MARCHING SQUARES

Thank you for your purchase! If you have any questions feel free to send me an email: onedragutin@gmail.com

This is simple guide that will help you get started using this system.

TILE

In order to have tiles or squares in your world you need to create a new tile by doing the following:

1. Create an empty game object in your scene
2. Add tile script
3. Make a prefab

Now when you have a tile you need to define how this tile will look in the world. To do that you need to assign a material with the texture assigned to it(texture should have the tile look).

Now you need to tell where is your tile in the texture by changing the 'Tile information' values.

The pivot point is bottom left. The offset X tells how many pixel you are offsetting from the bottom on the horizontal axis, same does the offset Y. Width and Height are just a size of the tile.

At the very bottom you can see how the tile will look in the game.
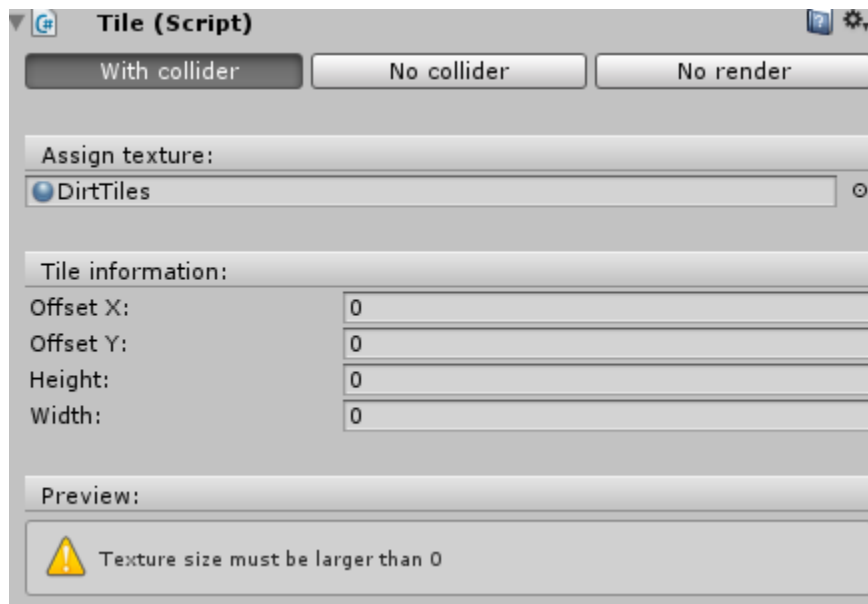
There is also three tile types that you can switch any time:

1. With collider – this tile will have a collider
2. No collider – this tile will not have a collider
3. No render – should be used for air tiles (no geometry nor colliders)

