IK Helper Tool

Game Engine: Unity
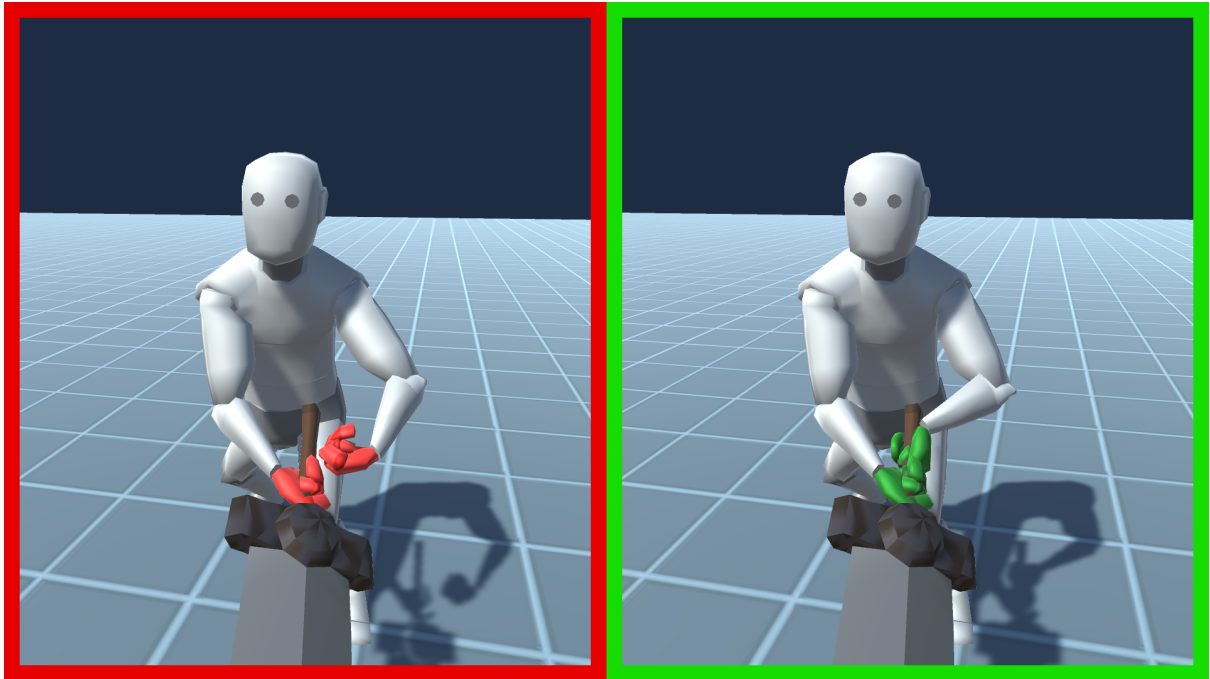
Kevin Iglesias - www.keviniglesias.com

Contact Support:
support@keviniglesias.com

[Online Documentation](#)

Overview

IK Helper Tool is a script that fixes mispositioned hands when retargeting animations (holding an item or prop) between two models with different arm rig length.

