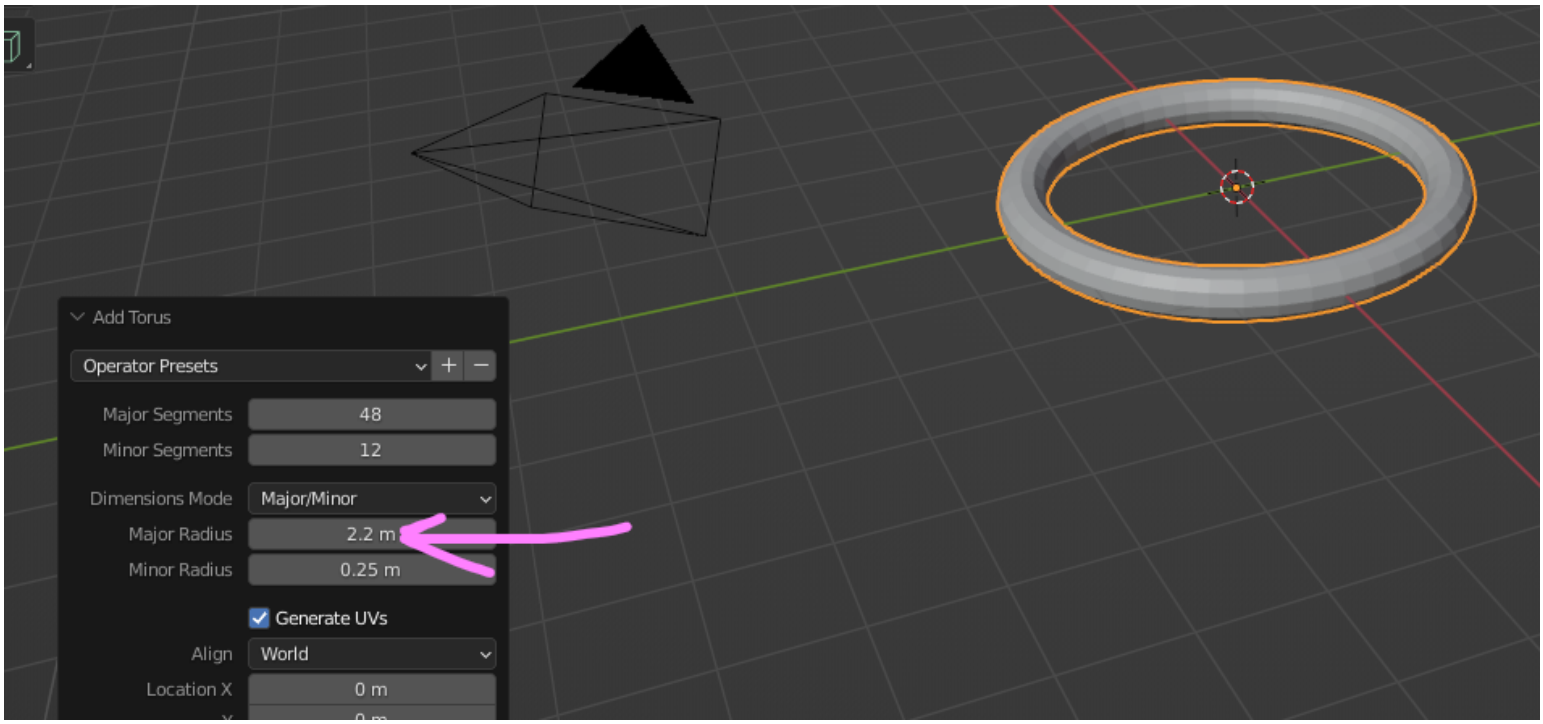
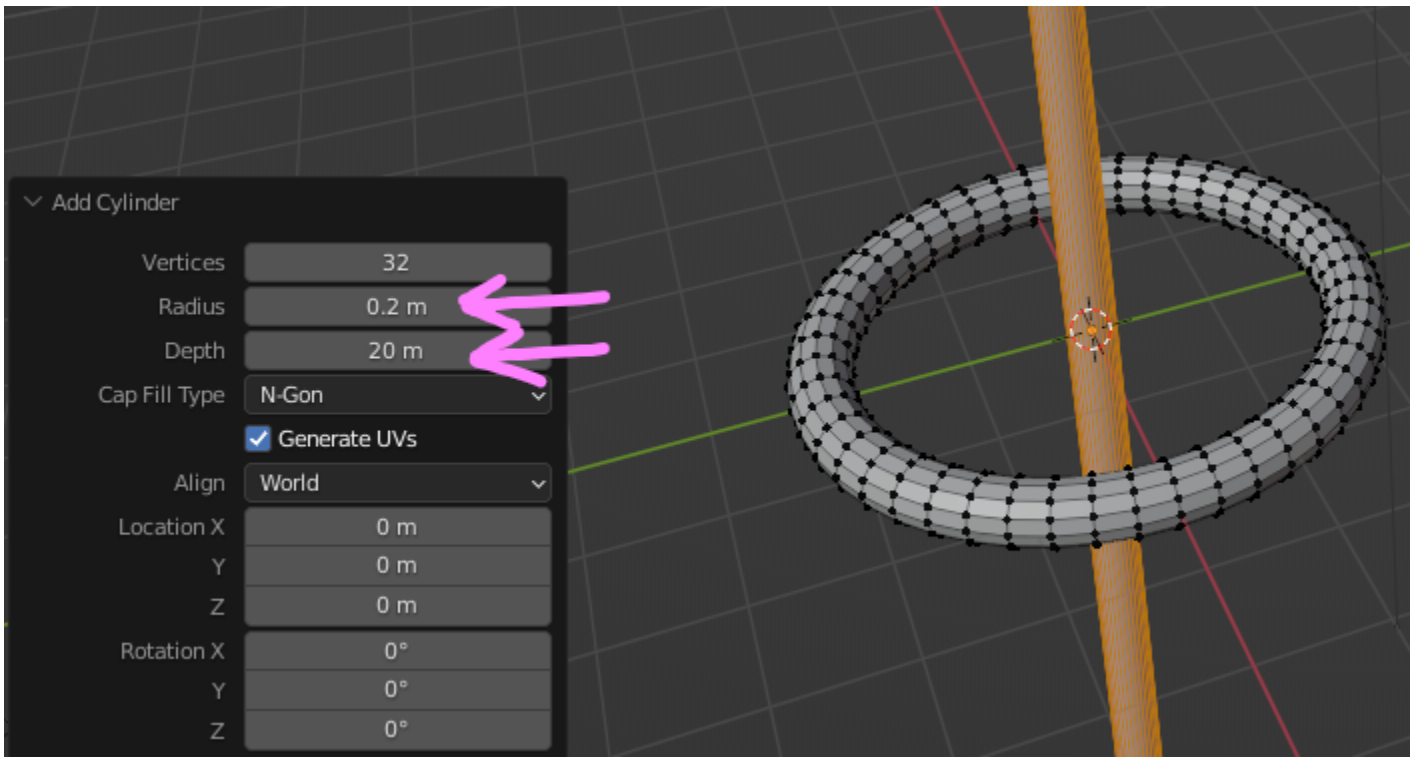
Learning to have the camera follow a path, track to an object, and using empties as targets for keyframes.
Learning to render an animation towards the end

Starting with a new scene, delete the cube and a torus (donut) mesh. Increase the outside radius of the torus so that we have a rage hoop type of shape

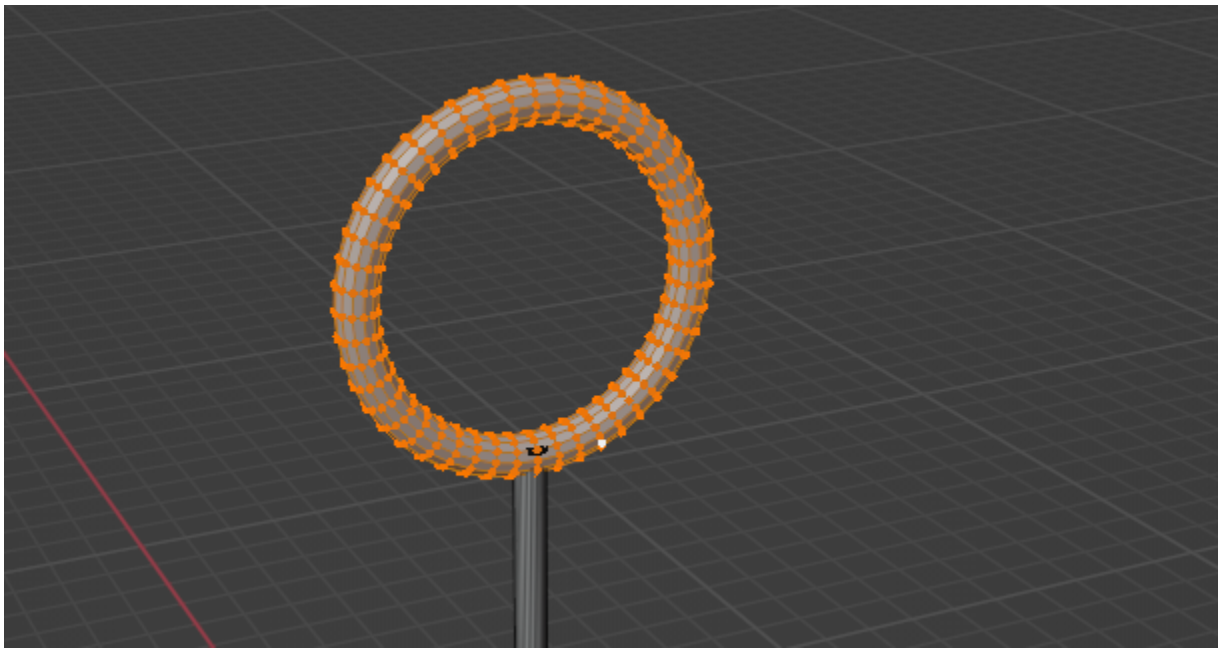


Tab into Edit mode for the torus and **deselect everything**
add a cylinder (note, we are adding the cylinder while inside edit mode for the torus)
Decrease the radius of the cylinder to 0.2 and increase the depth to 20



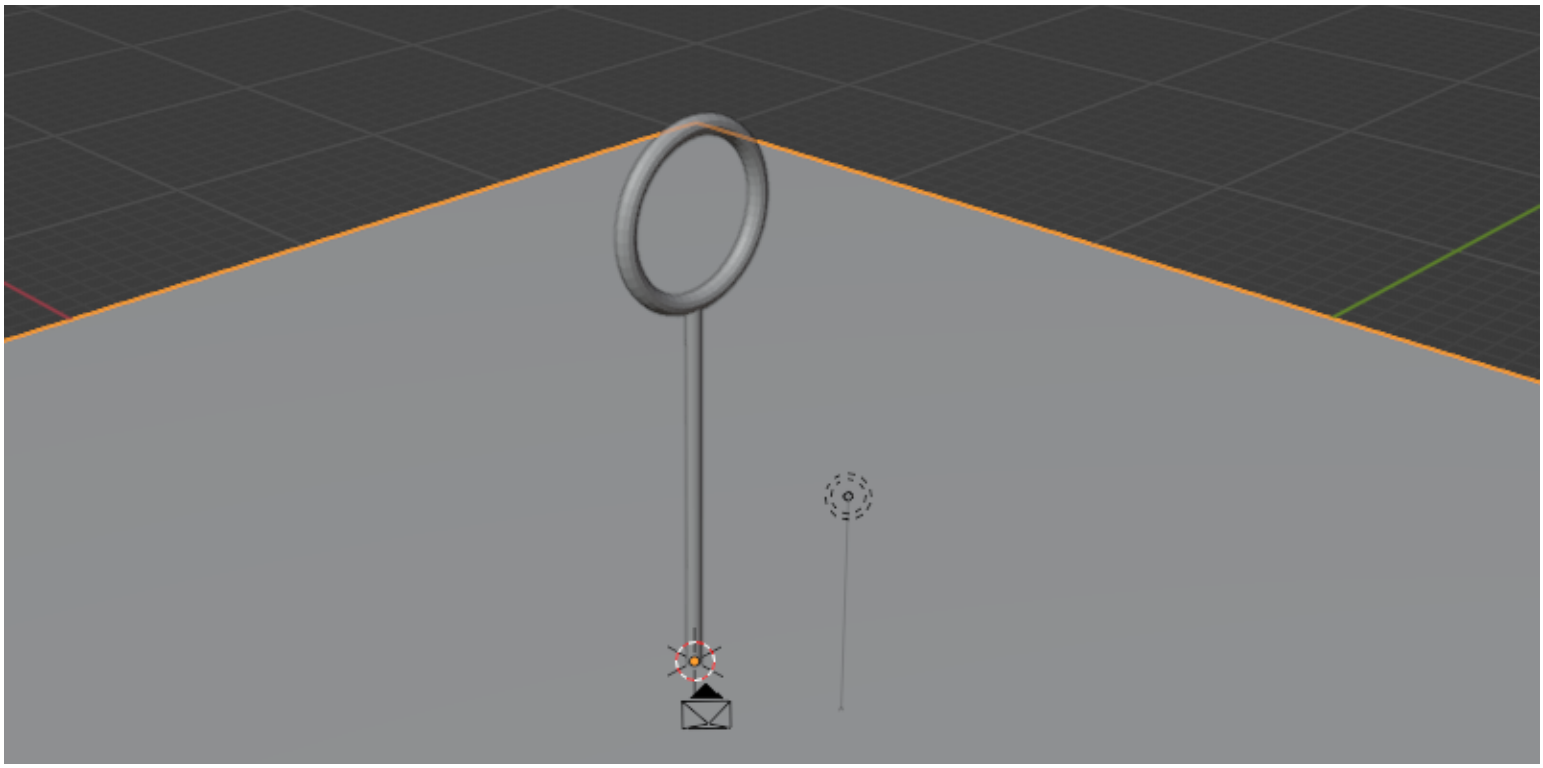
Click off of the cylinder now to deselect everything.
Since we have two separate objects in edit mode, we need to use an additional step when selecting one versus the other.
Select a vertex on the torus and then press the L key on the keyboard. This will select all of the linked vertices
We now have just the torus selected.

