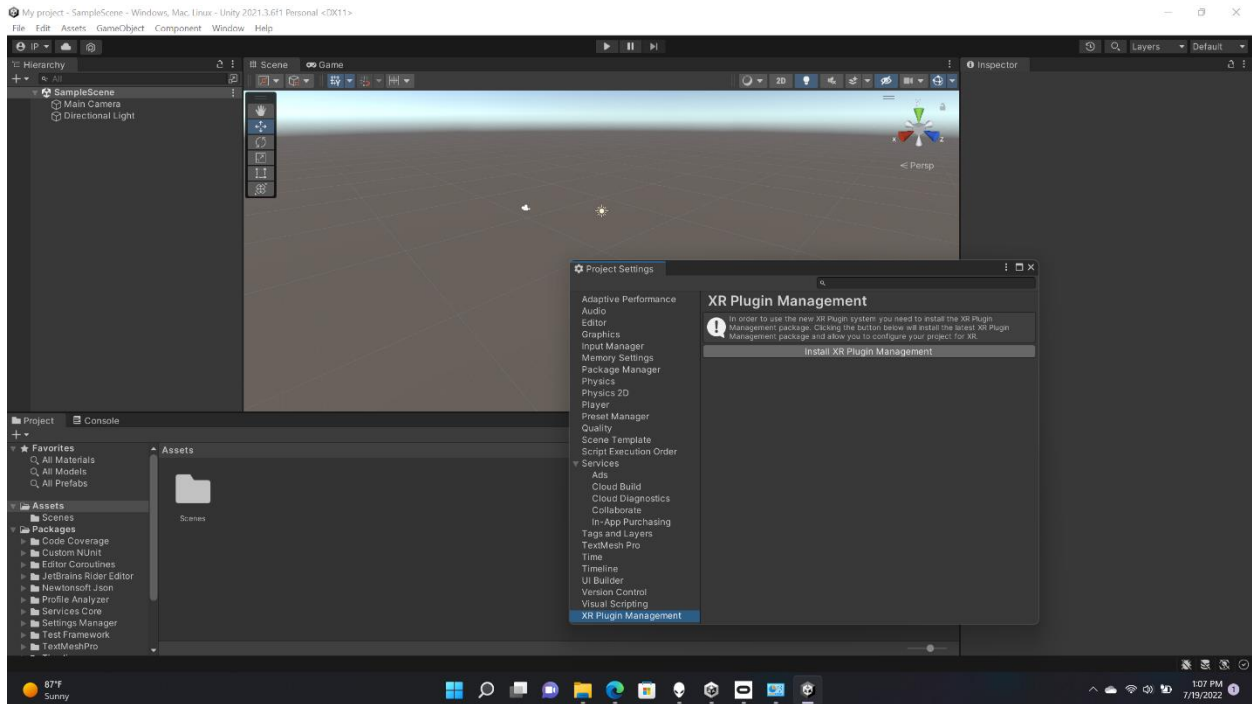
This tutorial was made with Unity version 2021.3.6f1, it'll probably work with others but be aware. A couple of things will be needed before you start, first download the oculus app on your computer and log in with a Facebook account. The oculus quest must be in developer mode and to do that you need a verified account. It's not too hard but you'll need either a phone number that can receive texts or a credit card that can be attached to the account. You can do this at [Oculus Developer Center](#) by logging in with the account the Oculus will be used with.

1. Install that version 2021.3.6f1 of unity if you don't already and add the following modules to it
 1. Android Build Support, OpenJDK & Android SDK & NDK Tools
2. Open a new 3D Unity project
 - It is **ESSENTIAL** that you name the project with your FIU username before it, for example "jsmith01-myproject". This is important because different projects with the same name can cause issues with deployment to the Quest 2. So please use the naming scheme above to keep the names of your projects unique and easily identified.

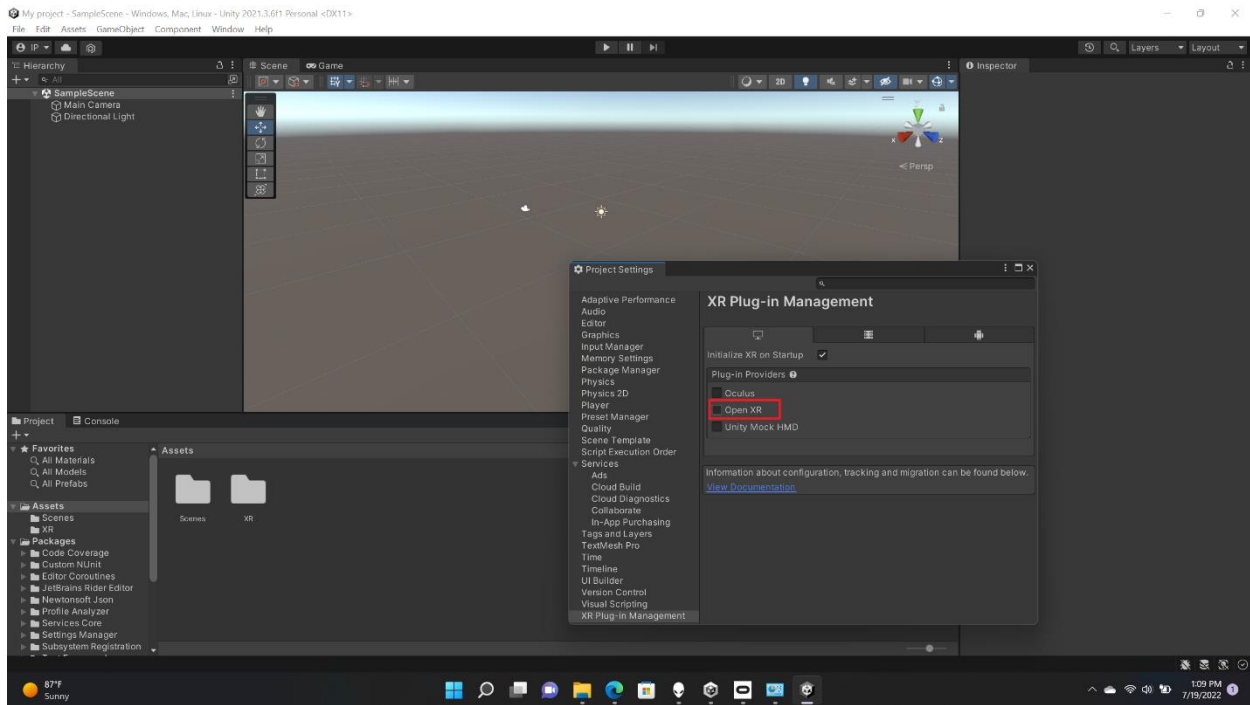
3. Once open, in the unity window page go to edit -> project settings



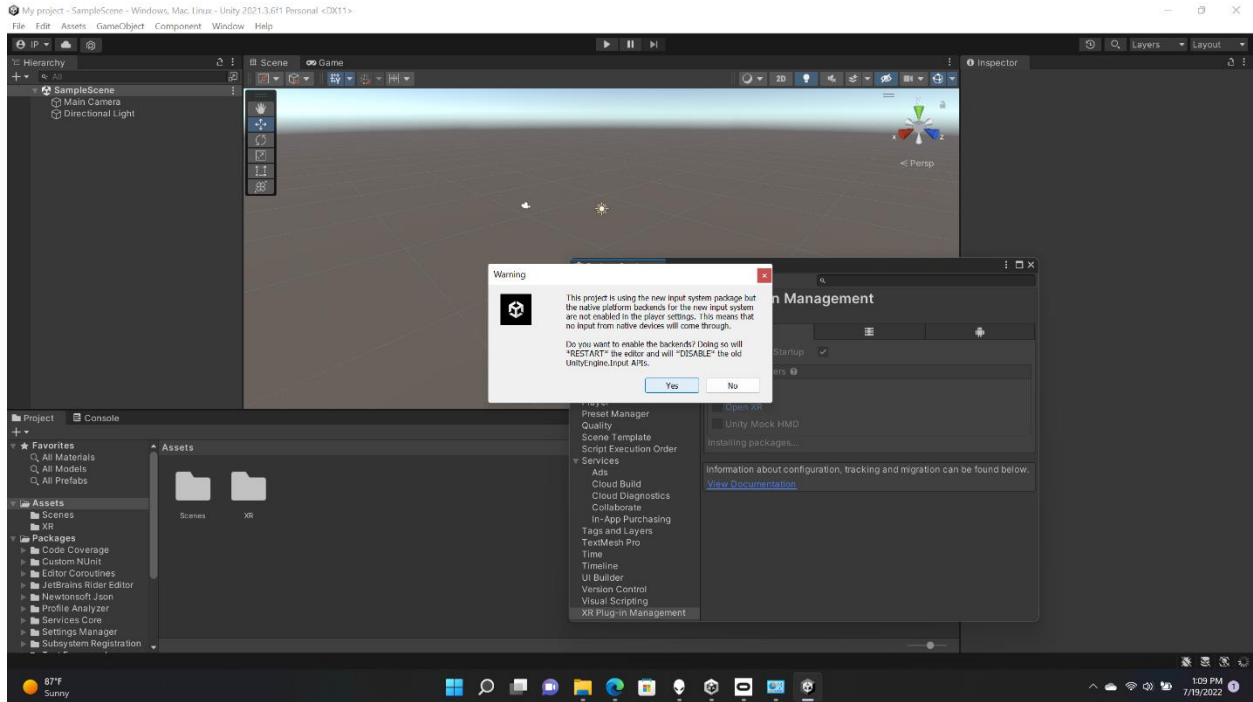
4. Go to XR Plug-in Management and click "Install XR Plug-in Management"



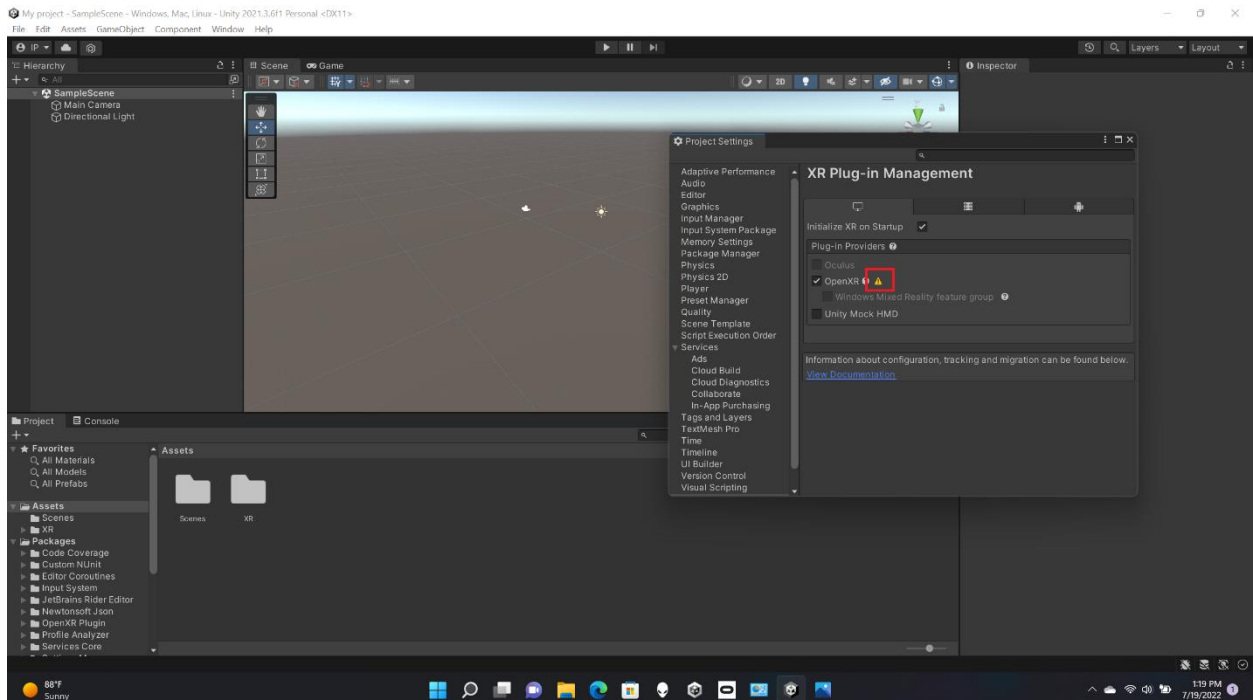
- You'll then be on an options menu for the XR Plug-in, you're on the option for Windows, Mac, and Linux right now, go ahead and click Open XR