Requisites

In order to make the script work you need:
- The script added as component to your character model (where your Animator Controller is).
- IK Pass enabled on your animation layer.
- A duplicated of the animation you want to fix (to make it writable)
- 2 empty GameObjects to use as IK effector and IK switch and both referenced in the IK Helper Tool script.
- Y position (from IK switch empty GameObject) property with value 1 added to the duplicated animation.
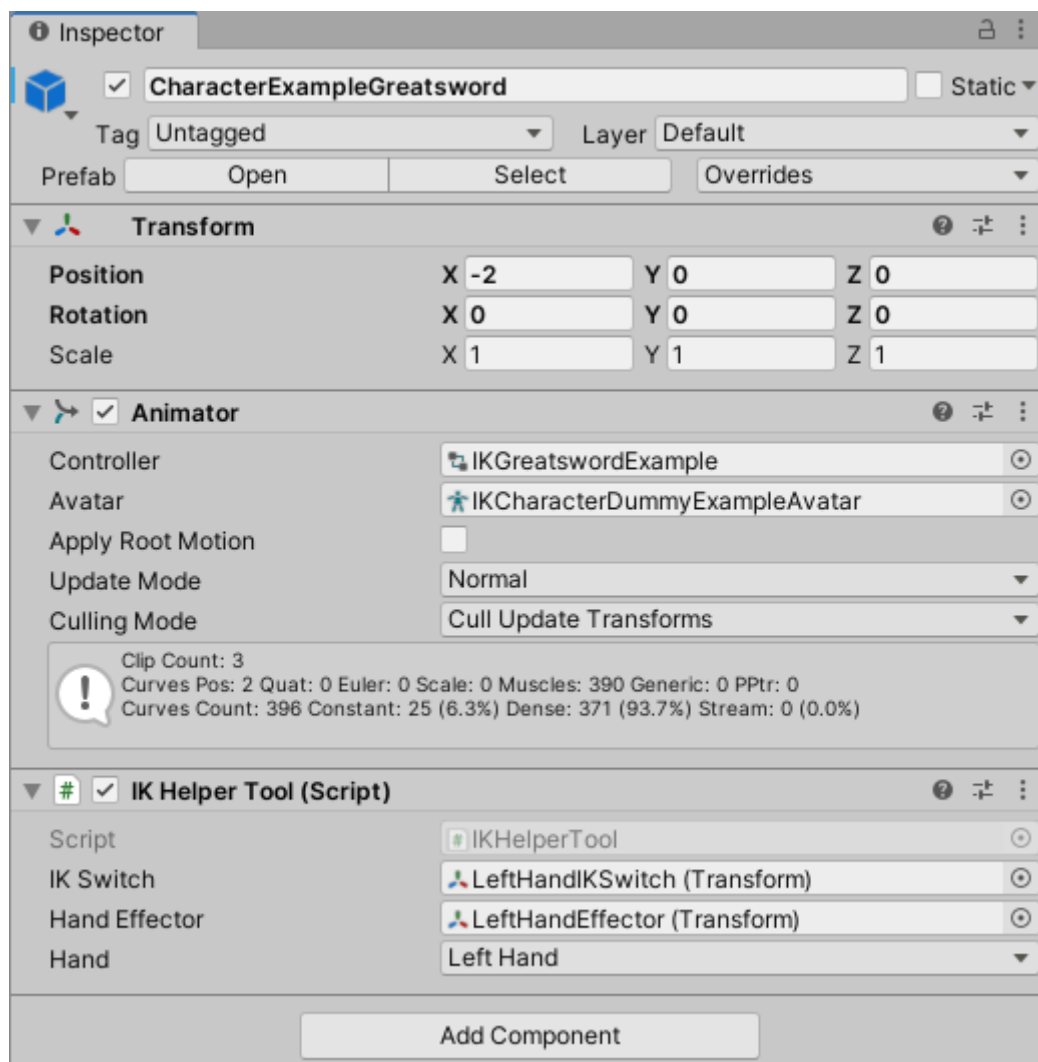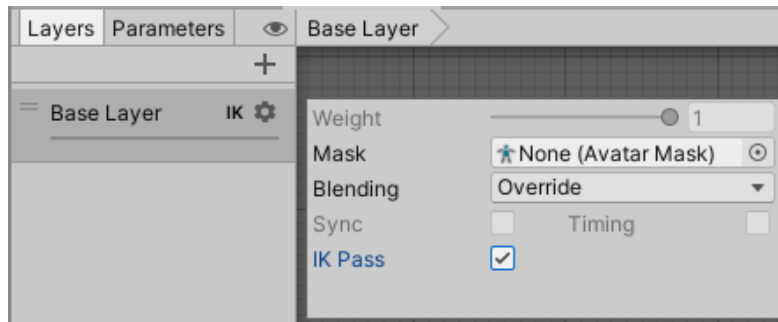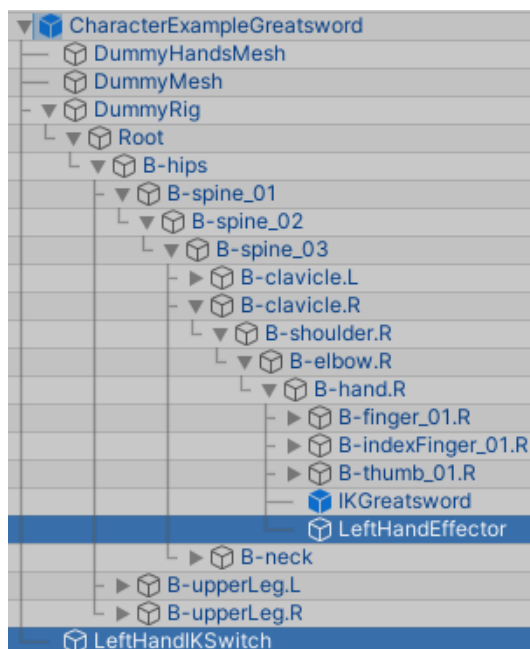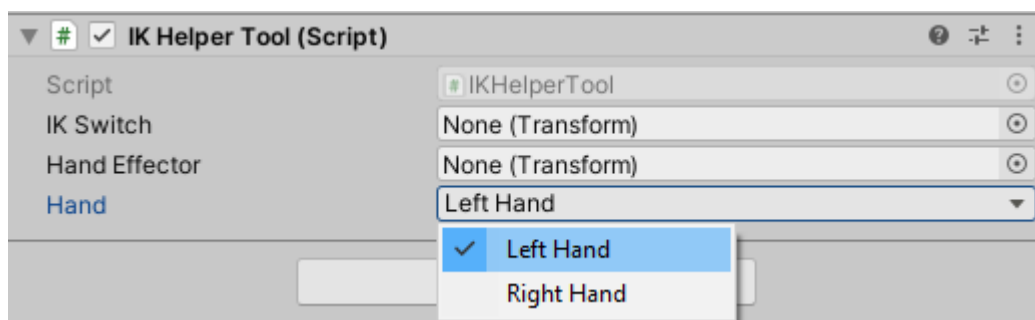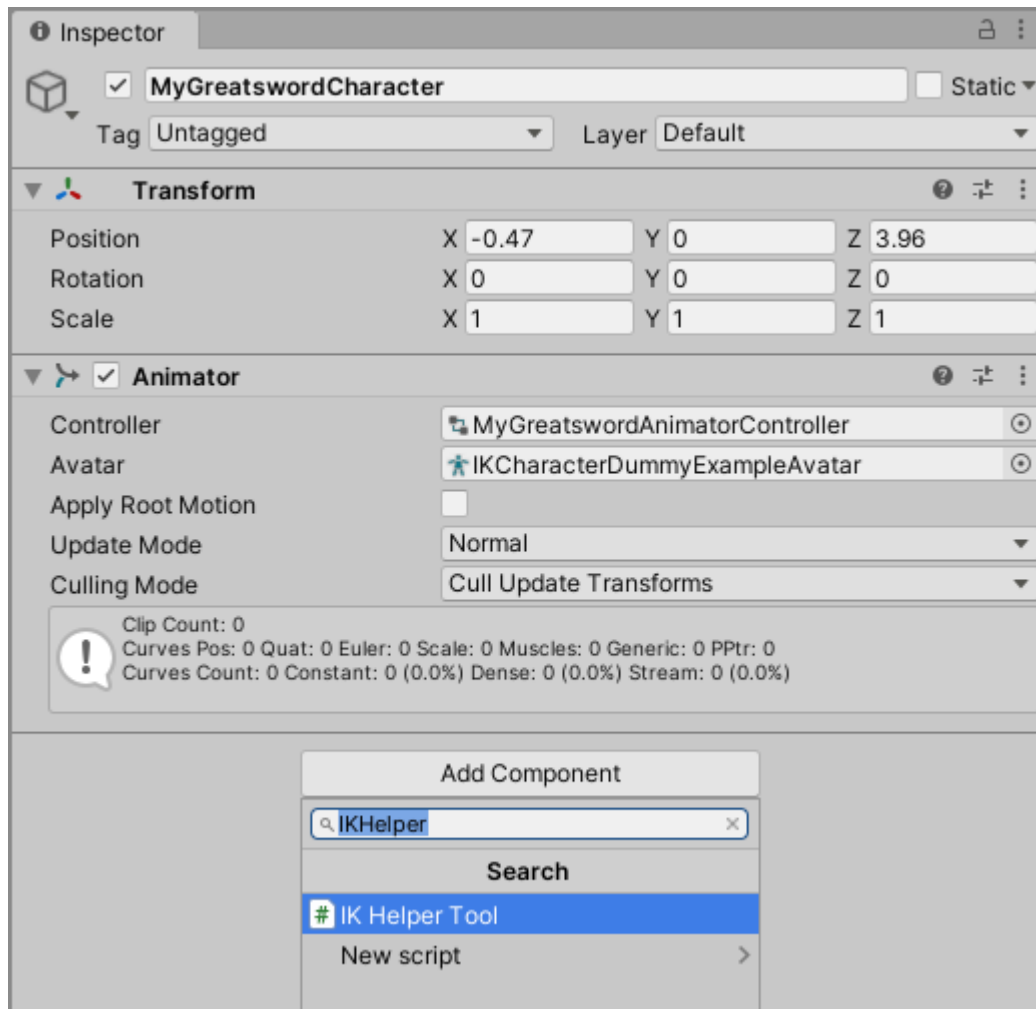
IK Pass enabled
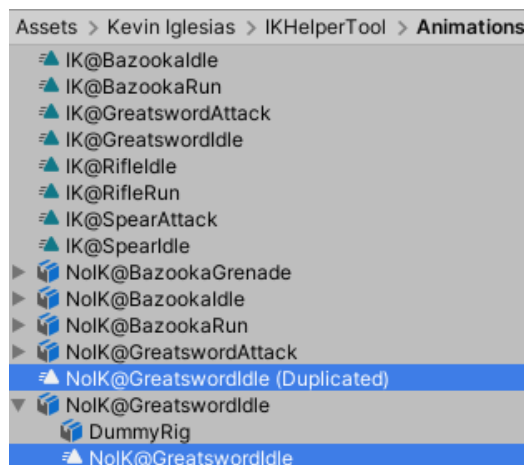
Duplicated animation (the duplicated is now writable)

