ANGLE V1.1

DOCUMENTATION

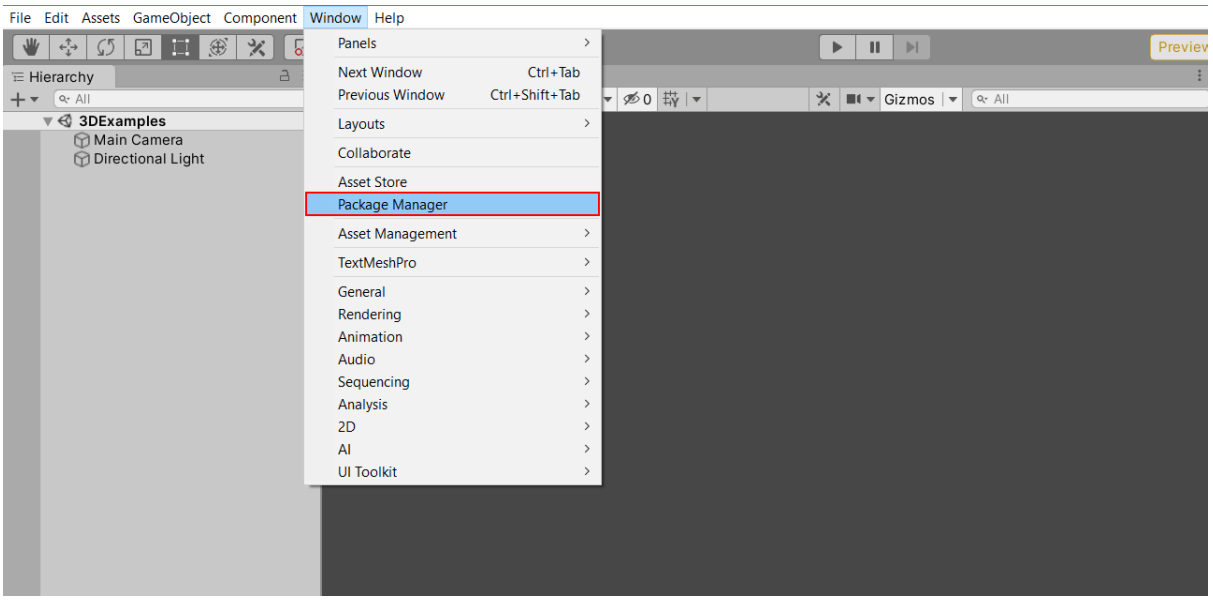
Hi, friend! Thank you for purchasing this Asset Pack!

In this asset you are waiting for: auto tiling, animated tiles, a lot of sprites, different VFX, animations and so on. However, first things first.

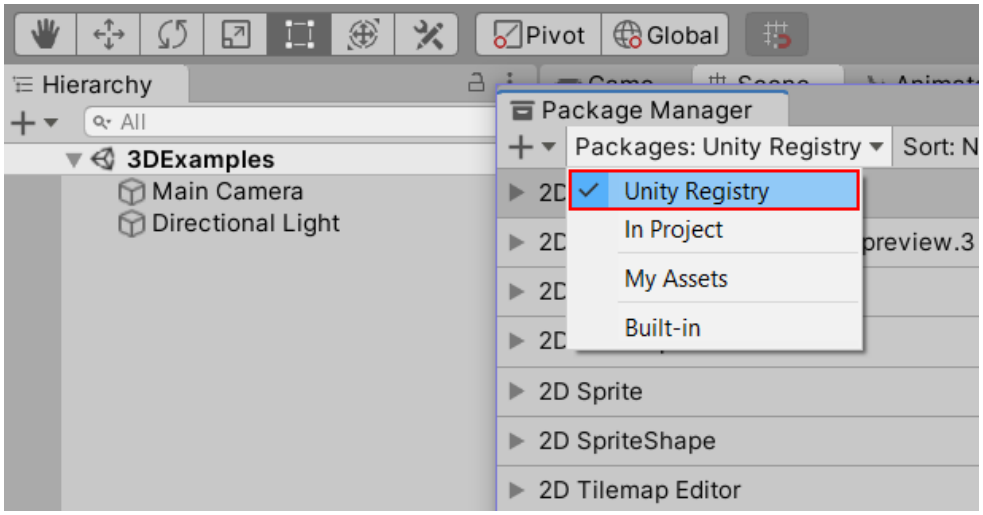
Packages

You need to check that all the necessary packages are installed. Namely: **2D Sprite, 2D Tilemap Editor, 2D Tilemap Extras.**

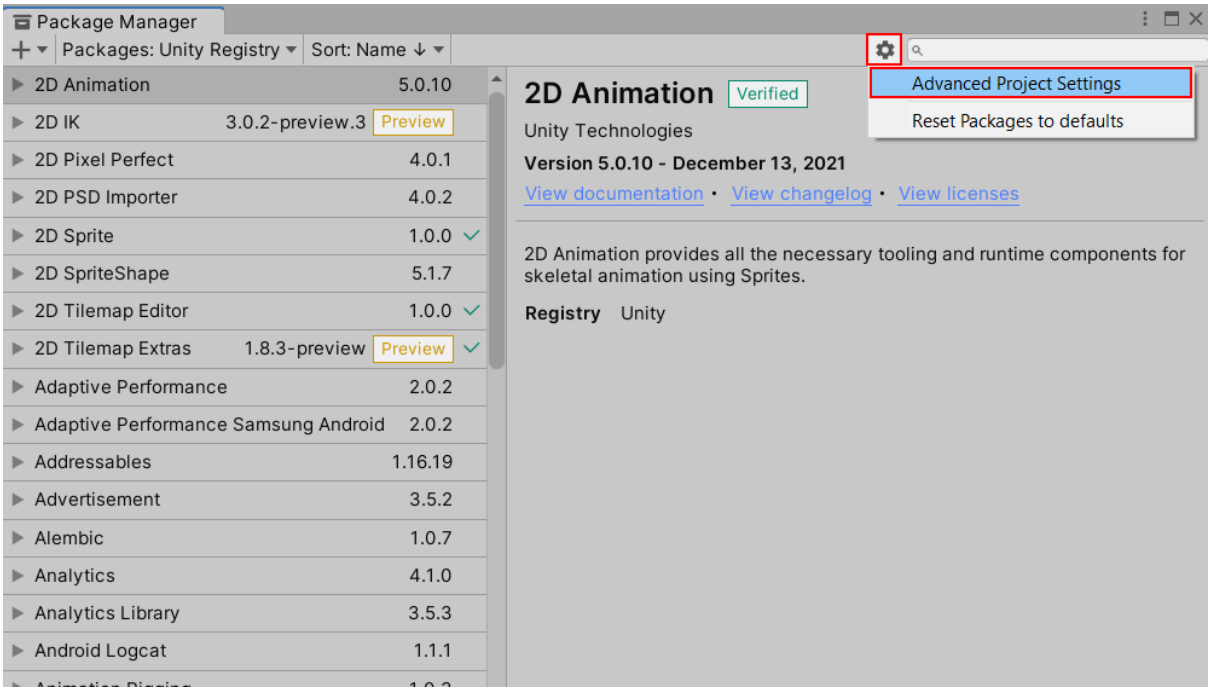
1) Open Package Manager.