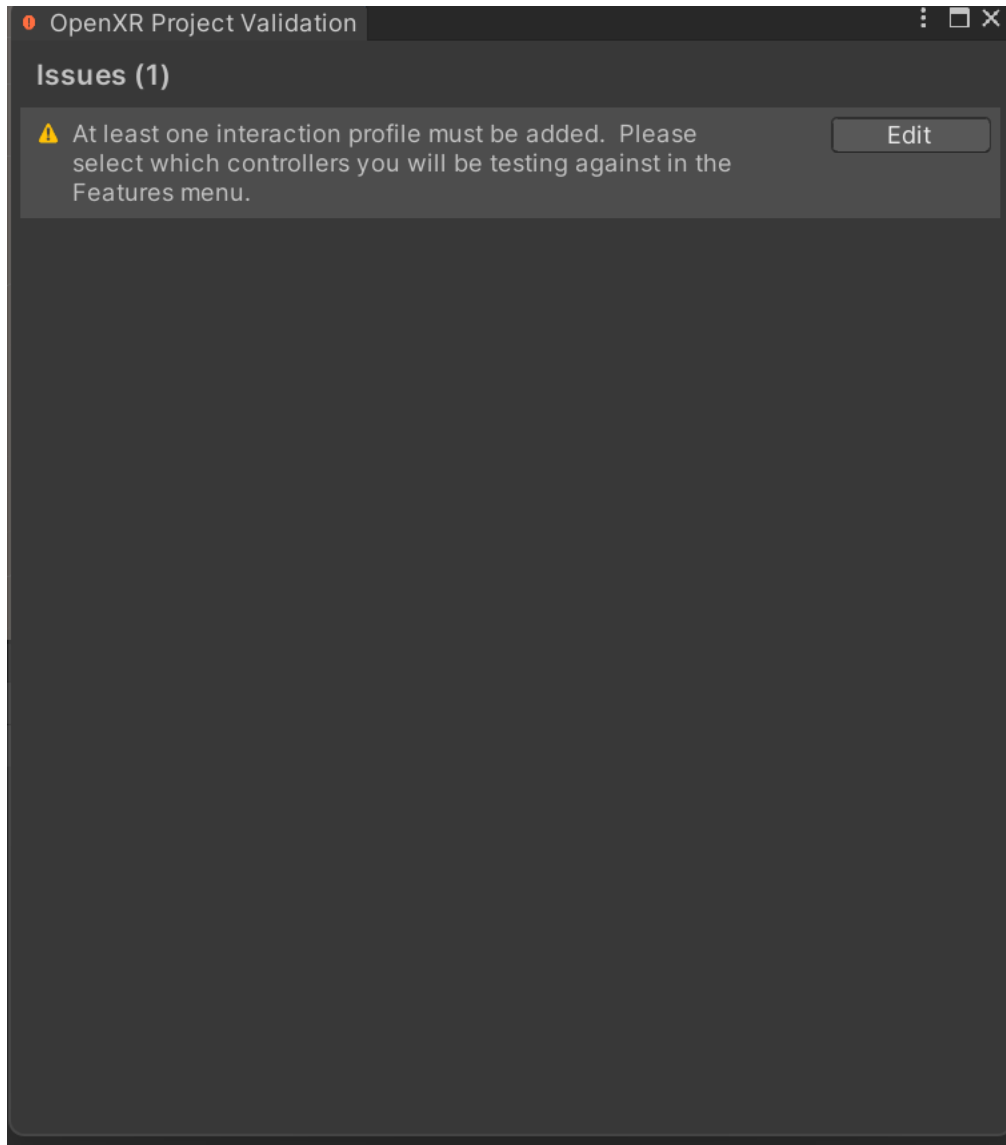
- Once done, wait a bit and you'll get a warning from Unity, just click yes; Unity will reboot itself just wait for a second



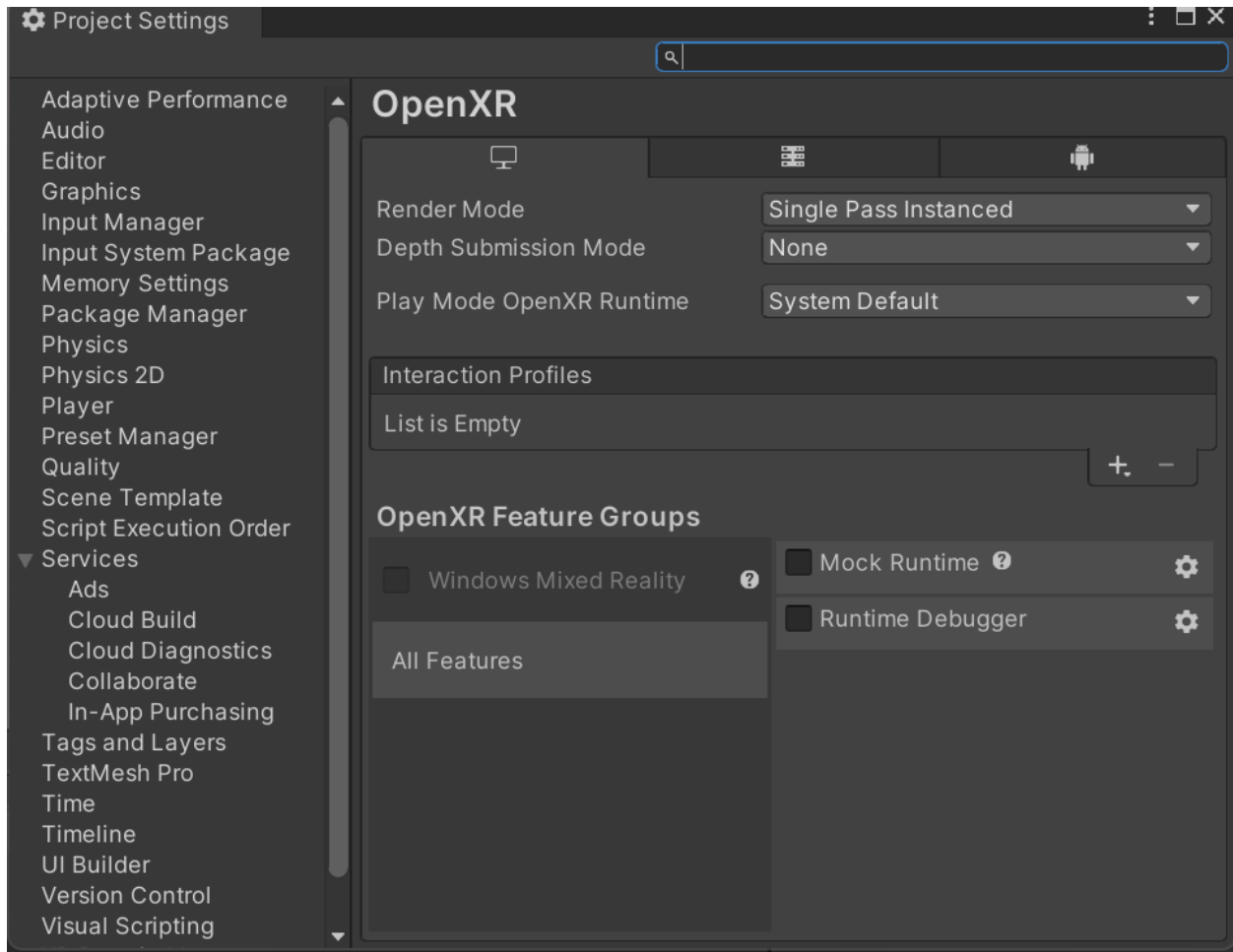
- When the Unity editor resets, you should notice a warning symbol next to the Open XR Selection. Click on that.



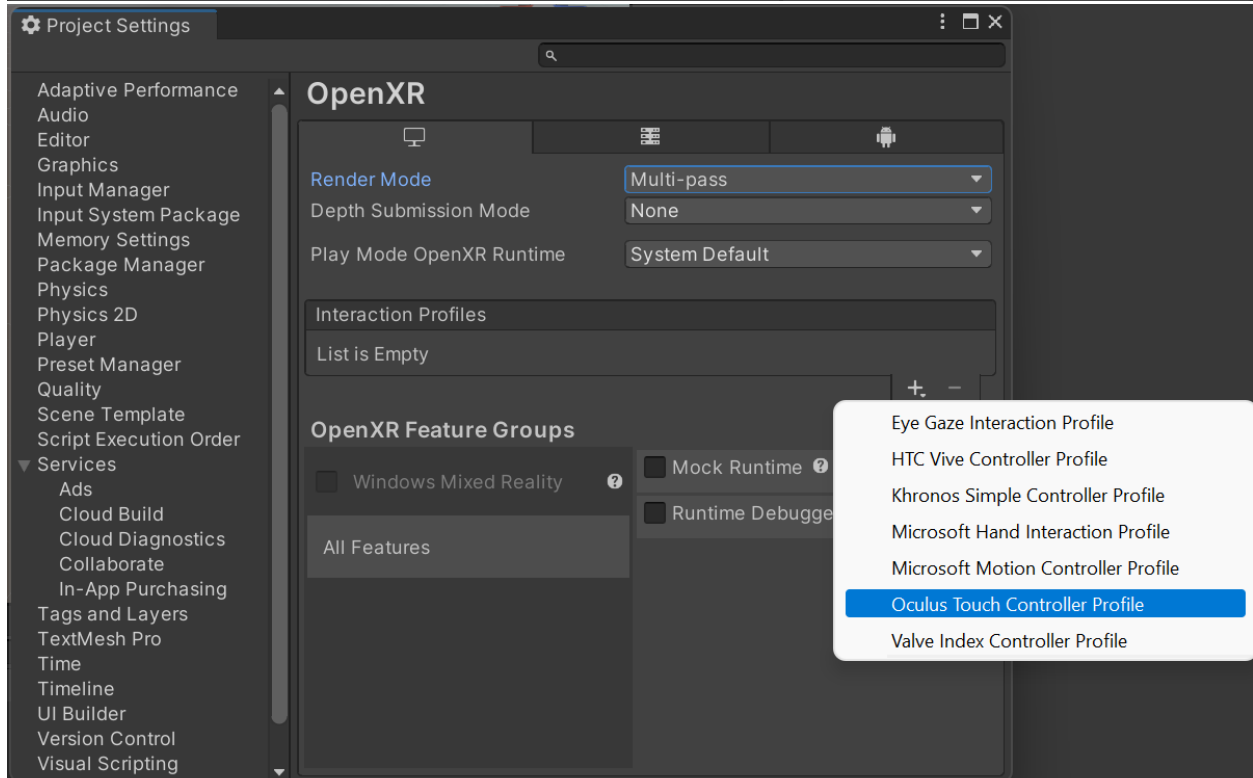
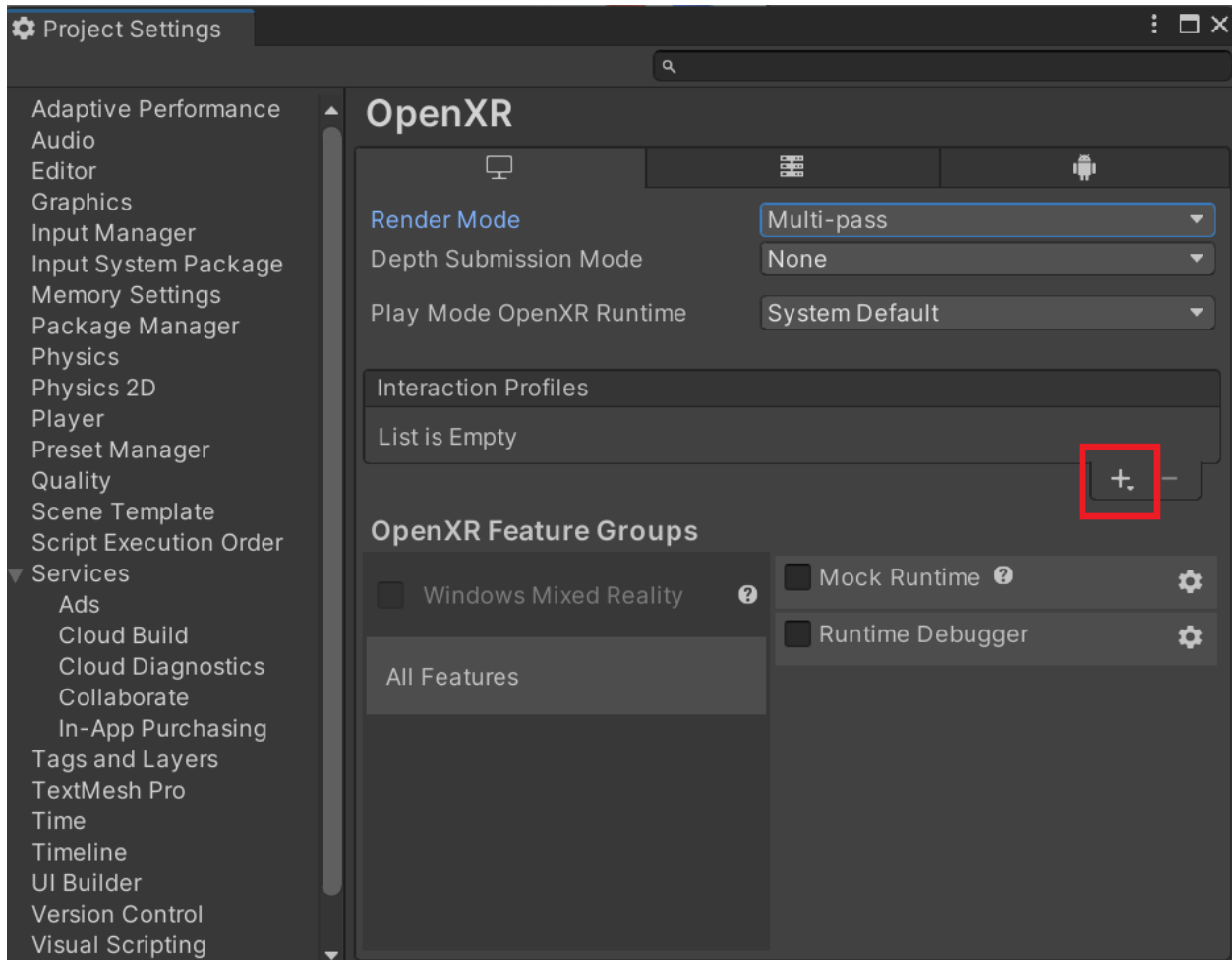
8. When you click on it, you'll see the following window pop up, select "Edit"



