

# Setup Your Own Bike

## Guide

Offered by AIKODEX

**Simple Motocross Physics** allows you to add your own bikes to your game. These bikes need not necessarily be dirt bikes, but it does help if your model looks similar to the architecture of the bike the asset supports.

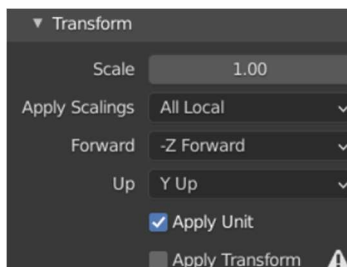
You will need to perform the following steps to setup your own bike:

### 1. Model Preparation



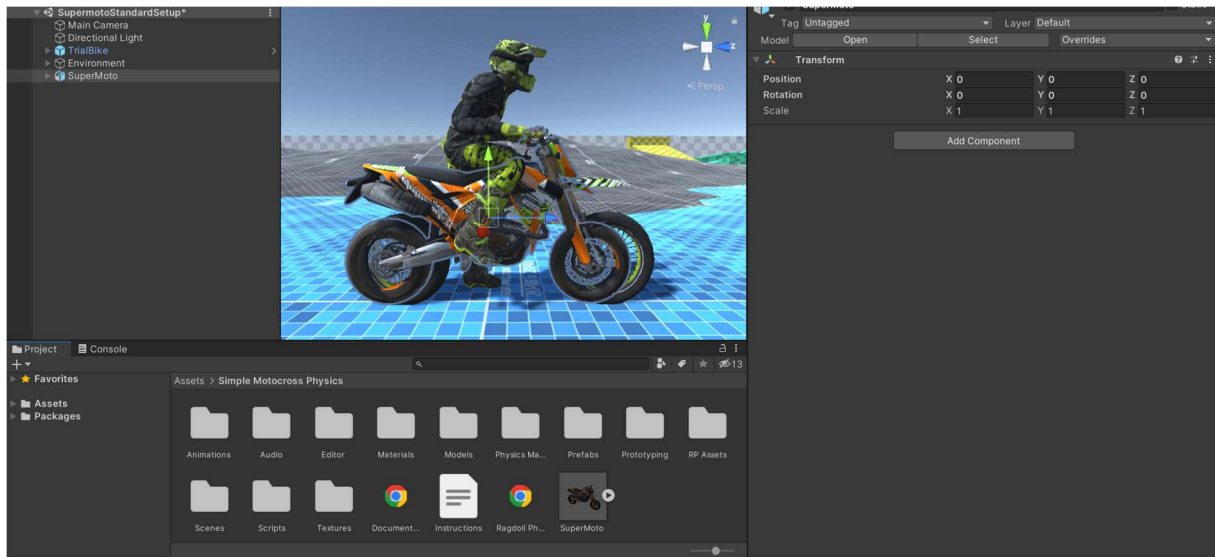
The model needs to have at least the parts shown above. The red circles indicate the pivot points of the model. The Z coordinate of all the pivots is 0. The scale of the bike is close to real life and is frozen as 1,1,1 in the modeling software. All the rotations are frozen to 0,0,0.

Please also make sure while exporting, you check the option Apply Transform.



In Unity, double check if the rotations are exactly as they are shown in your 3D modelling software.

## 2. Setup in Unity

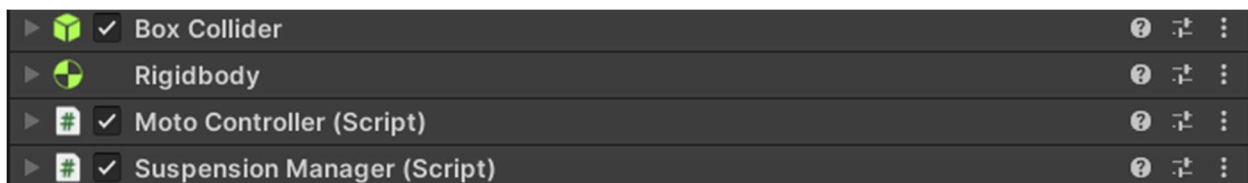


To make the process easier, open the scene that best represents your bike type. Here is the integration for SuperMoto.