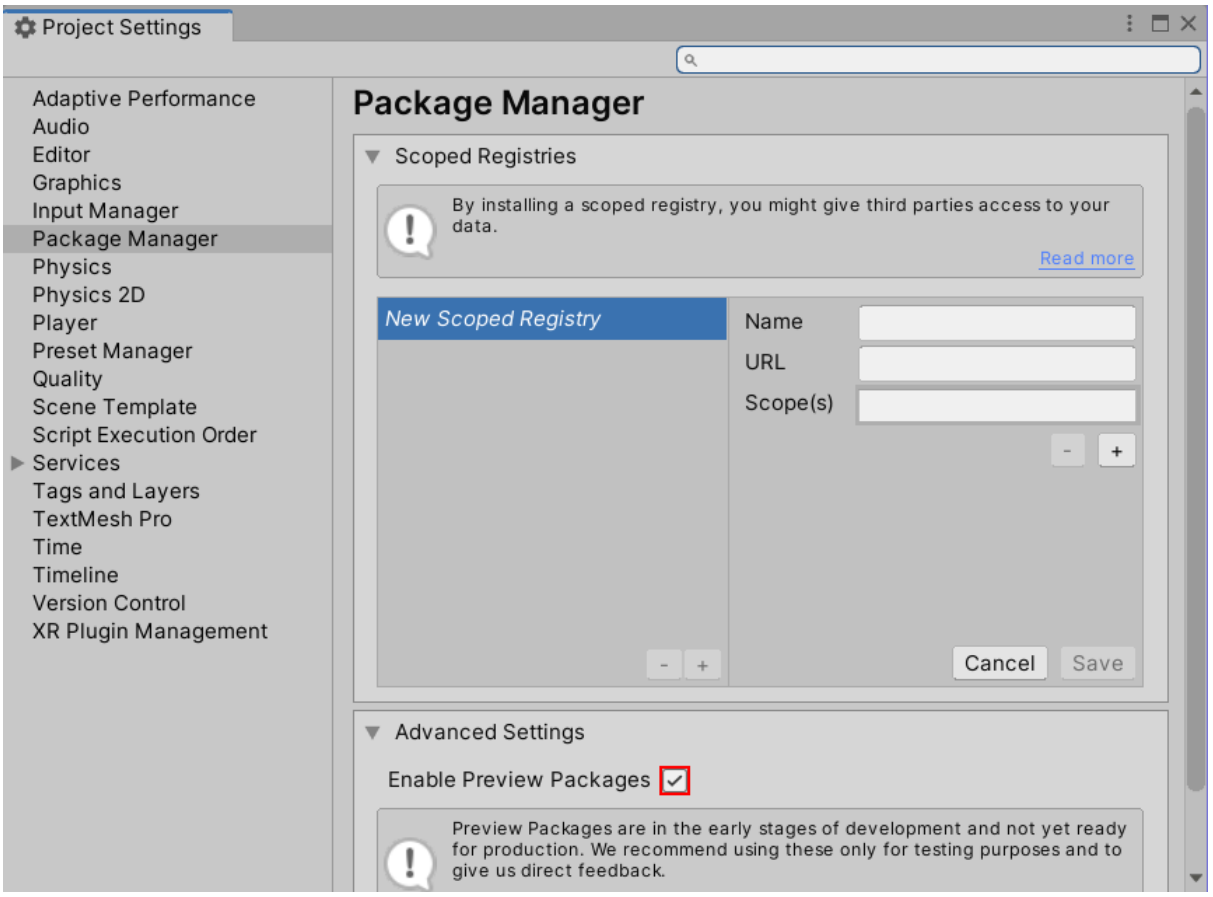
2) Go to the Unity Registry.



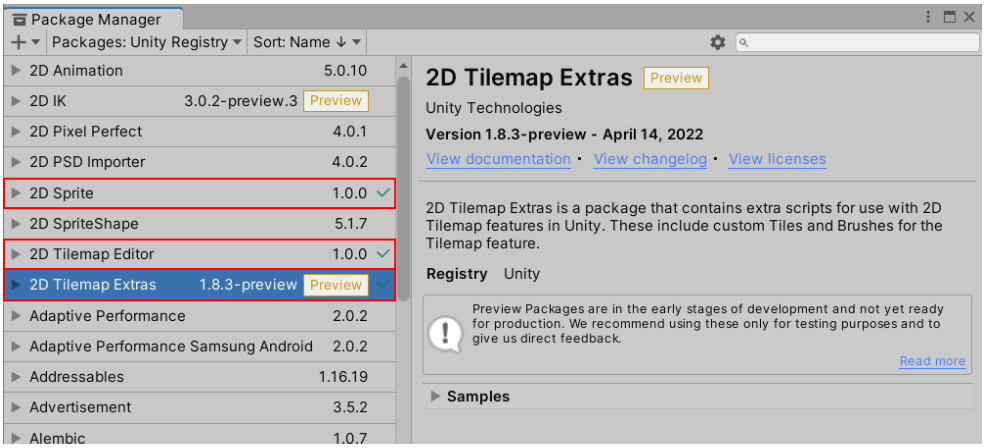
3) In the **Package Manager**, the **2D Tilemap Extras** package will not be visible, at least for now. In order to see it, click on the small gear in the right corner of the screen, and then on **Advanced Project Settings.**



4) Here, check the box next to **Enable Preview Packages.**



5) After that, we look for the packages we need and if they are not installed, install them.

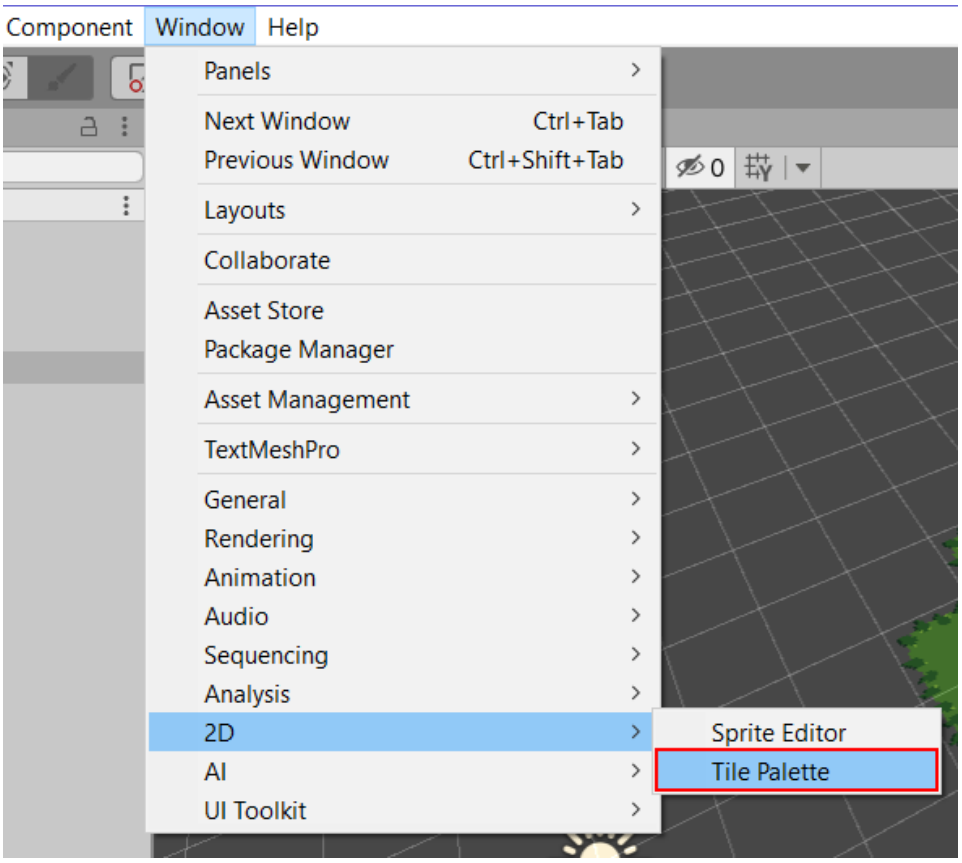


Here we will leave a link to a video tutorial on how to install **2D Tilemap Extras**, in addition: Auto Tiling in Unity - Tilemap Rule Tiles - 2021 Tutorial

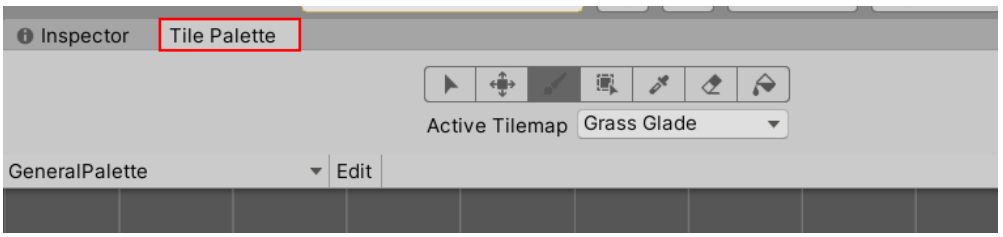
Tilemap

Next stage: you should add the **Tile Palette**.

Open Window - 2D - Tile Palette.