9. Your project window should have changed into this, first change the render mode from “Single Pass Instanced” to “Multi-pass”

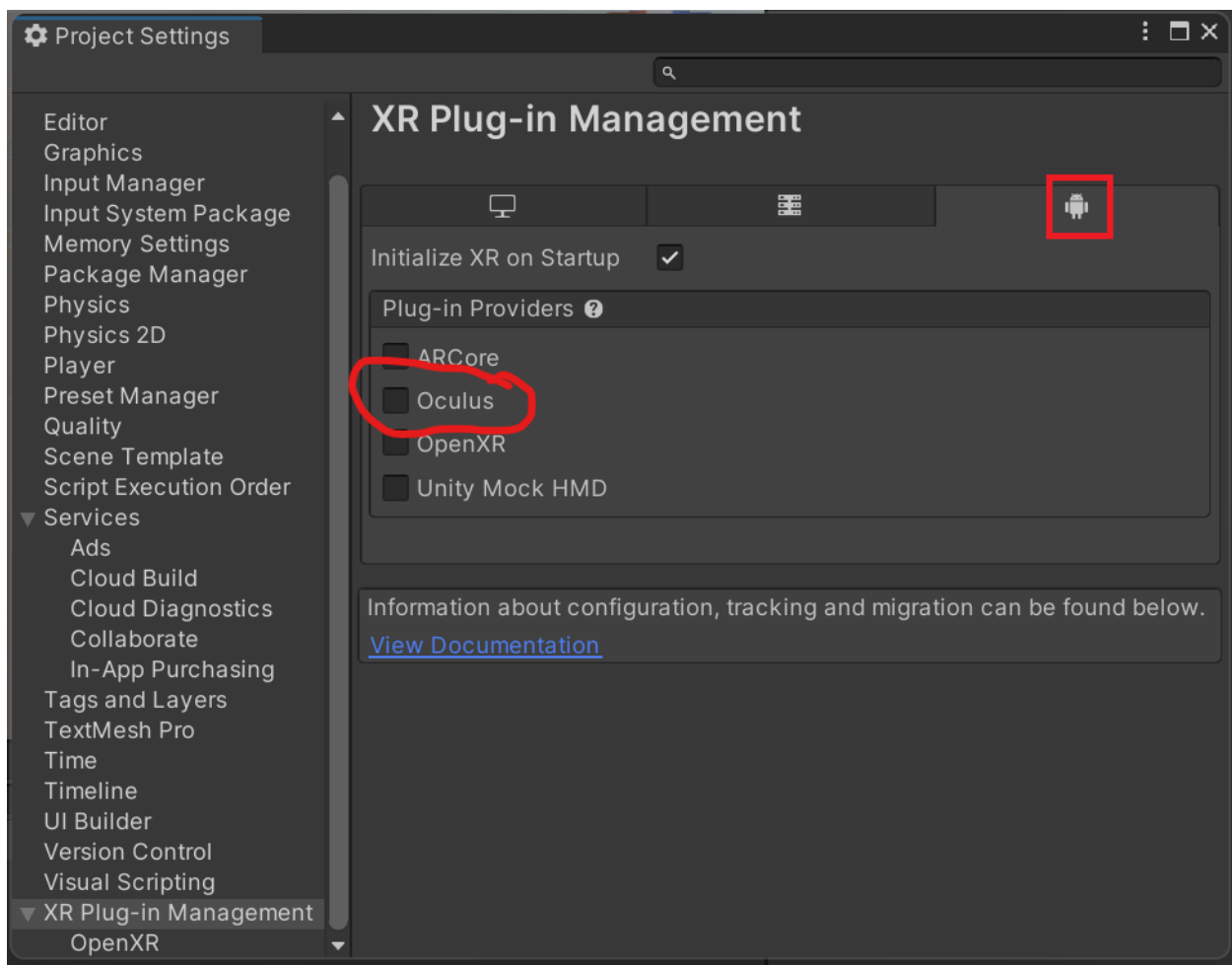


10. Then you'll want to press the + button under "Interaction Profiles" and select "Oculus Touch Controller Profile"

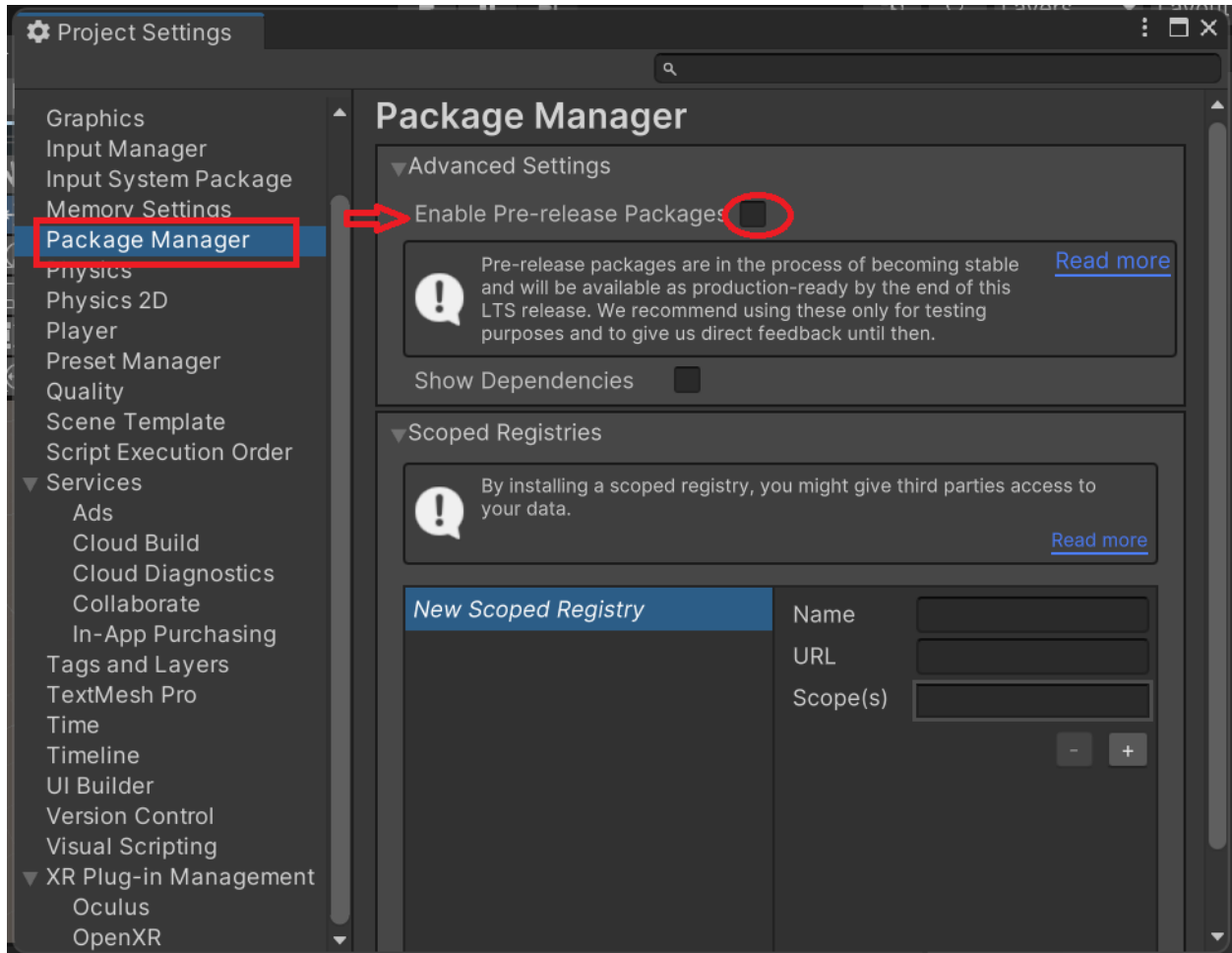


11. Now go back to “XR Plug-In Management”, we’re going to set up the Android settings now. The PC environment is for VR using the PC with the oculus wired in, the Android environment allows you to play applications on the quest 2 without the aid of a computer.

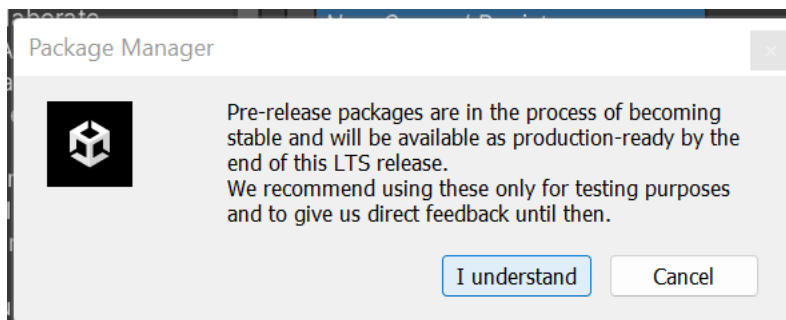
12. On the XR Plug-In Management menu, click the Android option and click the Oculus check box.



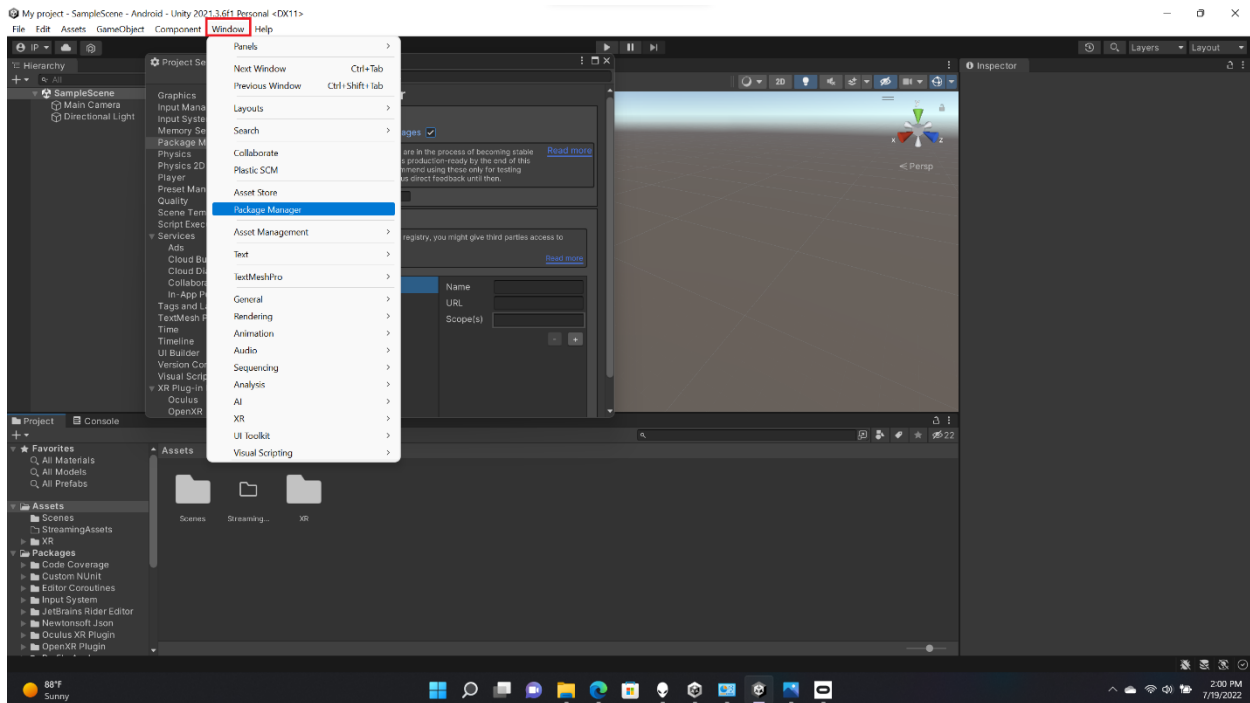
13. Then look on the left list in project settings for “Package Manager” Click it and you’ll see a check option under “Advanced Setting” called “Enable Pre-release Packages”, click that.



