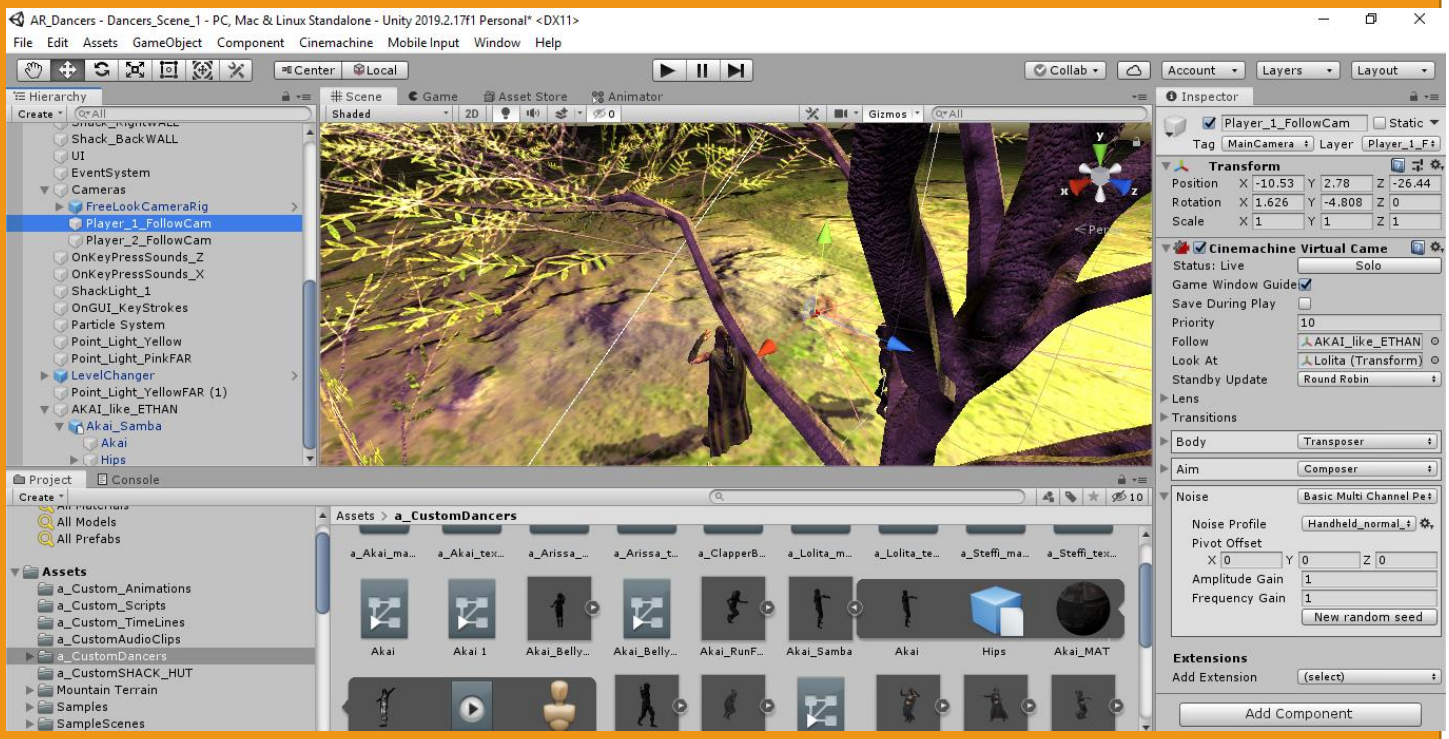
Absolute Beginners Unity Book

Chapter – II



GitHub repository with the **C# scripts** and the **.asmdef files** , there will be files which shall be extras and may not be required to follow along with the text in this PDF file.

https://github.com/RohitDhankar/Unity3D_2020_UnrealGamelabs_Dancers

For additional Notes refer Blog - https://datasciencewithrandpython.blogspot.com/search/label/Unity

GitHub Repository - https://github.com/RohitDhankar/Unity3D_2020_UnrealGamelabs_Dancers
Unity Blog Posts - https://datasciencewithrandpython.blogspot.com/search/label/Unity3D

This is the rough first cut of the Second Chapter of the – **Absolute Beginners Unity Book** , this is being written in near realtime by **Rohit Dhankar** (independent / self employed Unity game developer / animator / designer – working under the title of **Unreal Game Labs**) .

Social Media Contact details –

LinkedIn – https://www.linkedin.com/in/rohitdhankar/

GitHub - https://github.com/RohitDhankar

YouTube – https://www.youtube.com/channel/UC9J9N9CNv15s9U9Aejpza6g/videos

https://www.youtube.com/watch?v=3Rm0TlBx1O4&list=PLLpHcww3qjp7n6i2DznVnbDEmP9OENBqc

FaceBook - https://www.facebook.com/AugmentedReality-110417607128839

Its to be considered a Log of the , hidden quips that have been encountered during creating the very first basic games , game plays and animations with Unity2D and Unity3D.

This content as on date is totally raw and un-edited , shoddy to a great extent – its being put to help anyone who may need it .

There are lots of external to official Unity tutorials being referred , all references as on date are inline within the main text .

Content Disclaimer :-

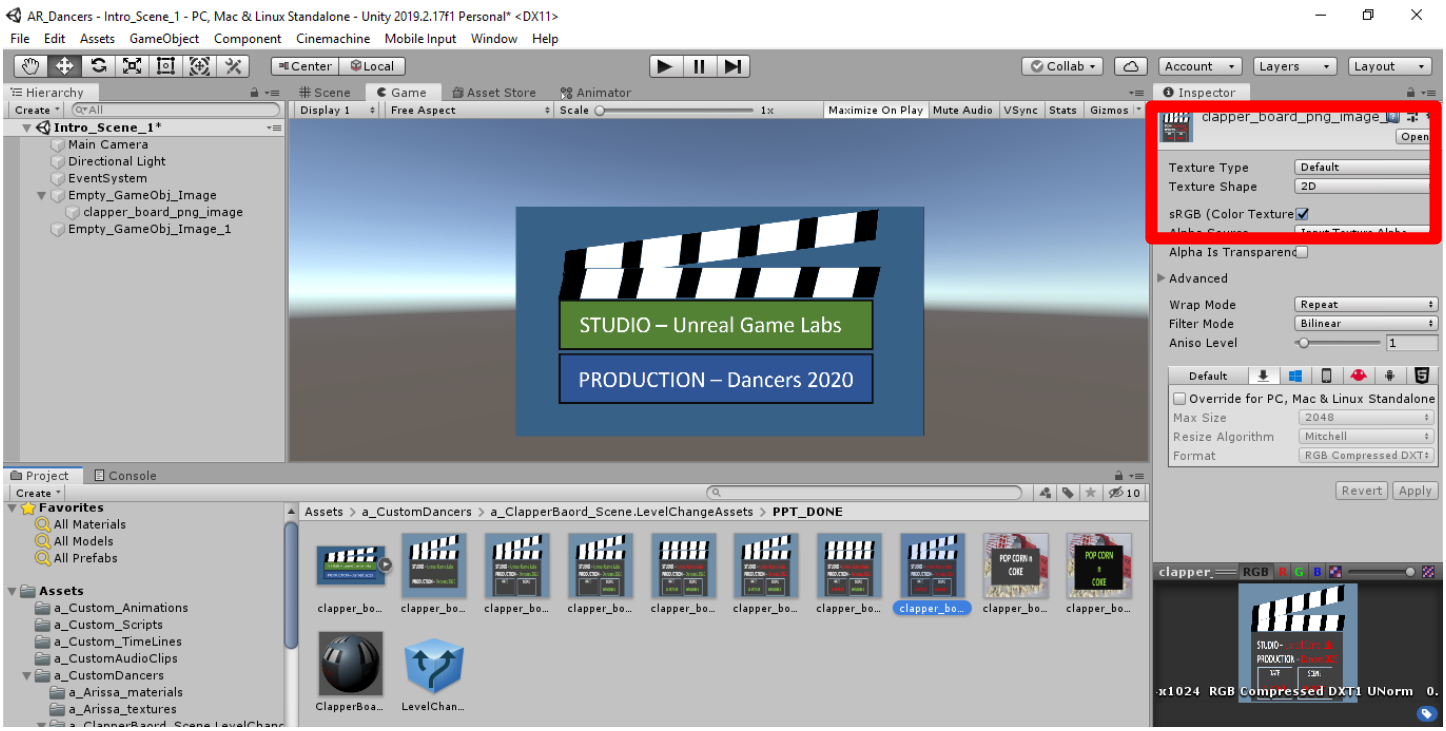
Under no circumstances is Rohit Dhankar liable in any way for any Content, including, but not limited to: any infringing Content, any errors or omissions in Content, or for any loss or damage of any kind incurred as a result of the use of any Content posted, transmitted, linked from, or otherwise accessible through or made available via this book.

GitHub Repository - https://github.com/RohitDhankar/Unity3D_2020_UnrealGamelabs_Dancers
Unity Blog Posts - https://datasciencewithrandpython.blogspot.com/search/label/Unity3D

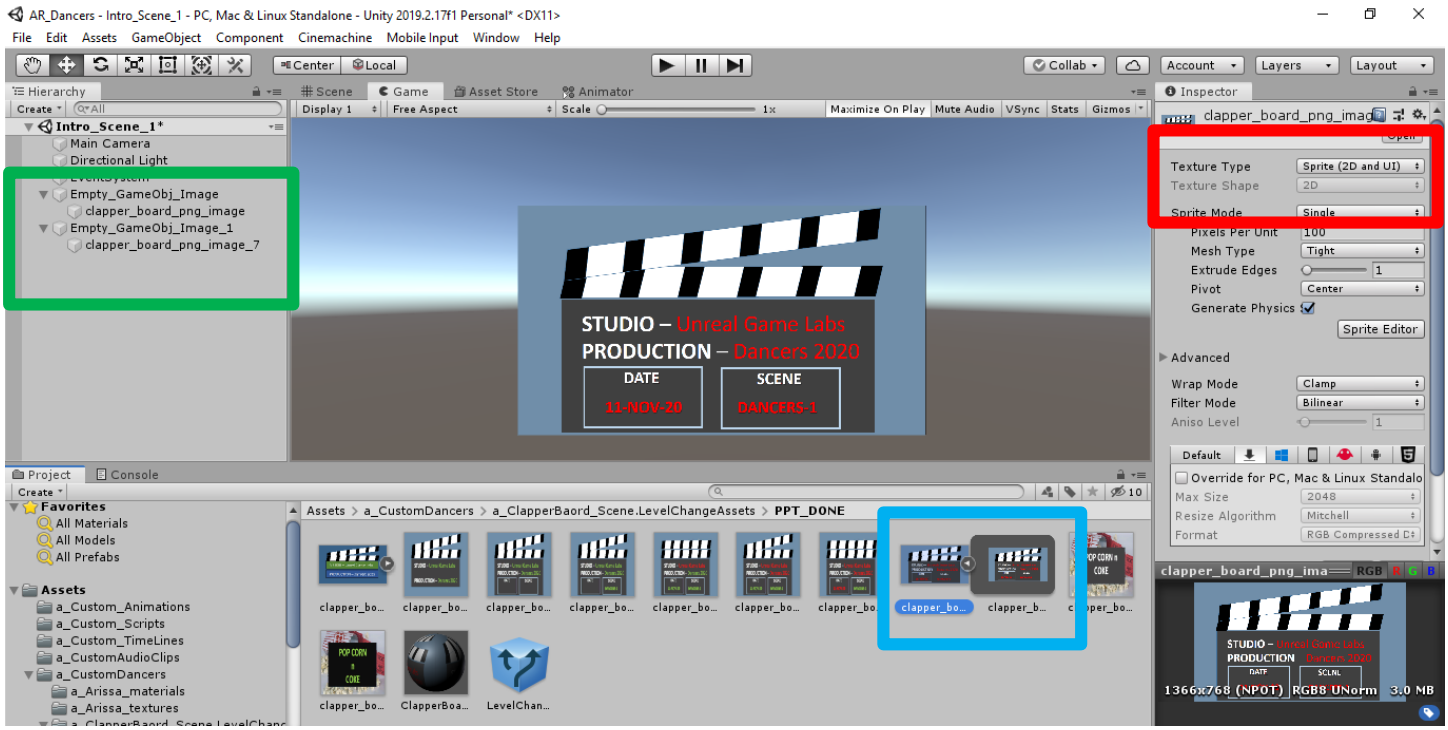
Unity 2019 – Adding a PNG Image as a Sprite

Specific to 2D images being added as Sprites within the Unity 2019 editor .

a. Add the PNG image to the Assets or any Sub DIR of assets .
b. Select image , in the INSPECTOR change the TEXTURE TYPE – from Default to Sprite 2D and UI
c. Drag image to below the Empty Game Object created earlier , image shall now appear within the Scene.