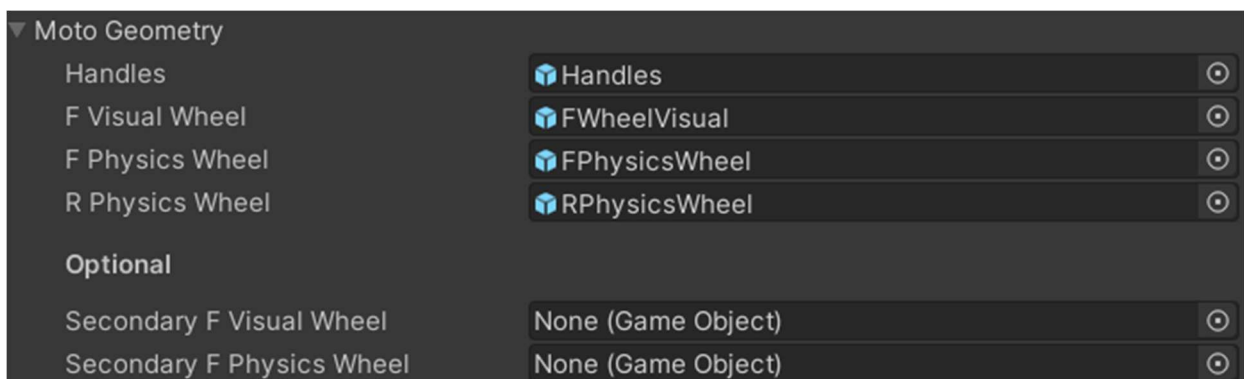
Copy all the components from the sample bike to the new bike.

This includes the following:

- Rigidbody
- Box Collider
- Moto Controller
- Suspension Manager



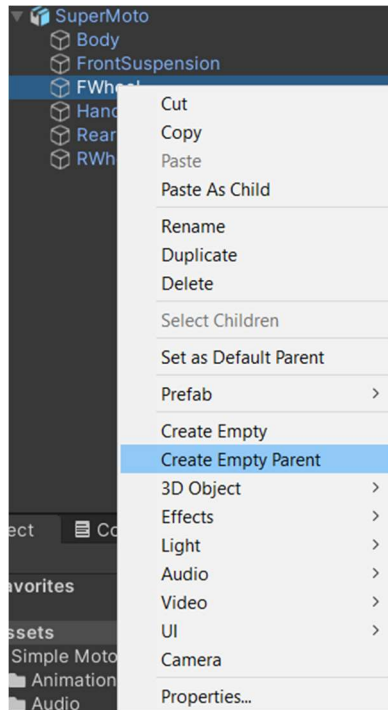
After you have completed this step, go over to the moto controller's moto geometry class and expand it:



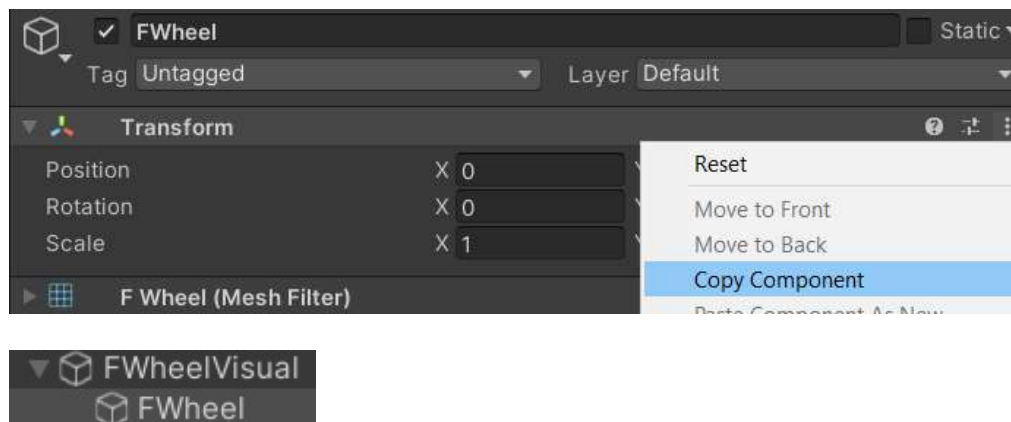
Clicking on the fields should still direct you to the sample bike's parts. Replace the **Handles** gameobject from your bike's hierarchy manually.  
The rest of the parts need to be constructed in Unity.