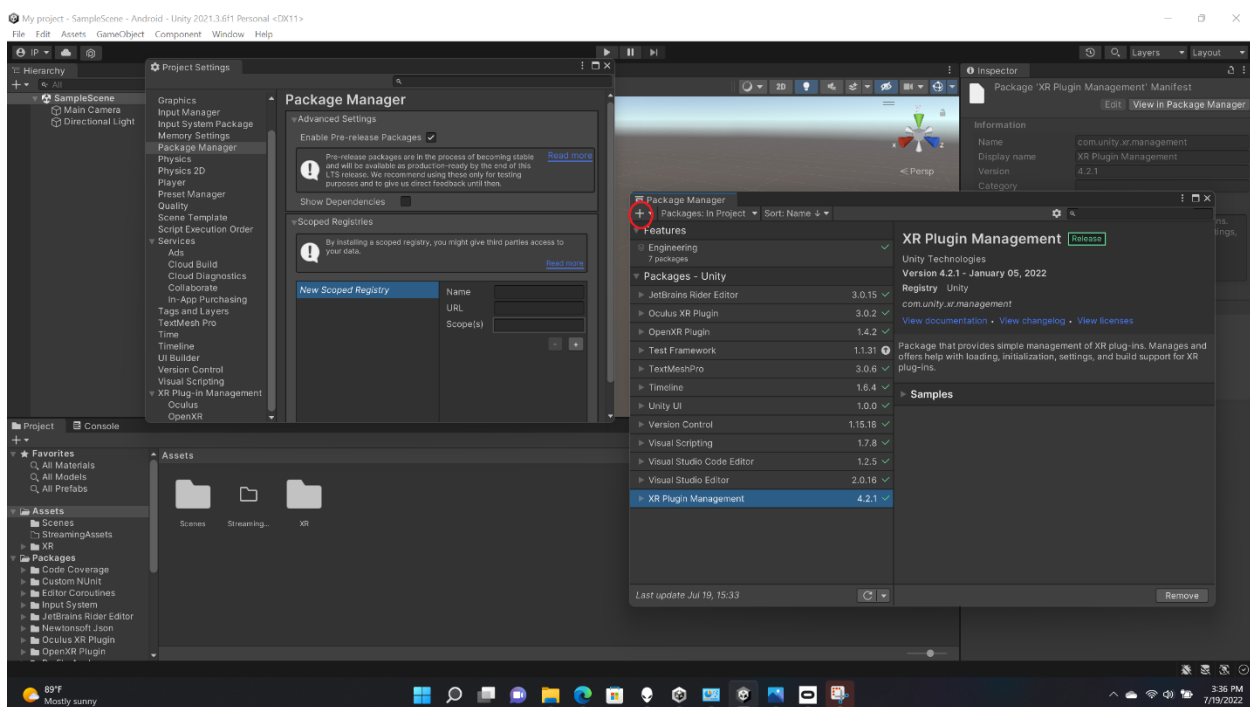
14. You’ll see this window pop up, just click “I understand”

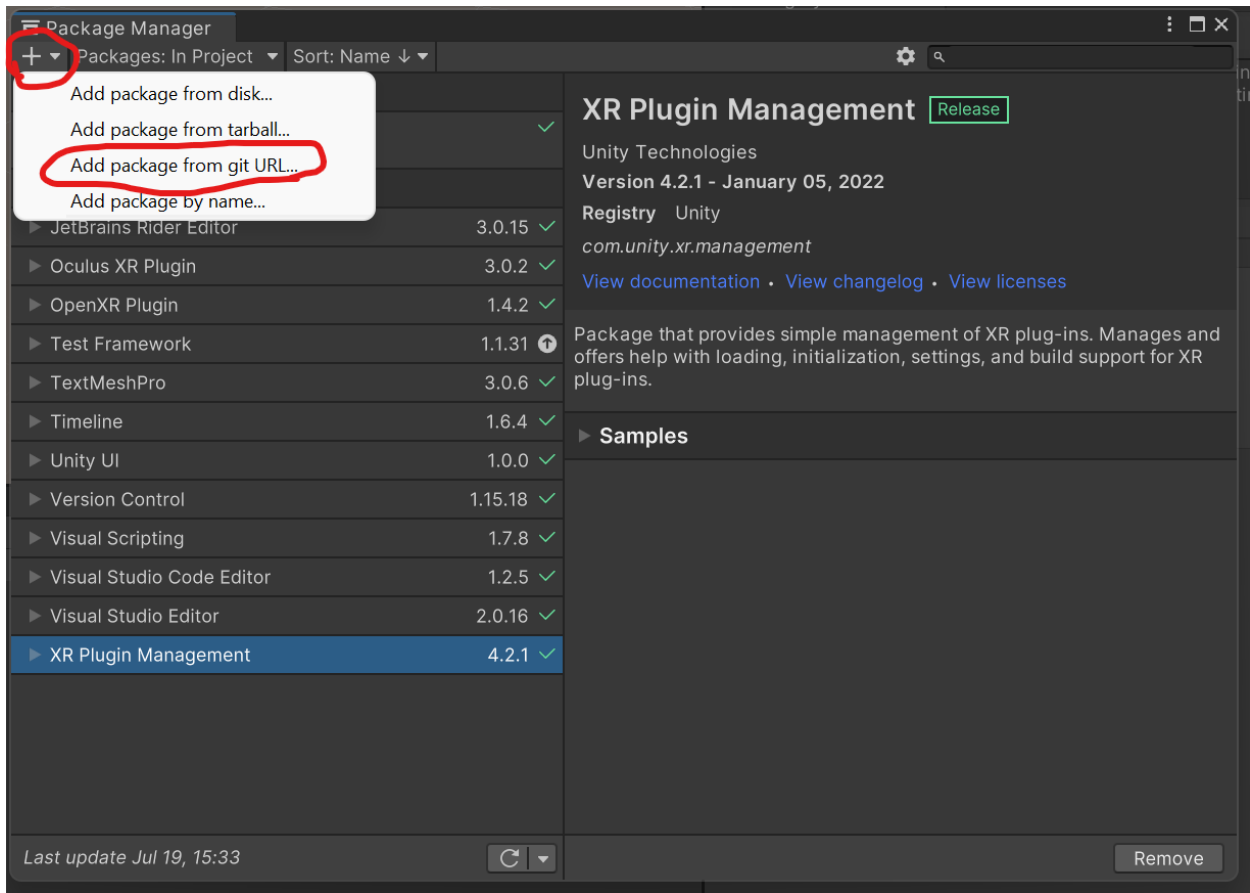


15. After clicking on “I understand” button, go to the top of the screen and click on “Window”, from the drop down go ahead and click on “Package Manager”

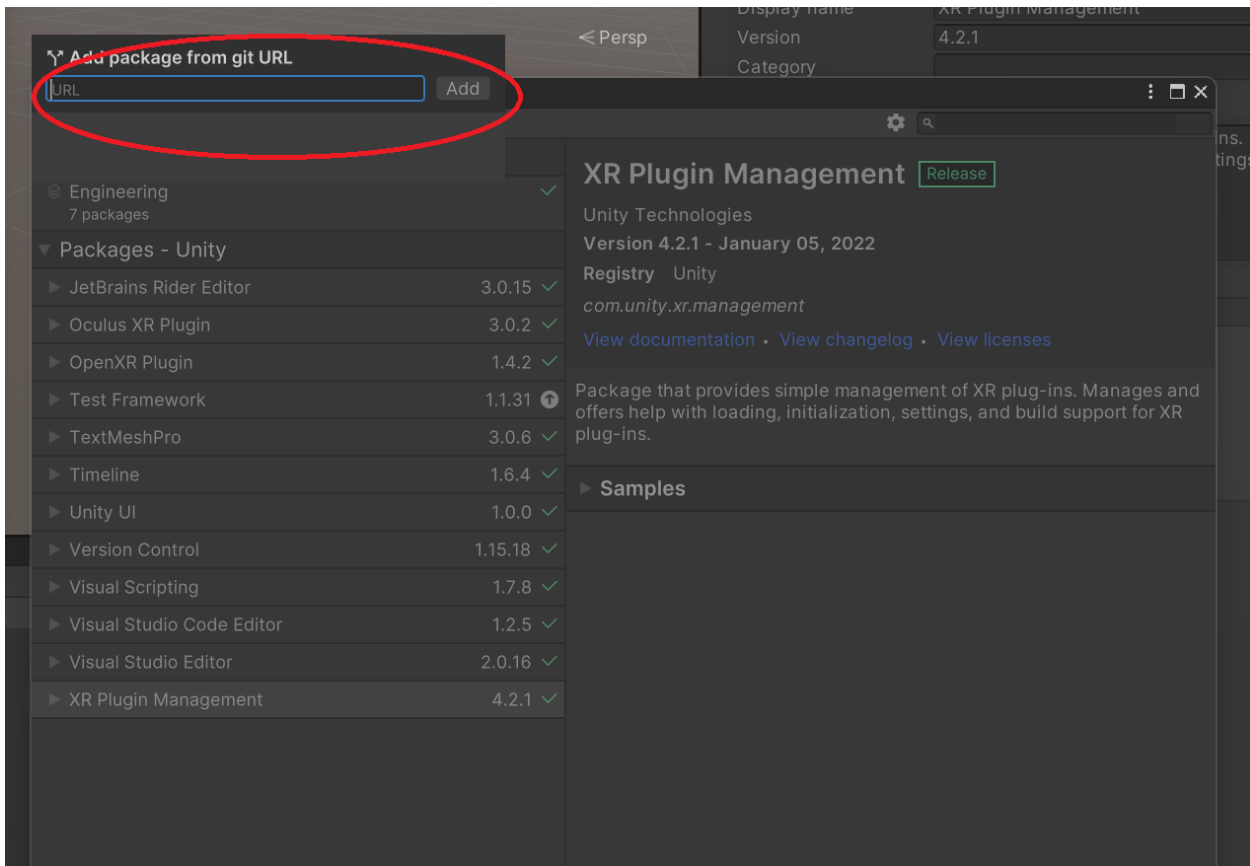


16. You should see a new window pop up. From here go to the top left corner of the new window and click on the “plus” (+) icon. When clicked, a drop down will appear, from this list go ahead and click on the one that says “Add package from git URL...”