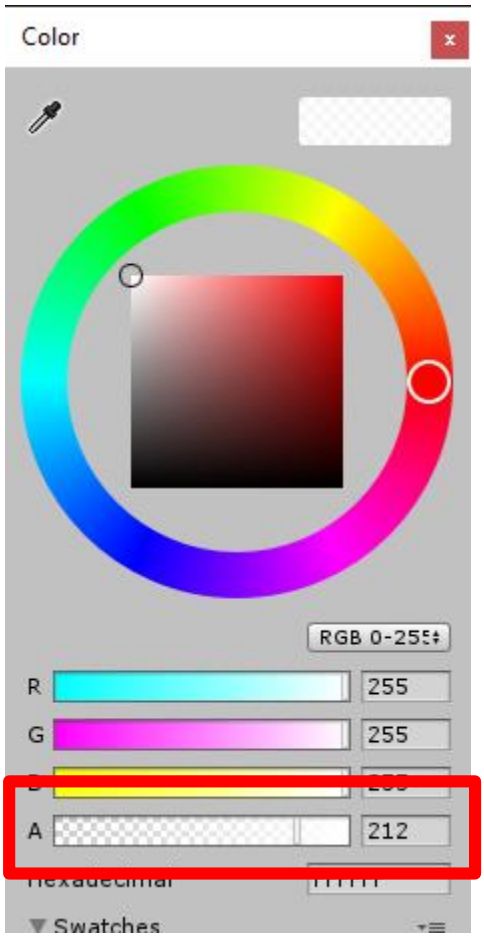
GitHub Repository - https://github.com/RohitDhankar/Unity3D_2020_UnrealGamelabs_Dancers
Unity Blog Posts - https://datasciencewithrandpython.blogspot.com/search/label/Unity3D

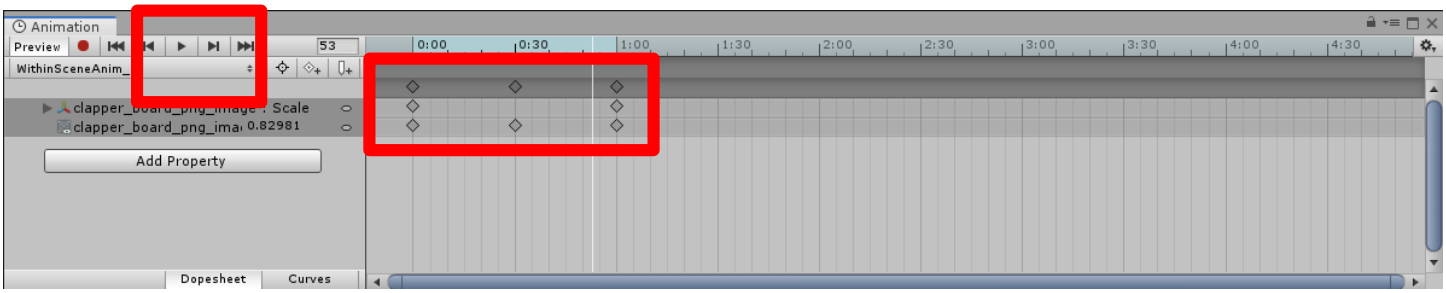


Choose the Image in the Inspector >> Color >> Change the ALPHA channel to make Image Transparent at the beginning of the Animation .

Change Alpha slider as required at various stages of the animation.

GitHub Repository - https://github.com/RohitDhankar/Unity3D_2020_UnrealGamelabs_Dancers
Unity Blog Posts - https://datasciencewithrandpython.blogspot.com/search/label/Unity3D

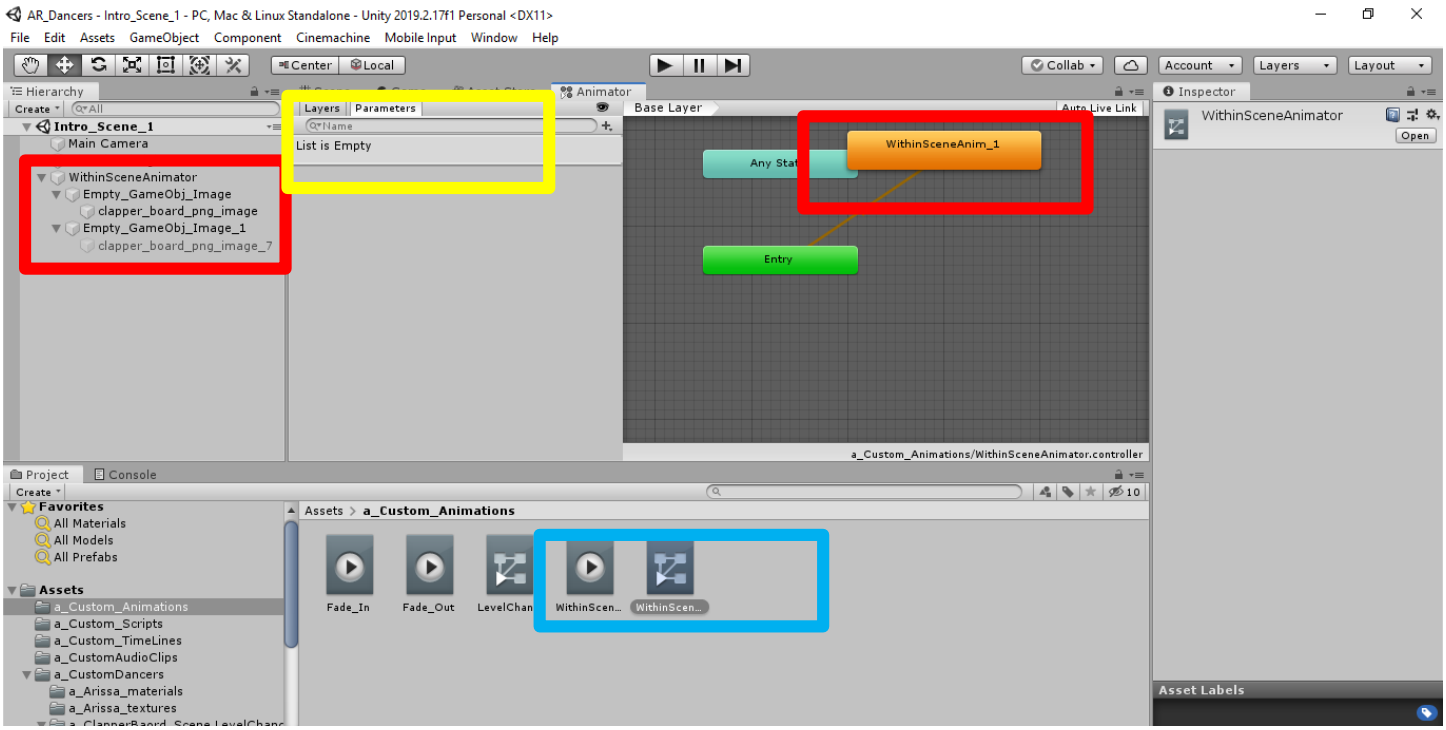




As seen above – we have added 2-3 KEYS to the Animation as seen in the DopeSheet above . Can check the animations now by clicking play .

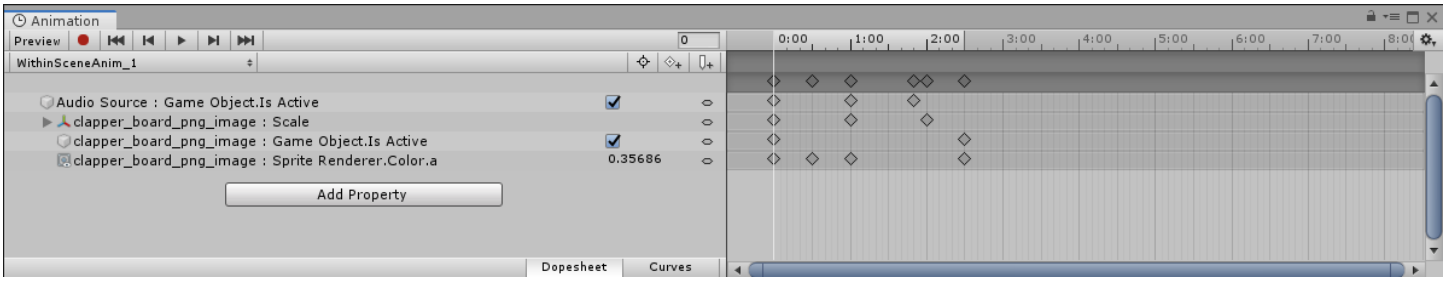
The animation thus created is saved within the **Custom_Animations DIR** created earlier, its named - **WithinSceneAnimator.anim** and has a **WithinSceneAnimator.controller ,** both seen below in the **BLUE RECT**. The Controller is as seen below in the **RED RECT**

GitHub Repository - https://github.com/RohitDhankar/Unity3D_2020_UnrealGamelabs_Dancers
Unity Blog Posts - https://datasciencewithrandpython.blogspot.com/search/label/Unity3D



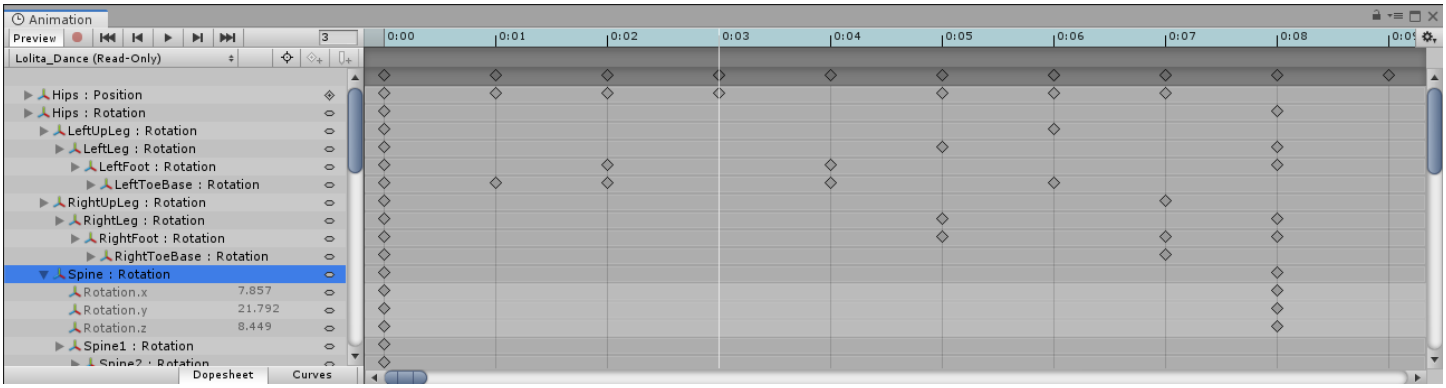
As seen above the ANIMATOR / CONTROLLER – has no PARMETERS yet – the List of PARAMETERS is EMPTY , highlighted above in the **YELLOW RECT**.

Use the **MOUSE SCROLL WHEEL** to reduce / increase the **DopeSheet TimeLine duration** , from within the Seconds view to the Minutes view .