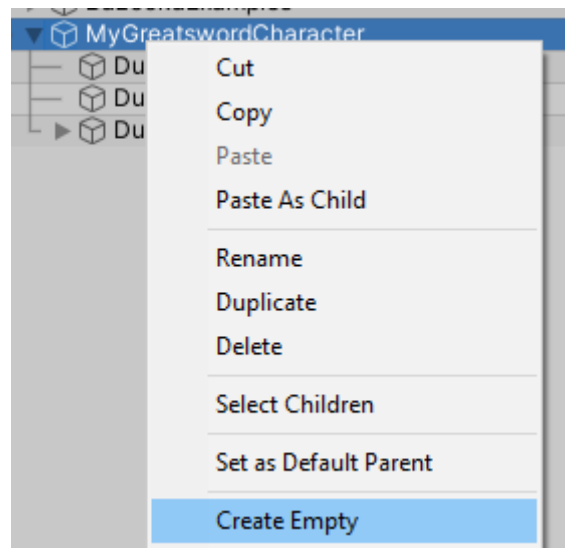IK effector and IK switch (empty GameObjects)

Configuration Guide

We start adding the IK Helper Tool script to our character.

In the 'Hand' field we can select which hand needs to be fixed. Usually there is one hand that controls both the object being held and the other hand.
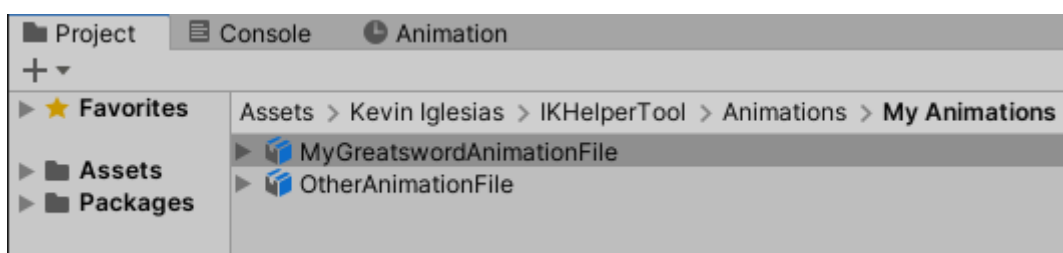
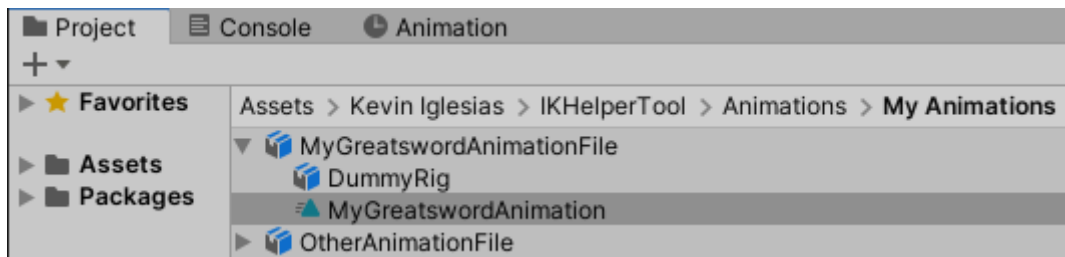Then, we need to create an Empty GameObject at the root of our character.

The name to set is important as we will use it later, for this example we will use '**LeftHandIKSwitch**'.
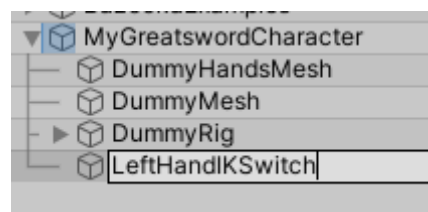
Now we need to duplicate the animation we want to fix. We will do it by looking for it in the Project tab (our project file browser) and we will select and unfold our animation file.

Notice that when unfolding our animation file we can see the clips and the actual animation, in this case '**MyGreatswordAnimation**'.

Now we can select the clip and pressing Ctrl + D (Windows) or ⌘ + D (Mac) we will create a duplicated version of the clip. We must do this because the original animation clip is bound to the animation file and is read-only.


We can now rename our duplicated clip to something different. In this case we added 'WithIK' suffix as this clip will be the one that will trigger the IK Helper Tool script.
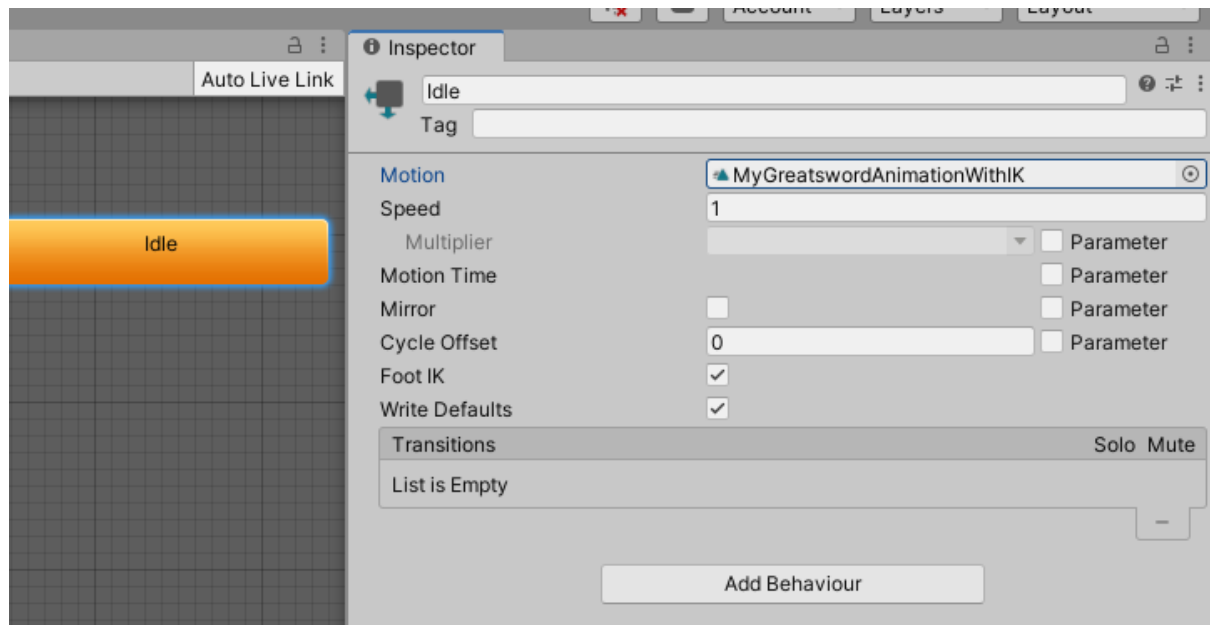