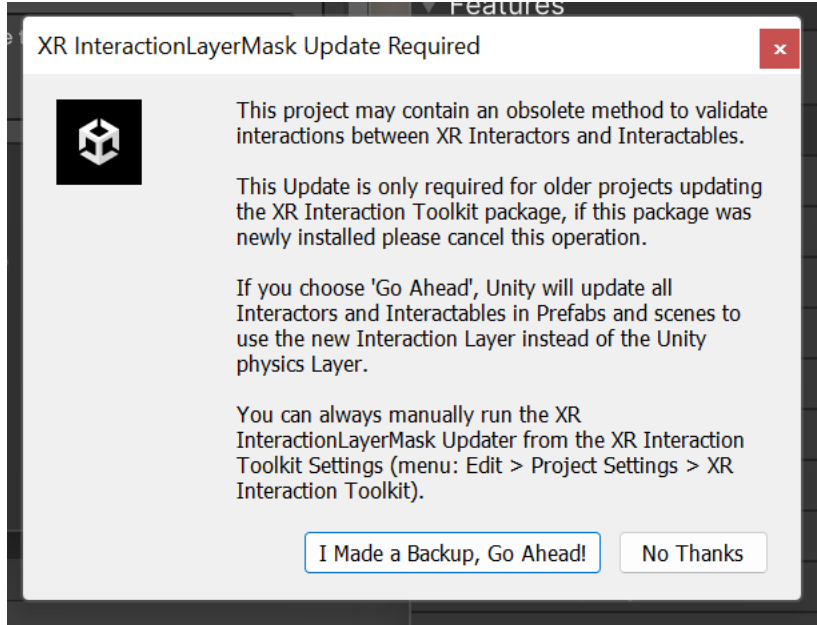




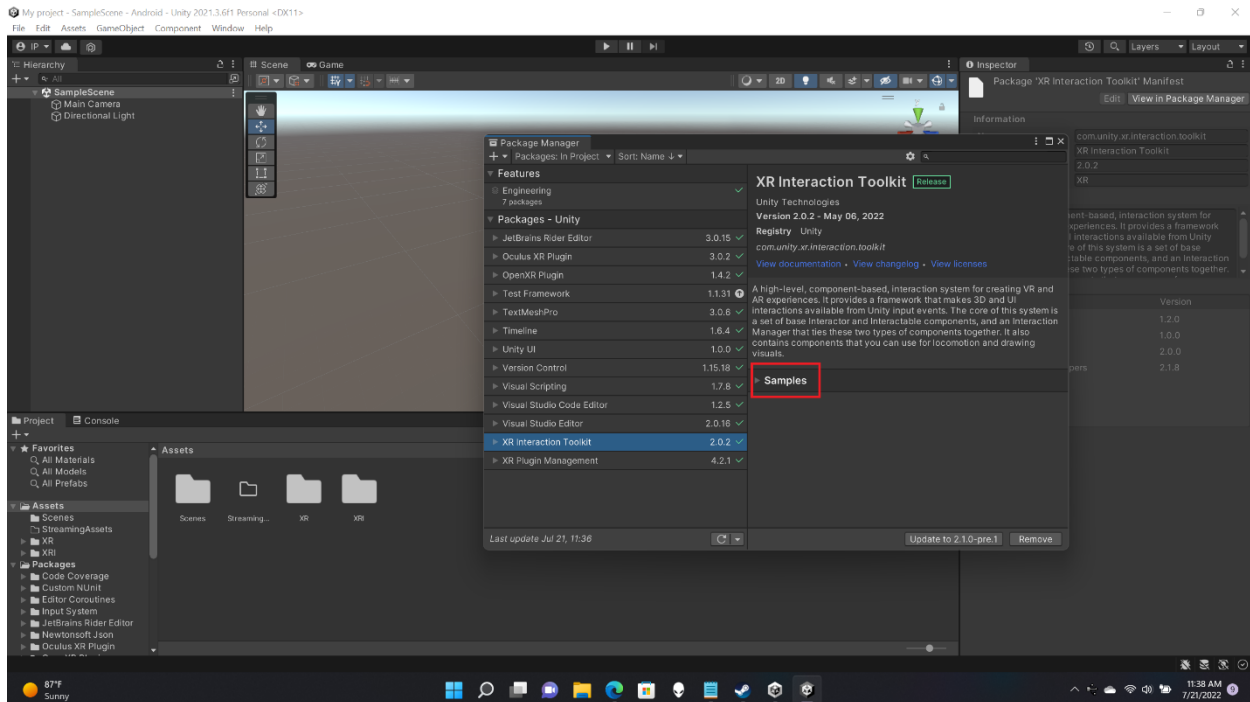
17. A text box will appear and you'll want to type "com.unity.xr.interaction.toolkit" into it.



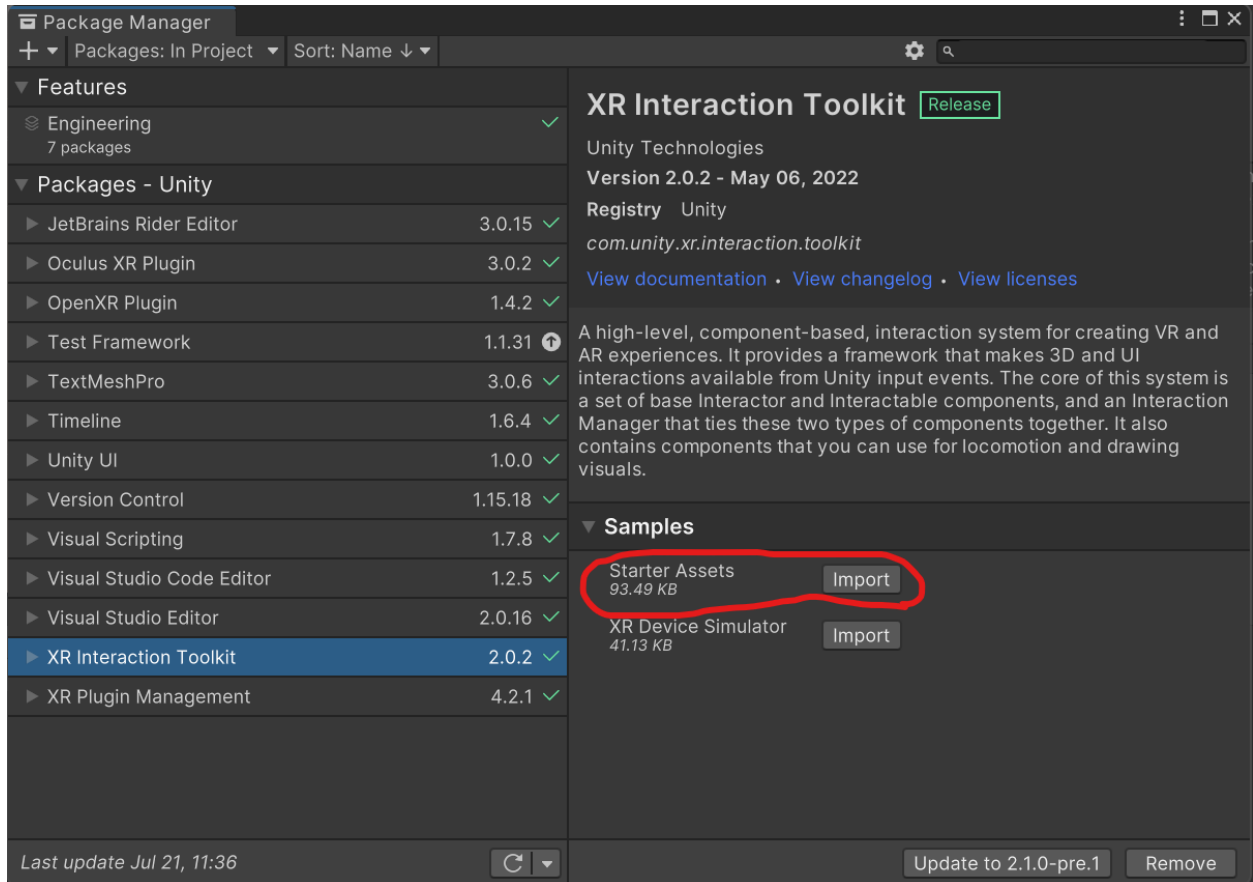
18. You'll get a warning just select "I Made a Backup, Go Ahead!"



19. After clicking the button "Yes I Made a Back Up" and after the packages have been installed, go click on "Samples" from the mini drop down

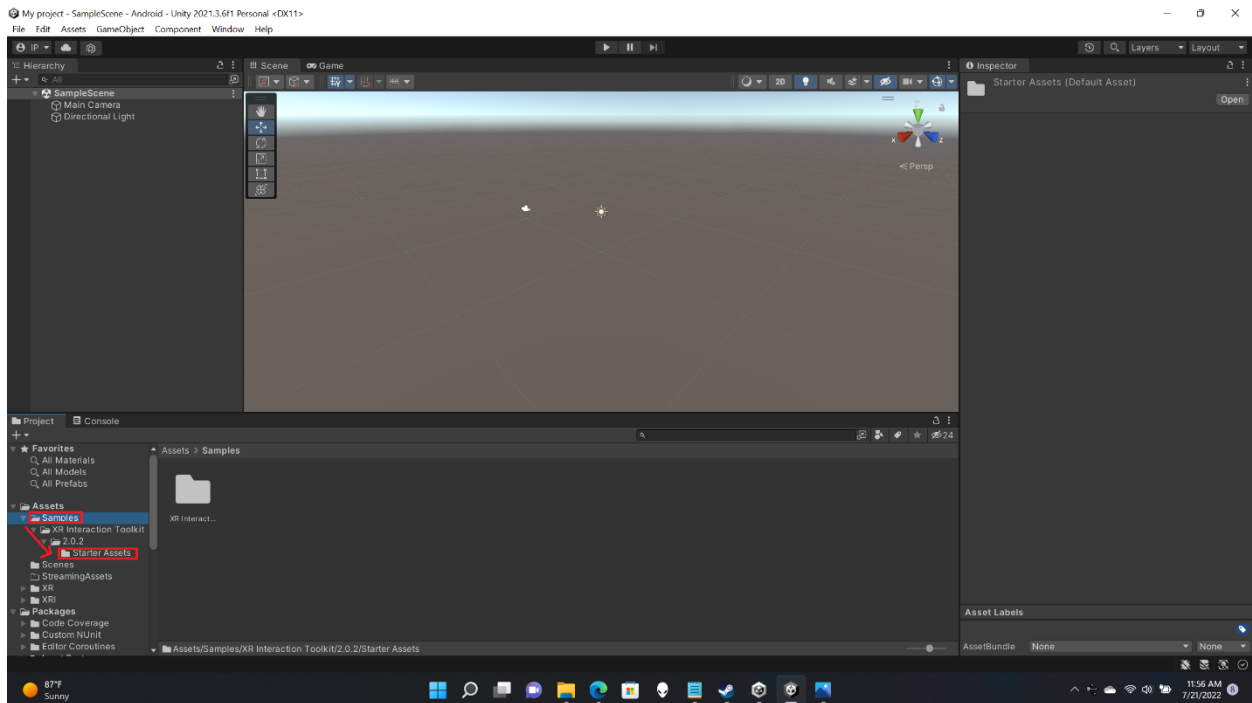


20. You will see “Starter Assets” go ahead and click on the “Import” button. When clicked Unity will go ahead and download the default input actions that we will be using for this project.



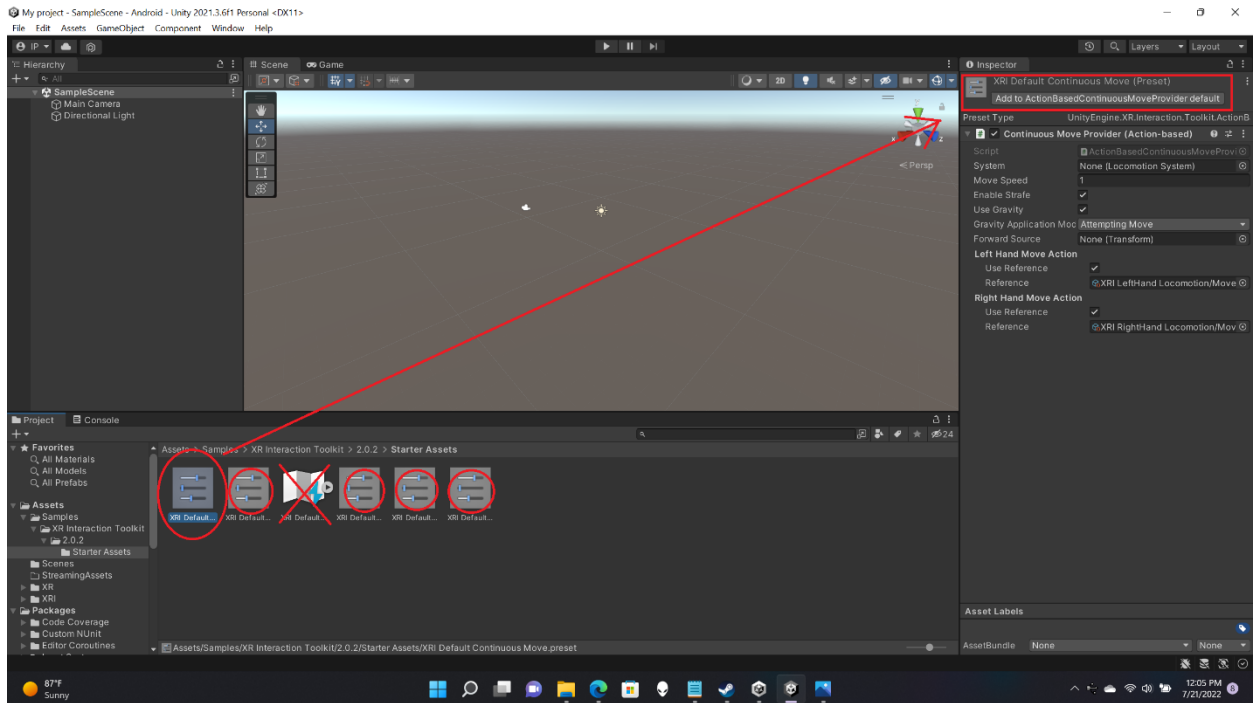
21. You can close out all pop ups now and go back to the main unity workspace

22. Go to the project folder and click on the Assets. From there go to Samples -> XR Interaction Toolkit -> 2.0.2 -> Starter Assets