GitHub Repository - https://github.com/RohitDhankar/Unity3D_2020_UnrealGamelabs_Dancers
Unity Blog Posts - https://datasciencewithrandpython.blogspot.com/search/label/Unity3D

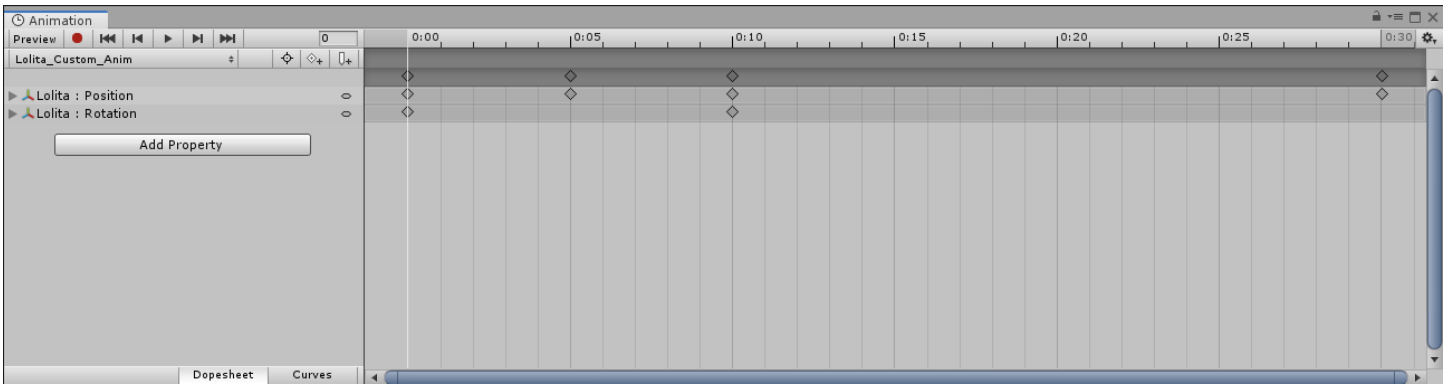
Looking through the Dope Sheet for Mixamo Models



The mixamo models have very many parts of the skeleton animated as is apparent from the dope sheet .

GitHub Repository - https://github.com/RohitDhankar/Unity3D_2020_UnrealGamelabs_Dancers
Unity Blog Posts - https://datasciencewithrandpython.blogspot.com/search/label/Unity3D

Add an Animation to the MIXAMO.com Models



As seen above have added an additional Animation – named it , **Lolita_Custom_Anim , this is very basic in comparison to the Mixamo animations .**

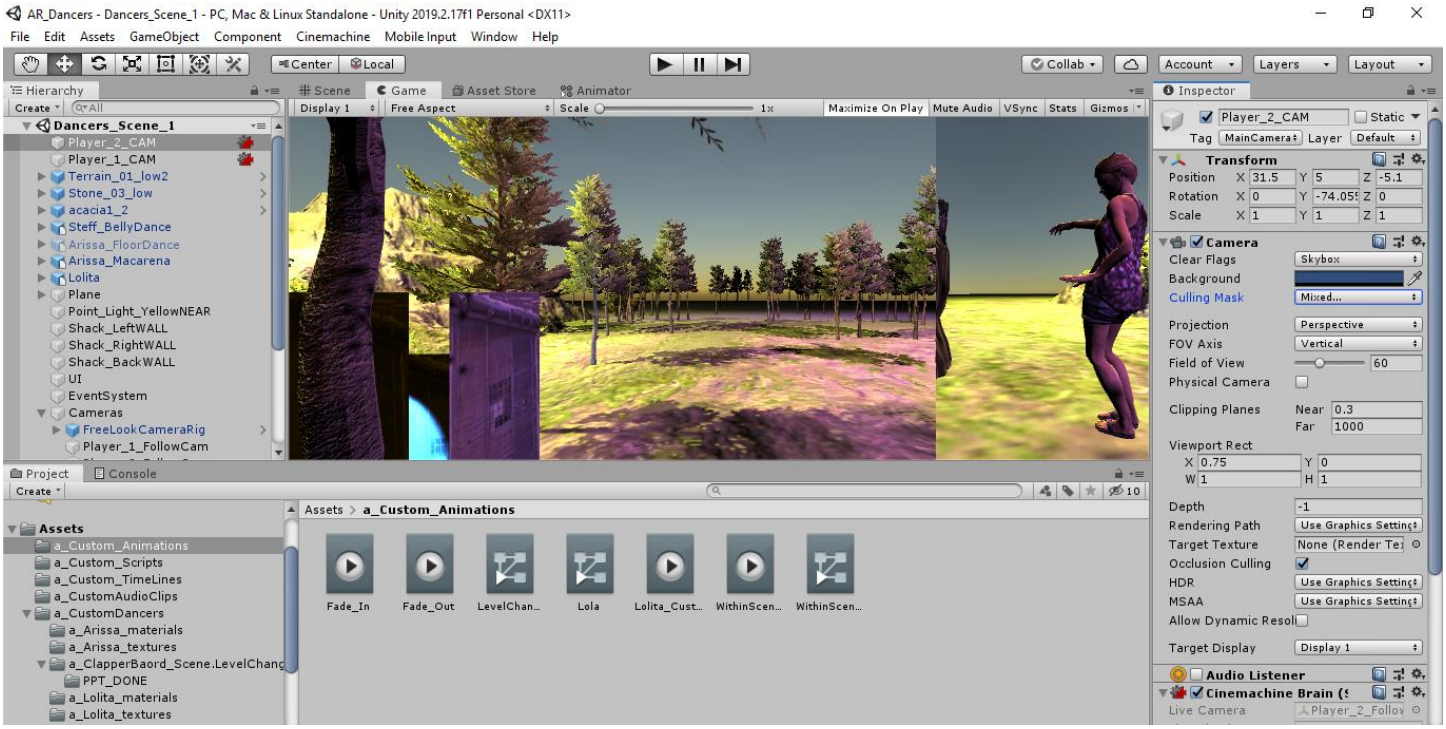
GitHub Repository - https://github.com/RohitDhankar/Unity3D_2020_UnrealGamelabs_Dancers
Unity Blog Posts - https://datasciencewithrandpython.blogspot.com/search/label/Unity3D

Revisit the Follower CAMS

This portion involves adding Virtual cameras to the scene in addition to the Main Camera and managing the Three cameras using the CineMachine Brain. Splitting the screen to view two Virtual Follower cameras, is to be considered optional as it may or may not be required by your gameplay.

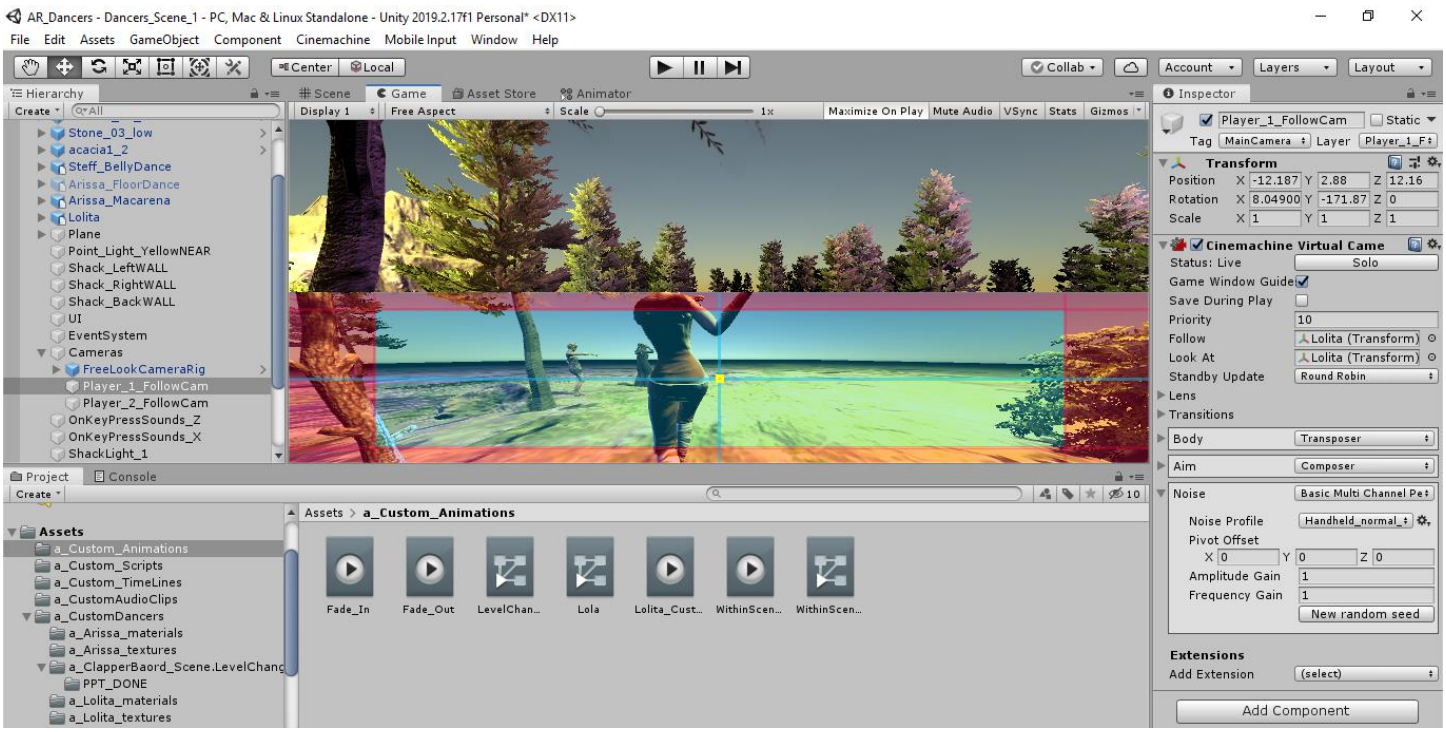
Primary Source – Unity Official Video = https://www.youtube.com/watch?time_continue=38&v=X6Ou8InMUbg

a) To the **Main Camera** – Add Component >> Cinemachine Brain
b) Rename MAIN CAMERA = Player Cam_1 , duplicate this Player Cam_1 and rename it **Player Cam_2**
c) Create Two **Virtual Cameras** , rename them **- Player_1_FollowCam** and **Player_2_FollowCam**
d) Drag and Assign the MAXIMO 3D Model – Lolita , to the - **Player_1_FollowCam ,** by dropping into the text box like space within **- CimeMachine VirtualCam >> Follow** and **CimeMachine VirtualCam >> Lookat.**
e) Exclude the Camera-1 from the Camera-2's CULLING MASK – by unchecking the - Camera-2 from within the Culling Mask Drop Down.
f)



GitHub Repository - https://github.com/RohitDhankar/Unity3D_2020_UnrealGamelabs_Dancers
Unity Blog Posts - https://datasciencewithrandpython.blogspot.com/search/label/Unity3D