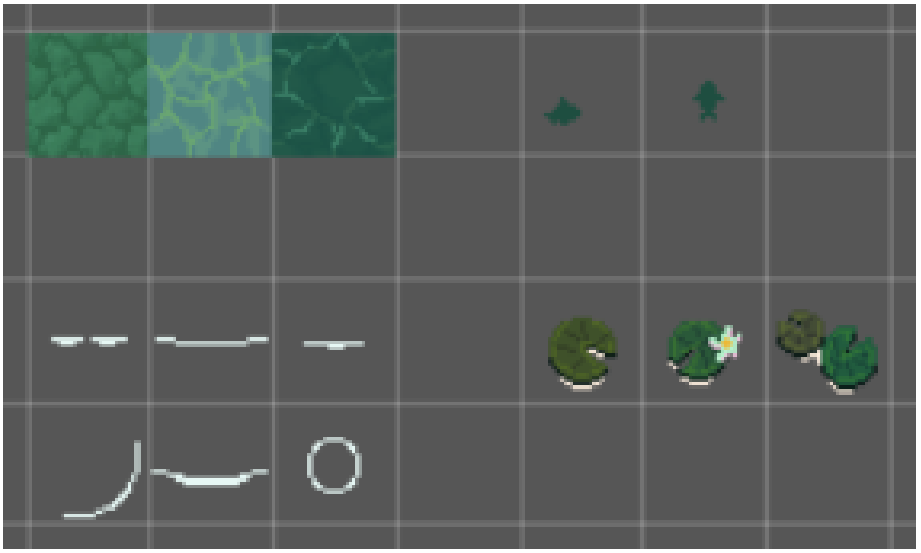
There is a new **Tile Polette** tab.



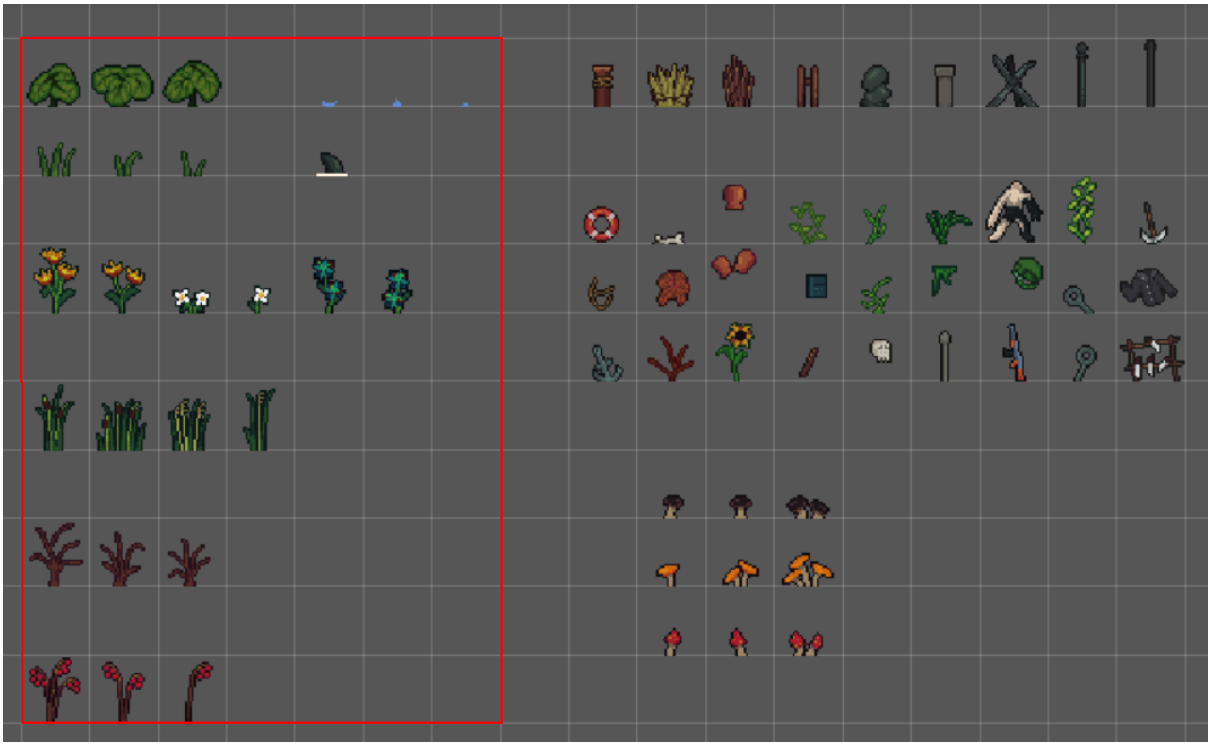
In general, the entire palette is divided into vertical and horizontal tiles. **The red frame highlights** horizontal tiles, while **the blue frame highlights** vertical tiles.

The horizontal tiles. They are dedicated **by red** and have the auto tiling function.

Tiles that are shown below are animated.



The vertical tiles. All selected tiles are animated.

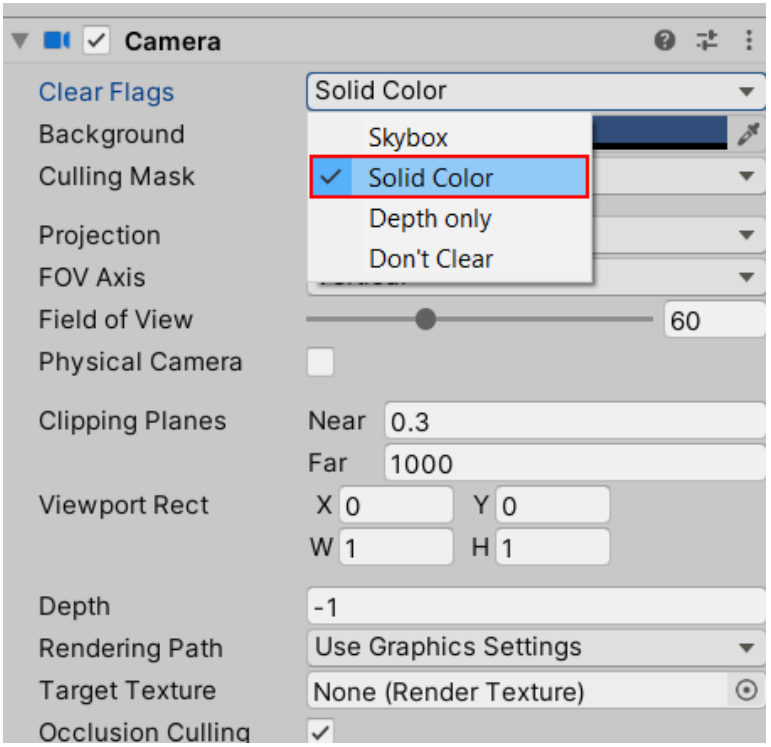


These tiles have auto tiling left and right:

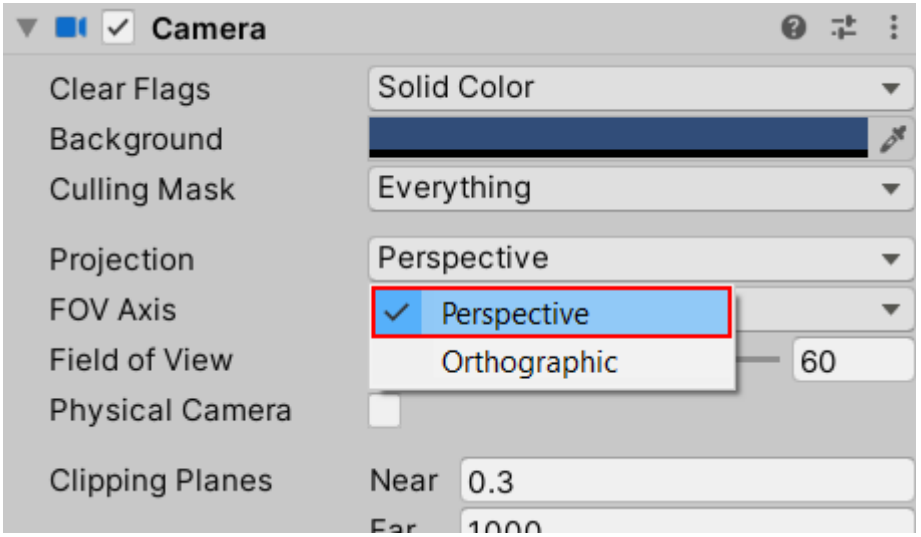
Once you have installed all the necessary packages and understood the tile palette, you can already get to work and use the asset. But if you want to use it in the 3D world, you should follow the next instructions.