Rotate the torus on the y axis by 90 degrees(press the r key followed by the y key and 90)
Grab the torus and move it to the top of the cylinder (press the g key followed by the z key and move it upwards)

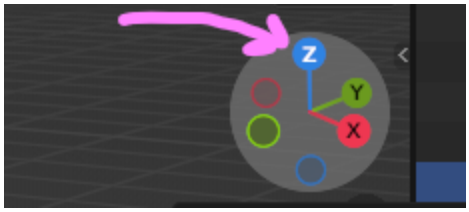


Tab out of edit mode and rename the torus in the hierarchy to Hoop

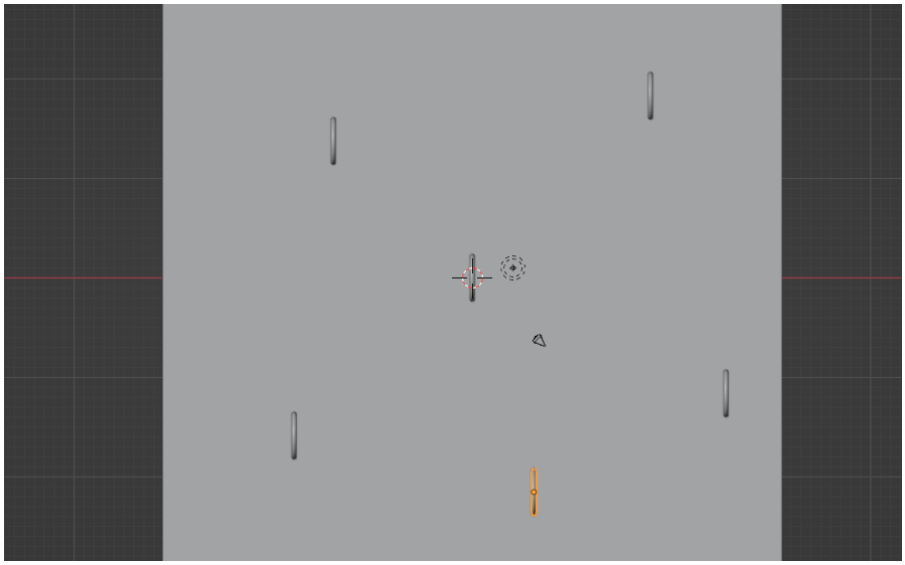
Next, let's add a floor plane to our scene
Shift + a → mesh → plane
Scale the plane so that it is nice and large relative to our hoop



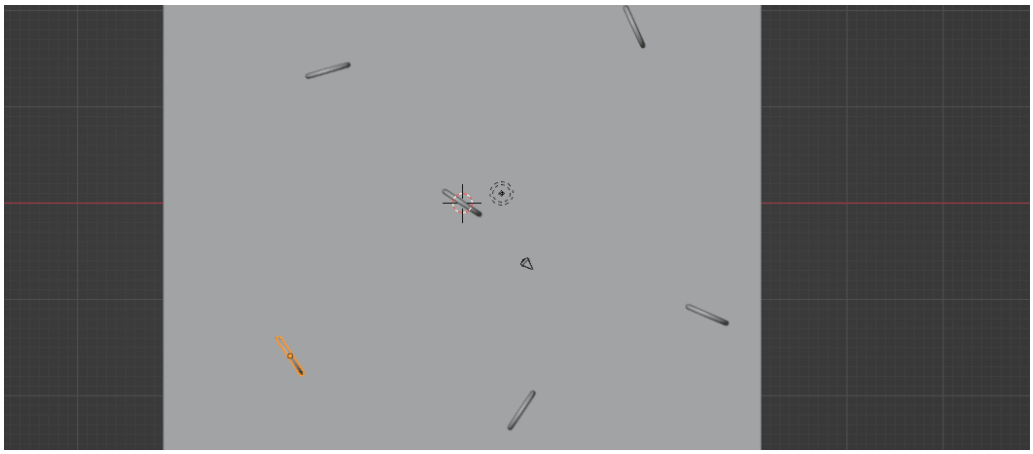
Next, view the scene from the overhead view looking down over our plane and hoop. Click the Z icon in the orbit gizmo to change the view



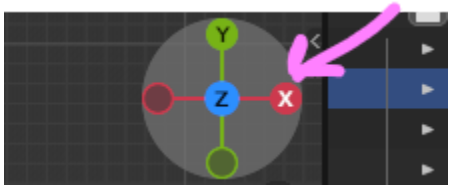
Let’s make some copies of our hoop object.
Select the hoop and duplicate it with shift + d
Place the new hoop a good distance from our original hoop. Repeat this 5 times scattering various hoops around our plane.



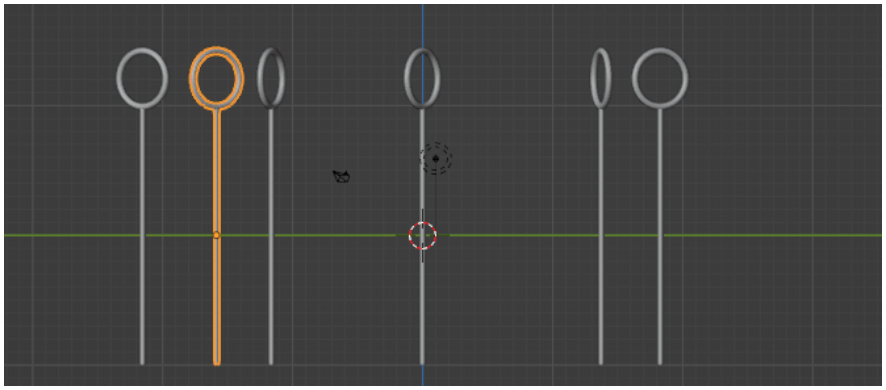
Next, rotate each hoop to a unique rotation on the z axis so that each hoop points in a different direction



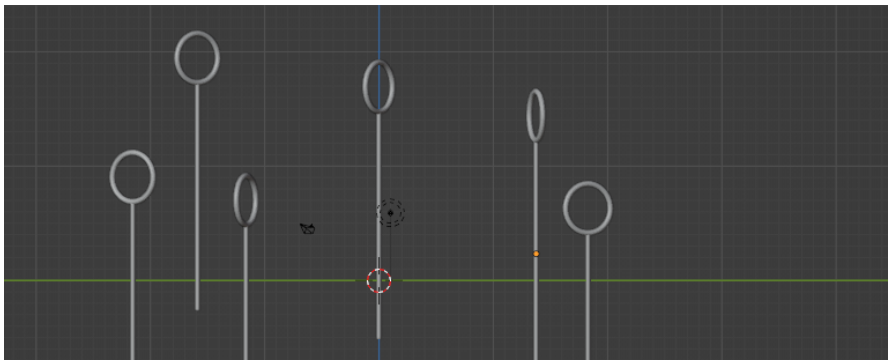
Finally, switch to the side view by clicking on the X icon in the orbit gizmo to change the view



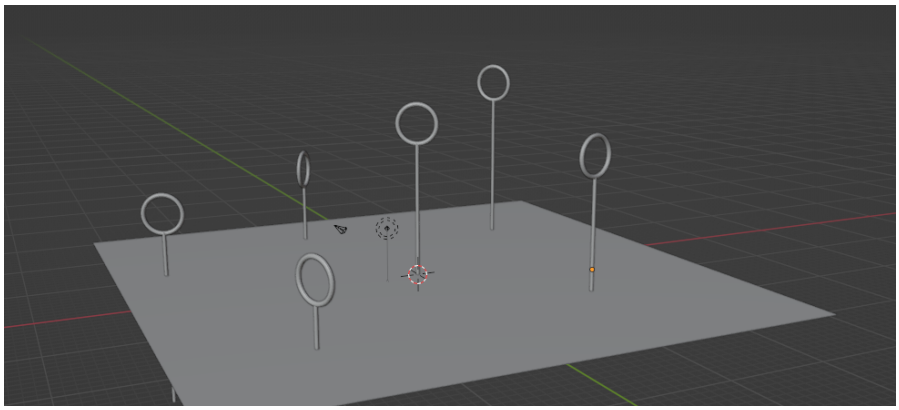
We can now see out hoops from the side view



Select each hoop one at a time and move each to a unique height. All hoops should remain above the plane.



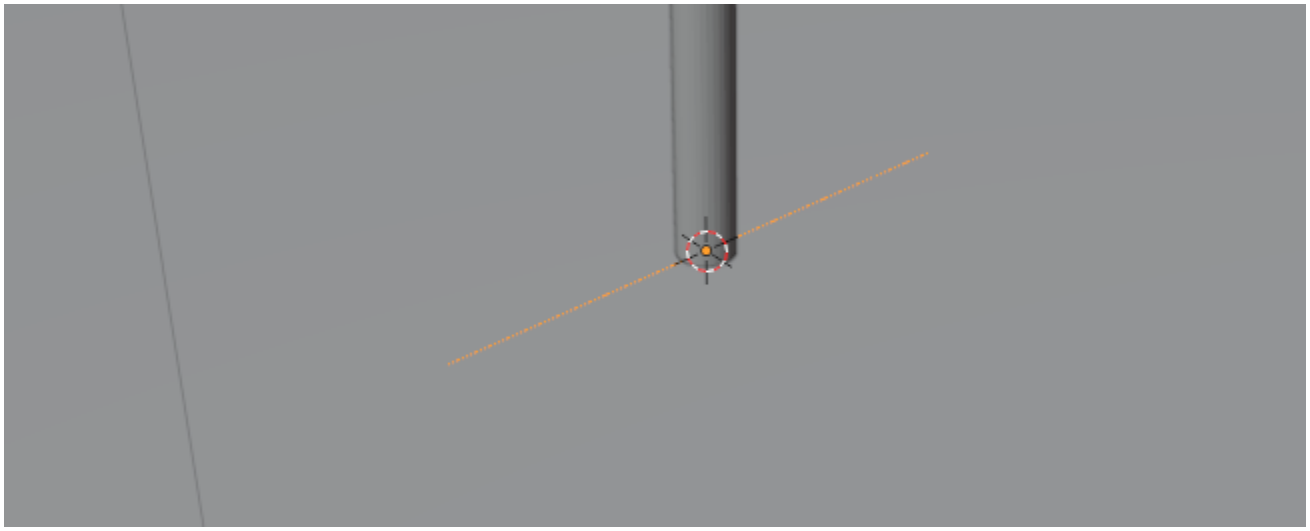
The scene should look something like this:



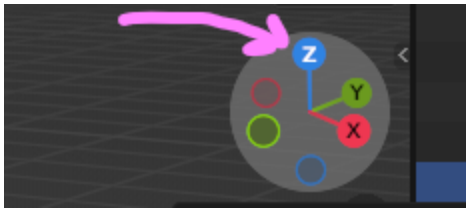
Next let’s add a path to our scene

Add → curve → Path

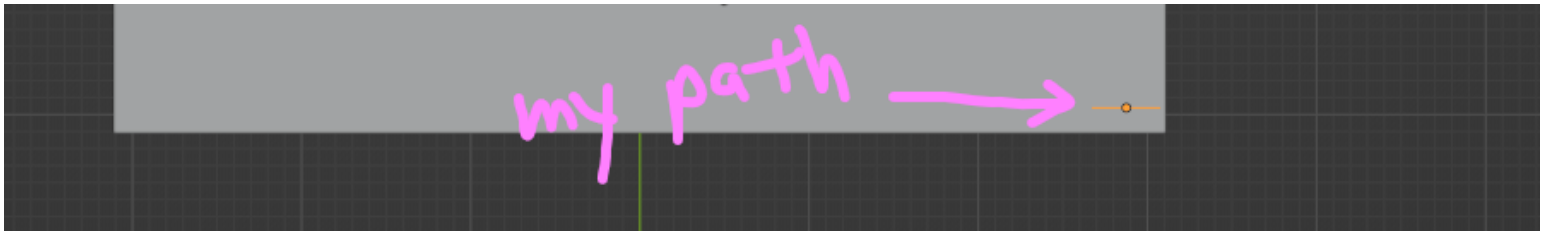
It might be difficult to see, but a path was added to the scene at the 3d cursor