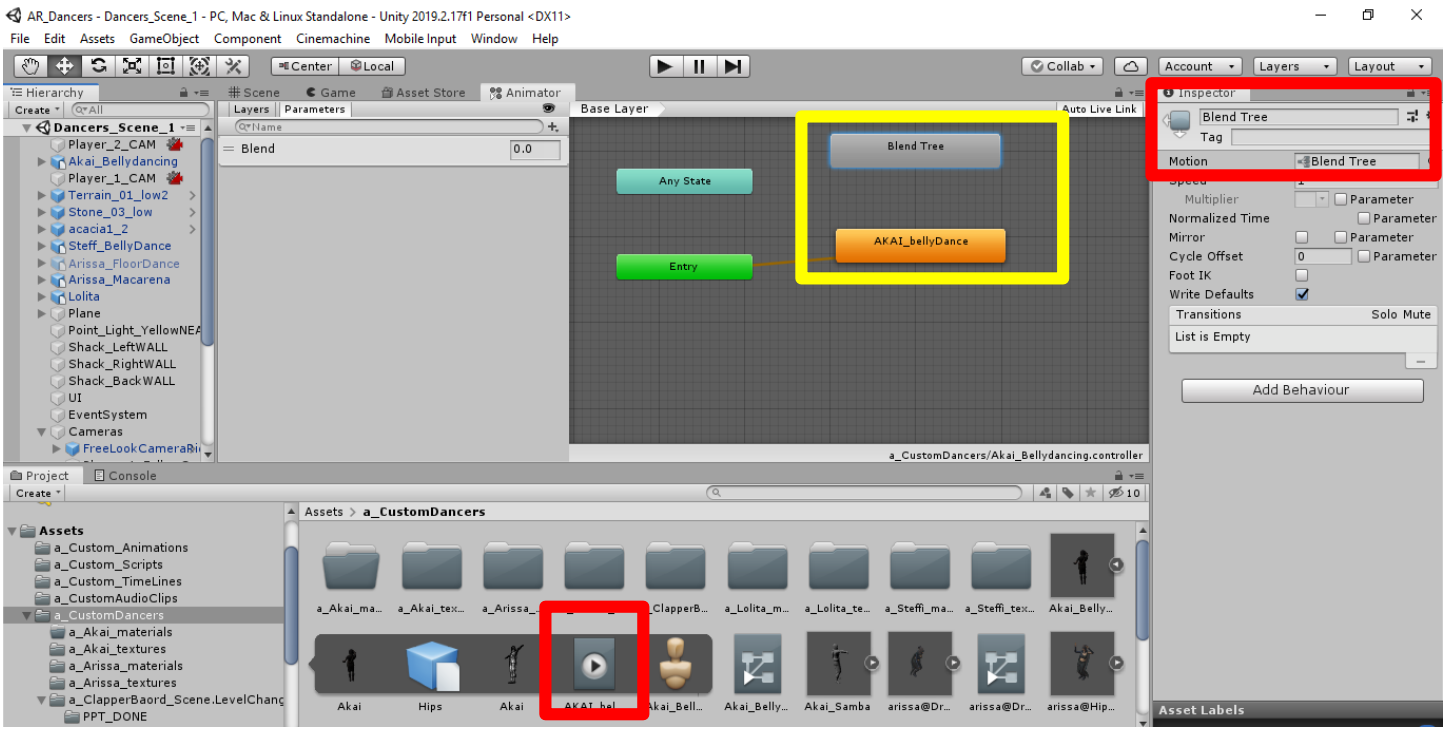
Seen below the - **Player_1_FollowCam**



GitHub Repository - https://github.com/RohitDhankar/Unity3D_2020_UnrealGamelabs_Dancers
Unity Blog Posts - https://datasciencewithrandpython.blogspot.com/search/label/Unity3D

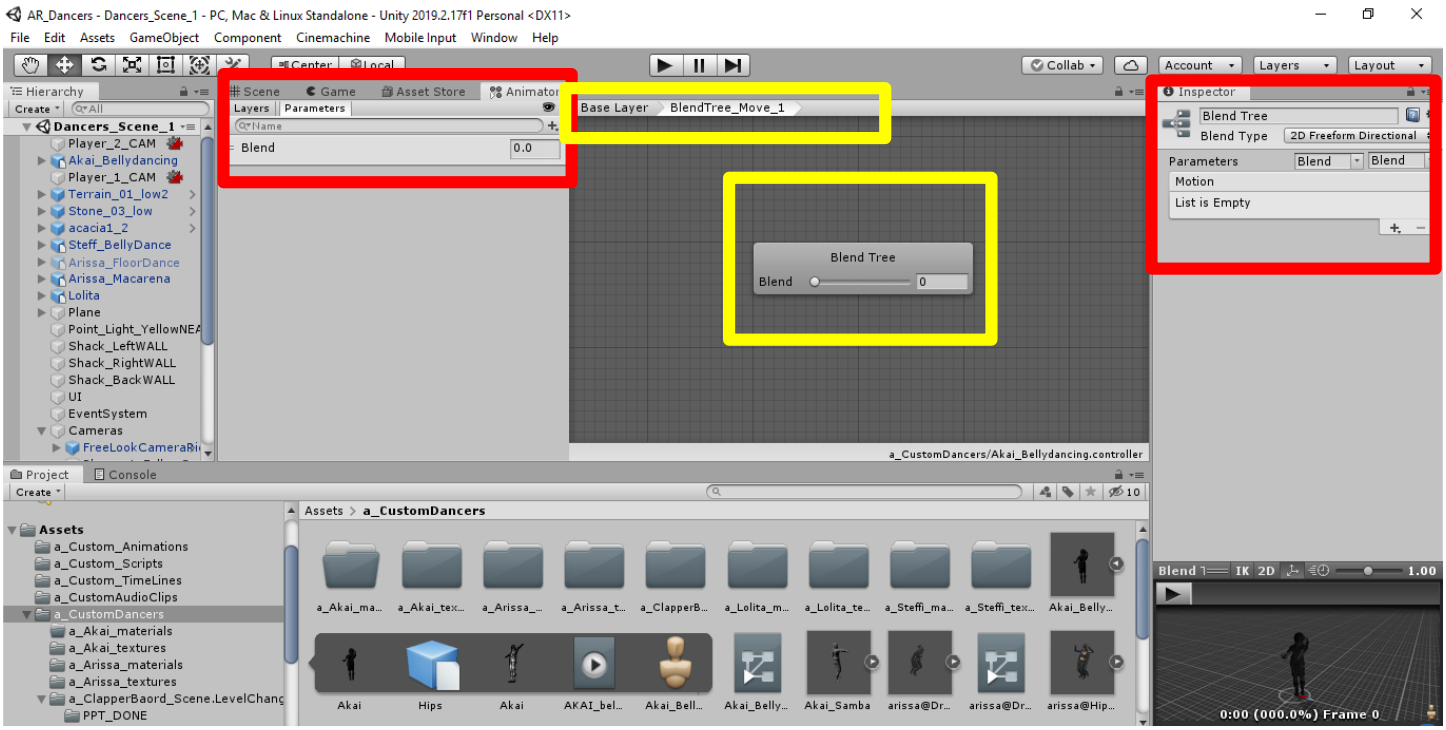
Add BlendTree Animations to the MIXAMO.com Models

Primary Source Unity Official tutorials and - https://www.youtube.com/watch?v=YgaLKrSApWM

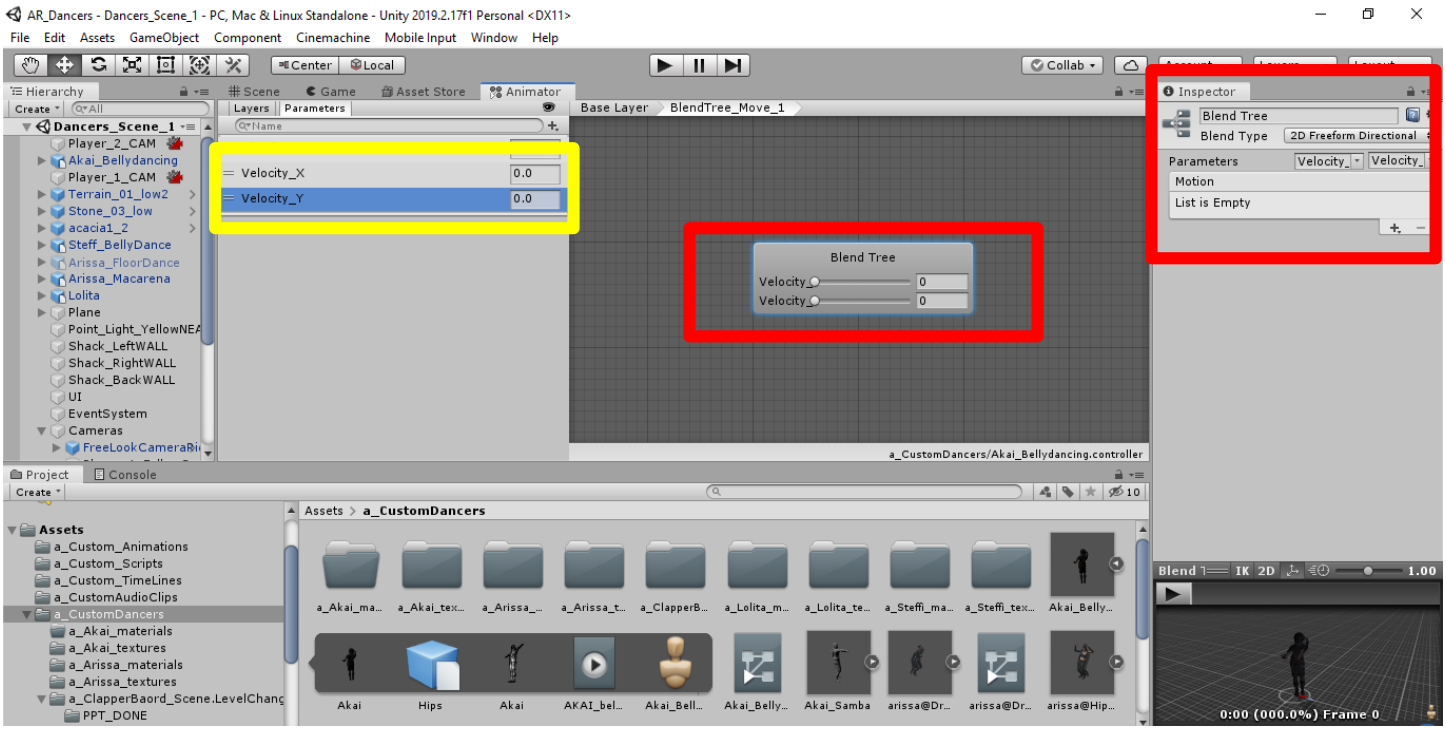


a) As seen above – created a New BLEND TREE .
b) Double Clicked on the BLEND TREE.
c) Choose from within INSPECTOR – **Blend Types** >> **2D FreeForm Directional**
d) Top Left corner within theANIMATOR TAB – Create a New PARAMETER .
e) Choose Parameters Type = Float
f) Create Two Such Floats – **velocity_X** and **velocity_Y**

GitHub Repository - https://github.com/RohitDhankar/Unity3D_2020_UnrealGamelabs_Dancers
Unity Blog Posts - https://datasciencewithrandpython.blogspot.com/search/label/Unity3D