3D

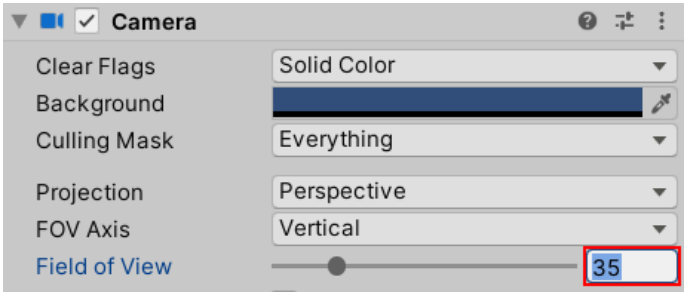
All actions in this section are performed in the **3DExamples** scene, you can open it and look at the result. First of all, create a new scene. Set up the camera. Set **Clear Flags** to **Solid Color.**



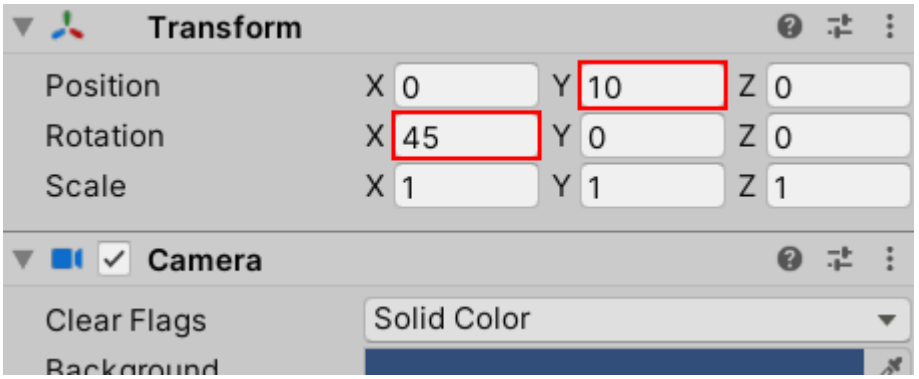
Set the camera to **Perspective** mode.



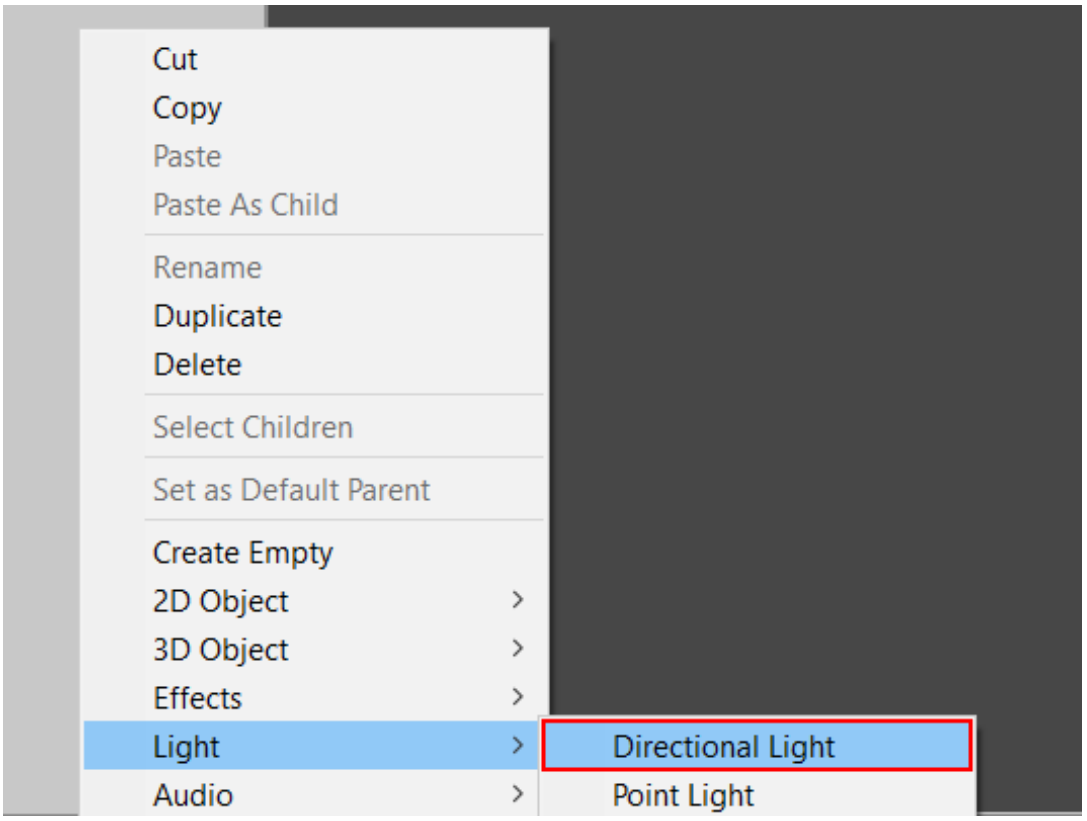
Field of View set to 35.



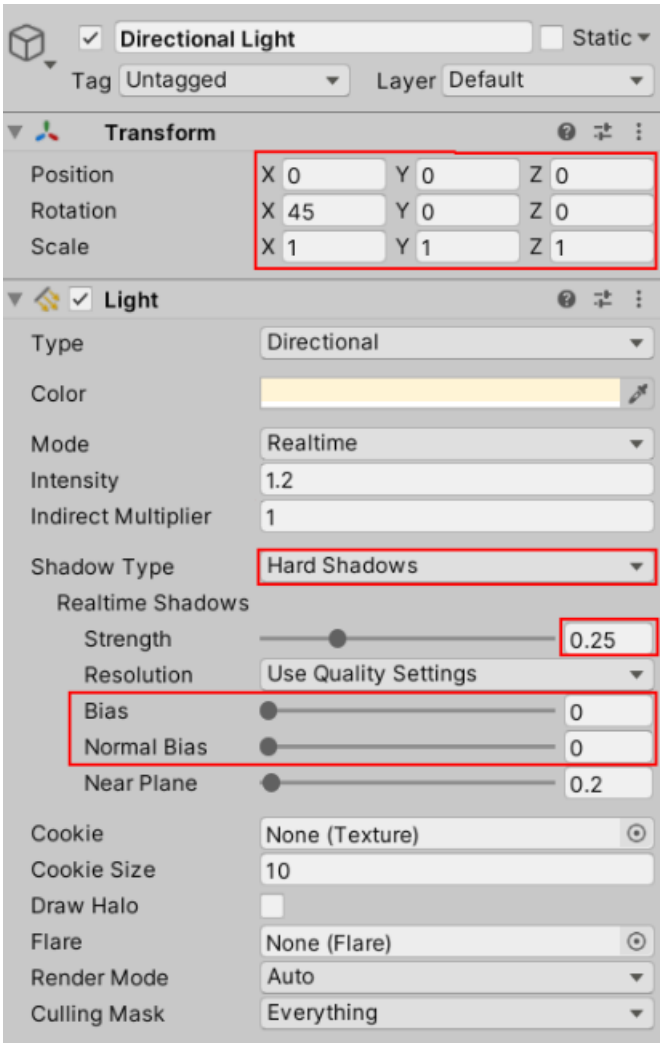
Now we rotate the camera by 45 degrees in the X axis and raise it to a height of 10 meters.



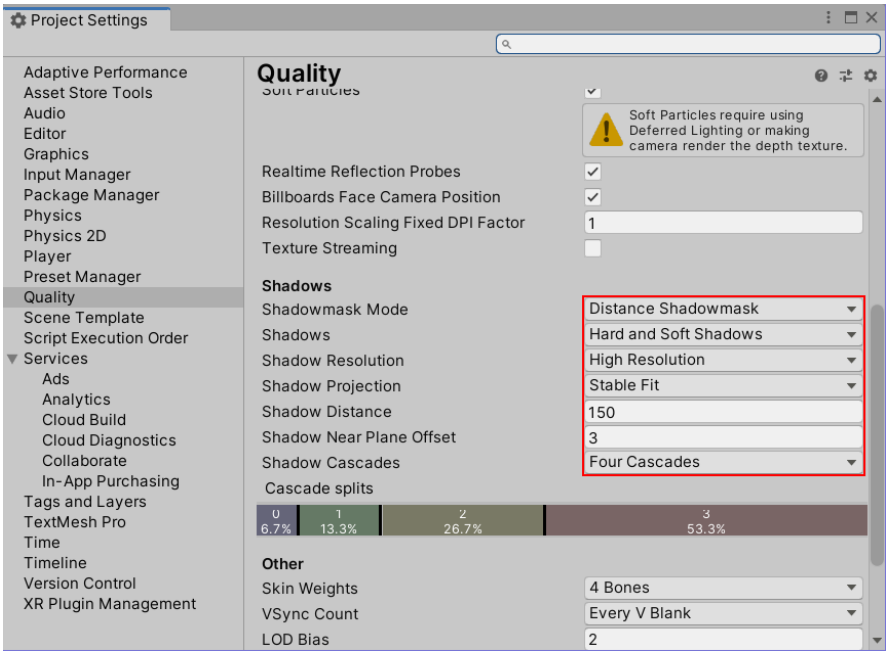
If you do not have a **Directional Light** on the scene, create it. Click on the right mouse button, in the menu point the mouse to **Light** and then click on **Directional Light.**