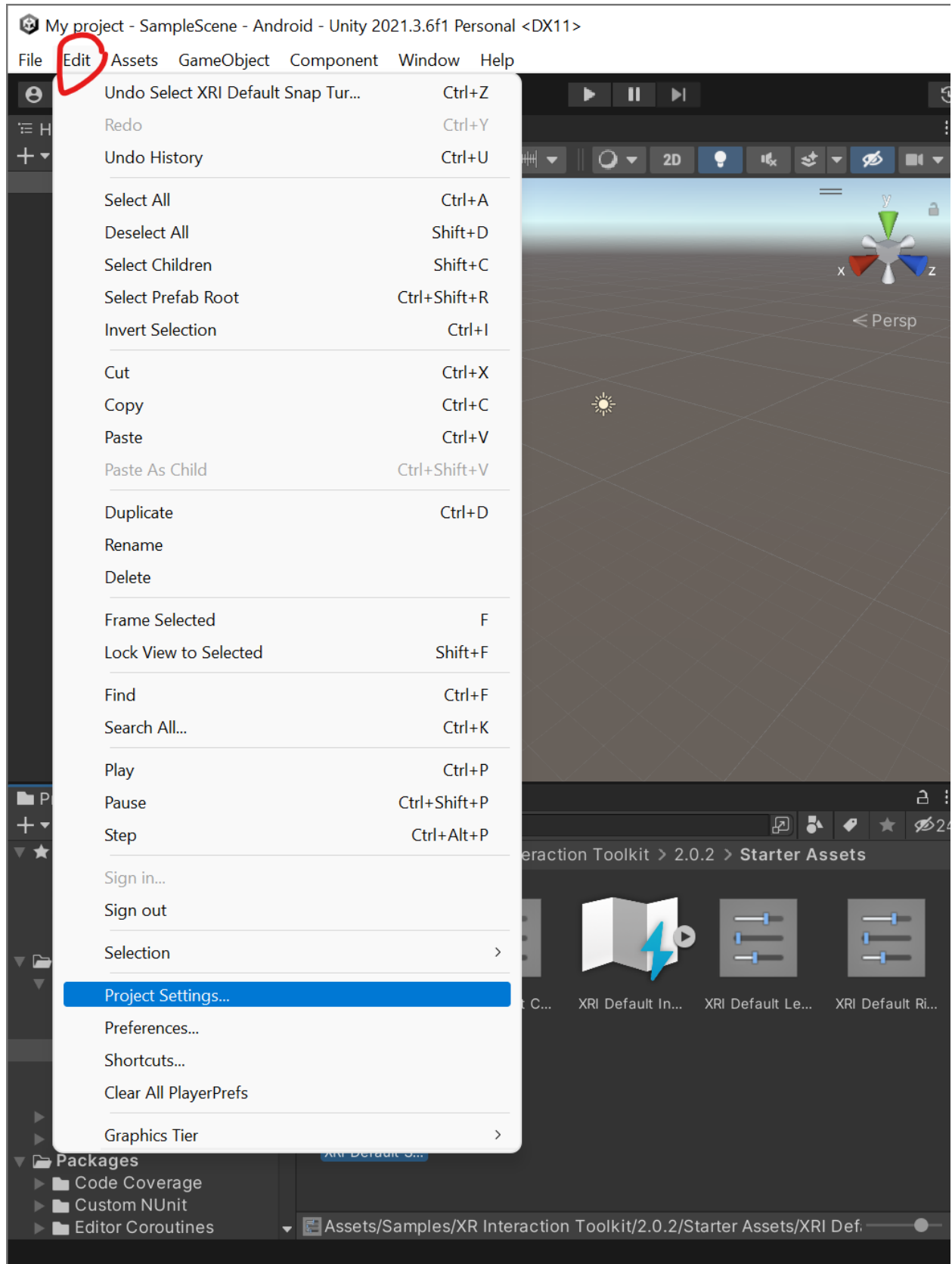


23. You'll see 6 options, ignore the one labeled XRI Default Input Actions for now and focus on the others

24. Click on EACH OPTION, go to the Inspector menu and click on the button that begins with “Add to ActionBased”. I.E., in XRI Default Continuous Move, the button is called “Add to ActionBasedContinuousMoveProvider default”. Do this for ALL files in this folder barring XRI Default Input Actions.

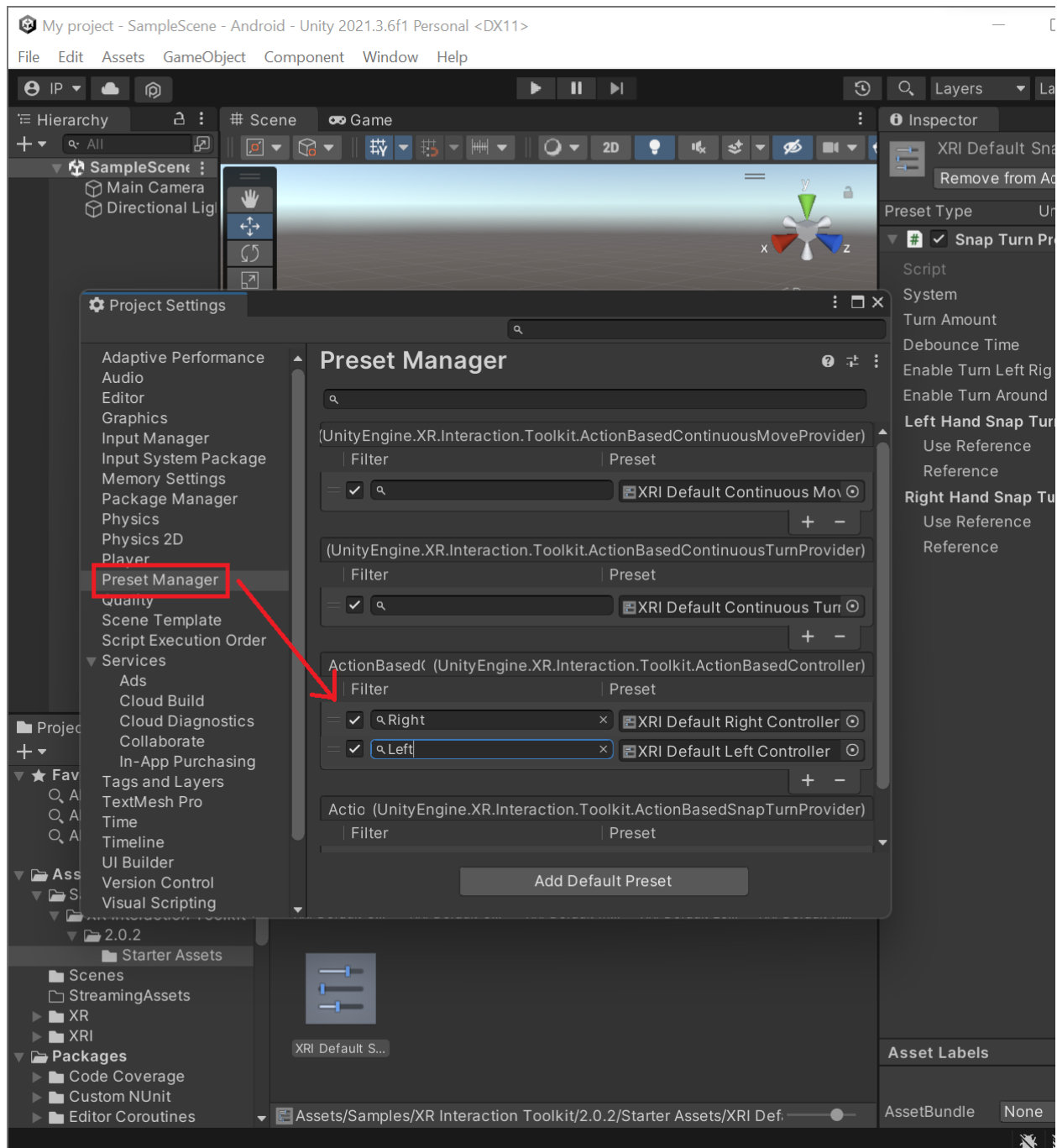


25. Now from here you will go to the top of the Unity screen and click on the “Edit” tab. In the drop down go ahead and click on “Project Settings”.



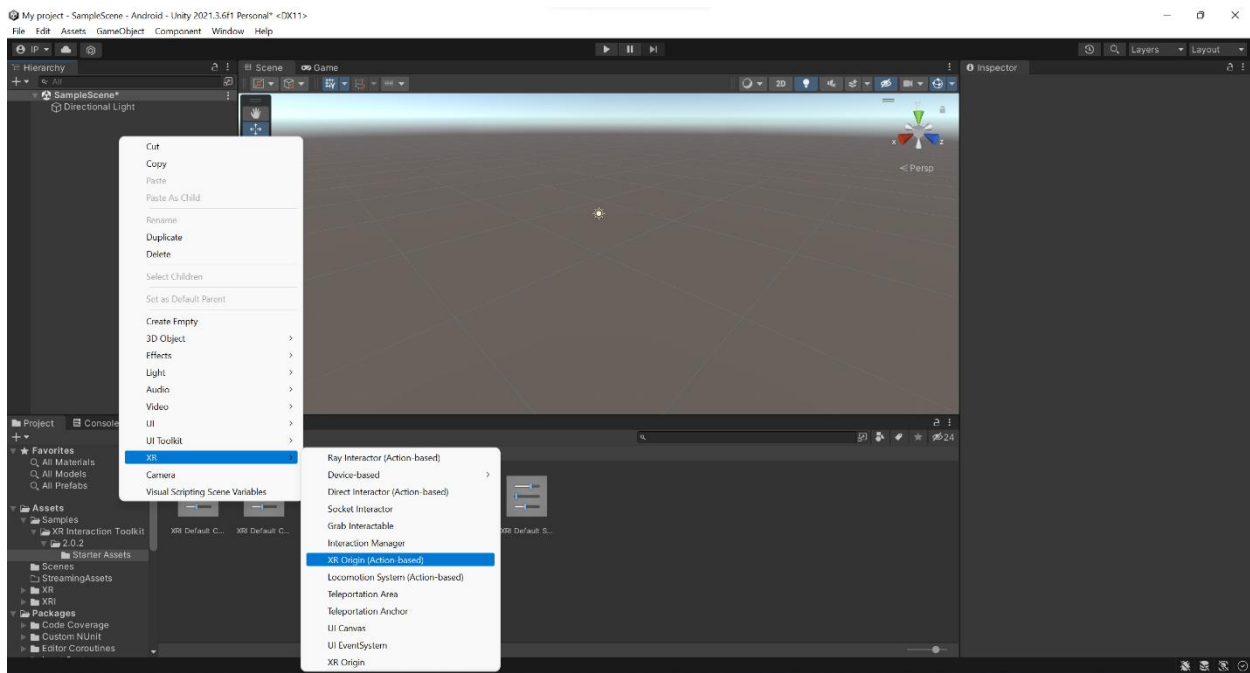
26. After clicking the project settings button, a pop up window will appear, from here you will go to the left of the pop up and go down to and click on “Preset Manager”.

27. After that look for the Action based option that deals with the left and right controllers. In the boxes under Filter, type “Right” and “Left” respectively in each box.

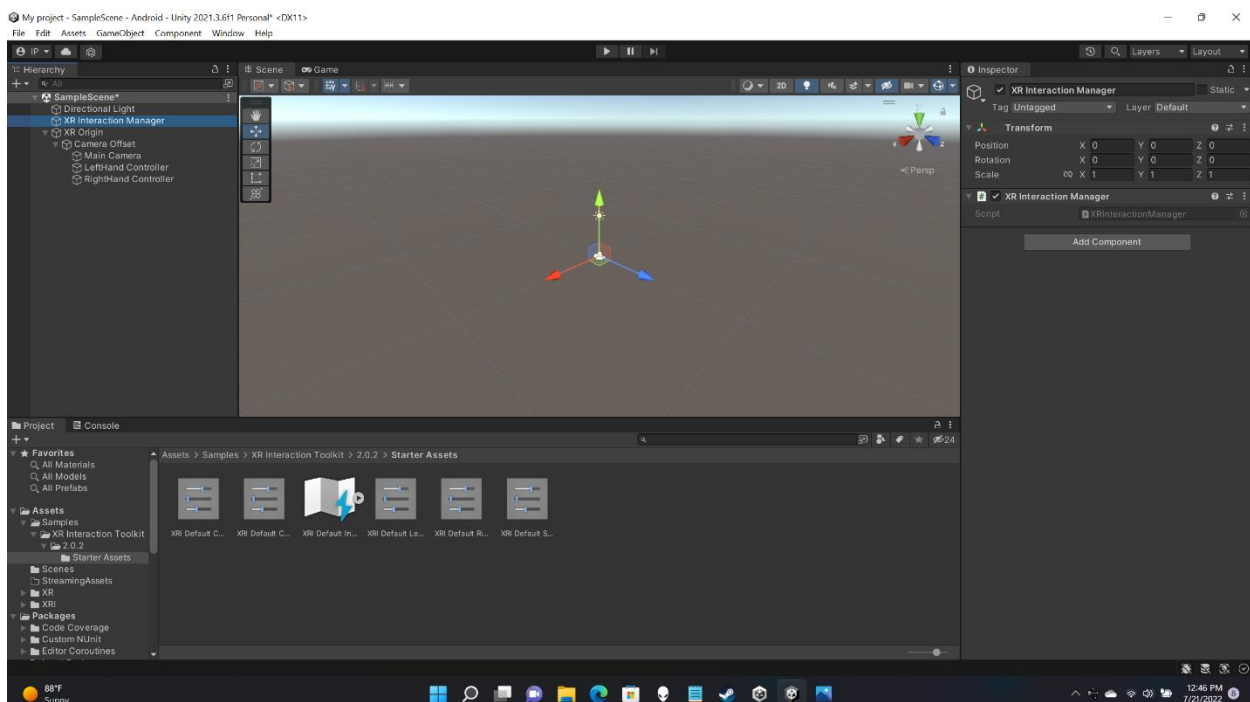


At this point you've set up the environment, this following part will be on the basic rigging required for the oculus to work