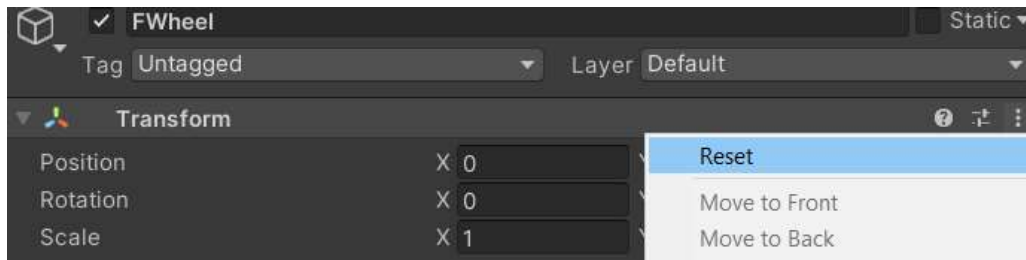
To prepare FWheelVisual make the following change in your front wheel game object:

Create an empty parent for the game object FWheel. Name it FWheelVisual.



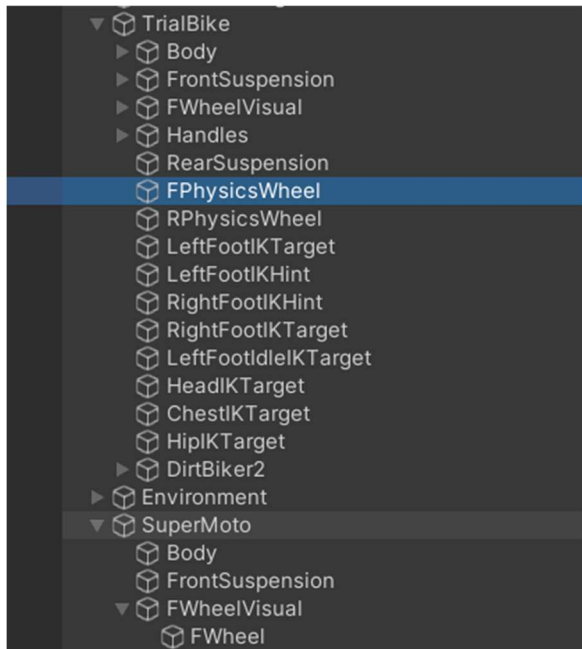
Copy the FWheel transform component onto its parent FwheelVisual. Then Reset the transform of FWheel.





Assign the FWheelVisual to the MotoGeometry public variable F Wheel Visual.