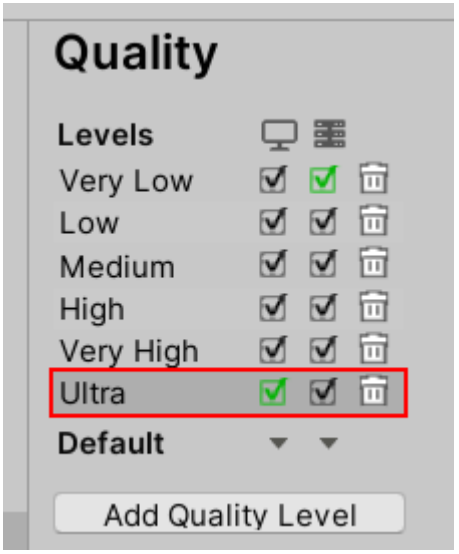
Set up as shown in the picture below. The most important values are **highlighted in red.**



Adjust shadows in **Quality settings**. **Edit - Project Settings - Quality.** This is what our settings look like. **Don't Forget About Optimization!**

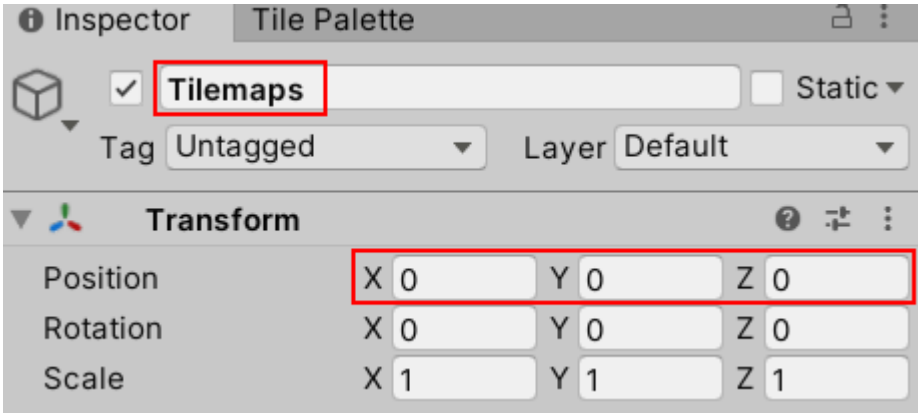


Attention! If you have missing shadows in your build, check for which platform you configured them. This can be checked in **Quality settings.**

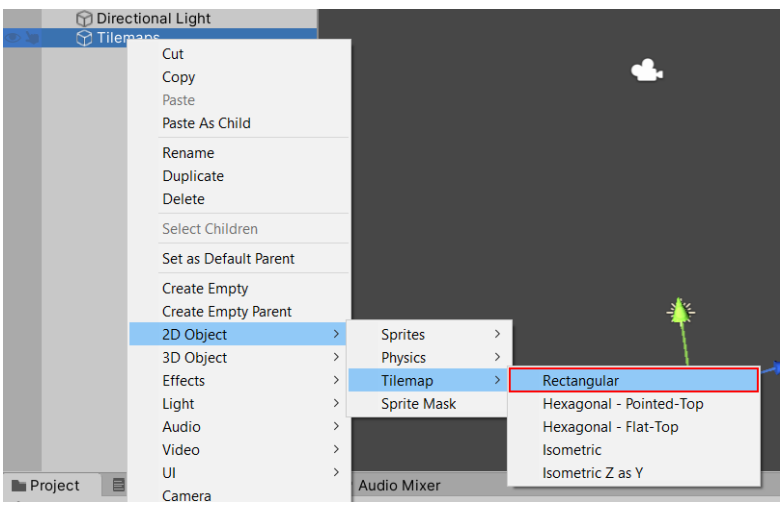


Green highlights those settings that, according to the standard, will be applied to this platform. Choose the one you want and also adjust the shadows.After setting up the light, we create an **Empty** game object on the scene.

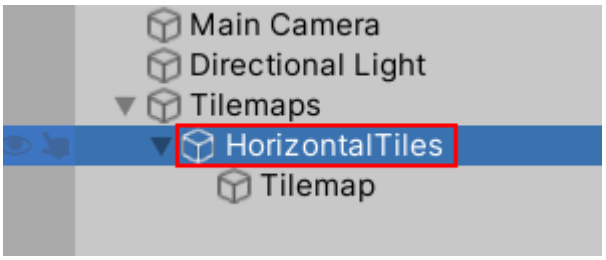
We call it **Tilemaps** and set values to all coordinates 0.



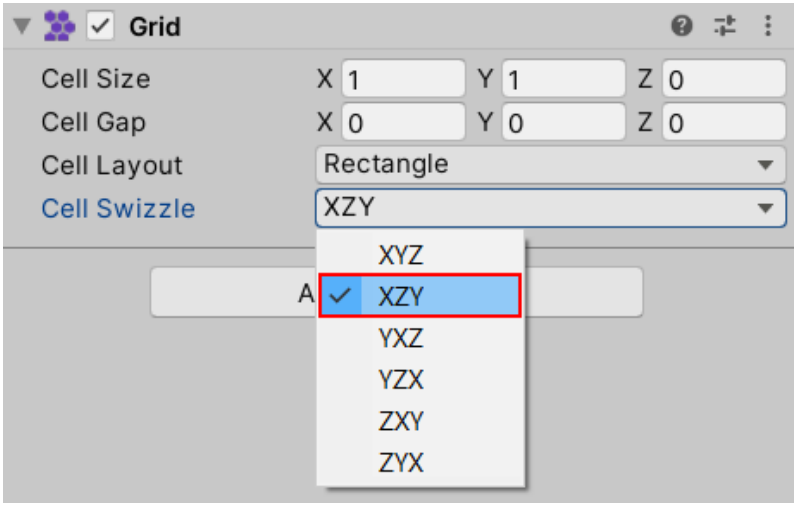
In this object we will store all the tile maps. Nextly, create a **Rectangular Tilemap.**



After that, we place it in the **Tilemaps** object and rename the **Grid** to **Horizontal Tiles.**

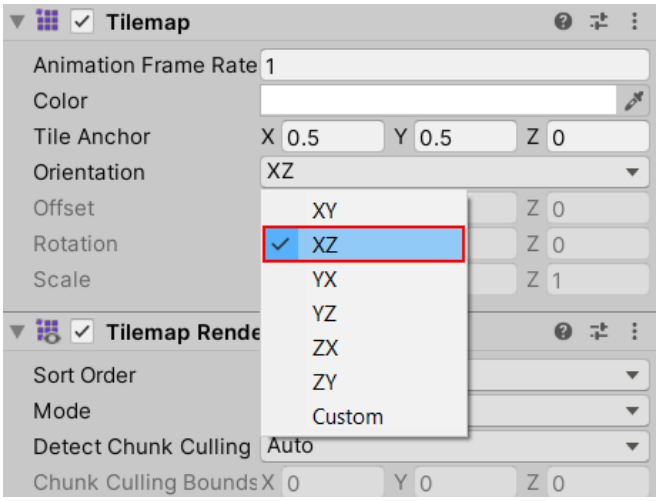


In the **Horizontal Tiles** in the **Grid** Cell Swizzle, set XZY.