Switch once again to the top view by clicking the Z icon in the orbit gizmo to change the view



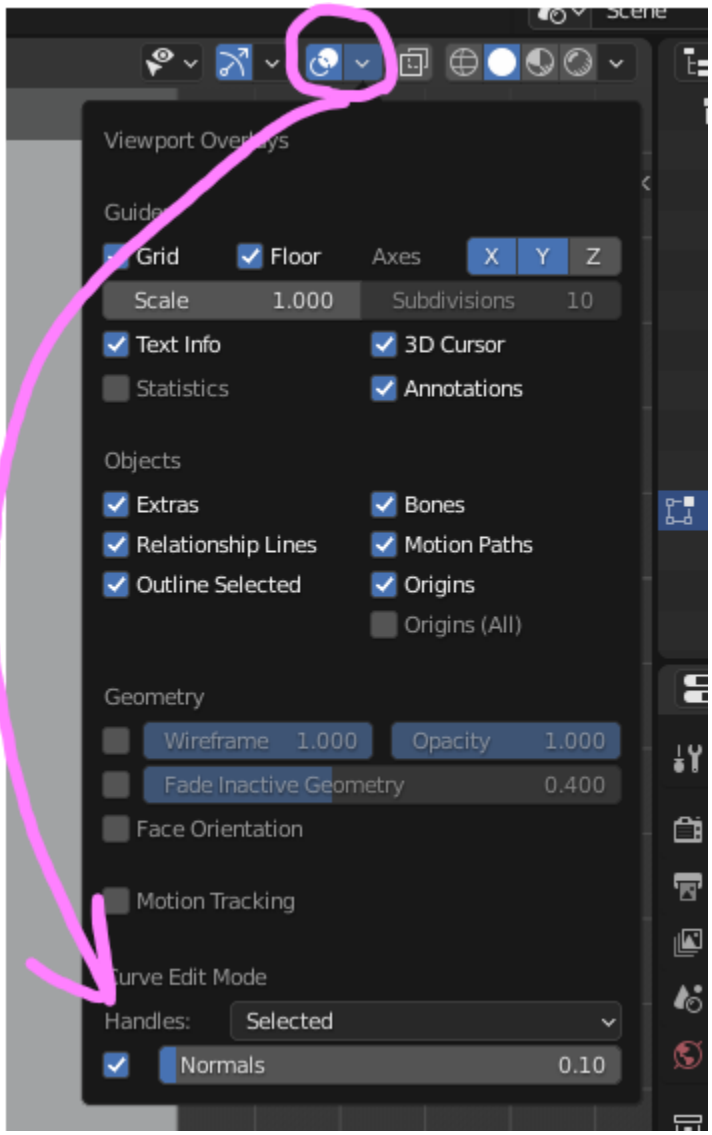
Placing the path curve will be easier from the top view.
Grab the path with the G key and drag it to the edge of the plane near the bottom right corner



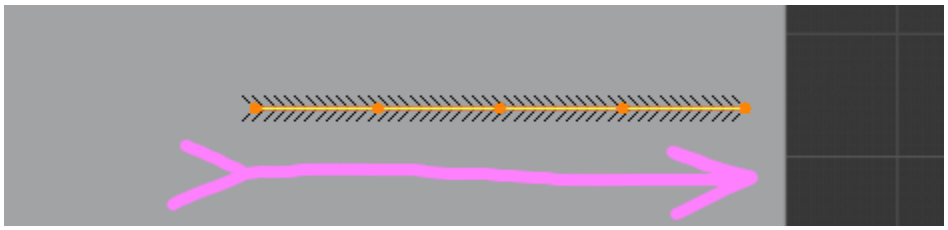
Enter edit mode with the path selected
The path now looks like it has a few vertices on it

Lets turn on the nirmals for the path so that we can see which direction it is pointing in.

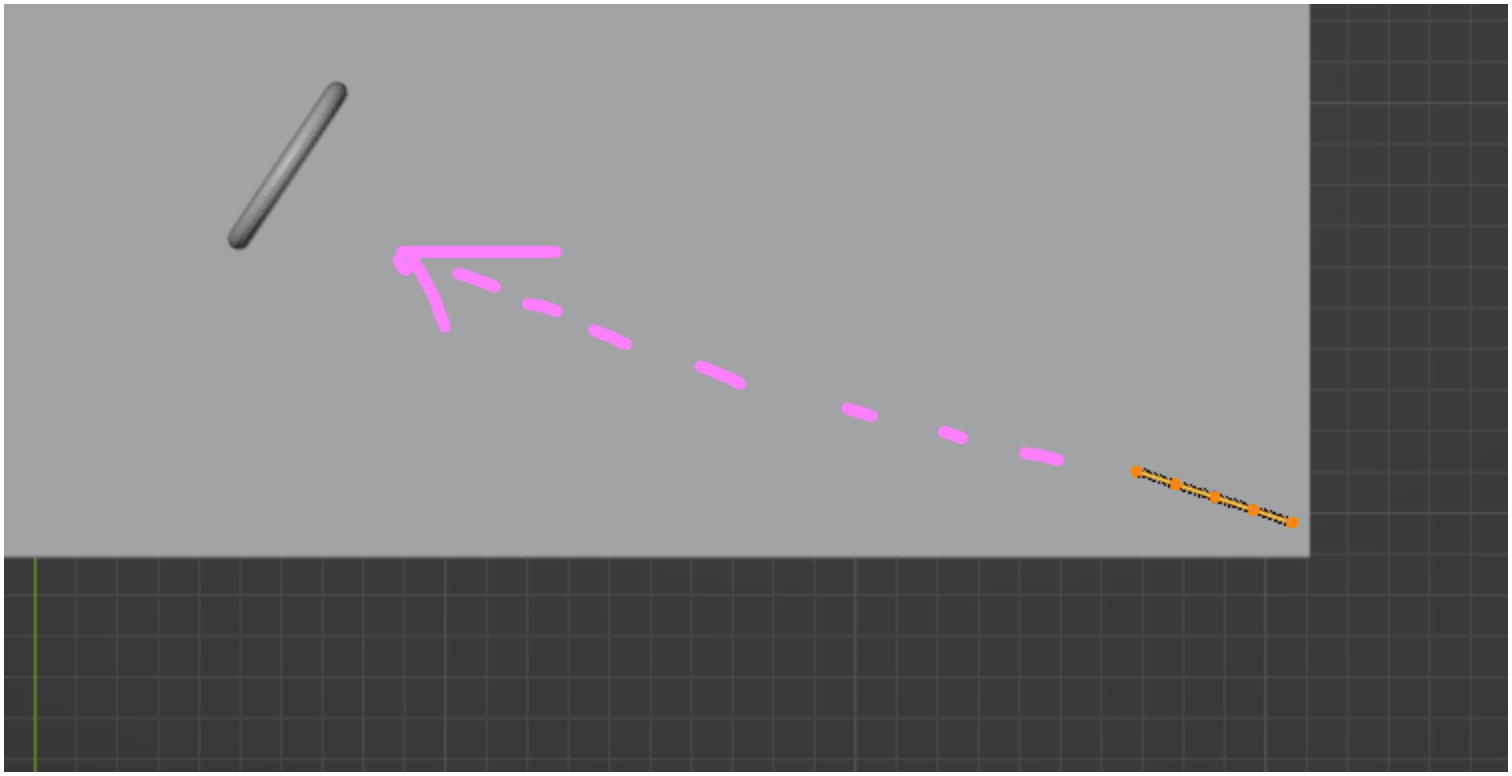
Click the overlay icon near the top right of the window and click the checkbox to enable the normals



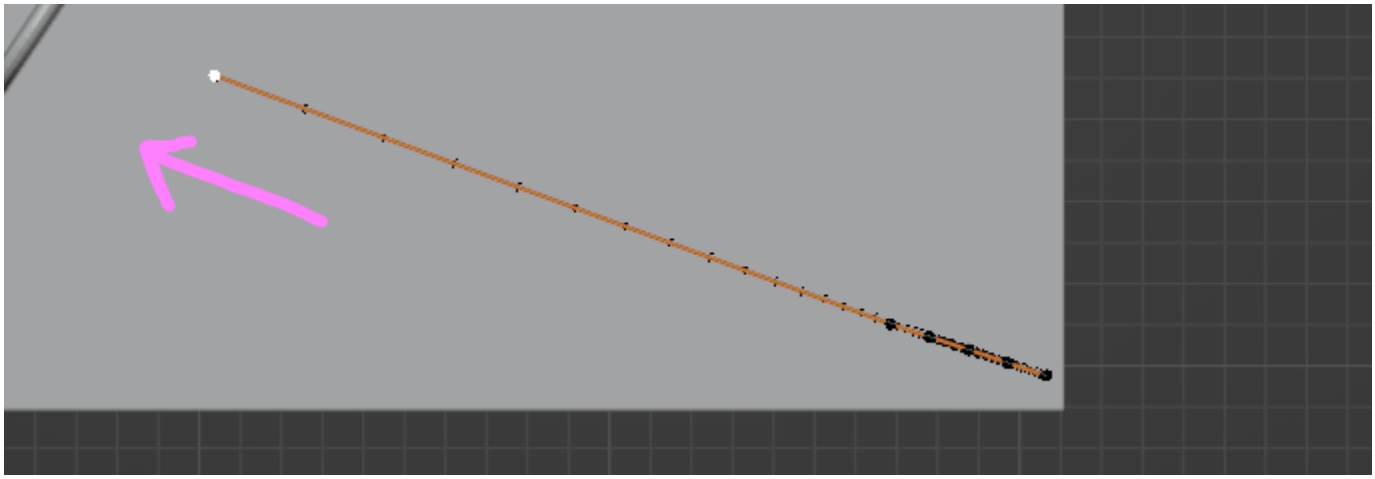
Our path now shows the direction that it is pointing in. Mine is pointing to the right



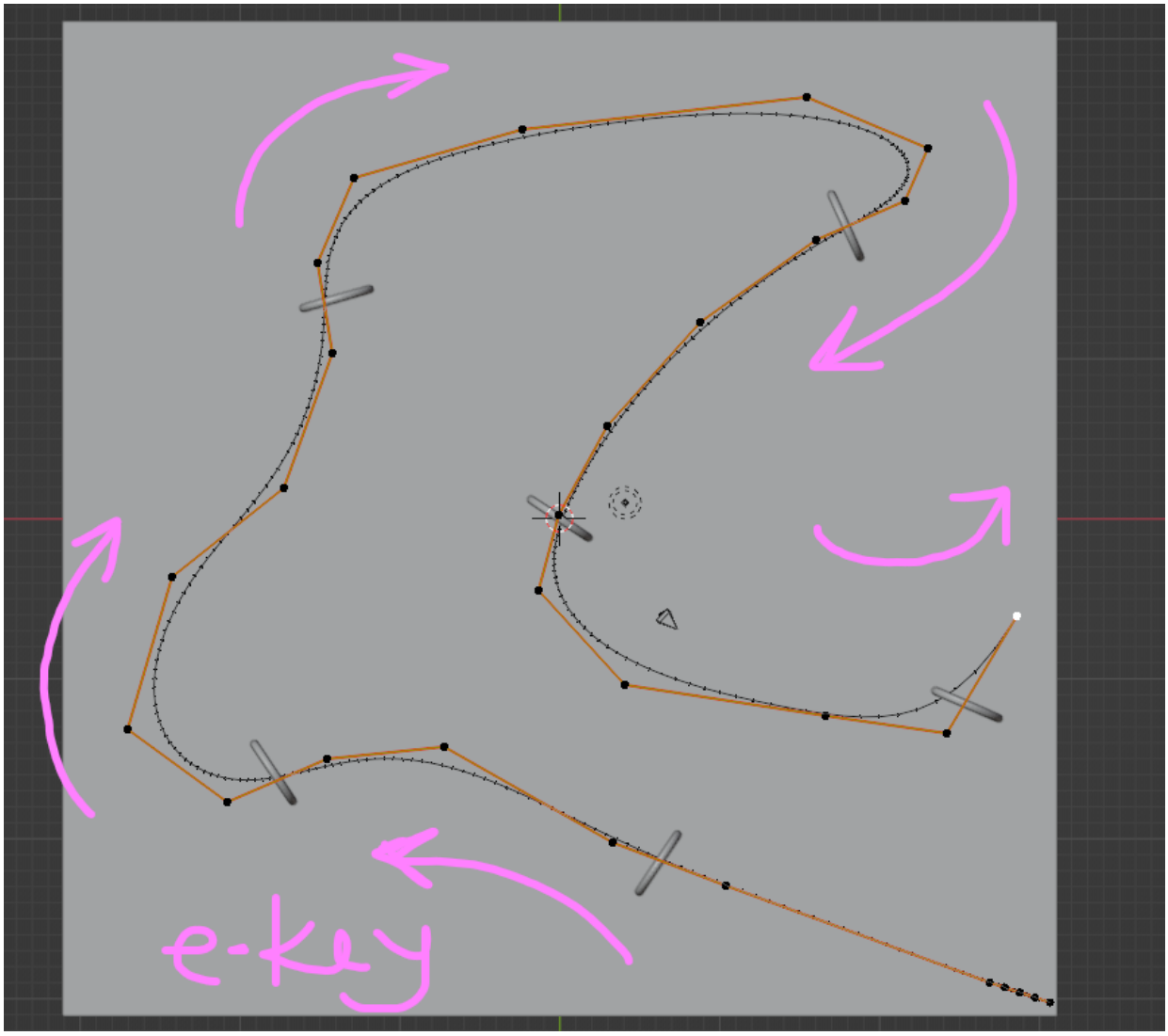
With all of the points of the path selected, rotate the path so that it points to the nearest hoop in your scene