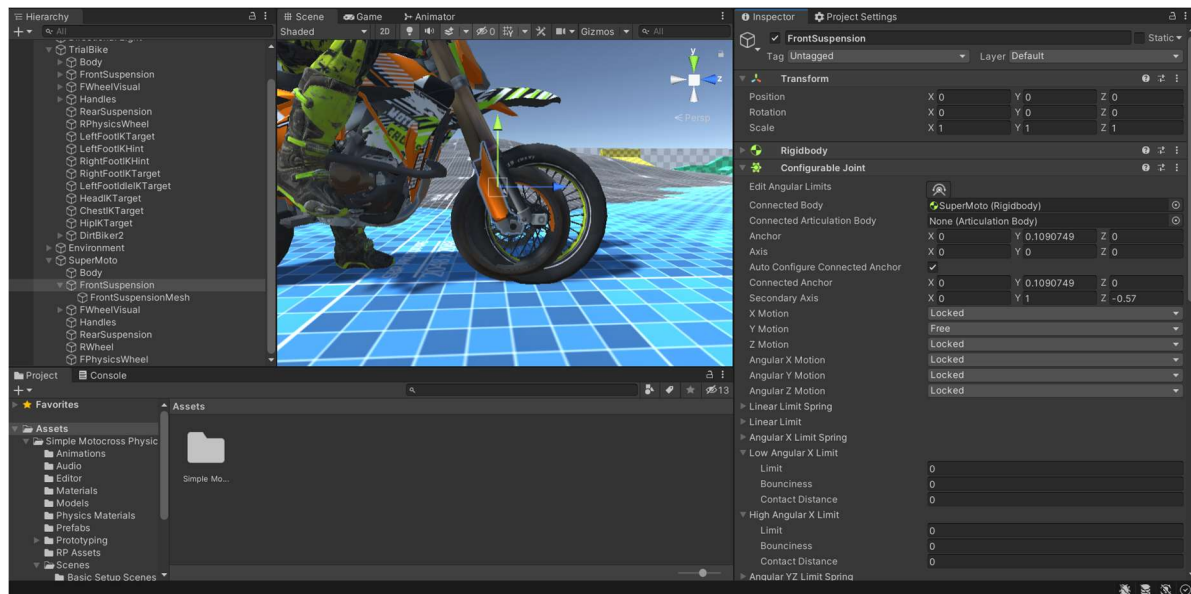
To make the FPhysicsWheel, drag the FPhysicsWheel from the sample into your model's hierarchy.



Following this, make sure the configurable joint's connected body in the FPhysicsWheel points to the Fsuspension.

To make Fsuspension, copy the components of the Fsuspension of the sample bike onto your model's Fsuspension.

Set Fsuspension's connected body to the root of the main gameobject.



Adjust the FPhysicsWheel's Wheelcollider to roughly match your front wheel.

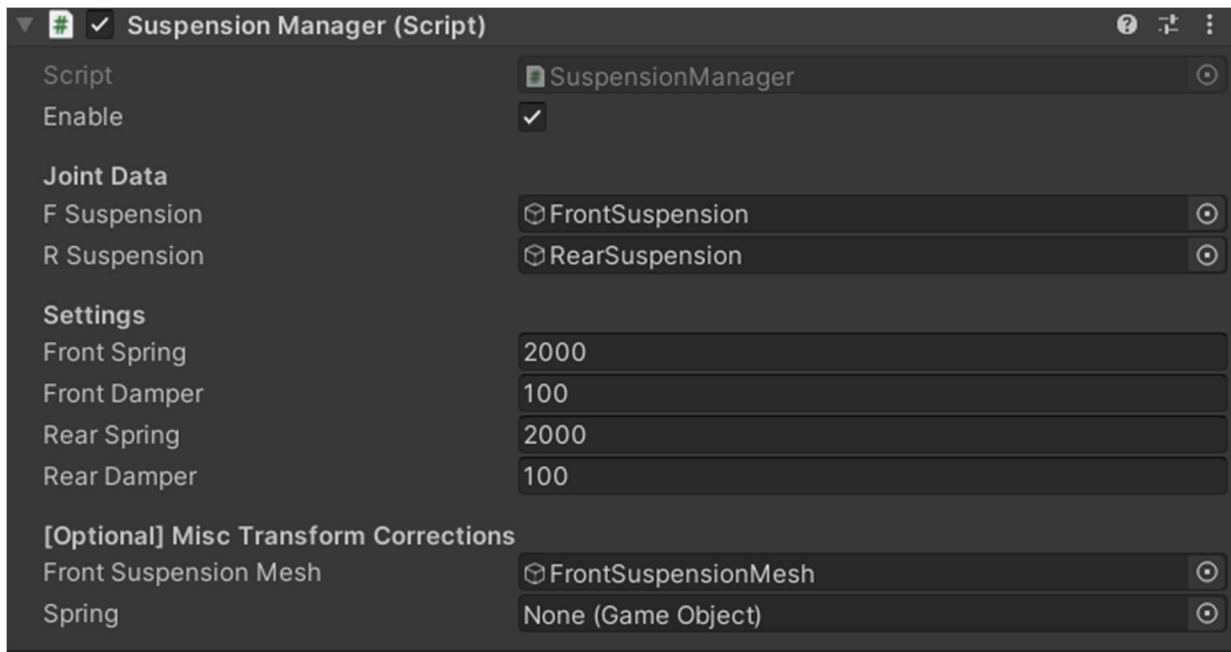
To make RPhysicsWheel, rename RWheel to RPhysicsWheel.