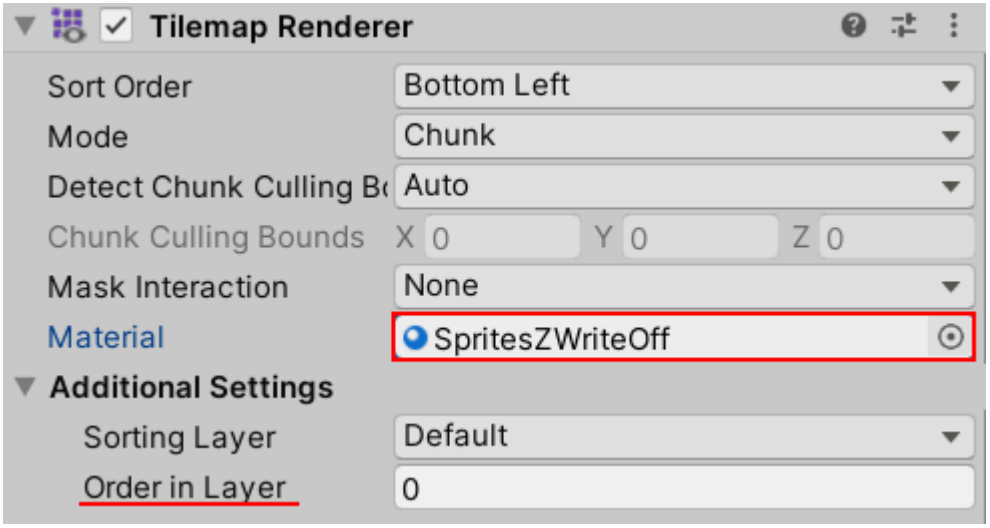


Rename the **Tilemap** gameobject inside **Horizontal Tiles** to **Grass Glade.** Here we will draw ordinary grass.

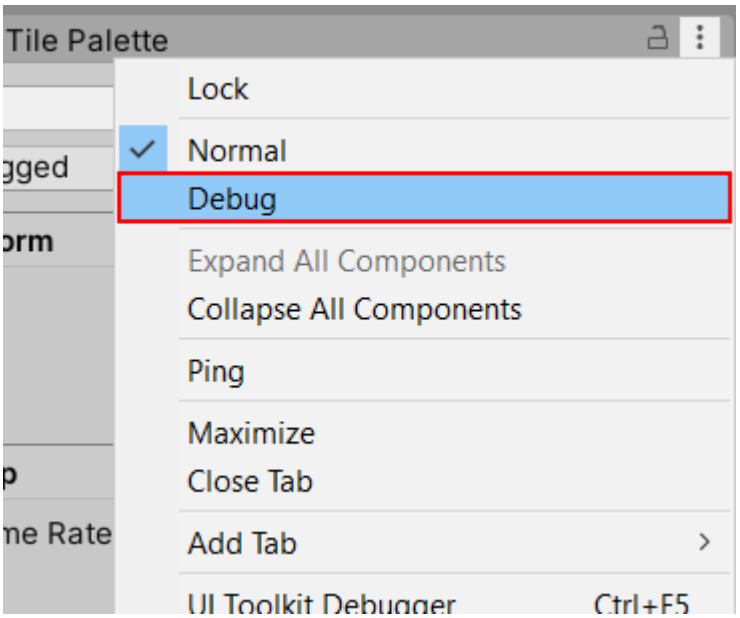
In the Tilemap component, change the Orientation value to **XZ.**



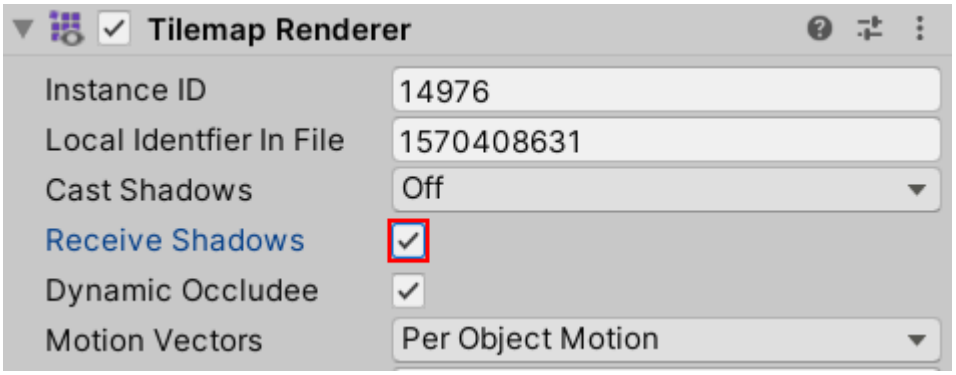
And in the Tilemap Renderer component, we set the **SpritesZWriteOff** material, it is located in the **Materials** folder in this asset. This material supports 3D lighting, its main difference from the **SpritesZWriteOn** material is that it only sorts the geometry by **Order in Layer.**



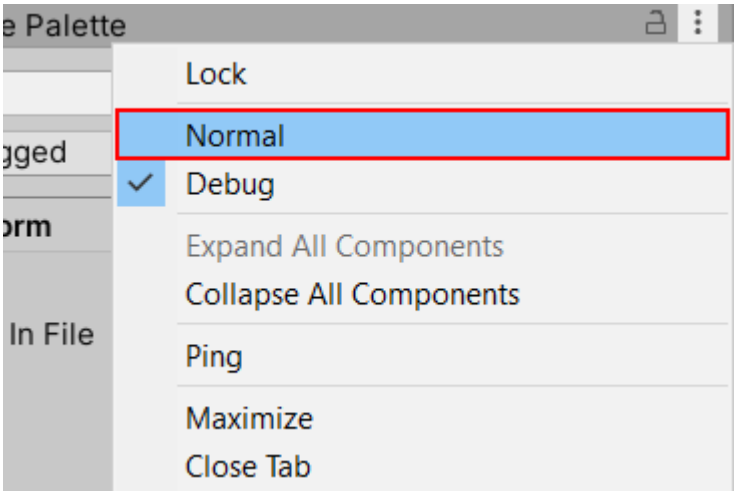
Make the grass receive shadows. Click on the three dots in the corner next to the lock. And enable debug mode.



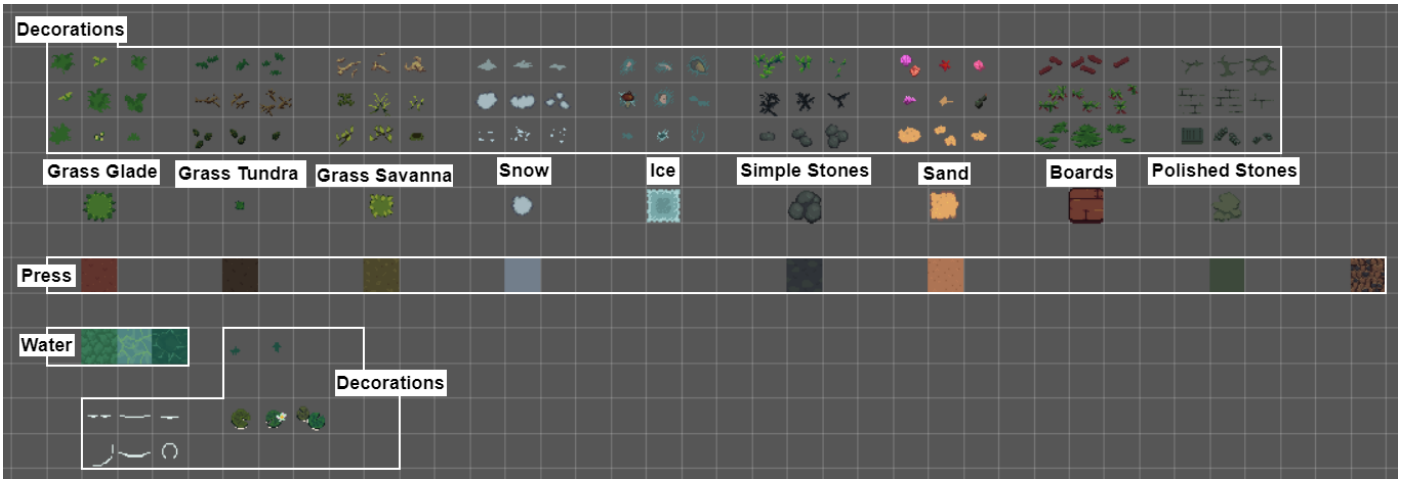
There are new parameters of components that are not visible in normal mode. In the Tilemap Renderer component, in the **Receive Shadows** check the box.



If you need, you can return to normal mode.