Now we are going to our Animator Controller and the first thing we must do is enable the IK Pass option at the Layers settings.
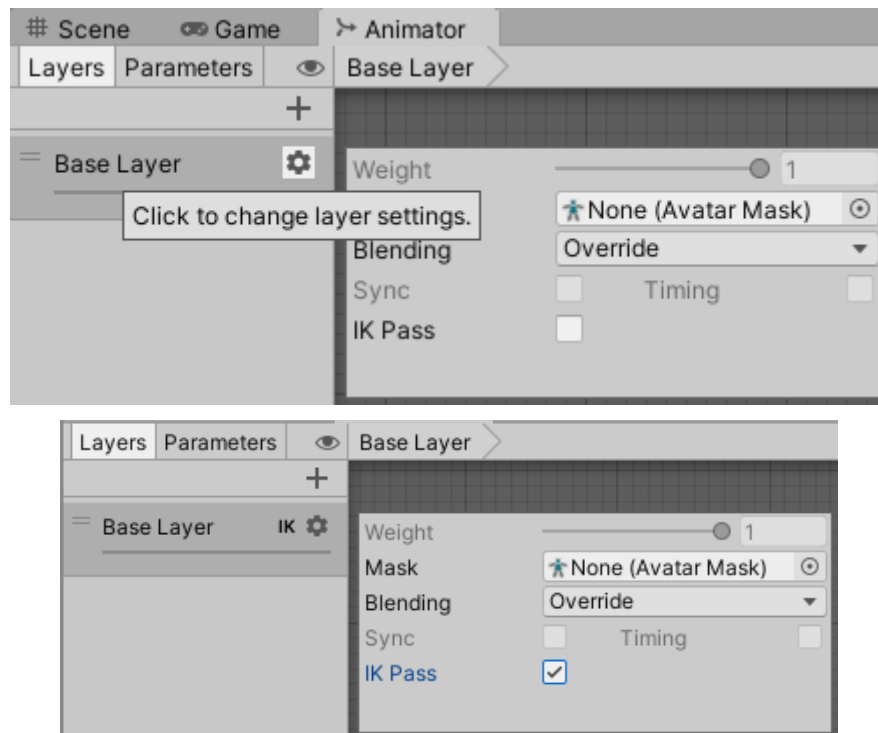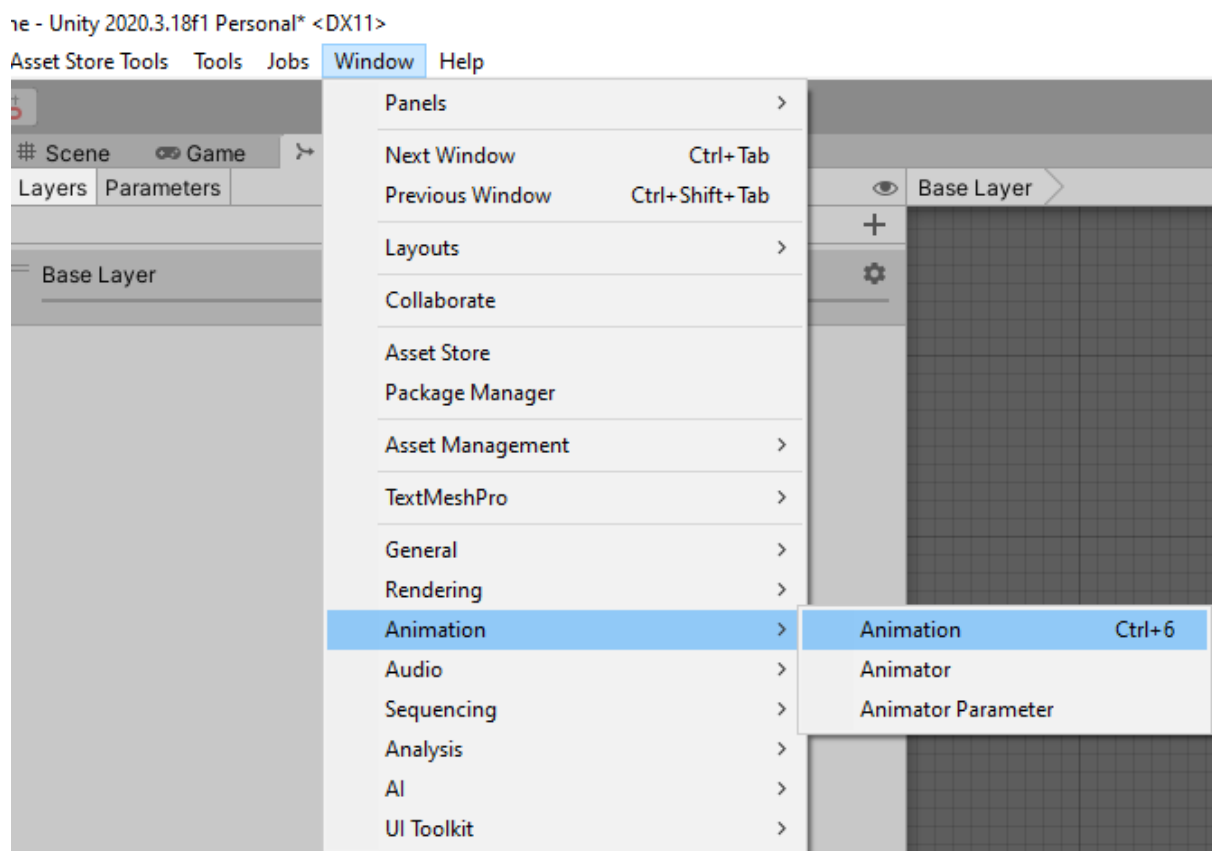
Then we are going to locate our animation state and replace the animation clip with the new one we renamed with the suffix '**WithIK**'.