Copy the components of the RPhysicsWheel from the sample bike onto your RWheel or now RPhysicsWheel.

Set the configurable joint's connected body to RSuspension. (RSuspension can be made by copying components Rigidbody and a Configurable Joint from the sample bike)

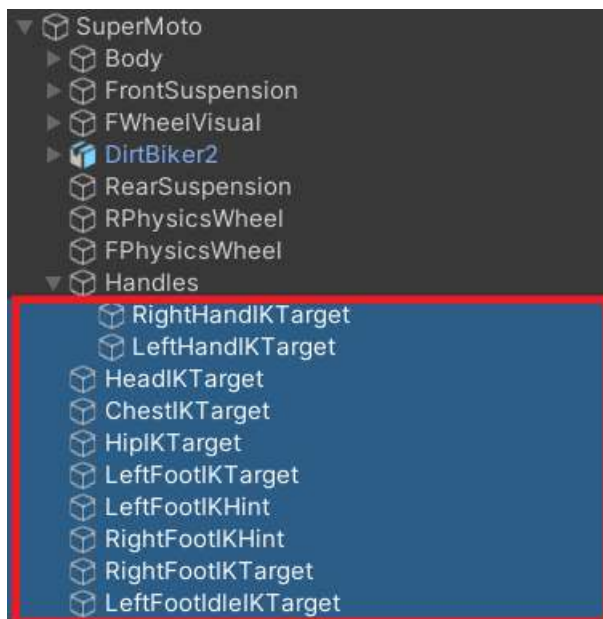
Adjust the sphere collider of RPhysicsWheel to roughly match the radius of the tyre.

Do not forget to assign the RPhysicsWheel and FPhysicsWheel in the MotoController.cs. Also assign the Fsuspension, Rsuspension and the FSuspension Mesh in the Suspension Manager script in the main root gameobject.