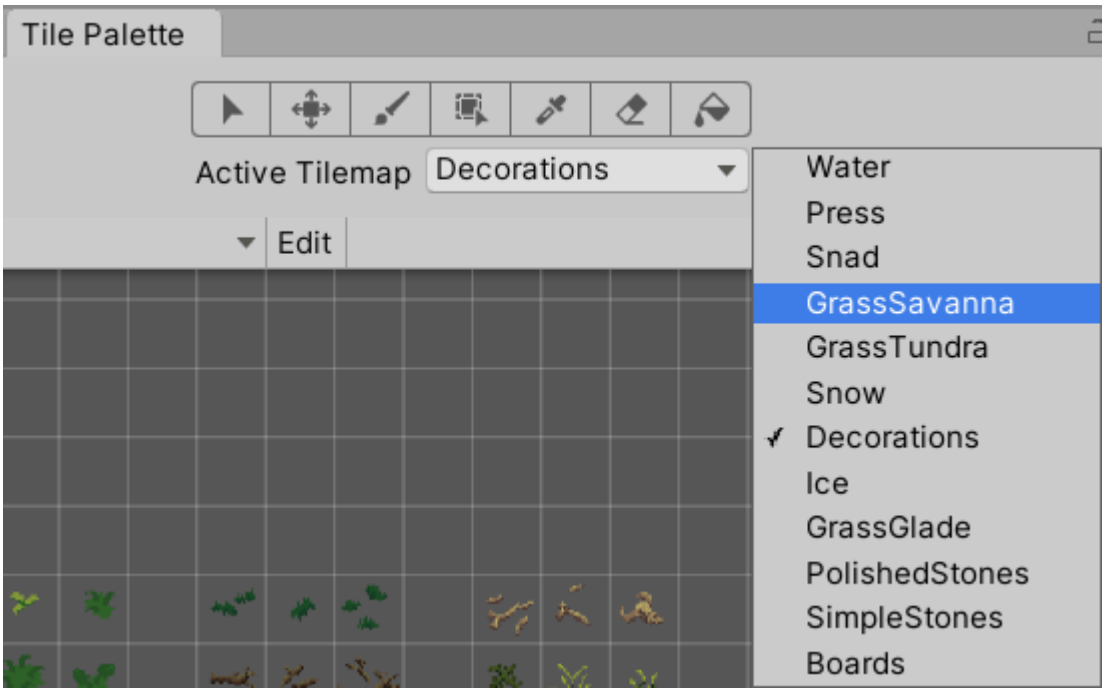
If you want to draw something other than grass, you should to duplicate this tilemap several times and rename it. Here are all the names depending on what you will draw in **Horizontal Tiles.**



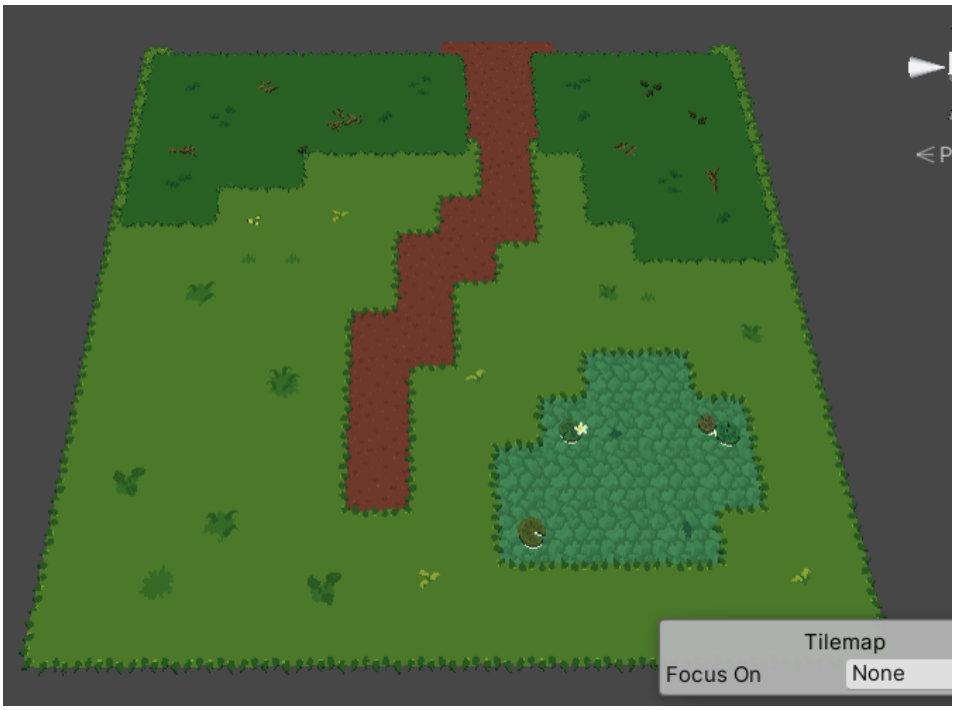
Here's what it looks like in the hierarchy.

You need to distribute the rendering order. To do this, in the Tilemap Renderer component, you need to change the **Order in Layer** values. We recommend you to use these values for each tilemap.

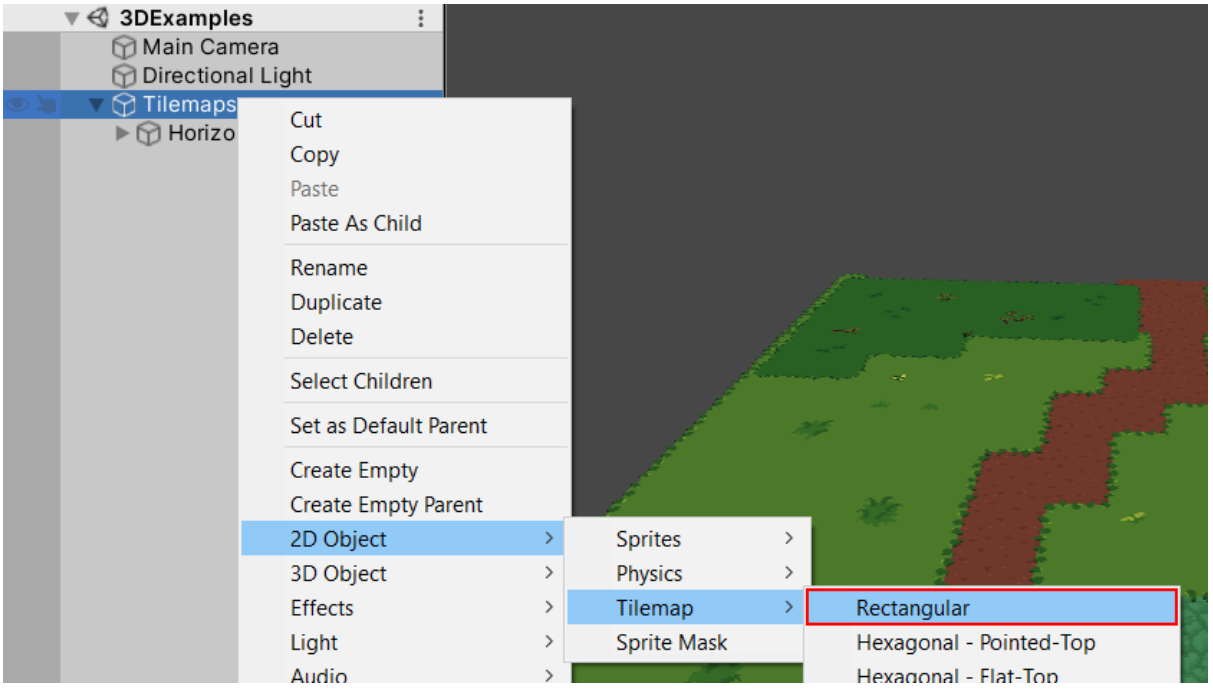
You can draw on each tile map now.



Here's what we ended up with. We remind you that you can see all this in the **3DExamples** scene in the **Scenes** folder of this asset.

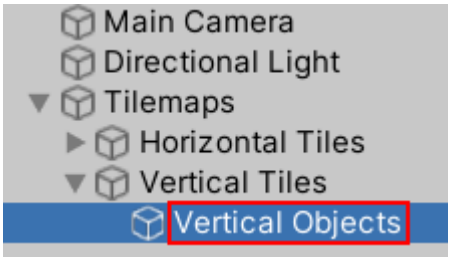


Let's add some more vertical tiles. We create a separate Tilemap for them. And also drag it to the **Tilemaps** object.