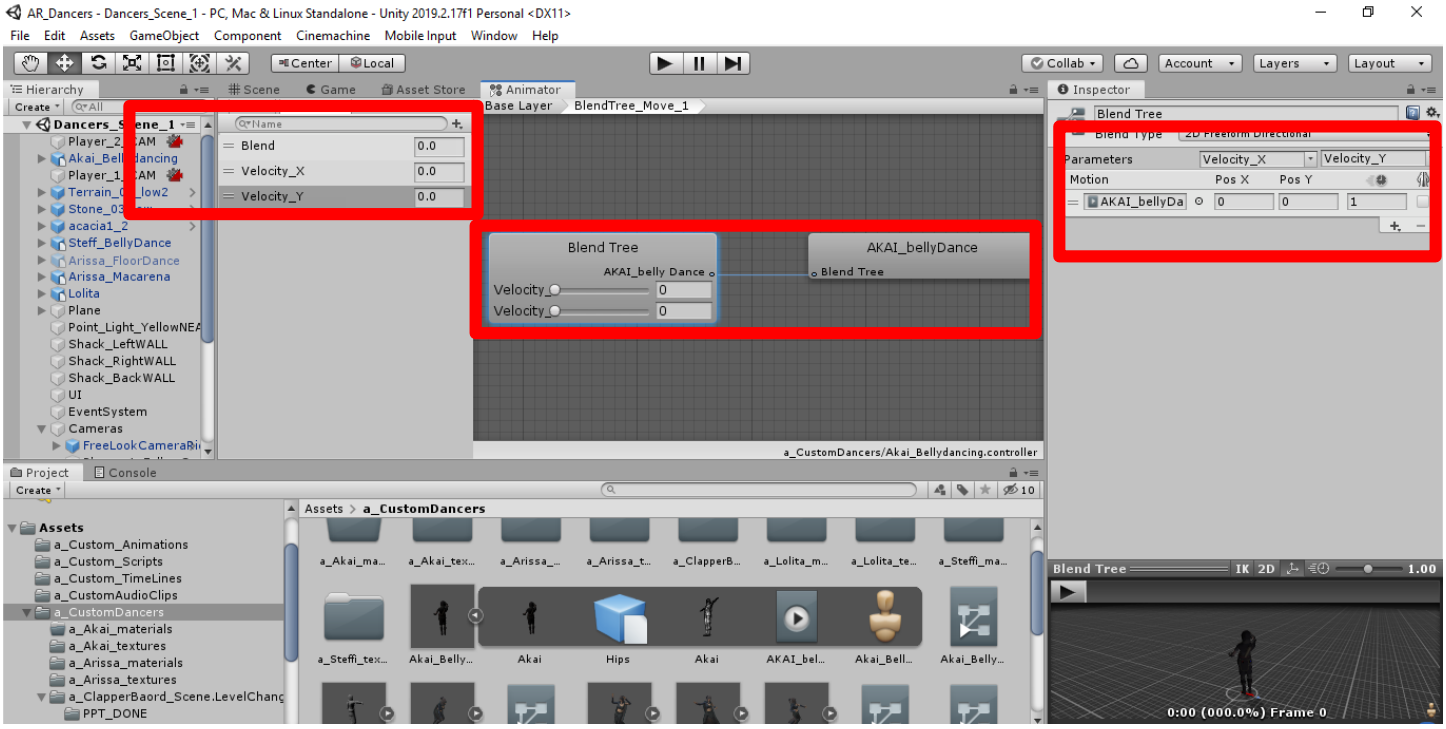


Go to Inspector – below Blend Type there are two Drop Downs – where we can choose the PARAMETERS we just created. Velocity_X and Velocity_Y.

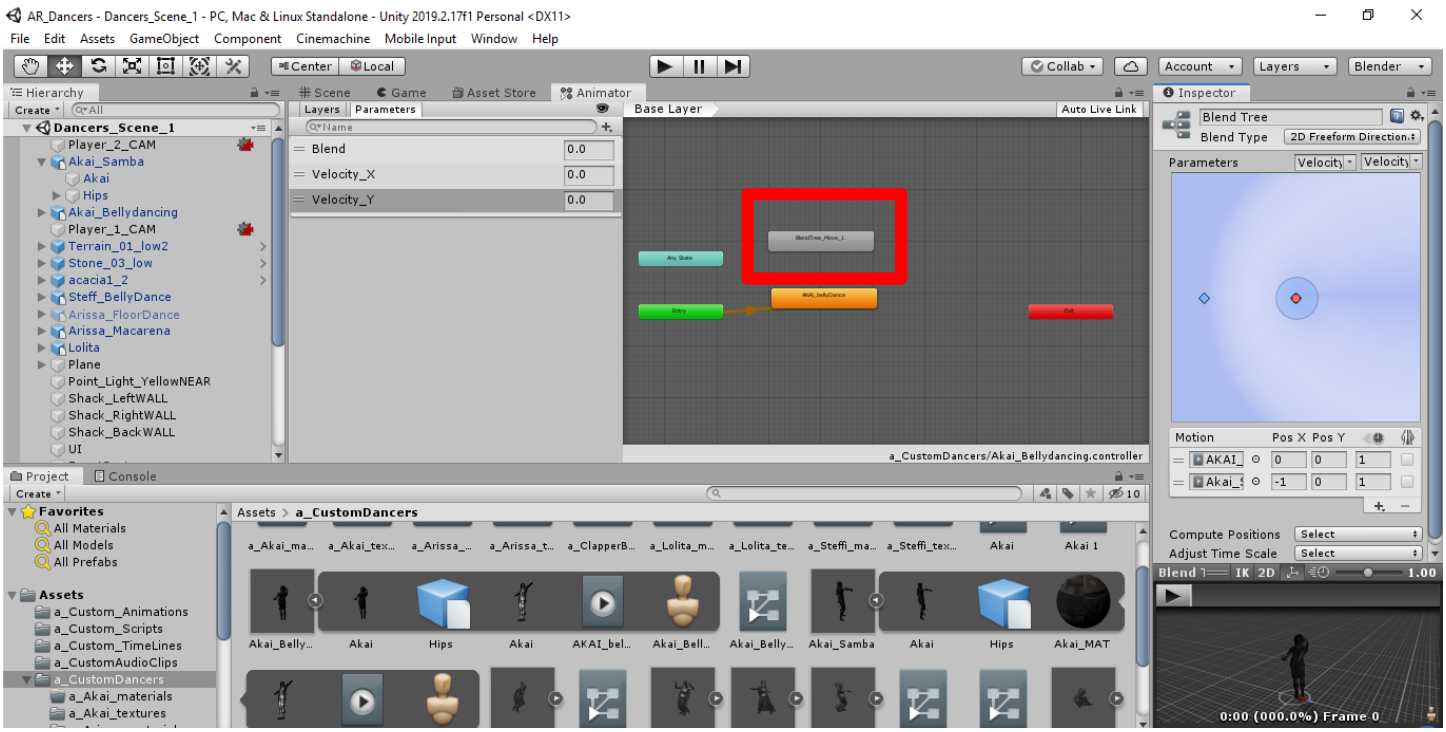


GitHub Repository - https://github.com/RohitDhankar/Unity3D_2020_UnrealGamelabs_Dancers
Unity Blog Posts - https://datasciencewithrandpython.blogspot.com/search/label/Unity3D

Under the PARAMETERS within the INSPECTOR for the BLEND TREE – we have an option to add a – MOTION , this Motion has a space for -



As seen below – the layout has gone out of Sync here .

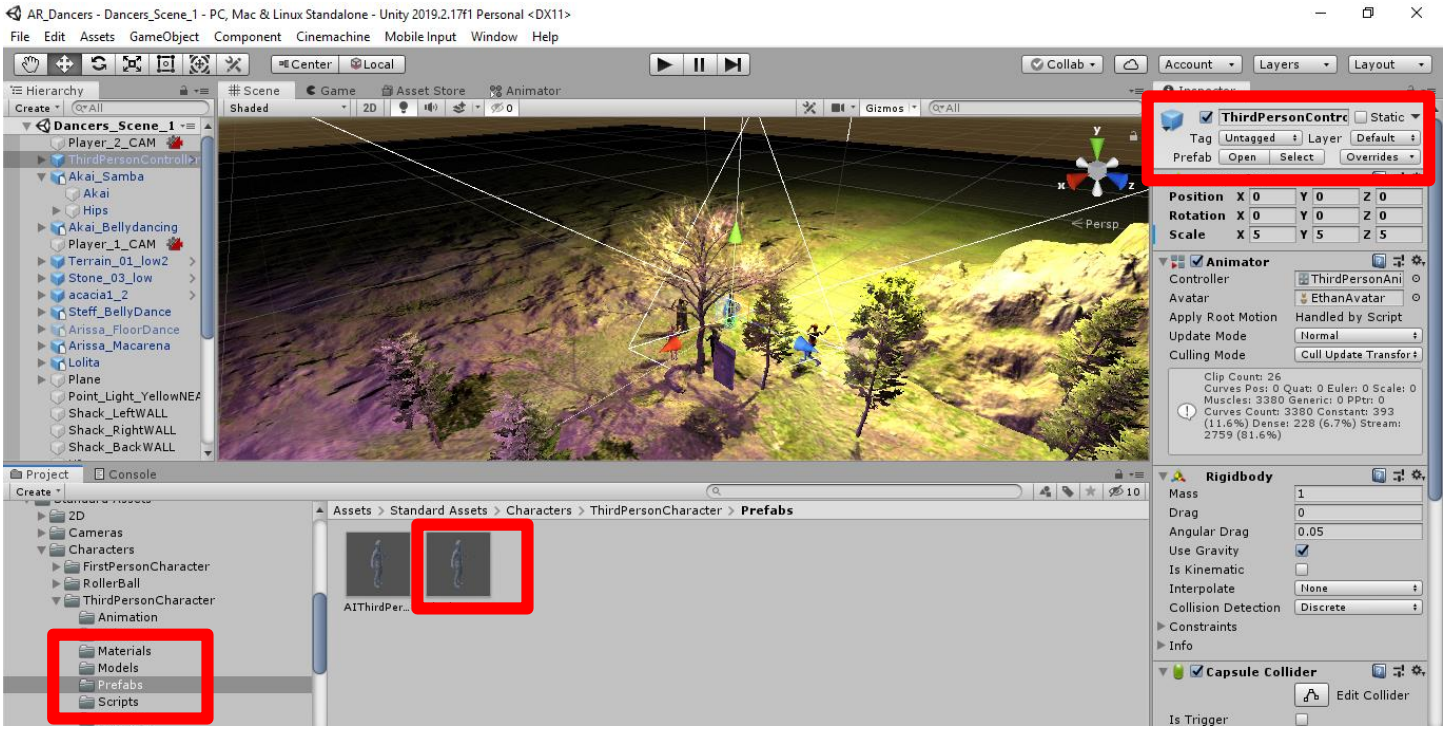


GitHub Repository - https://github.com/RohitDhankar/Unity3D_2020_UnrealGamelabs_Dancers
Unity Blog Posts - https://datasciencewithrandpython.blogspot.com/search/label/Unity3D

We can get back to the desired LAYOUT – by clicking on the BLENDTREE – Greyed Out Rectangle – which is highlighted above within the RED RECTANGLE highlighter.

Stealing Animations from the Standard Assets – ThirdPersonController

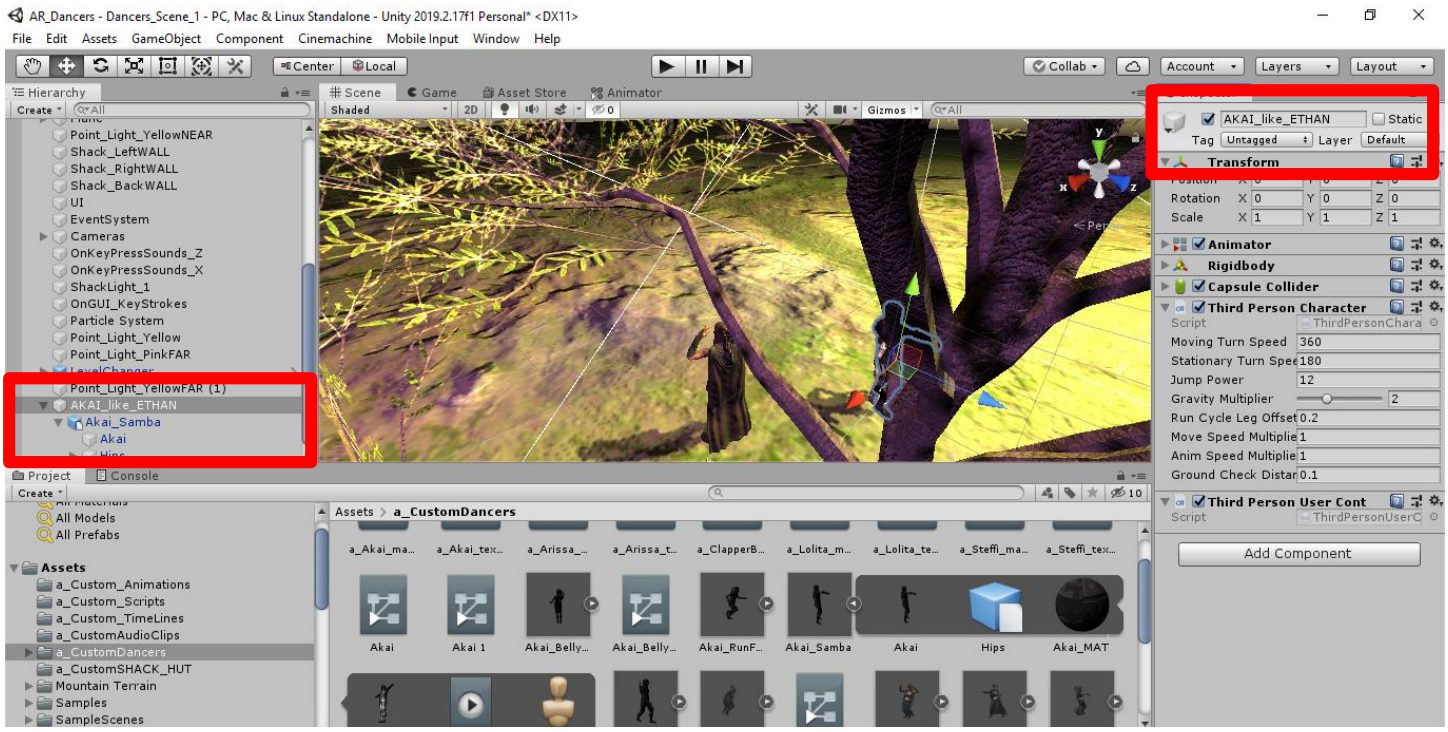
Source – **YouTuber - Aaron Hibberd** - https://www.youtube.com/watch?v=7NktwerZFro



…

GitHub Repository - https://github.com/RohitDhankar/Unity3D_2020_UnrealGamelabs_Dancers
Unity Blog Posts - https://datasciencewithrandpython.blogspot.com/search/label/Unity3D

Have copied the Component - **ThirdPersonUserControl** , which is a C# script - ThirdPersonUserControl.cs , which again in turn created the Component **ThirdPersonCharacter.cs** , but the values arent the same thus again went to Ethan and Copied the Values .