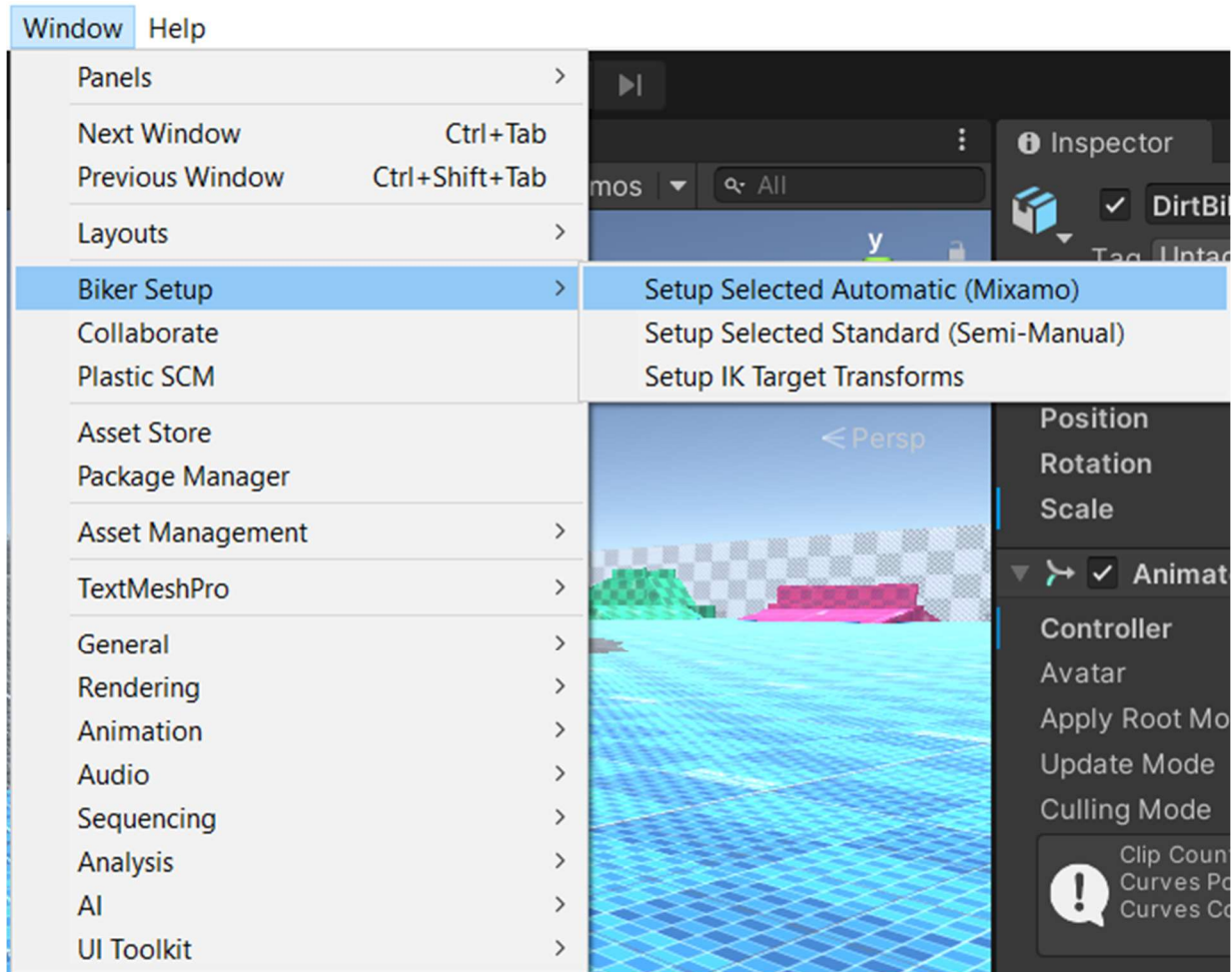
### 3. Rider Setup on the new Bike:

Transfer the following transforms from the sample bike into the hierarchy of the new bike:

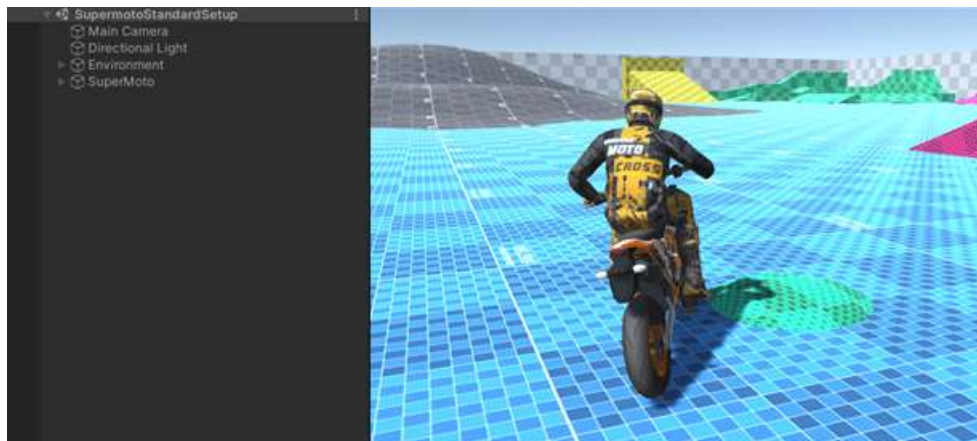


Select the (Mixamo) biker model and go to Window > Biker Setup > Setup Selected Automatic (Mixamo)





Hit play and test out the scene.



If you have any questions, please feel free to contact us at [info@aikodex.com](mailto:info@aikodex.com).

AiKodex