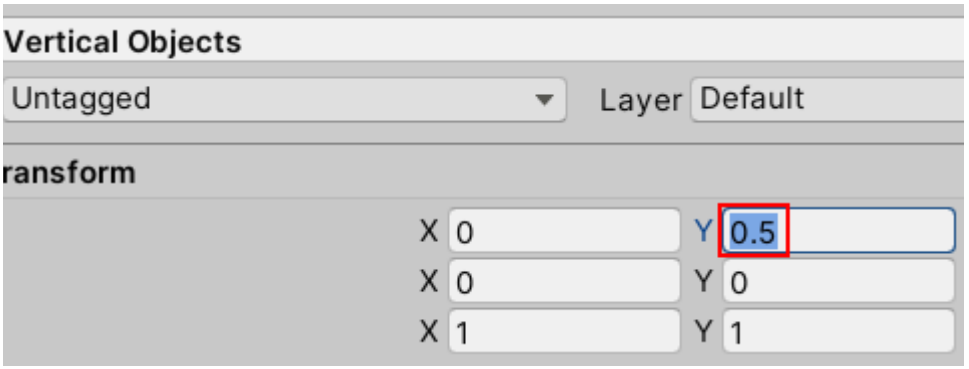


Rename **Grid** to **Vertical Tiles.**

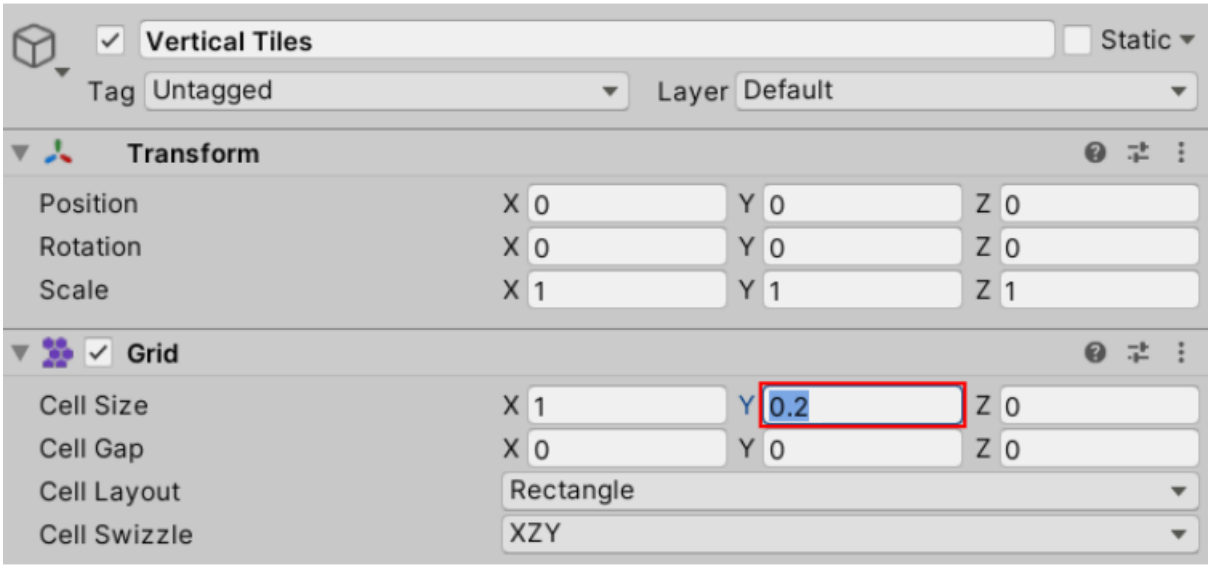
And rename **Tilemap** to **Vertical Objects.**



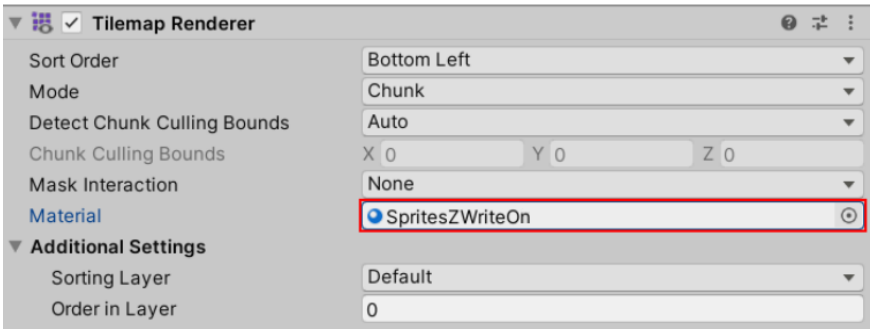
Also raise the **Vertical Objects** 0.5 meters up.



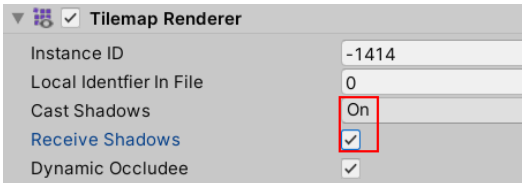
Then we switch back to the **Vertical Tiles** object. And in its Grid component, change the value of Cell Size by Y to 0.2. And in the Grid Cell Swizzle, set XZY.



After we select the **Vertical Objects** object and set the **SpritesZWriteOn** material in its Tilemap Renderer component. It draws the geometry based on the position, and not on the **Order in Layer.**



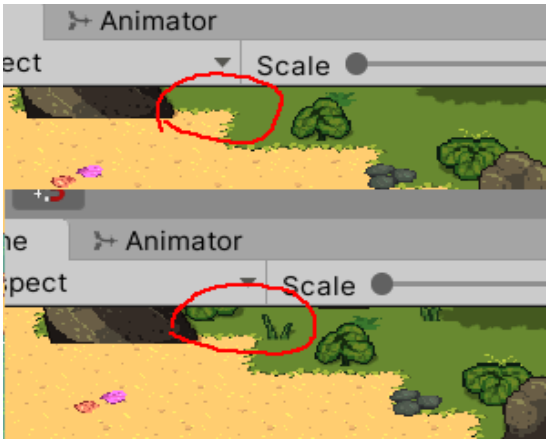
Then we turn on the debug mode again and change the **Cast Shadows** values to On and check the **Receive Shadows** checkbox.



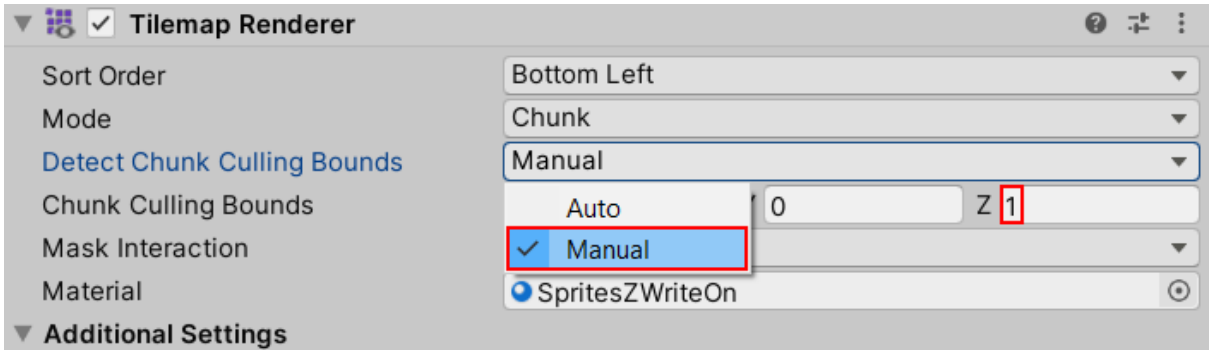
Vertical Objects cast and receive a shadow. Finally, we can start drawing vertical tiles.

Here's **what** we got.

Note: some vertical tiles may be rendered late, as in the picture below.



In order to avoid this, you need to change the Detect Chunk Culling Bounds in the Tilemap component from Auto to Manual. And change the Z value in Chunk Culling Bounds. Up to value until the problem disappears. In our case, this is 1.



Sprites

To add a sprite, for example a tree, you need to transfer it to the scene. Put it in the right position, add **SpritesZWriteOn** material to it. And in debug mode, change the values of **Cast Shadows** to On and check the box **Receive Shadows**.

Note: SpritesZWriteOn and SpritesZWriteOff materials do not support translucent objects.

Completion

We hope that this document has explained to you how to work with this asset. We would like to add that the **Prefabs** folder contains some visual effects that you could see in the trailer. You can always write to us by mail. Also don't forget to write your reviews on the Unity Asset Store page. Good luck to everyone and see you soon!

Email: ooosuperpositionprinciple@gmail.com

Website: https://sites.google.com/view/superprinciple