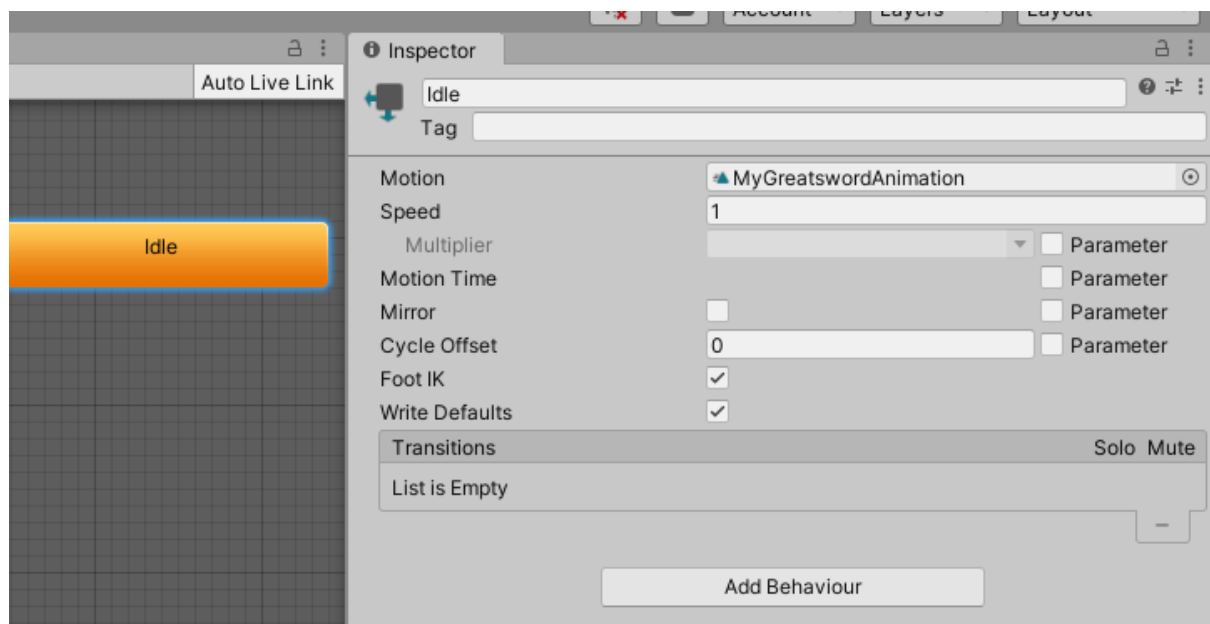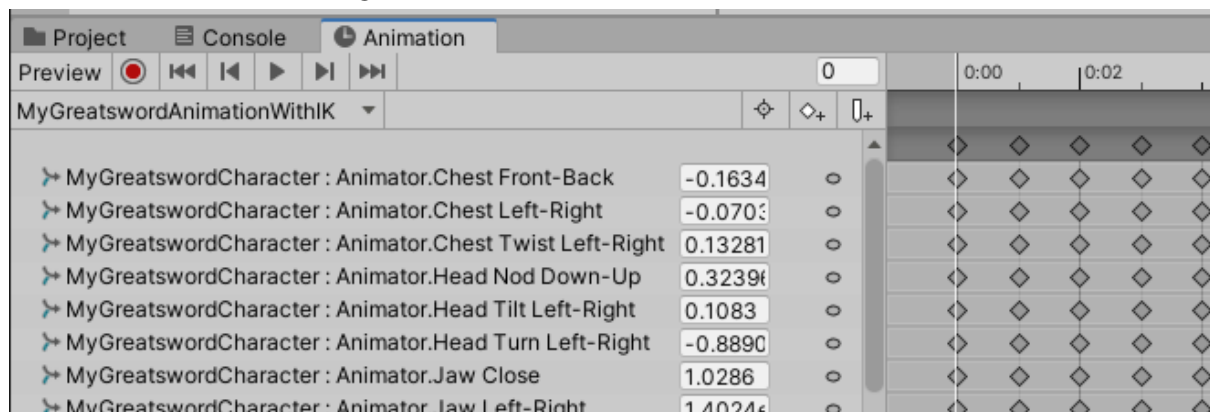
For the next step we need to select our character and, while it is selected, we are going to open the Animation window using the Unity top menu bar (Window > Animation > Animation).