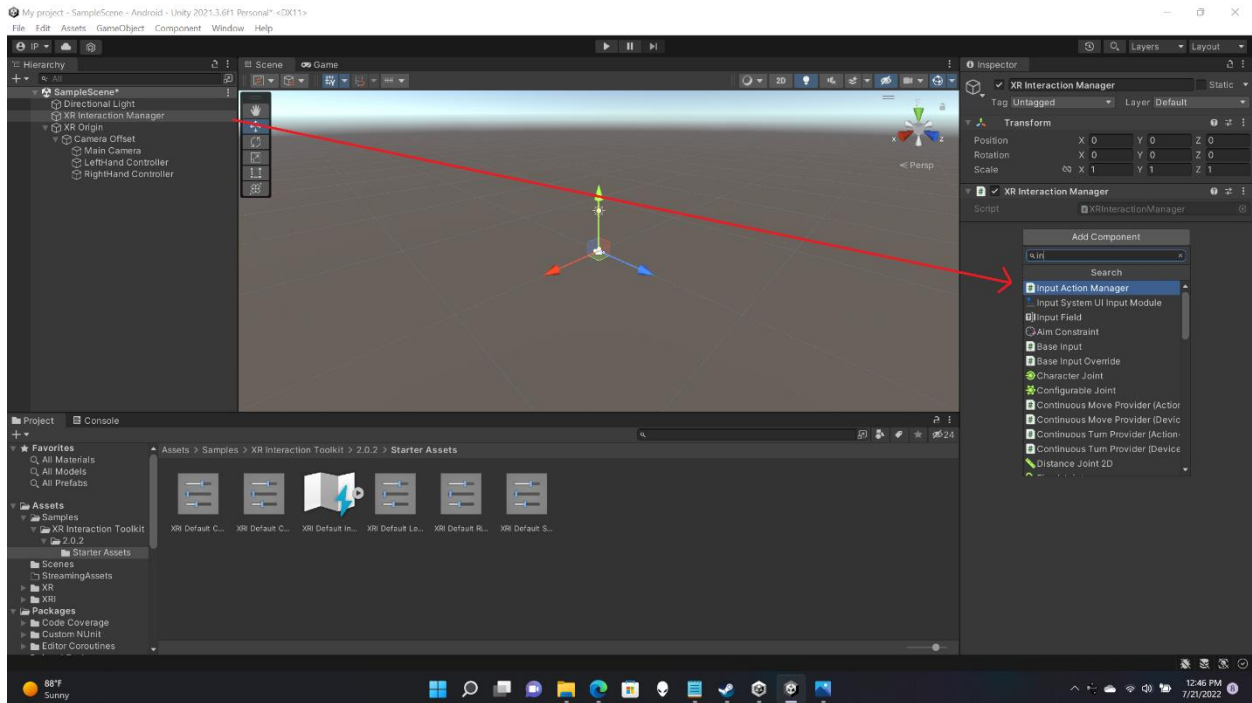
1. If you right click on the Hierarchy like you would to create any object in Unity you'll notice a new option, "XR", go through that menu and click the option "XR Origin (Action-based)"



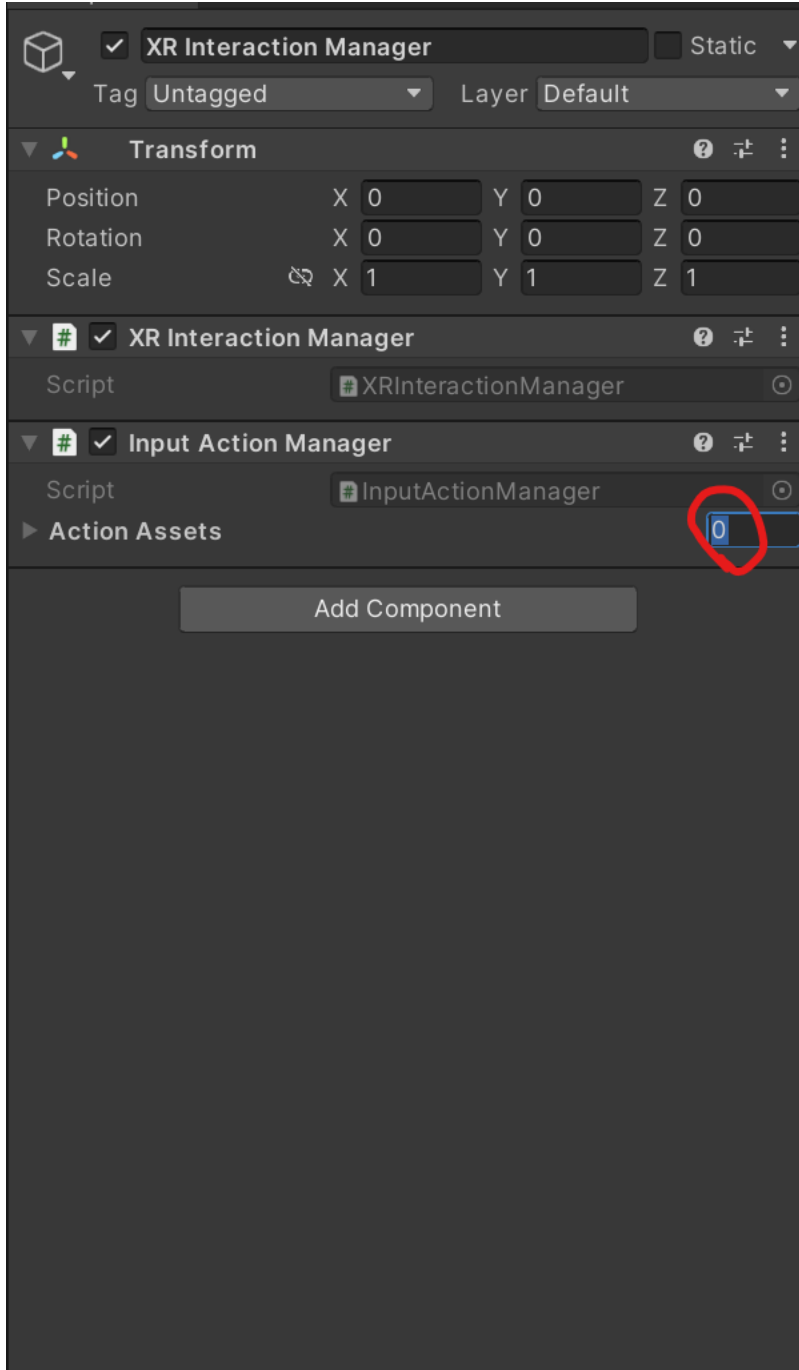
2. Two things should appear, XR Interaction Manager and XR Origin



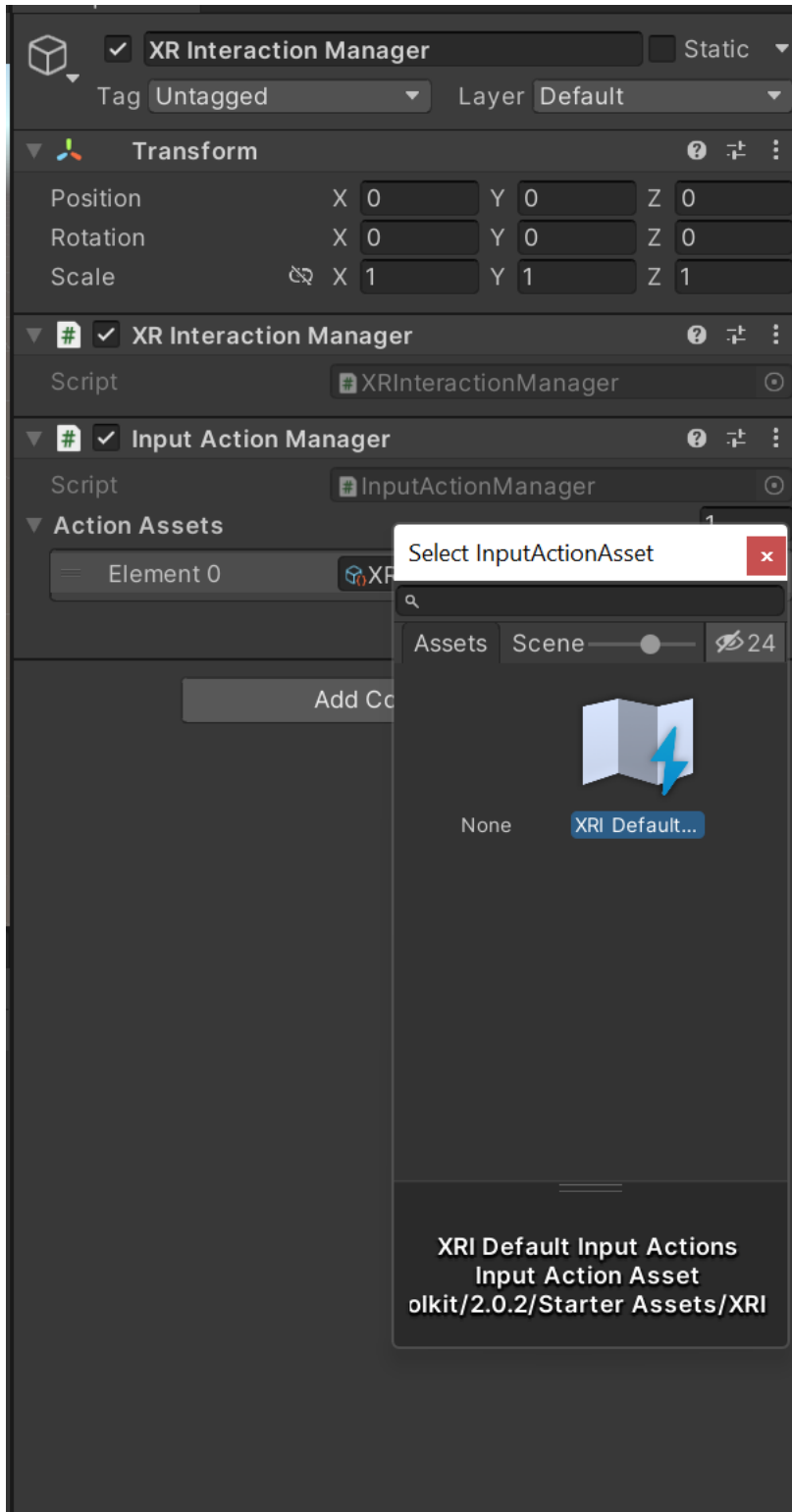
3. Go ahead and click on the “XR Interaction Manager” Game object that is located in the Hierarchy tab. Then go to the inspector window of this game object and click on the button “Add Component”. Type InputActionManager and add that component.



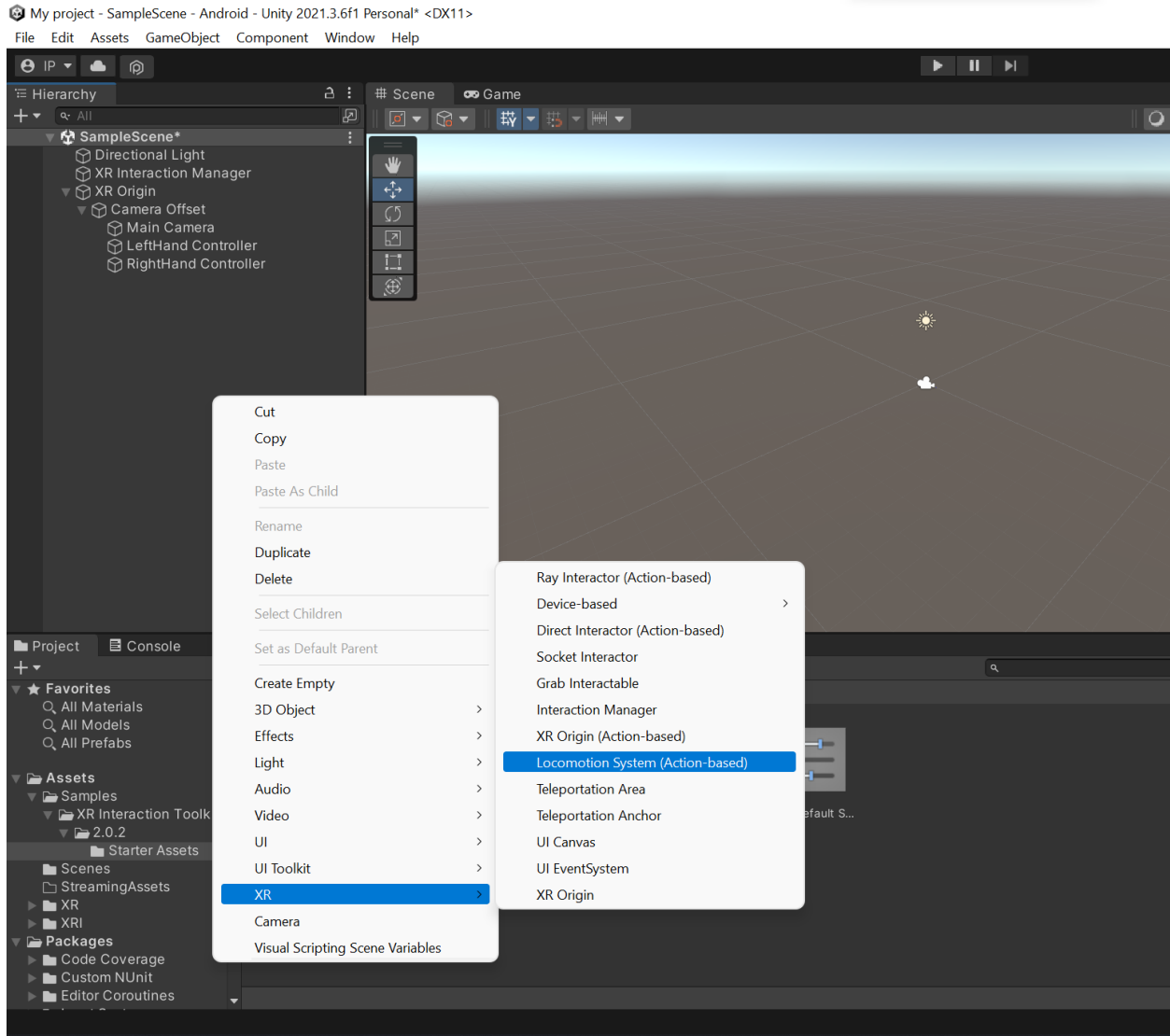
4. Look for the Action Assets options and change that “0” to a “1”