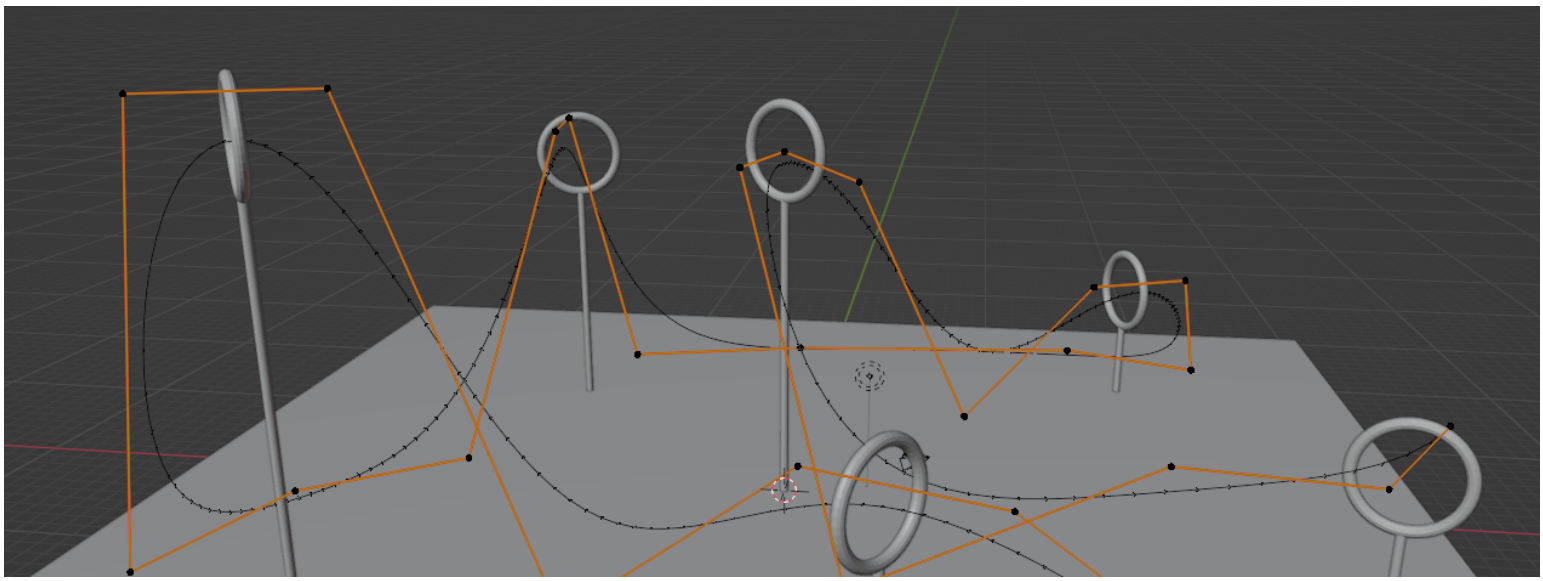
Next, click the first point of the path and extrude it with the e key towards the hoop.



Zoom out to fit your scene in the window. Note we are still in top view.
Extrude the new points additional times using the e key thereby weaving a line to all of the hoops.



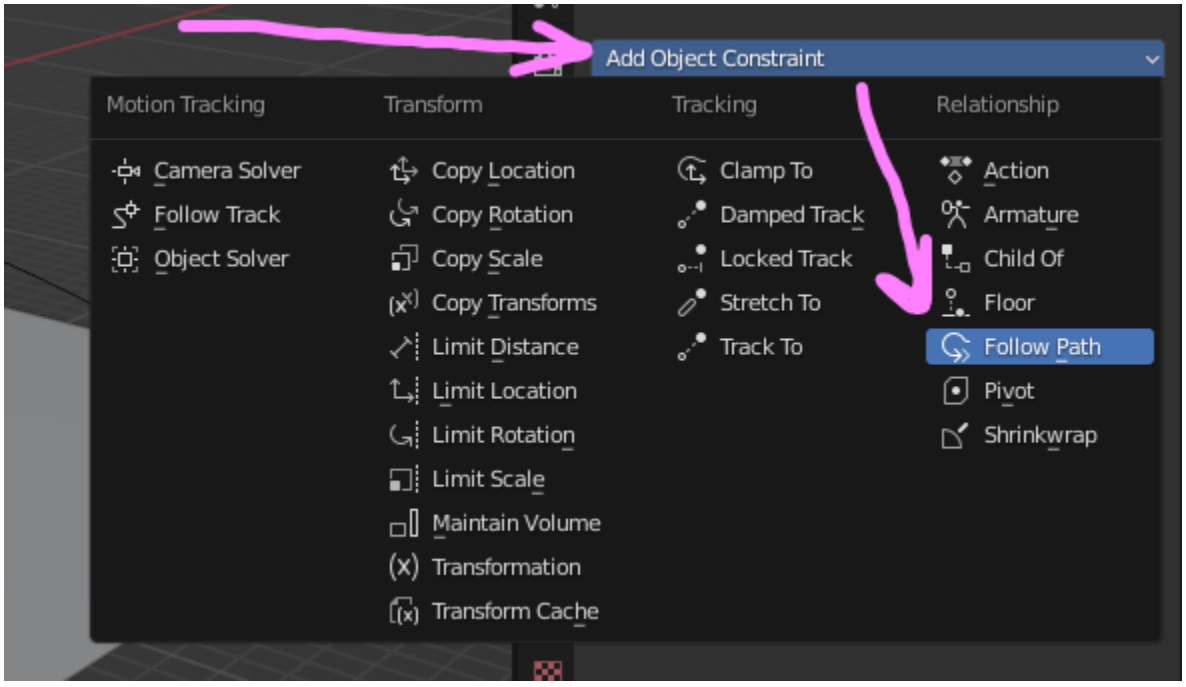
The path created now weaves a line to all of the hoops.
Orbit the scene a little and we can see that the path is completely flat along the plane.
Click the points in the path and move them upwards on the z-axis to make the line pass through the ring of each hoop.



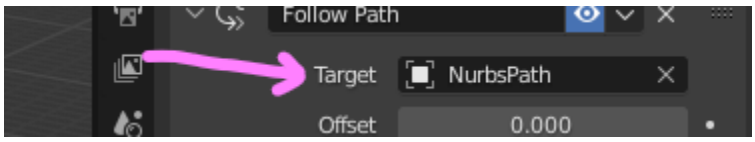
The path should look like a continuous thread that has been woven through the eyes of various hoops.
We will now make our camera follow this path
Tab out of edit mode and back into object mode.
Lets assign a constraint to our camera.
Constraints are sets of rules that we can add to objects to give them motion or limits

Select the camera in the scene.
With the camera selected, click on the constraints tab. It looks like a belt around two wheels

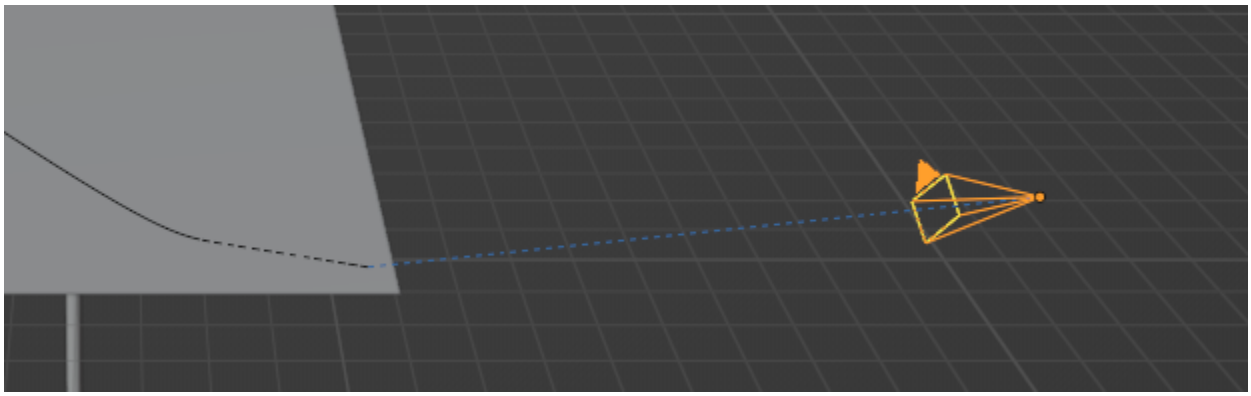
Add a “Follow Path” constraint to the camera



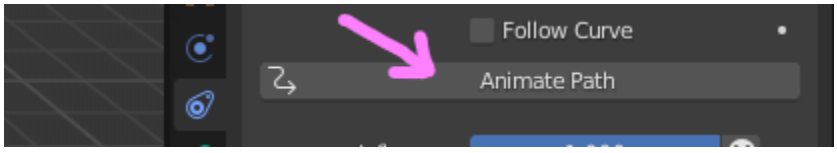
In the target setting assign the Path that we created a moment ago. It might be called NurbsPath



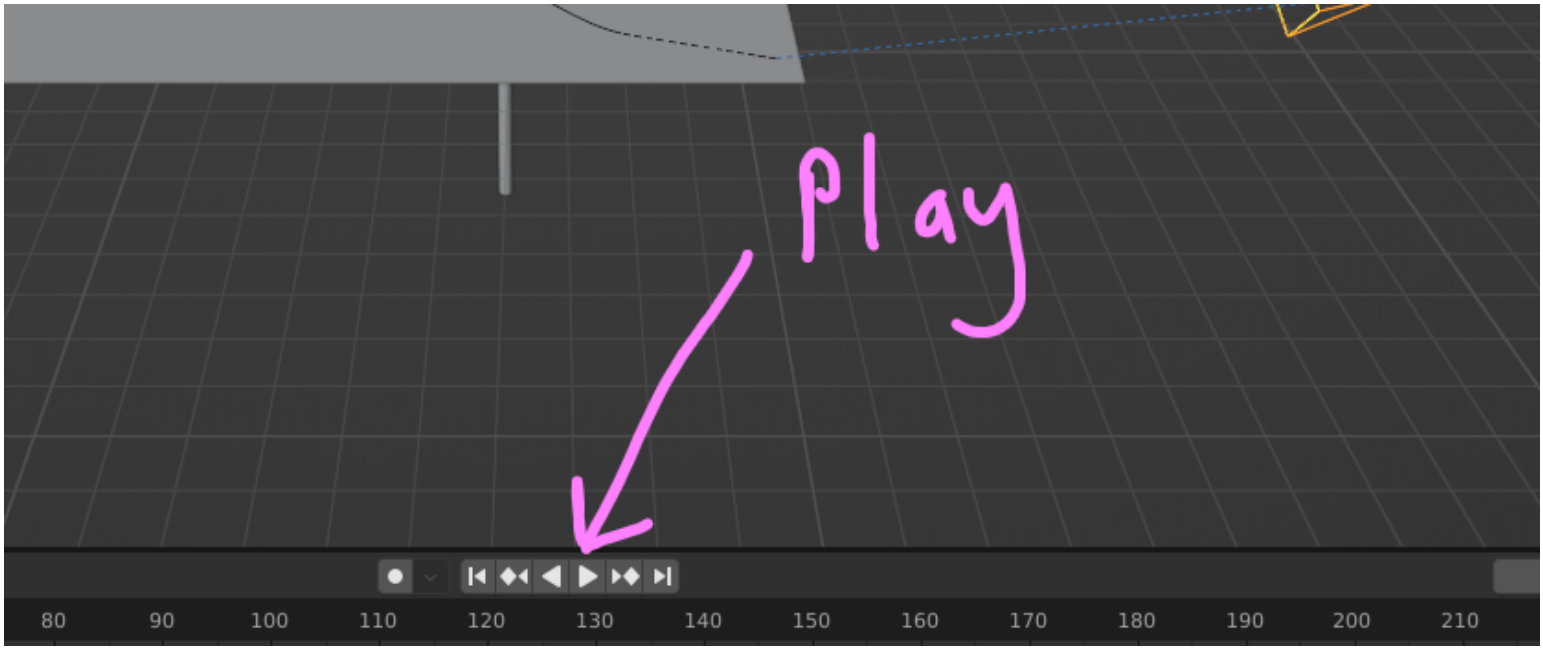
You will see that the camera has now moved to an area near the start of the path and a dashed blue line indicates that the camera follows a constraint



Click the animate path button



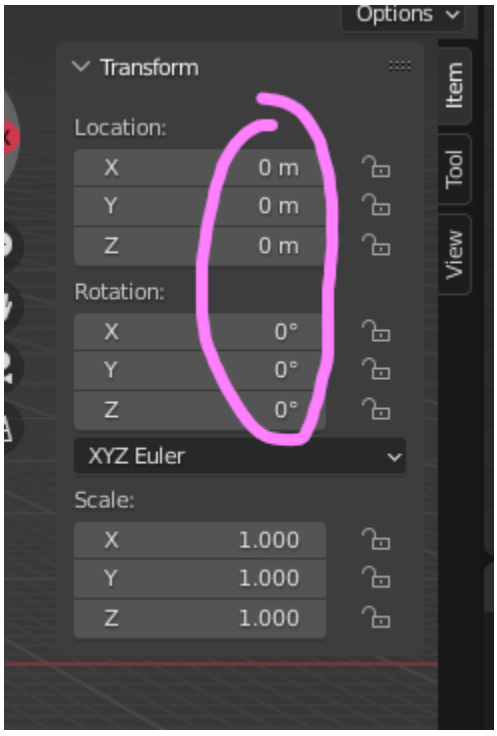
And press play on the timeline below the 3d view



You will see that the camera follows the path in an animated fashion. We will adjust the settings in a moment. First press the pause button to stop the movement. You may press the back arrow with the vertical line to jump back to the start frame.