We will see something like this:


Carefully scroll down to bottom and make sure you don't edit any values, we can mess up the animation. If that happens we can make a new duplicate from the original animation again.

At the bottom we will see a button called '**Add Property**'. Click on it, look for the empty GameObject we created before (LeftHandIKSwitch) and select to add Position property (under Transform).
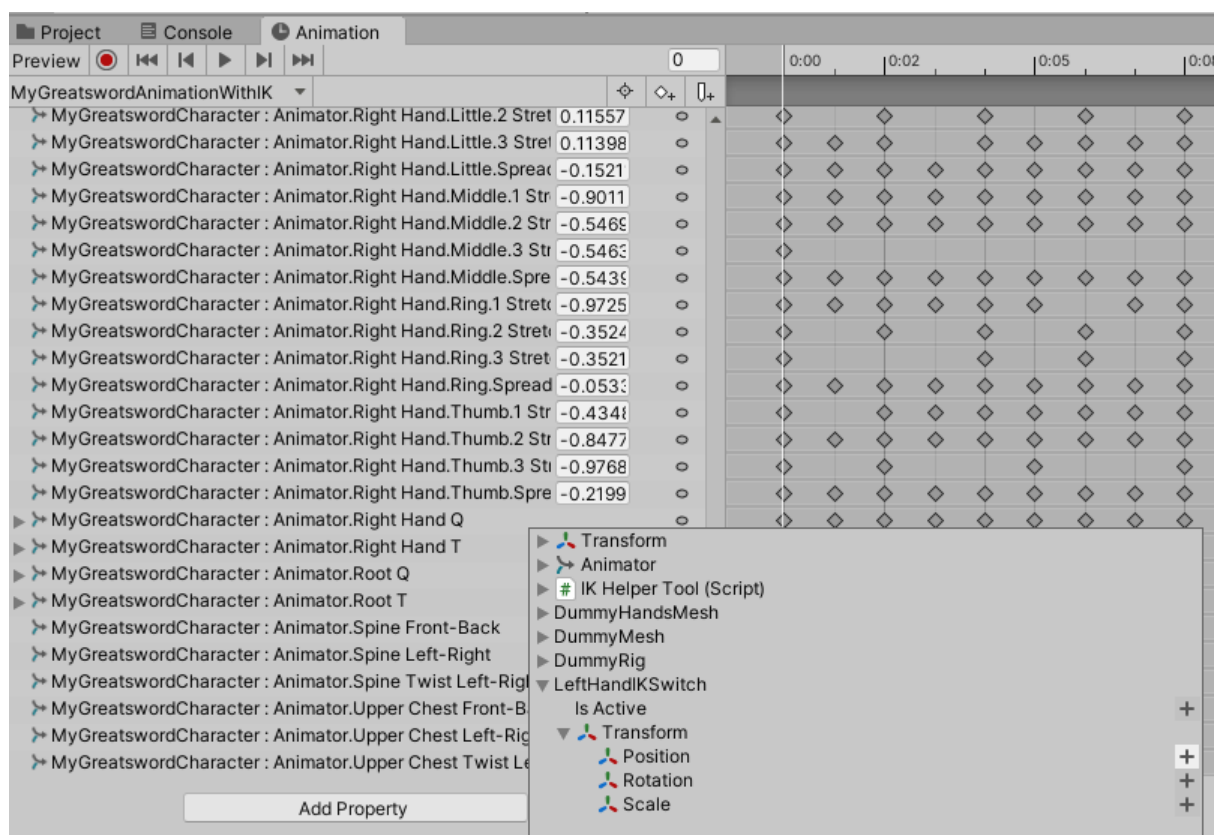
Then we will set the value of Position.y to 1 in the first and the final frame.
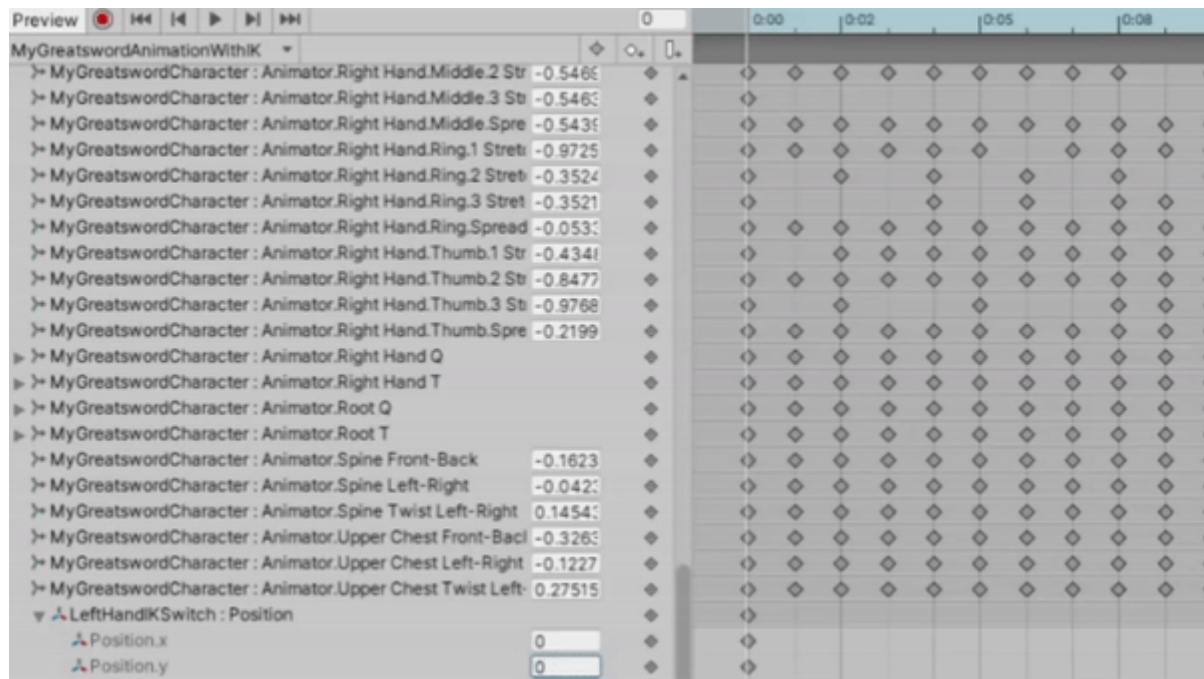