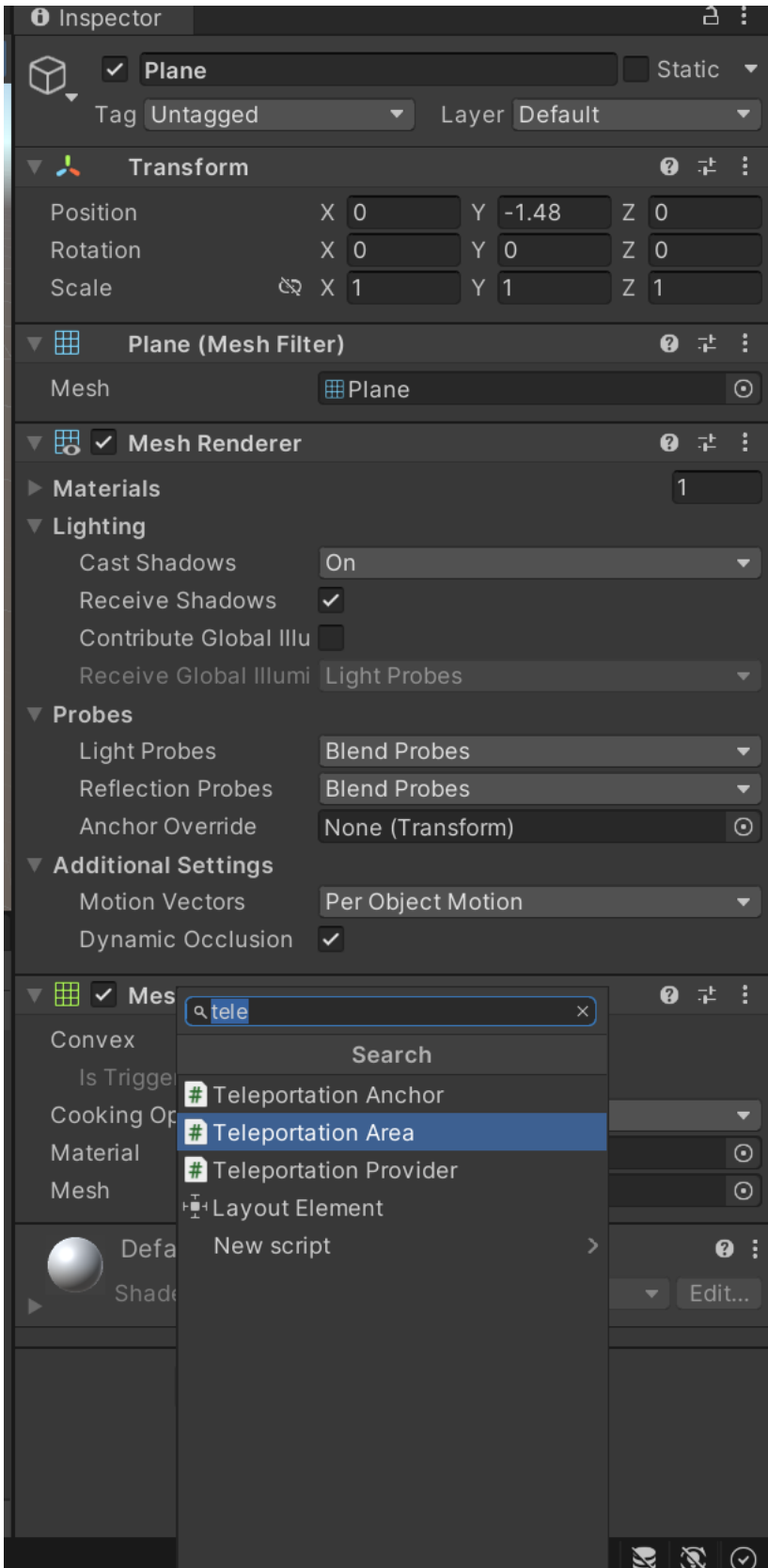
- Click the arrow next to Action Assets and you should see an option that says Element 0, click on that option and choose the option that says XRI Default Input Actions.



6. Move the mouse to the Hierarchy tab, right click, move the mouse to XR and in the pop up click on “Locomotion System (Action-Based)”.



7. After this step we click on any game object you intend to make a floor and on its inspector window go ahead and add the following component “Teleportation Area (Script)”. In this case it’s a plane but it could be any 3d object you create in Unity’s Hierarchy.



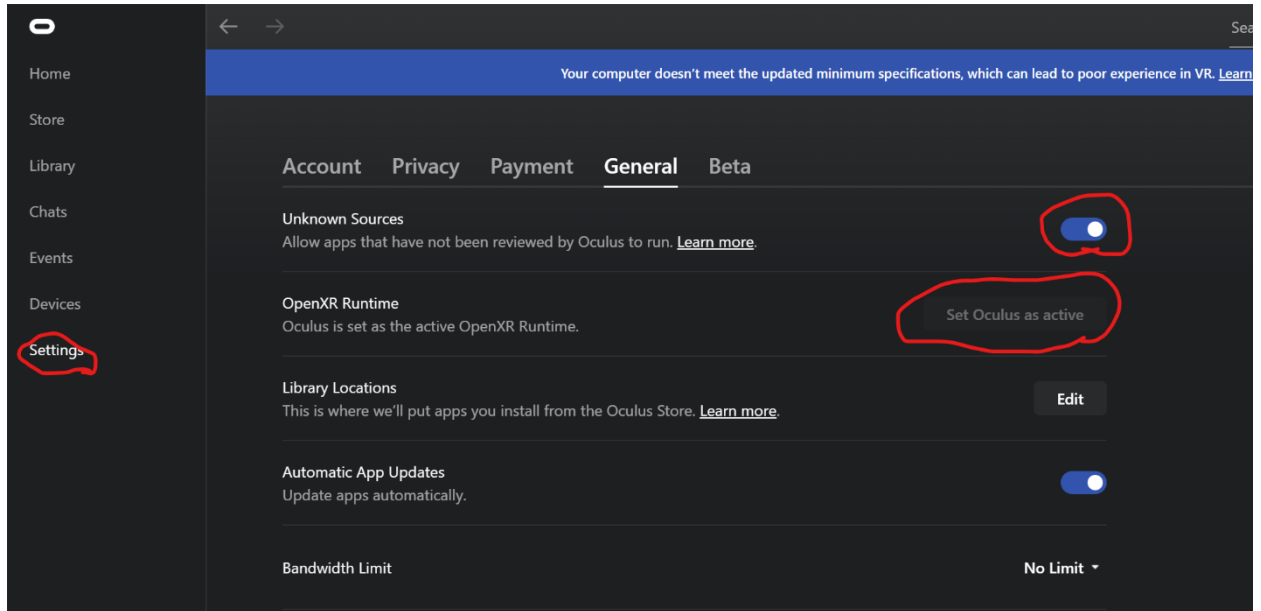
At this point you're done! You should be able to press play in the Unity engine and provided you've opened the oculus app and have a Quest plugged in; the Quest should pick it up. You should be able to look around and teleport on your 3d object with the face buttons.

Troubleshooting:

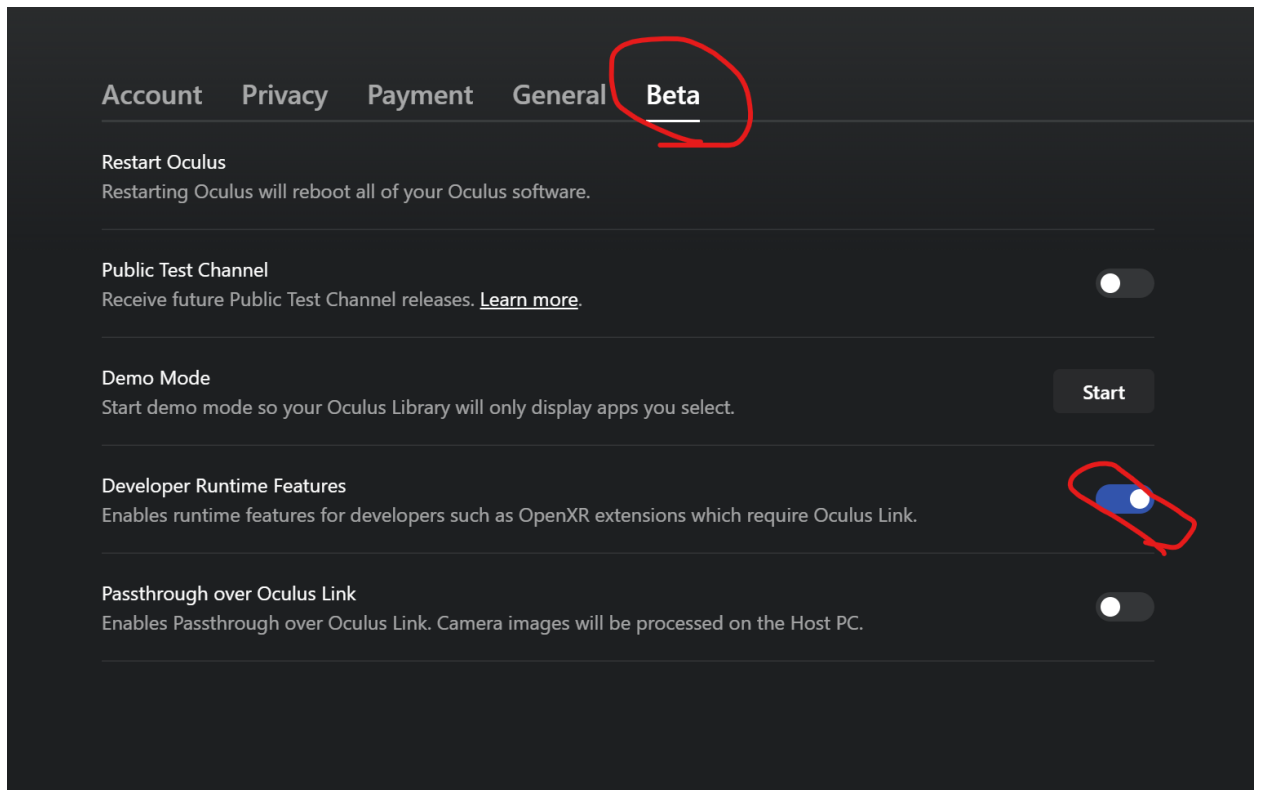
If unity is working fine, but you're having trouble getting anything to appear on your headset, take the following steps

- Ensure the Oculus App is open
- Ensure that the headset is in desktop mode by looking inside. If you're in a white environment with a grid on the floor, you're in desktop mode
 - If you're not in desktop mode, go to settings and click the big option to right of the Guardian option, and then press launch
 - If you do not see the option, it means your headset is not properly linked to your computer, please call an instructor over.

- Open the Oculus App, go to Settings -> General, and click the button that says, “Set Oculus as active” and ensure that Unknown sources is toggled on.



- Then click on the Beta option and check if Developer Runtime is toggled on.



- If the device is still not working, it could be that your headset is not in developer mode. Ask an instructor to help you with that.