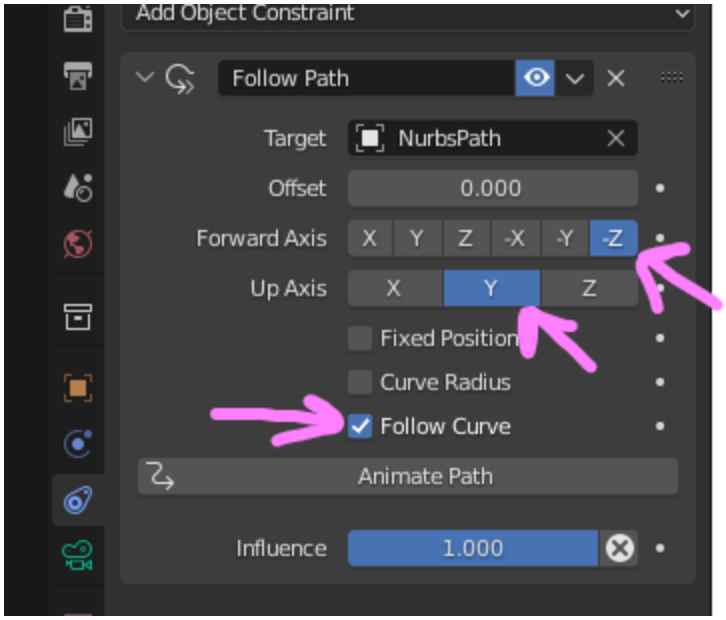
Let’s fix the offset that the camera has from the path.

To do this, reset the position and rotation of the camera by setting the cameras values to 0. Press the n key to see the values and set them to 0 for the position and rotation for the camera.

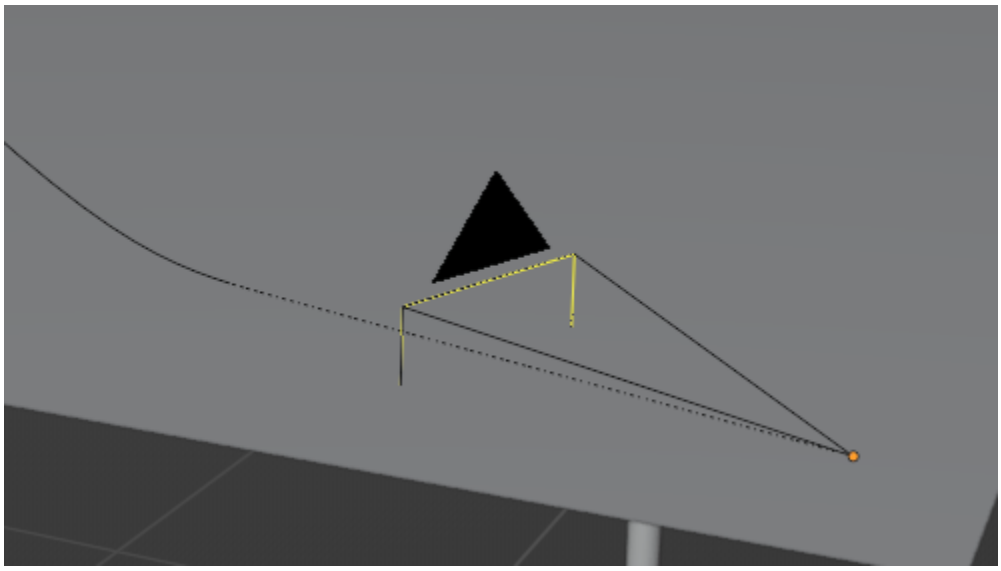


Now the camera is where it should be on the path, but it is pointing in the wrong direction. Let's have the camera face along the path as it travels down it.

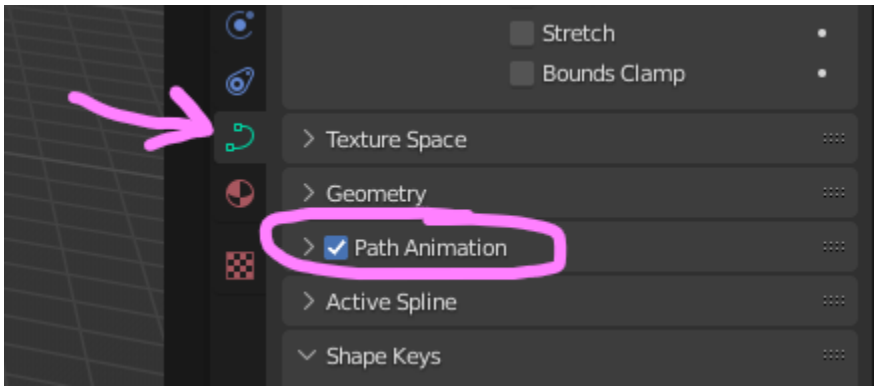
In the camera’s Follow Path constraint, click the “follow curve” checkbox and set the Up position to the camera’s y axis and the forward axis to the camera’s -z axis. See the settings below:



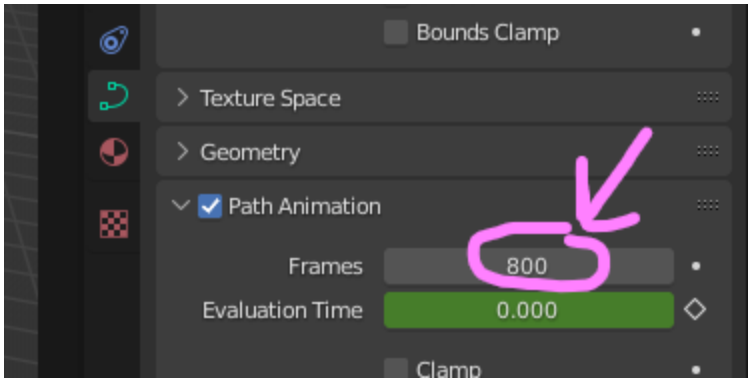
The camera now points in the correct direction along the path



Press the play button and watch the camera travel along the path with the position pointed correctly.
There are a few things that we want to fix.
The camera is moving too fast along the path. Let’s slow it down
Because we animated the follow path constraint, now our curve has some animation properties assigned to it too.
Click the path curve and click the green curve properties icon in the properties area.



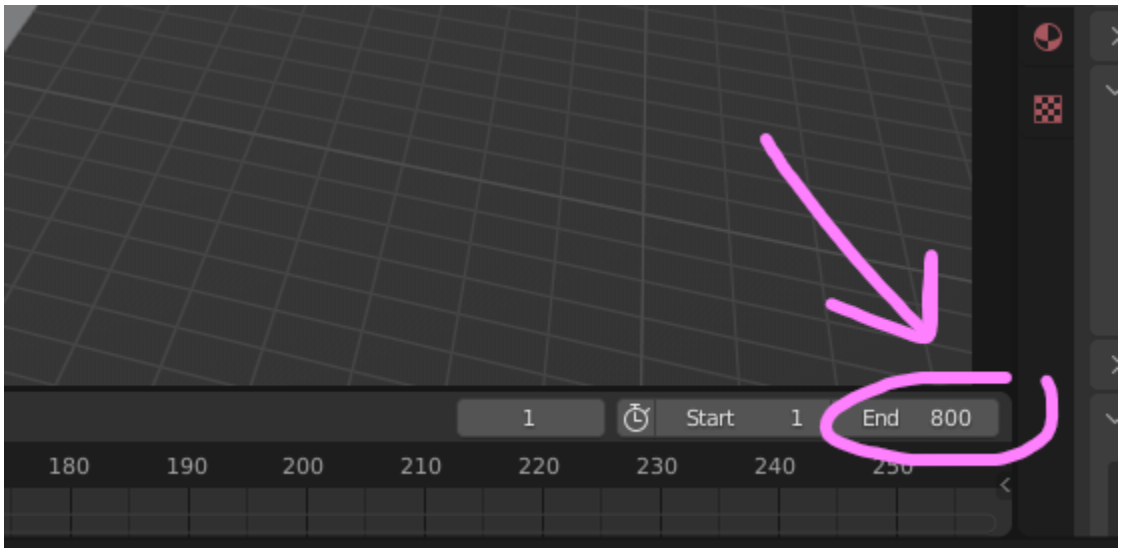
You will see the checkbox in the path animation setting. This is thanks to the constraint we added to the camera earlier.
Open the Path Animation panel. Change the Frames value to 800. This is how many frames our follow path constraint will last.



With the frames set to 800, replay the animation. The camera now moves slower along the path. It completes the journey along the path in 800 frames.

We need to increase the number of frames visible in out preview as well. The default value of 250 frames is not enough to watch the entire 800 frame animation, therefore we must increase the range of the animation preview window.

Set the end frame to 800 in the animation preview area below



We can now preview the full 800 frames of the animation

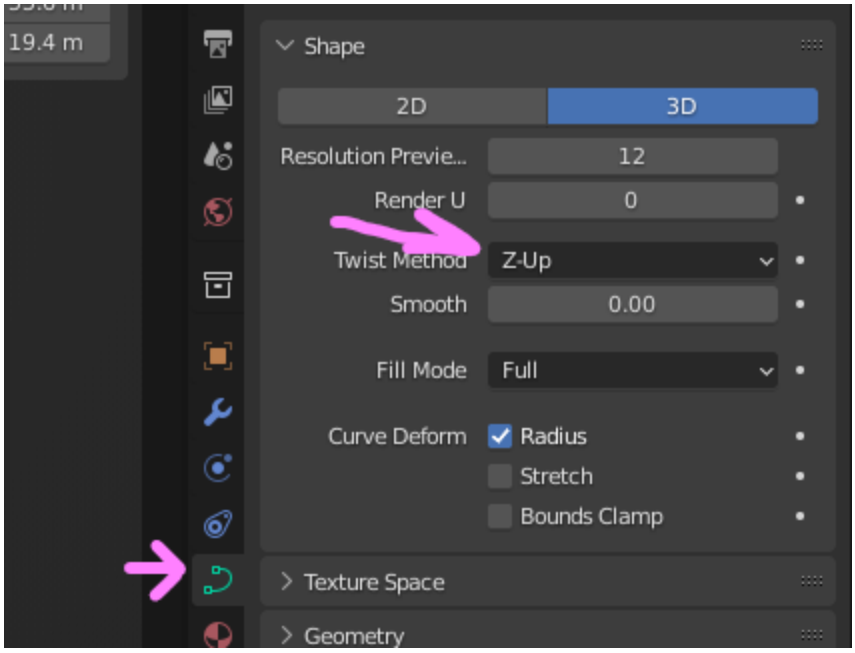
Watch your camera run through the full path a few times.

Switch to camera view to watch the animation from the perspective of the camera, like a roller coaster.