Created the follow cam earlier - now made the follow cam to follow the Ethan like ThirdPersonCharacter created, named - AKAI_like_ETHAN.

Another option for attaching a camera to the AKAI_like_ETHAN character is to get a - MultiPurposecameraRig from the StandardAssets , and making that Follow the - AKAI_like_ETHAN. This is as suggested by the **YouTuber - Aaron Hibberd** in this video - https://www.youtube.com/watch?v=7NktwerZFro

This should work just fine if we want to make it into a third person , scene. But im aiming at a camera , not really a FollowerCam , that can be moved around like a First Person , to create the feel of a First Person model walking through the level , with a Free/ MultiPurposecameraRig , in place of the persons own eyes view - thus obviously a FirstPerson view.

C# script that controlling movements of Ethan **ThirdPersonCharacter.cs** , is at , my local DIR Path = ~ Assets\Standard Assets\Characters\ThirdPersonCharacter\Scripts\ThirdPersonCharacter.cs

The C# script cloned from the script above - when copying the Component / C# Script - ThirdPersonUserControl.cs is a the path - Assets\Standard Assets\Characters\ThirdPersonCharacter\Scripts\ ThirdPersonUserControl.cs

Also , we copy the components off the ETHAN and into the AKAI_like_ETHAN , the scripts being used are the same scripts from the same local location within standard assets . Thus better to copy the scripts into another local directory and make changes to them. Testing this bit now … On creating copies of the Scripts in another local DIR at path … get to see the warning within the Console

Assets\a_Custom_Scripts\a_EthanCopied\AICharacterControl.cs(19,38): warning CS0436: The type 'ThirdPersonCharacter' in

GitHub Repository - https://github.com/RohitDhankar/Unity3D_2020_UnrealGamelabs_Dancers
Unity Blog Posts - https://datasciencewithrandpython.blogspot.com/search/label/Unity3D

'C:\Win_UnityProjs\AR_Dancers\Assets\a_Custom_Scripts\a_EthanCopied\ThirdPersonCharacter.cs' conflicts with the imported type 'ThirdPersonCharacter' in 'Assembly-CSharp-firstpass, Version=0.0.0.0, Culture=neutral, PublicKeyToken=null'. Using the type defined in 'C:\Win_UnityProjs\AR_Dancers\Assets\a_Custom_Scripts\a_EthanCopied\ThirdPersonCharacter.cs'.

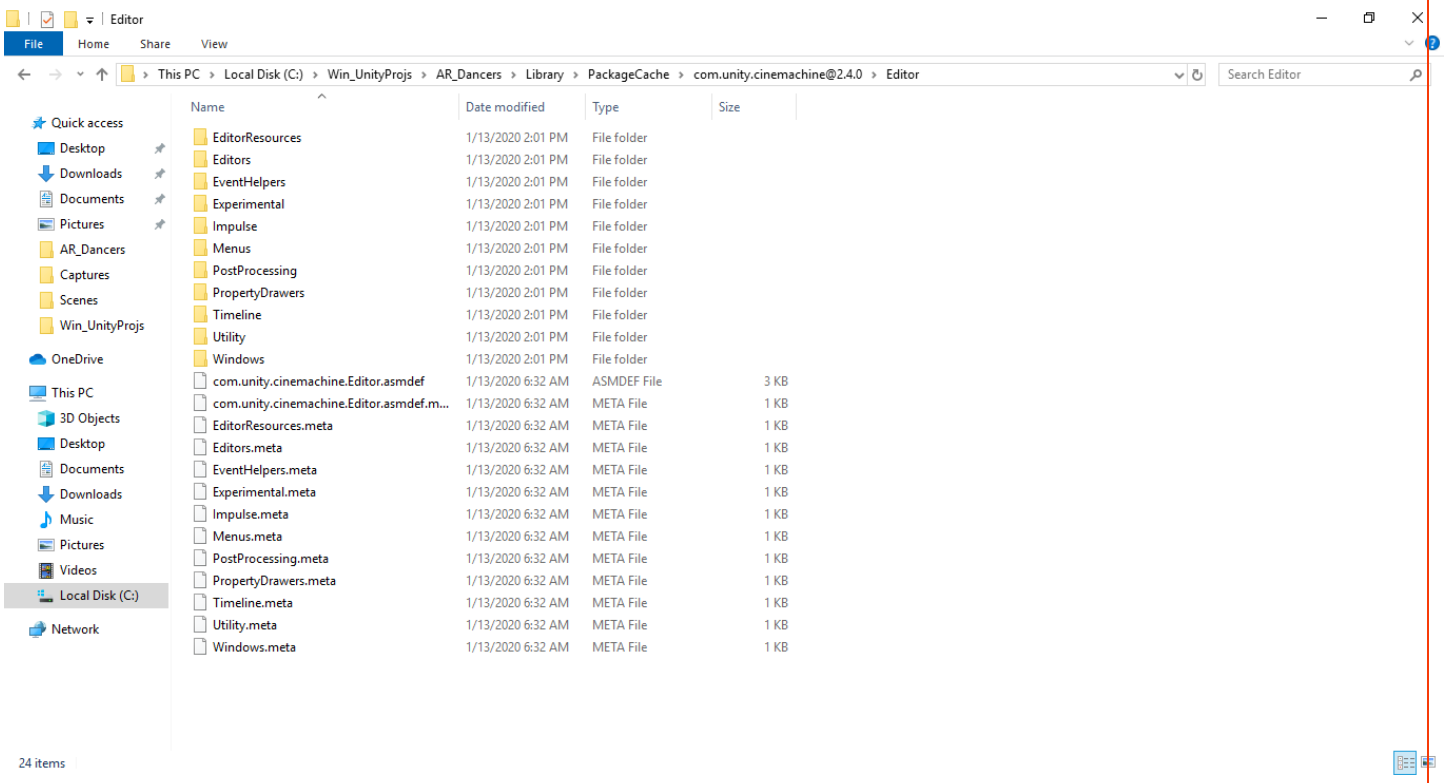
Assets\a_Custom_Scripts\a_EthanCopied\ThirdPersonUserControl.cs(32,40): warning CS0436: The type 'ThirdPersonCharacter' in 'C:\Win_UnityProjs\AR_Dancers\Assets\a_Custom_Scripts\a_EthanCopied\ThirdPersonCharacter.cs' conflicts with the imported type 'ThirdPersonCharacter' in 'Assembly-CSharp-firstpass, Version=0.0.0.0, Culture=neutral, PublicKeyToken=null'. Using the type defined in 'C:\Win_UnityProjs\AR_Dancers\Assets\a_Custom_Scripts\a_EthanCopied\ThirdPersonCharacter.cs'.

Here above **Assembly-CSharp** , is the compilation of C# code scripts into a Package Structure or a JSON defined loose coupling / assembly of Scripts , in particular directory, if during runtime or Compilation , there are any CLASS / TYPE conflict's found they are highlighted in the Editor Console

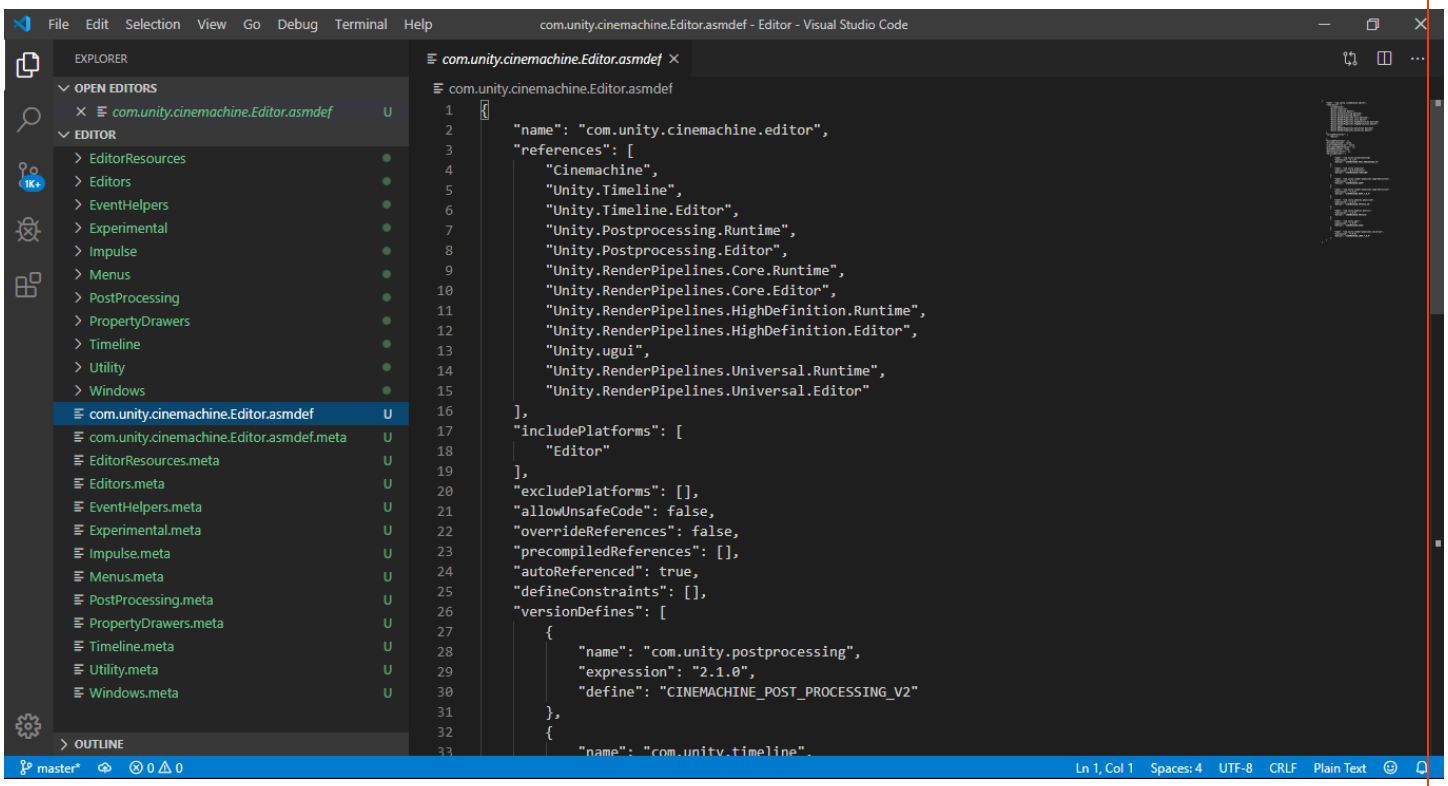
Further Reading – Unity Official Documentation - https://docs.unity3d.com/Manual/ScriptCompilationAssemblyDefinitionFiles.html#AsmdefFileFormat