The last thing we need to do is create a new empty GameObject and parent it to the hand that holds our item or weapon. We will call it 'LeftHandIKEffector'. Make sure both LeftHandIKSwitch and LeftHandIKEffector are referenced in the script:
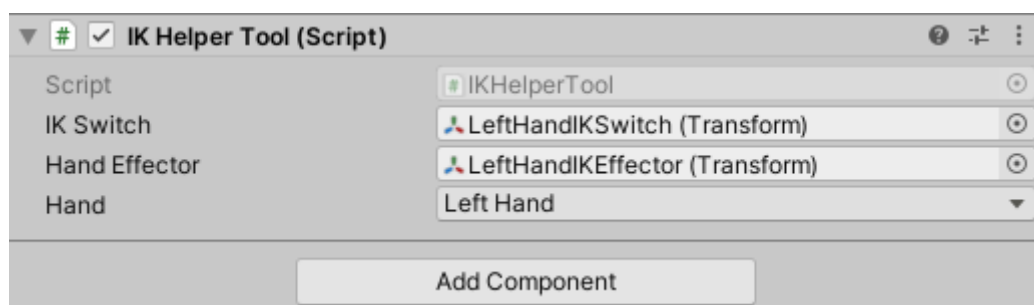
If everything went well we should have the left hand following the LeftHandIKEffector when playing our scene.

We can now edit (while in Playmode) the LeftHandIKEffector to make the hand actually hold our item. After that, while still in Playmode, we copy the Transform attributes from our effector, we can now exit Playmode and paste them back.

One tip is to set the animation speed to 0 in the animation state. This way editing the LeftHandIKEffector position and rotation is easier. Just make sure you set it back to the original animation speed when you finish editing.
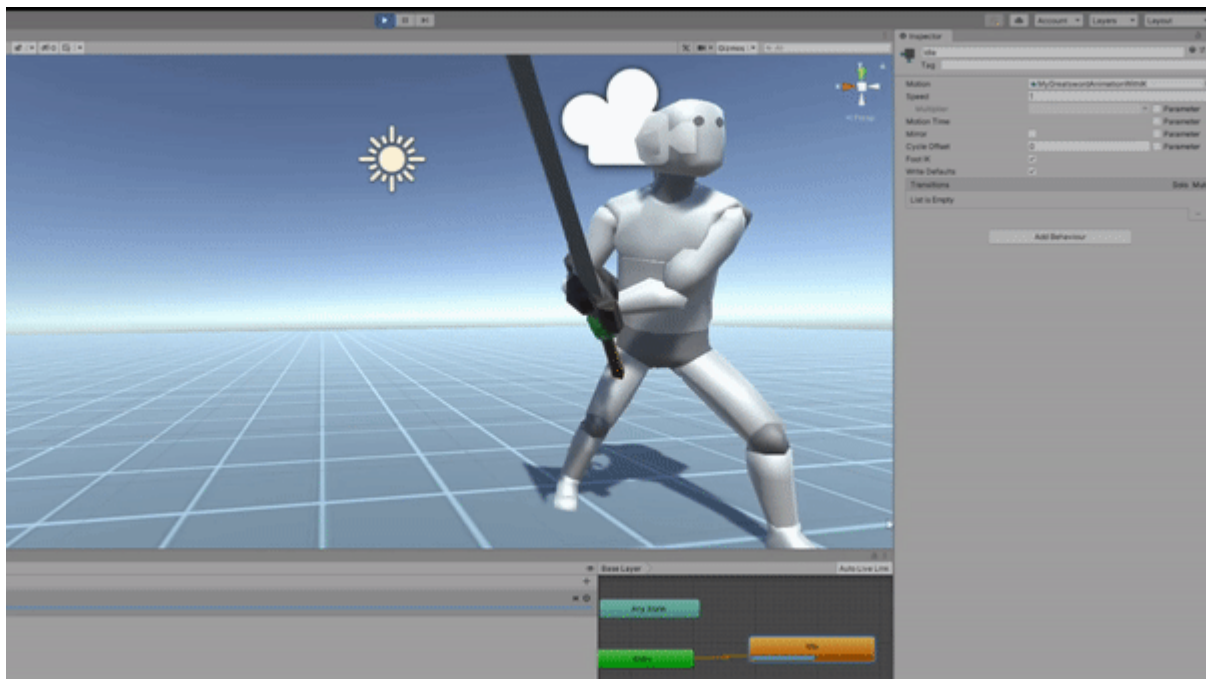
Contact Support

For support, questions or suggestions regarding this product send me an email to:
support@keviniglesias.com

Kevin Iglesias - www.keviniglesias.com

- **More Unity Assets**

- **Facebook**

- **Twitter**

- **YouTube Channel**