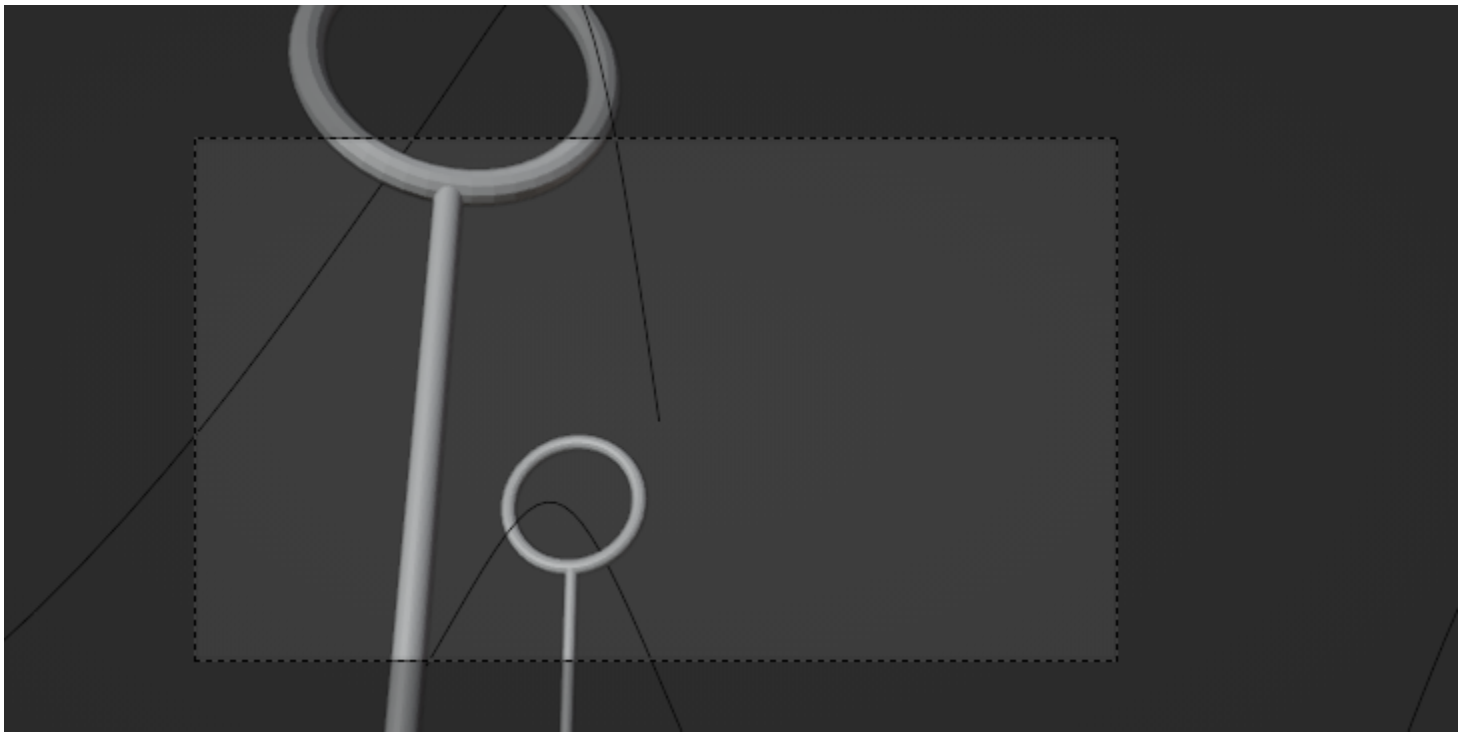
You may notice that the camera twists and turns in directions that place the view upside down or sideways.

This rotation information is contained within the path. One easy fix it to enable the Z-Up twist method in the curve shape properties.

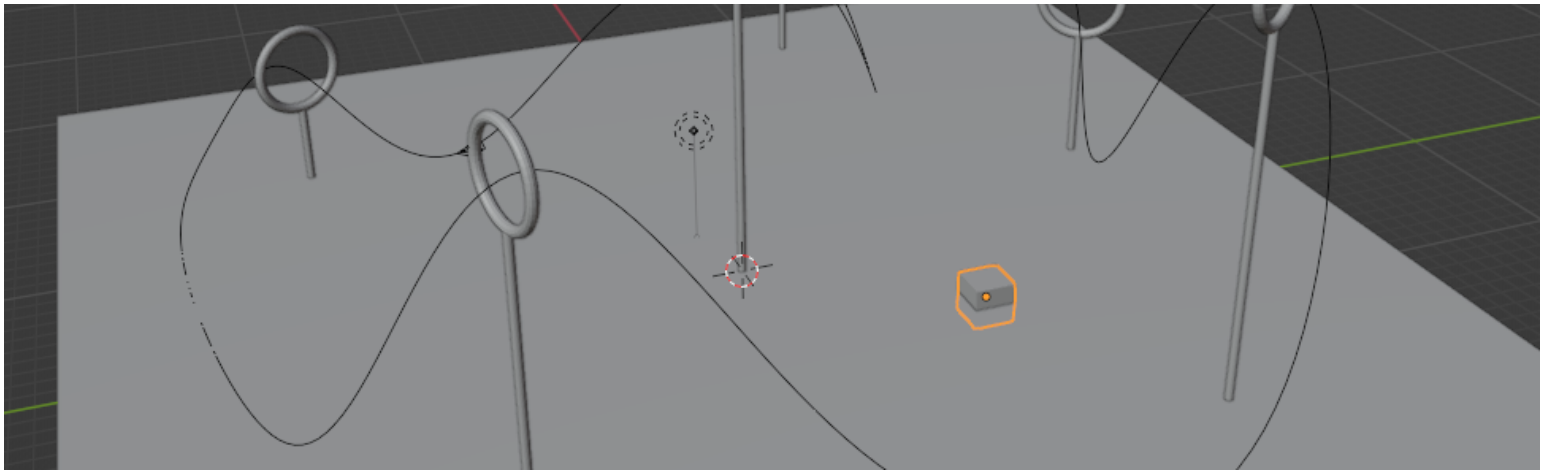
Click the twist method and change it to Z-up



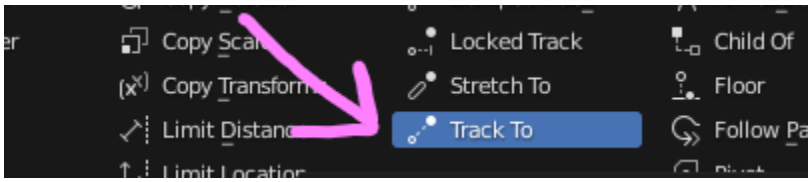
Now the camera z axis always points upward, preventing it from twisting upside down.



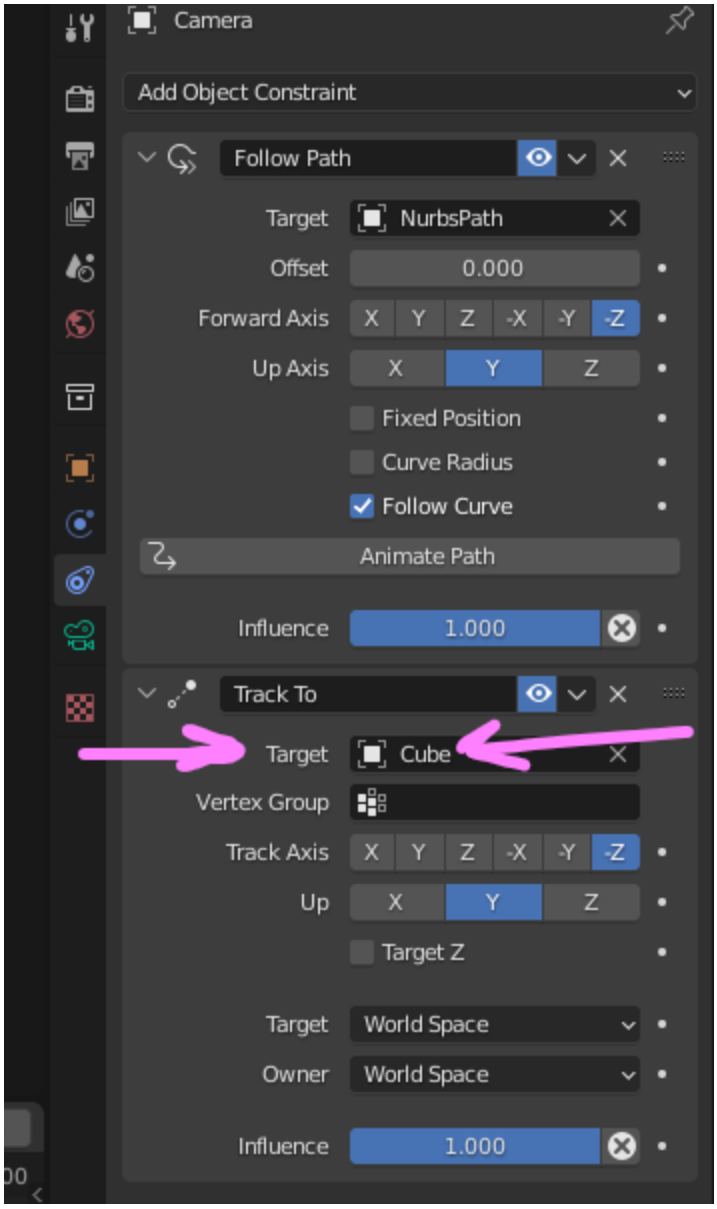
This is a good moment to take a break and save your work.
Next, let's explore how we can point our camera at an object.
Add a cube to the scene and place it somewhere on the floor plane of the scene amongst the base of the hoops



Let’s have the camera always pointing to this cube
Select the camera and add another constraint from the constraints menu
This time add a “Track To” constraint.

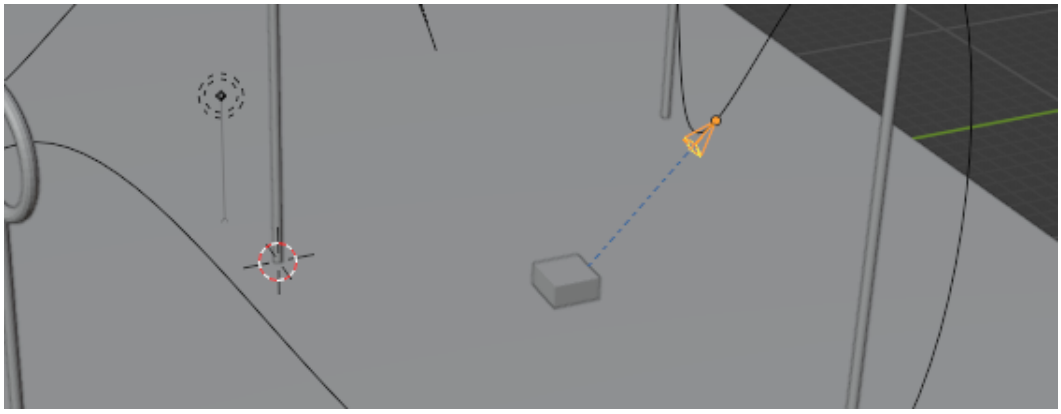


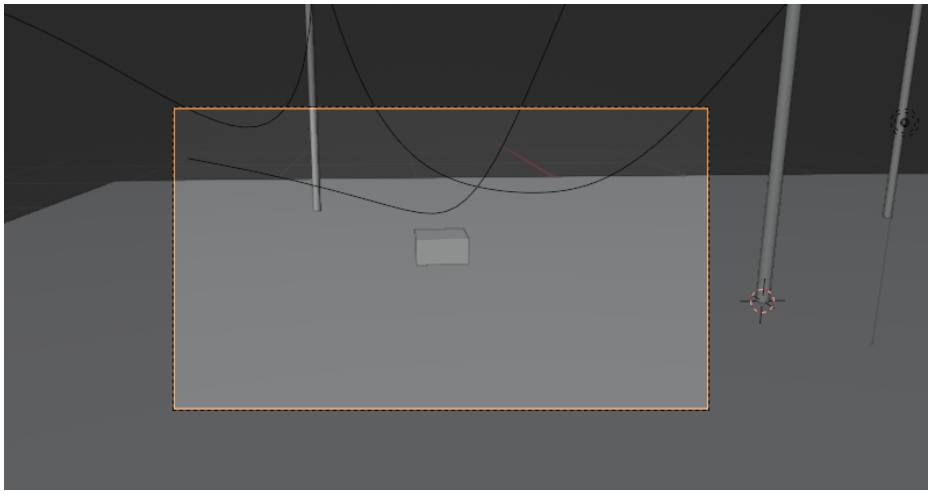
Note that the camera will now have two constraints, the “Follow Path” from earlier and now the “Track To” that we just added.



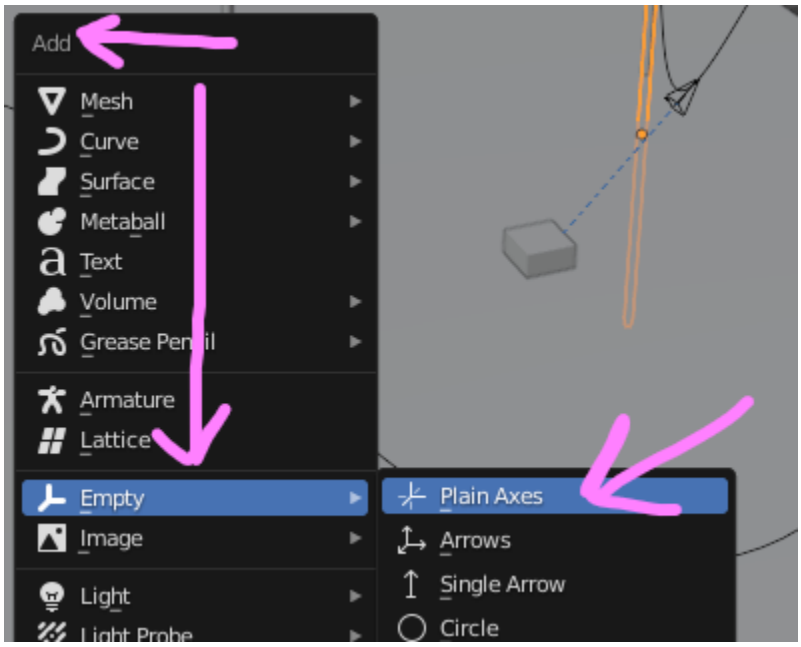
The camera now has a blue dashed line indicating that the constraint is applied. Also, the camera now always points at the cube.