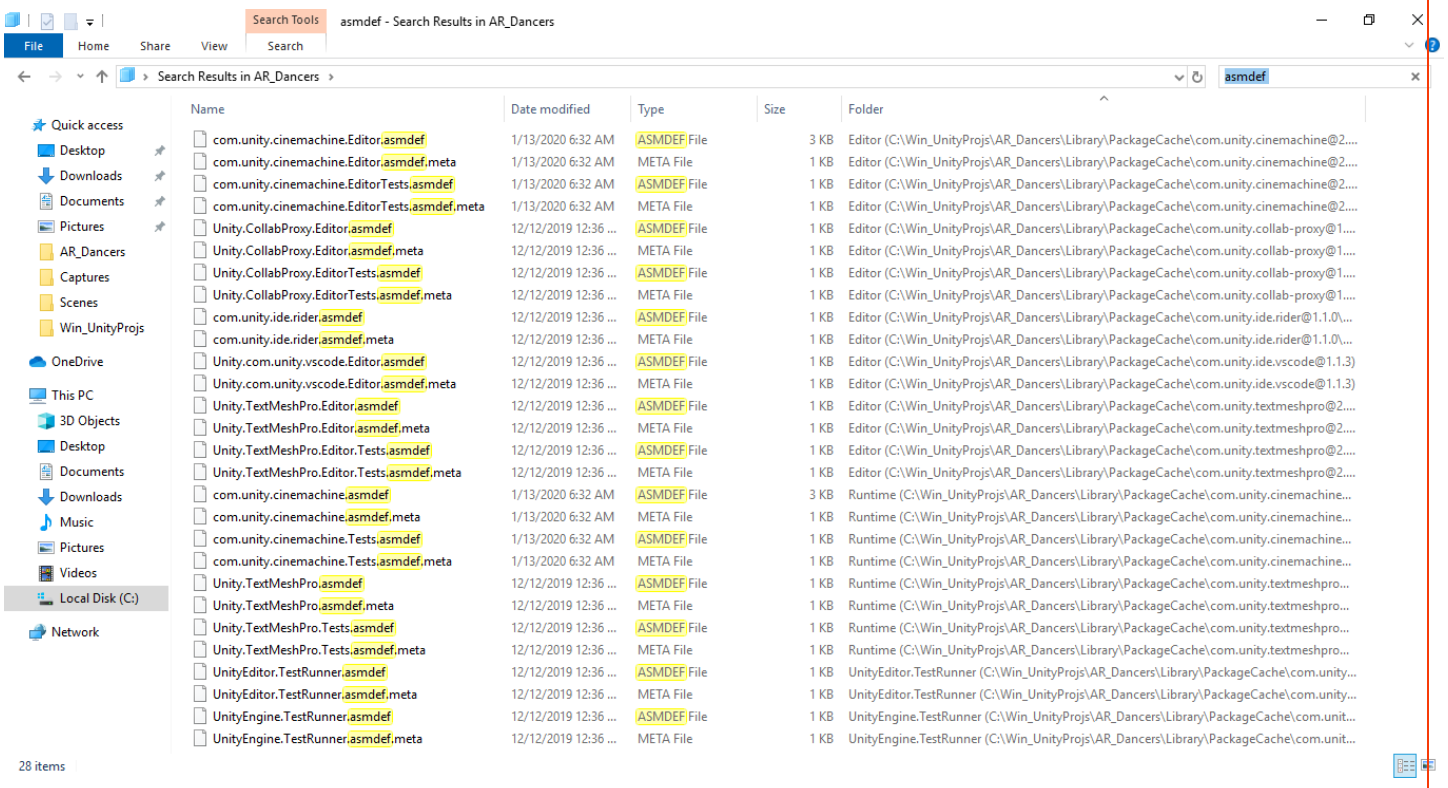
https://docs.unity3d.com/Manual/class-AssemblyDefinitionImporter.html

Seen below are screencaptures of some of the **.asmdef** files within my current Unity Project



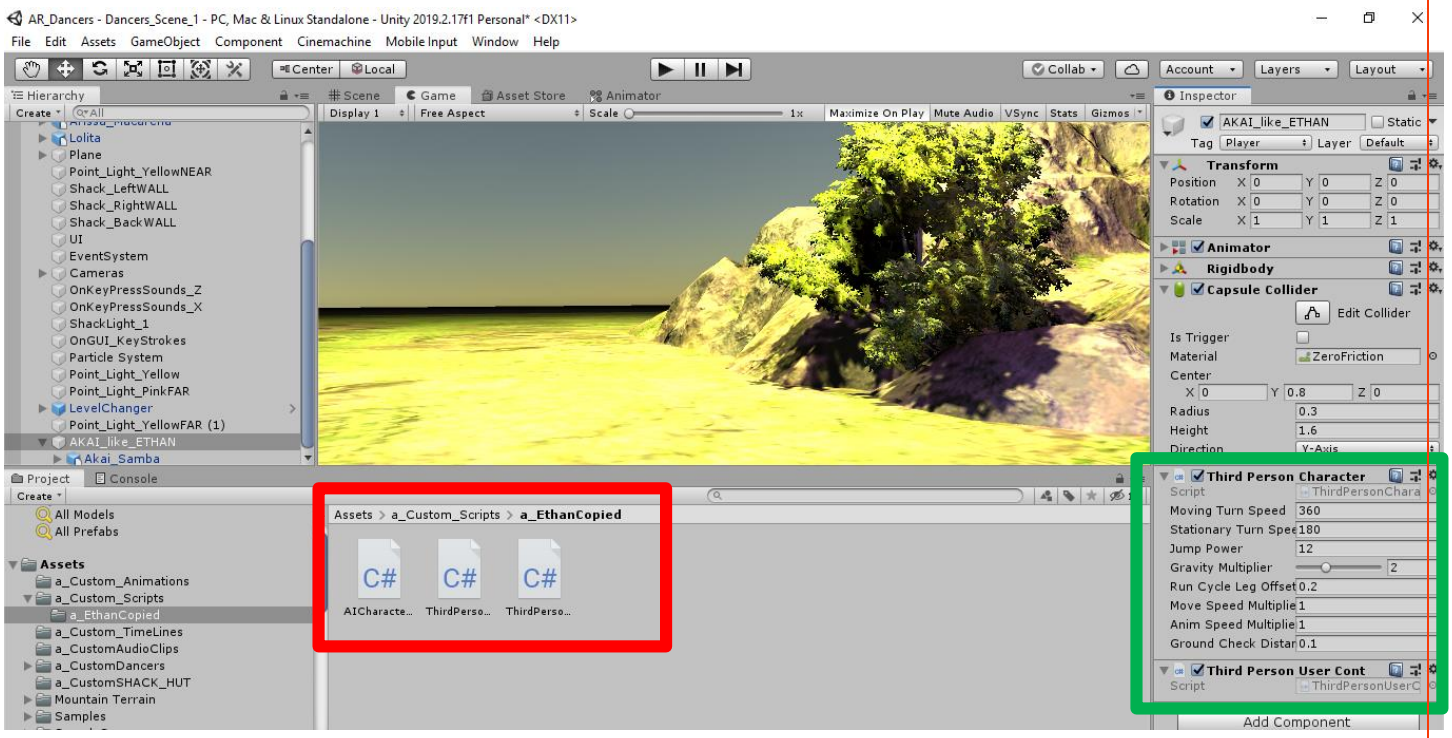
GitHub Repository - https://github.com/RohitDhankar/Unity3D_2020_UnrealGamelabs_Dancers
Unity Blog Posts - https://datasciencewithrandpython.blogspot.com/search/label/Unity3D



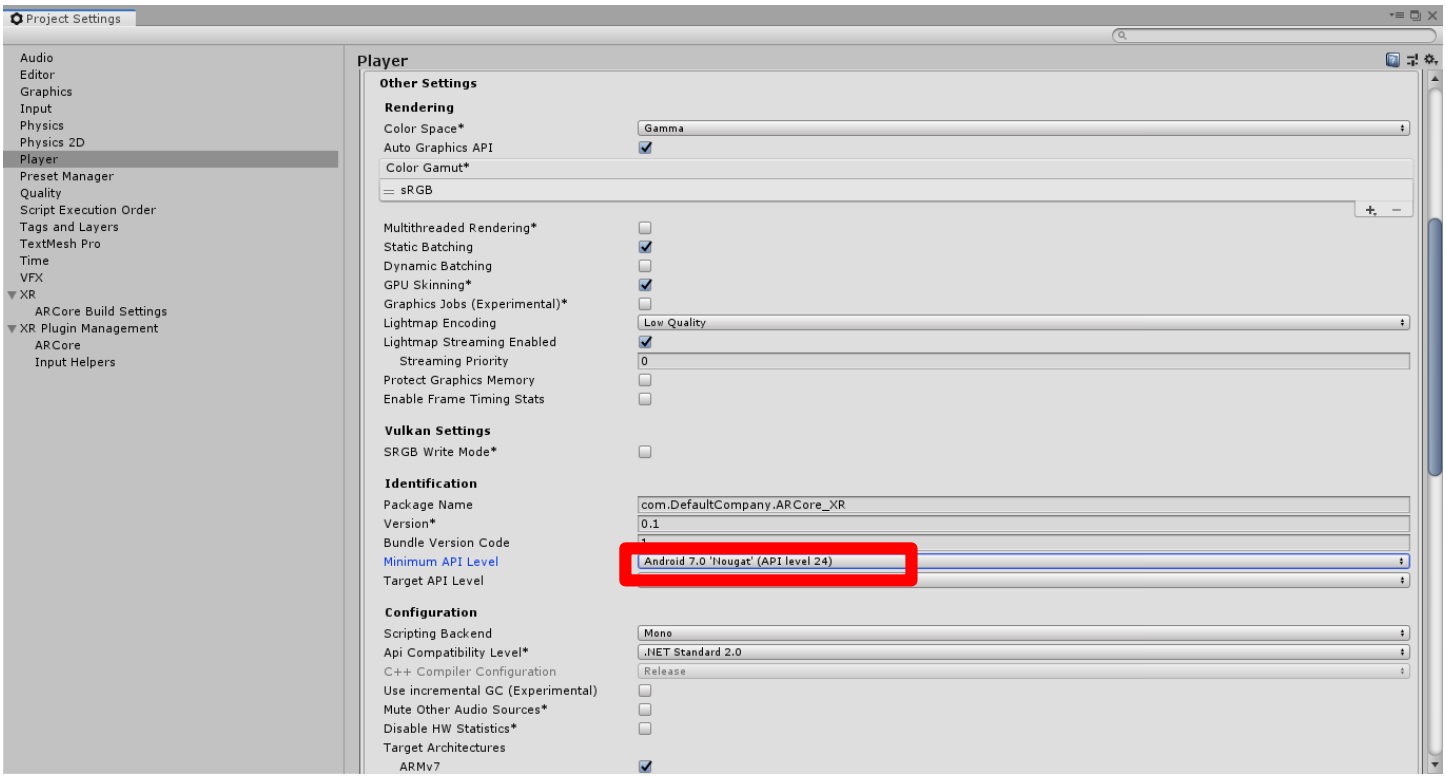


So have created another directory and copied the Ethan Character scripts into it and then placed them onto the INSPECTOR for the - AKAI_like_ETHAN, as seen below.