Try playing your animation now and watch the camera always track to the cube.

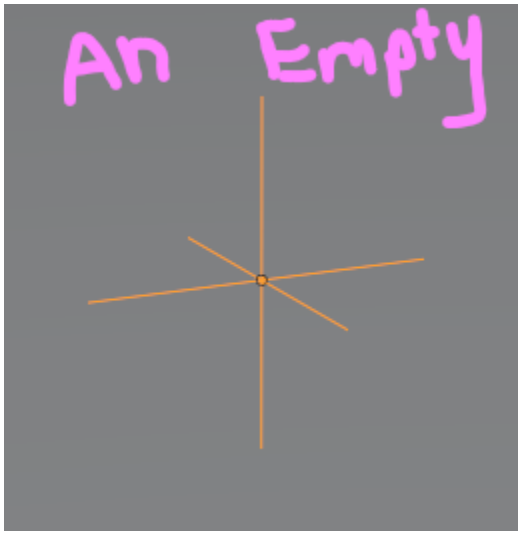




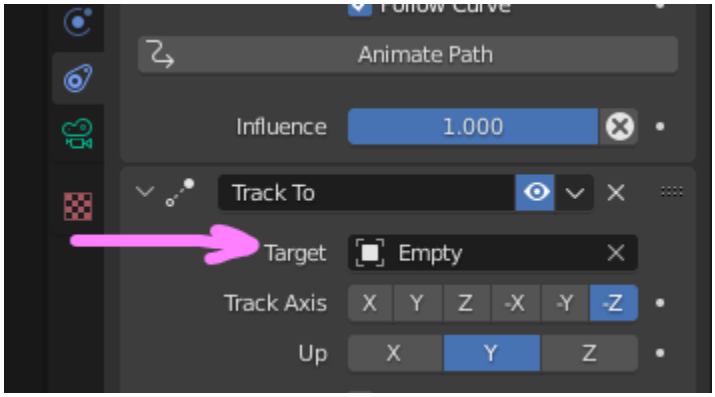
Next let’s implore how an empty can be used as the tracking target rather than the cube
Pause the animation if it is still playing.
Add an empty to your scene:
Add → Empty → Plain Axes



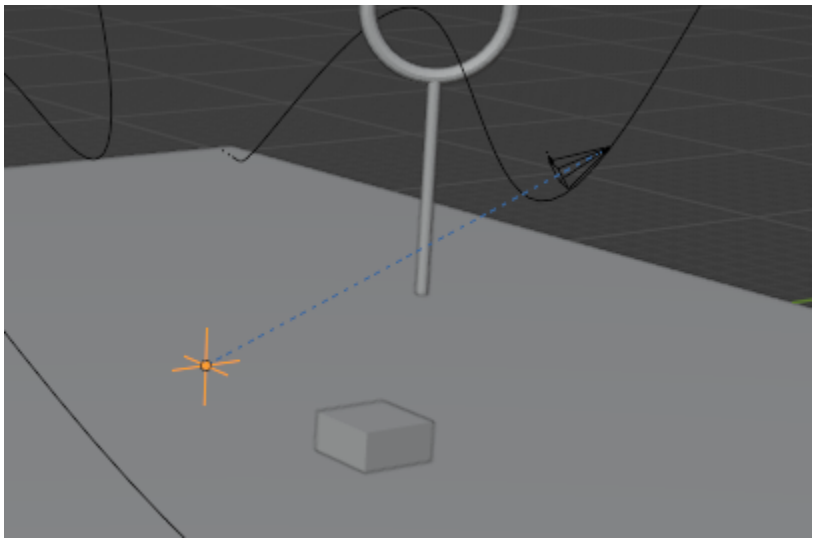
The empty object is now in the scene at the 3d cursor. The empty looks like a plain set of axis lines.
Essentially an empty is what it sounds like: an empty object. It doesn't get rendered and is just a point object with a position, rotation, and scale assigned to it. It can be used for a variety of things.



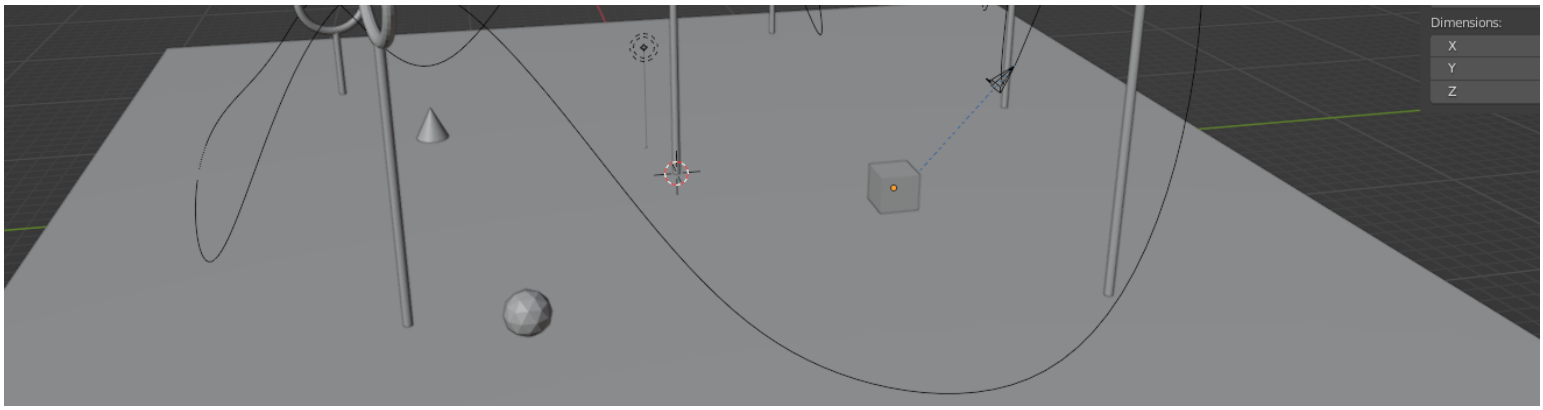
Let’s assign the target of the “track to” constraint currently on the camera to always point at the empty instead of the cube.
In the constraint set the target to empty



Now we can set the empty to any particular place we would like our camera to point. This makes it much easier to frame shots during an animation when everything is moving

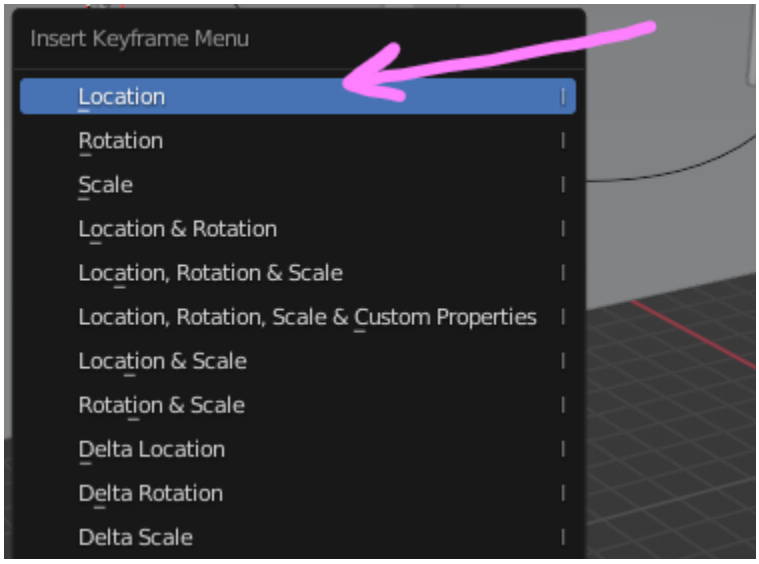


Move the empty so that it overlaps with the cube and the camera is now pointing at the cube again.
Lets add a few more shapes to the scene
Add a Cone to the scene and add a ICO Sphere to the scene. Place them on the floor plane.

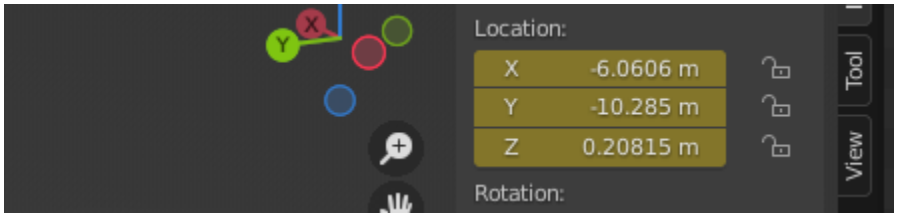


Lets animate the empty using keyframes. The idea is that we will have the camera point towards each of the objects at different times during the animation.
Lets start with the cube.
Place the empty directly on the cube.
Move the animation slider at the bottom to frame zero.

If the full 800 frames are not visible in the animation area, use the middle mouse wheel to zoom and click and drag the middle mouse wheel to pan the view.
The time marker should be at 0 (zero)
Select the empty
Press the “i” key on the keyboard to add a location keyframe. Note your mouse must be hovering over the 3d view for the menu to appear when pressing the i key

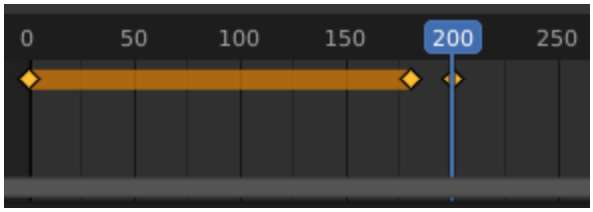


We have now ascribed a location to the empty at frame zero of the animation.
Next move the time slider to frame 180. Once again insert a location keyframe for the empty at frame 180 by pressing the “i” key on the keyboard.
When a keyframe is inserted you will see the keyframed values appear yellow

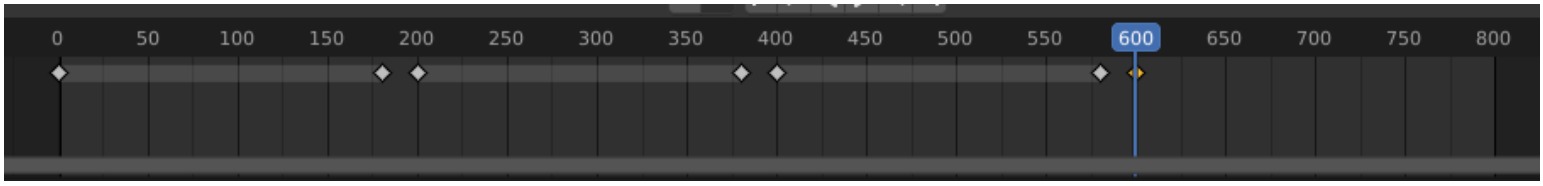


The timeline will also show a small diamond icon indicating that a keyframe is inserted. The highlighted line between the two keyframes indicates that nothing has changed between them. In this case, the location of the empty hasn’t changed between frame 0 and frame 180

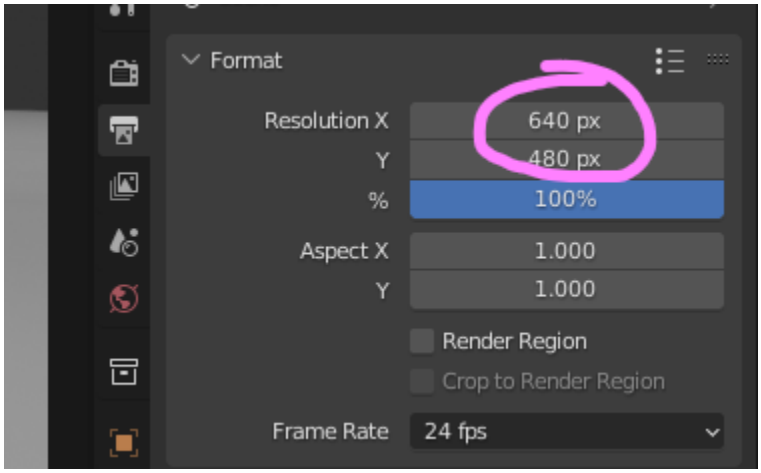
Next, move the time slider to frame 200.
Now move the empty from the cube over to the cone. The empty should be directly on the cone. Insert a location for the empty at this frame (200) by pressing the “i” key and selecting location from the menu.
We now have a new keyframe at frame 200. This time the position of the empty changed and we don not have the yellow highlighting between the keyframes