GitHub Repository - https://github.com/RohitDhankar/Unity3D_2020_UnrealGamelabs_Dancers
Unity Blog Posts - https://datasciencewithrandpython.blogspot.com/search/label/Unity3D



GitHub Repository - https://github.com/RohitDhankar/Unity3D_2020_UnrealGamelabs_Dancers
Unity Blog Posts - https://datasciencewithrandpython.blogspot.com/search/label/Unity3D



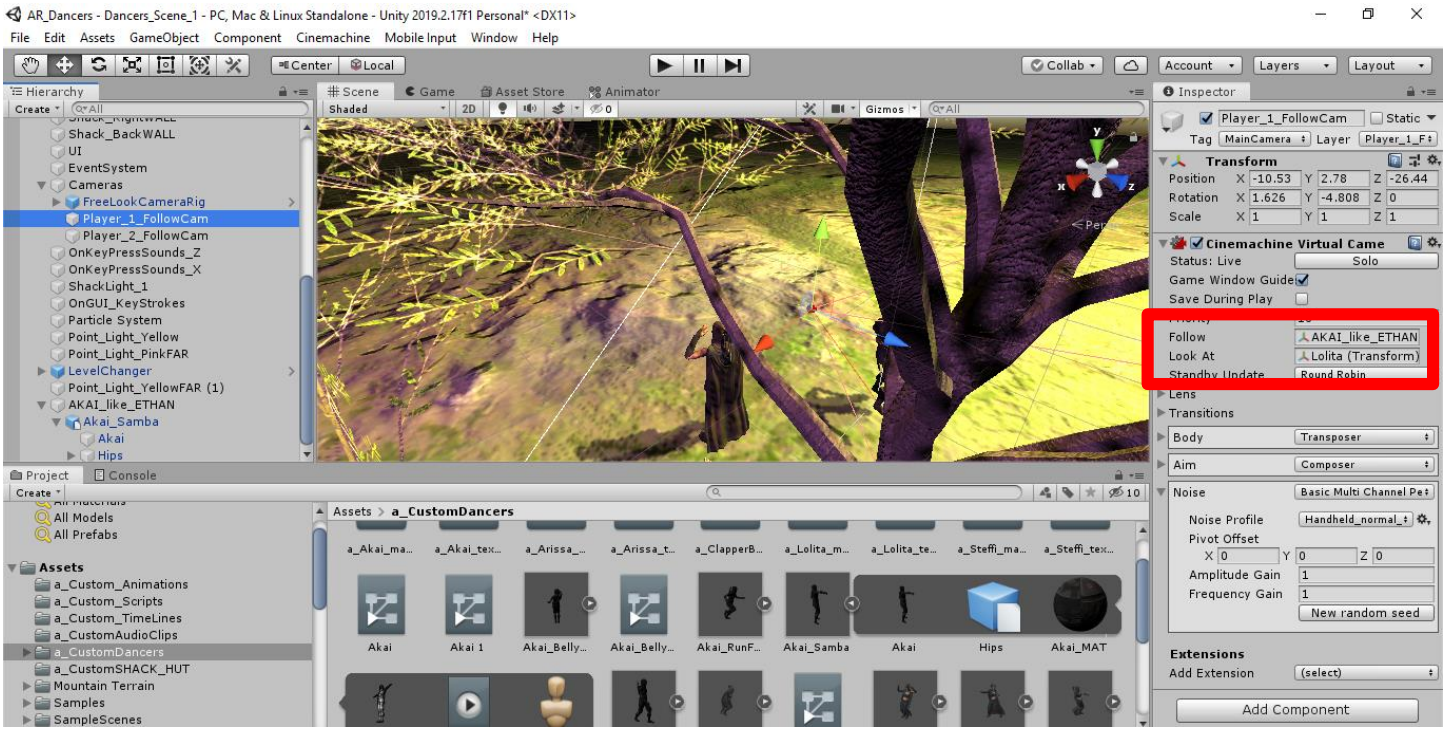
GitHub Repository - https://github.com/RohitDhankar/Unity3D_2020_UnrealGamelabs_Dancers
Unity Blog Posts - https://datasciencewithrandpython.blogspot.com/search/label/Unity3D

Again Revisit the Follow Cameras

Experiment with Follow Cam Following – Akai (Supposedly Samba dancing) , but this Follow cam is all the while actually looking at Lolita .

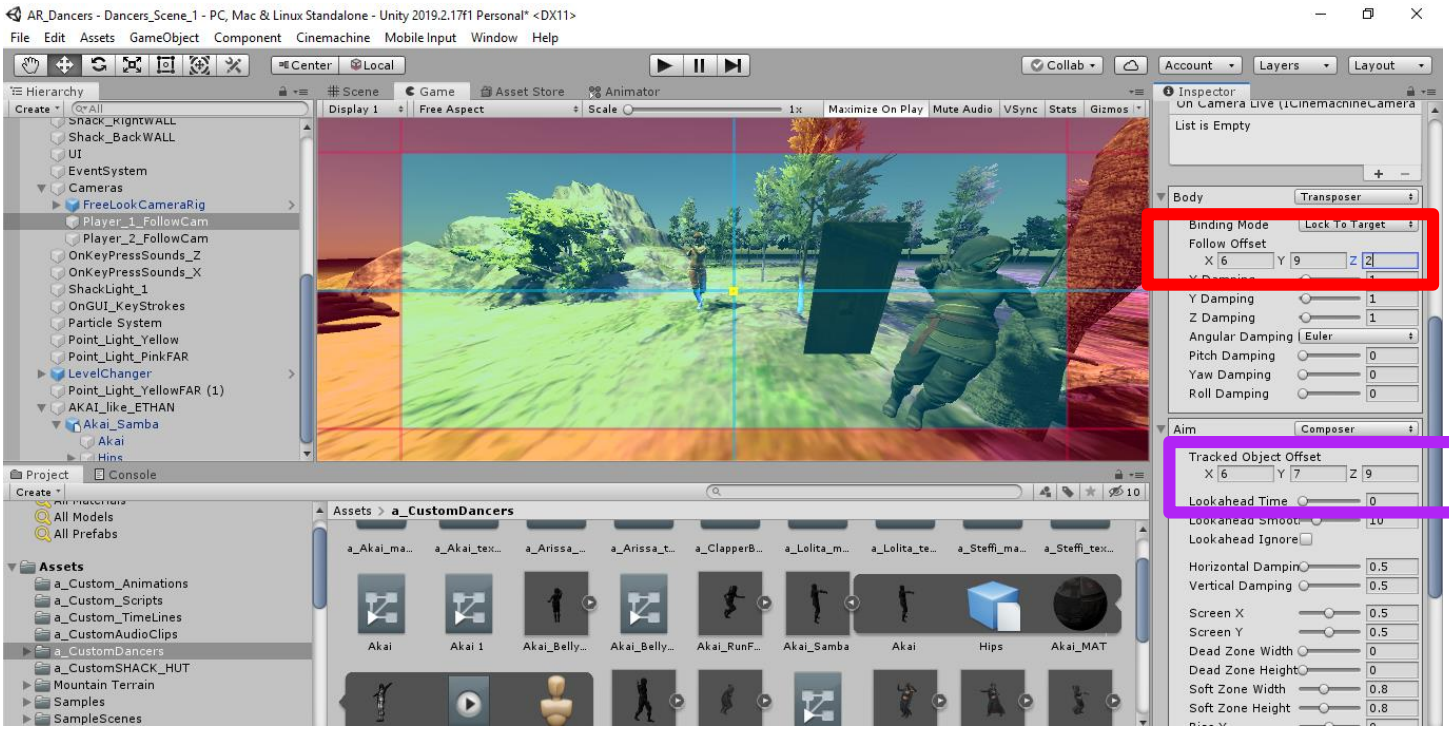
This effect is simply achieved by - dragging the - Akai model onto the - **Player_1_FollowCam >> Inspector >> CineMachine Virtual Camera >> Follow**

As seen below the Follow and LookAt values are different



See the experimental effect here --
https://www.youtube.com/watch?v=Olxqda6SocU&feature=youtu.be

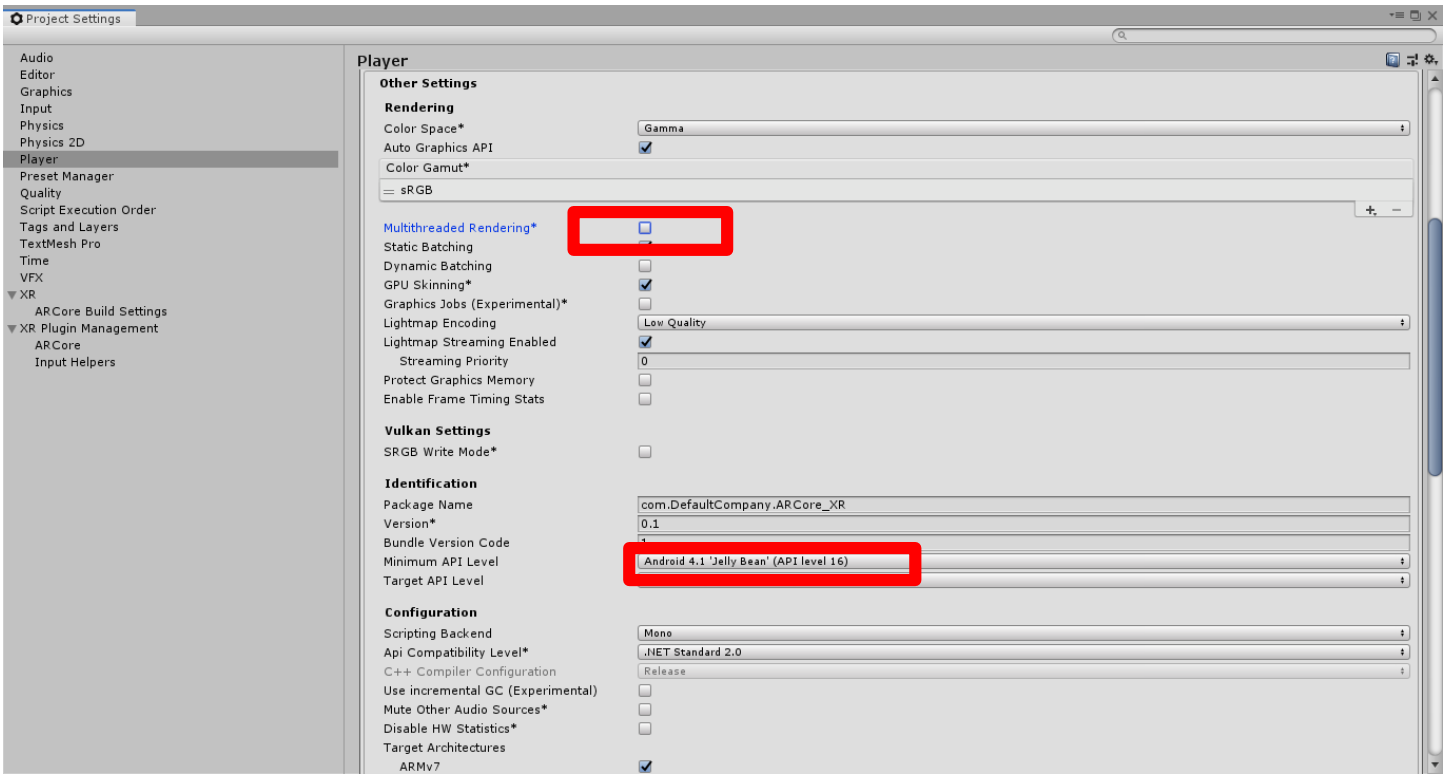
GitHub Repository - https://github.com/RohitDhankar/Unity3D_2020_UnrealGamelabs_Dancers
Unity Blog Posts - https://datasciencewithrandpython.blogspot.com/search/label/Unity3D



GitHub Repository - https://github.com/RohitDhankar/Unity3D_2020_UnrealGamelabs_Dancers
Unity Blog Posts - https://datasciencewithrandpython.blogspot.com/search/label/Unity3D

ARFoundation ARCore for Android

(Work In Progress)



As seen above choosing Project Settings >> Player Settings >> Android

1. Uncheck - Multithreaded Rendering
2. Choose Minimum API Level as – Android 7.0 and later
3.

GitHub Repository - https://github.com/RohitDhankar/Unity3D_2020_UnrealGamelabs_Dancers
Unity Blog Posts - https://datasciencewithrandpython.blogspot.com/search/label/Unity3D