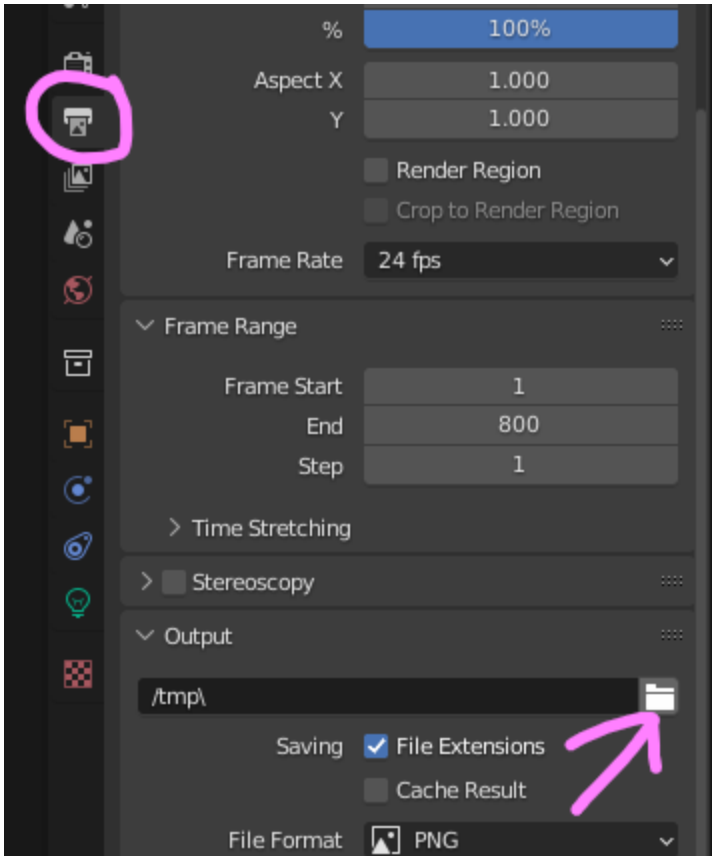
If you place a keyframe in the incorrect spot, click the keyframe and grab it with g to move it to the correct spot on the timeline. Keyframes can be deleted by clicking the diamond icon and delete key
Next move the time slider to frame 380.
Insert another location keyframe for the empty at this point by pressing the “i” key.
Next move the time slider to frame 400.
Also move the empty so that it is directly on top of the ICO sphere. Once the empty is correctly placed, insert a new location keyframe for the empty at frame 400.
We are almost finished, just a few more keyframes to add
Move the slider to frame 580.
Add another location keyframe to the empty at this frame by pressing the “i” key
Move the time slider to frame 600
Move the empty so that it sits directly on the cube again
Insert a location keyframe for the empty at frame 600
We are now finished inserting keyframes!
Your timeline should look like this:



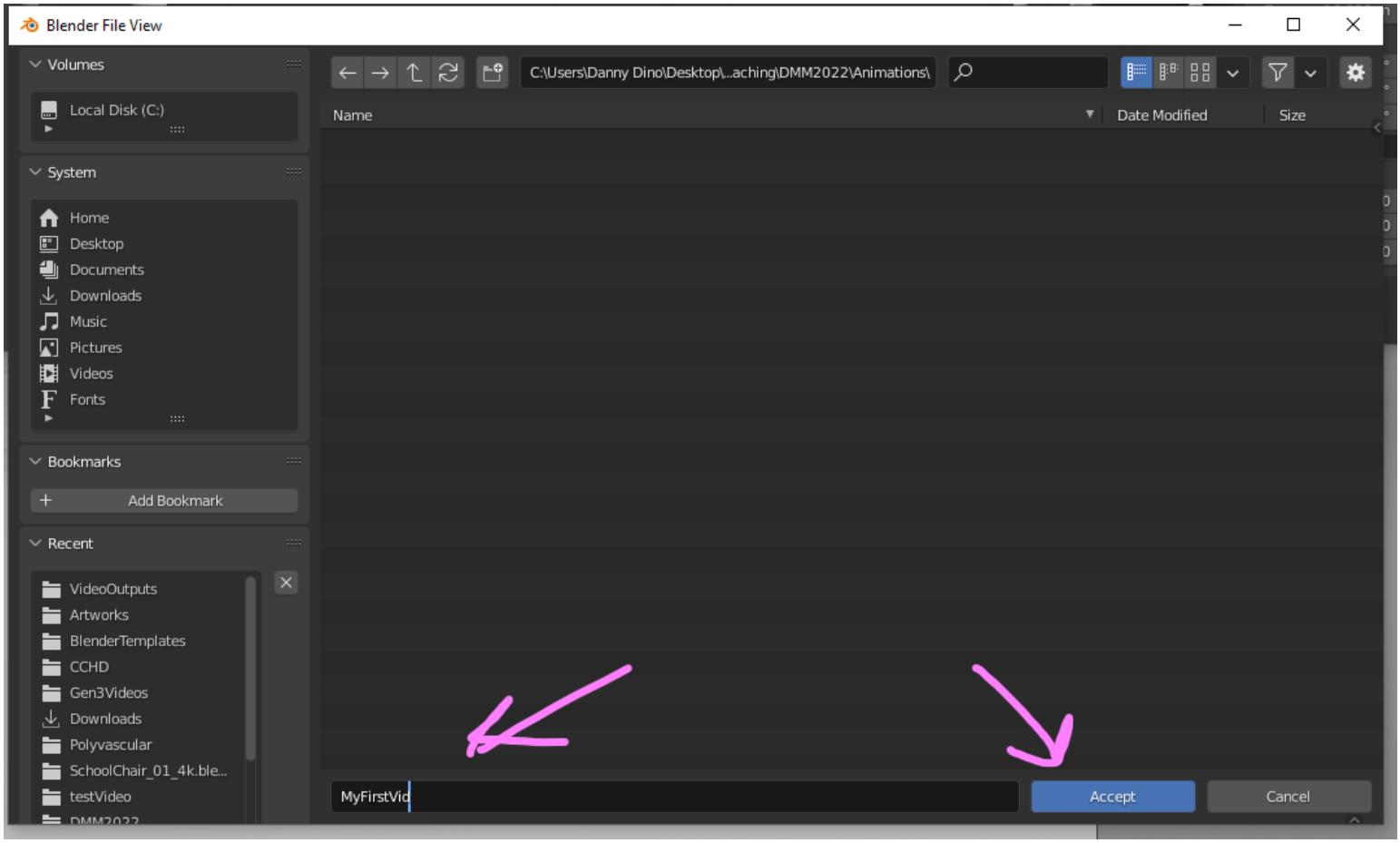
Finally, preview your animation by pressing the play key. You should see your camera point to the different shapes as it follows the empty object keyframed at various locations around the scene.
Watch your animation play from the camera view to see how the movement looks.
—
Rendering out the animation
We will be using the realtime rendering engine EEVEE to render the animation. This is faster for our first time doing it.
Set the render resolution to 640 for the x and 480 for the y. 640 x 480 is a small size that is good for low quality videos.



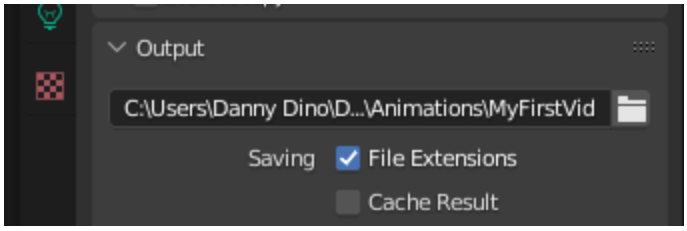
Next, select a location for the rendered video in the output panel
Click the folder icon where it says /tmp\



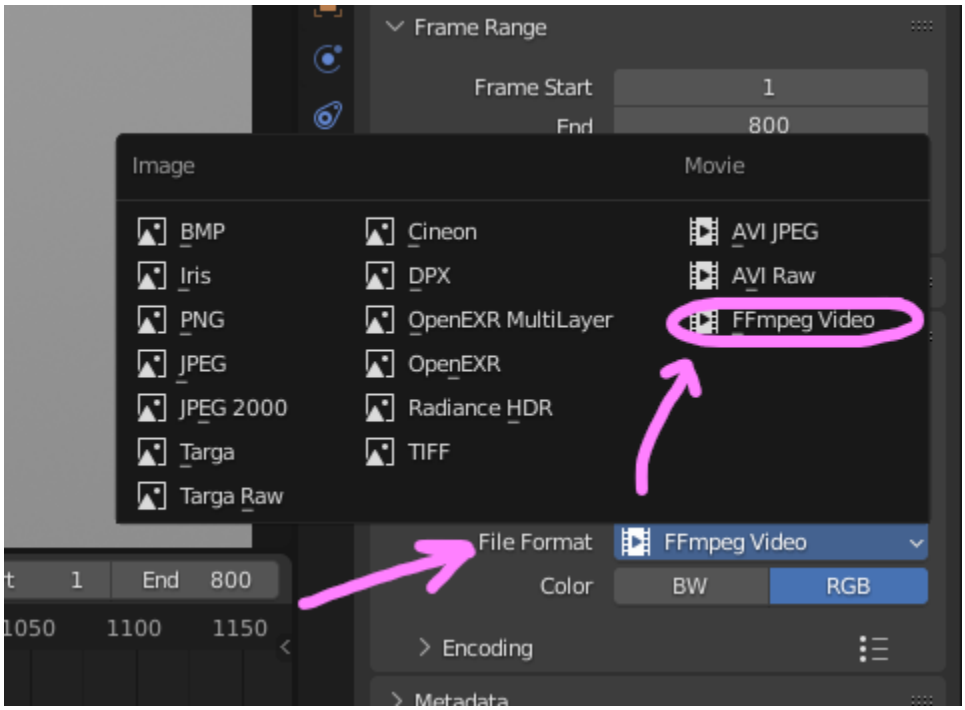
And select a location in your file system where the file will be saved and with the name it will be saved.
Notice that we are not putting any file extensions in the name of the saved output



The output field will now reflect the new save location



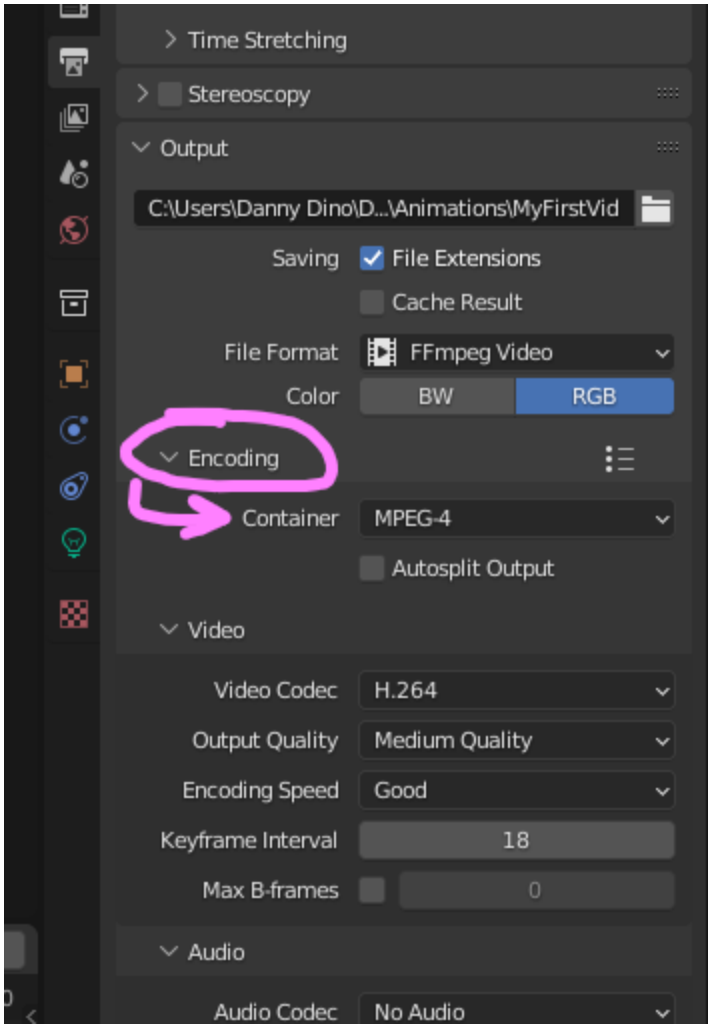
Set the File Format option to FFmpeg video



In the encoding panel, change the container to MPEG - 4 from Matroska.
Matroska is open source and free everywhere. MPEG - 4 is more widely supported on the web.
With those settings in place, render out the animation video
Render → Render Animation
It amy take several minutes to render the animation.
Afterwards, navigate to the folder where the animation was saved and try viewing it. The video should open in your system video player.

Congratulations, you have rendered your first 3D animation.

Save your file.
Spend some time adding lights, materials and textures to your scene and render it out again!
The second render will replace the first one unless the file name is changed before clicking the render animation button.


Time Stretching
Stereoscopy
Output
C:\Users\Danny Dino\D...\Animations\MyFirstVid
Saving
File Extensions
Cache Result
File Format
FFmpeg Video
Color
BW
RGB
Encoding
Container
MPEG-4
Autosplit Output
Video
Video Codec
H.264
Output Quality
Medium Quality
Encoding Speed
Good
Keyframe Interval
18
Max B-frames
0
Audio
Audio Codec
No Audio