MARCH TILE

You could already create a world using the basic tiles. Here is how the demo world would look with no marching tiles.
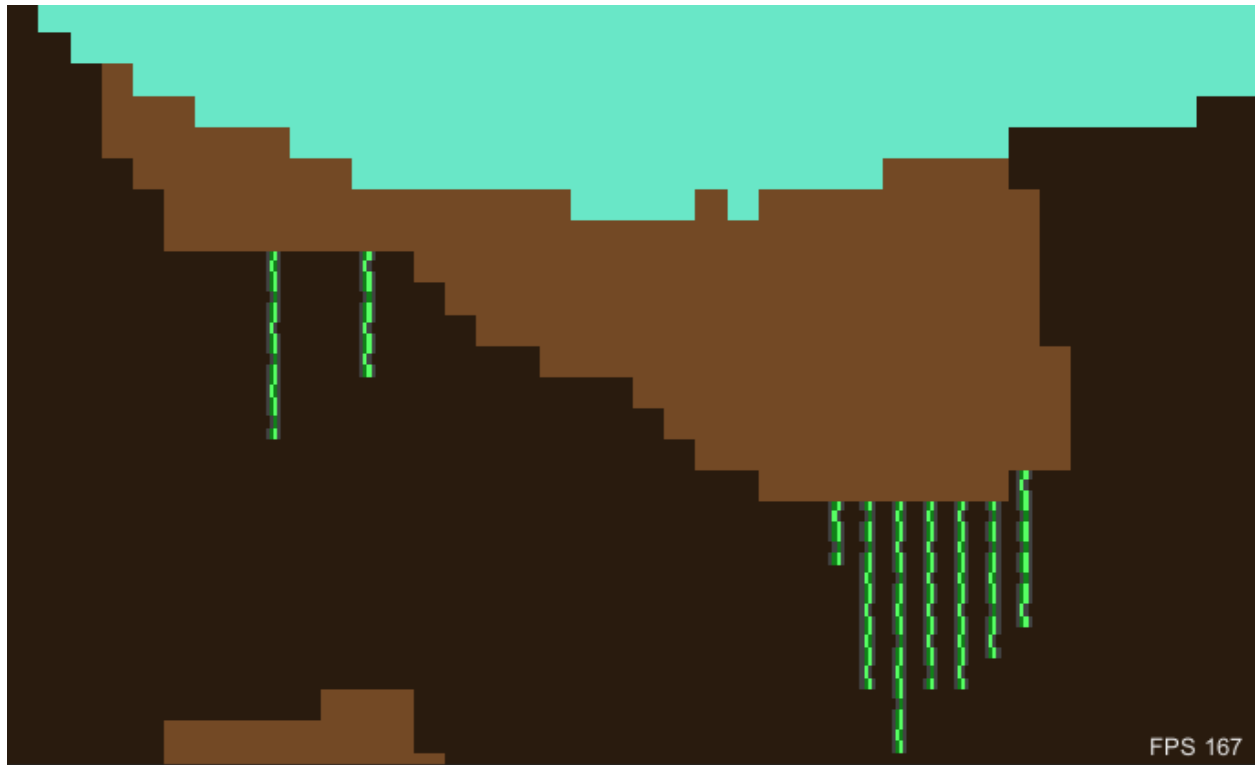
As you can see it somewhat looks good, but it has 'square' look. Which we will fix by adding marching tiles.

To create marching tile do the same steps like you did with the tile, only this time add march tile script instead. Then again you need to define marching tile look (same as tile).

Scripts are almost the same, but there is a difference. In the march tile script there is left, bot, right, and top array and one parent tile variable. These all will store a reference to the root or basic tiles.

Top

Left Parent Right

Bot
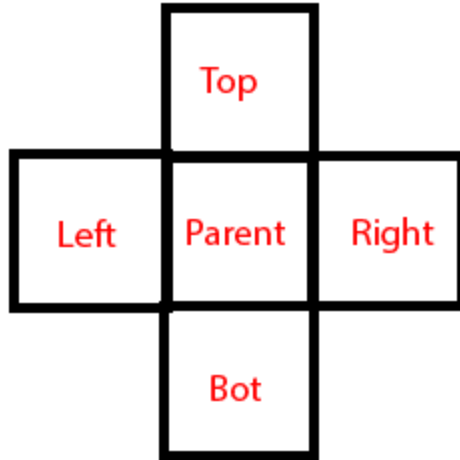
Parent is a tile which should be checked for condition.

Top is all tiles that if found above this tile will trigger the condition.

Left is all tiles that if found left to this tile will trigger the condition.

…

Same thing for the right and bot.

Every tile in the array could be represented as this:

If 1 tile in the array left or If 2 tile in the array left or if 3 tile in array left

And

If 1 tile in the array right or If 2 tile in the array right or if 3 tile in the array right
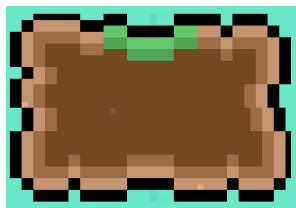
…

Same thing for the top and left.

Only then, the condition will be triggered.

Note that there could NOT be two same conditions, if that happens the second condition will not be added. The console would show you a message if that happens.

To better understand the marching tiles here is the march tile example:
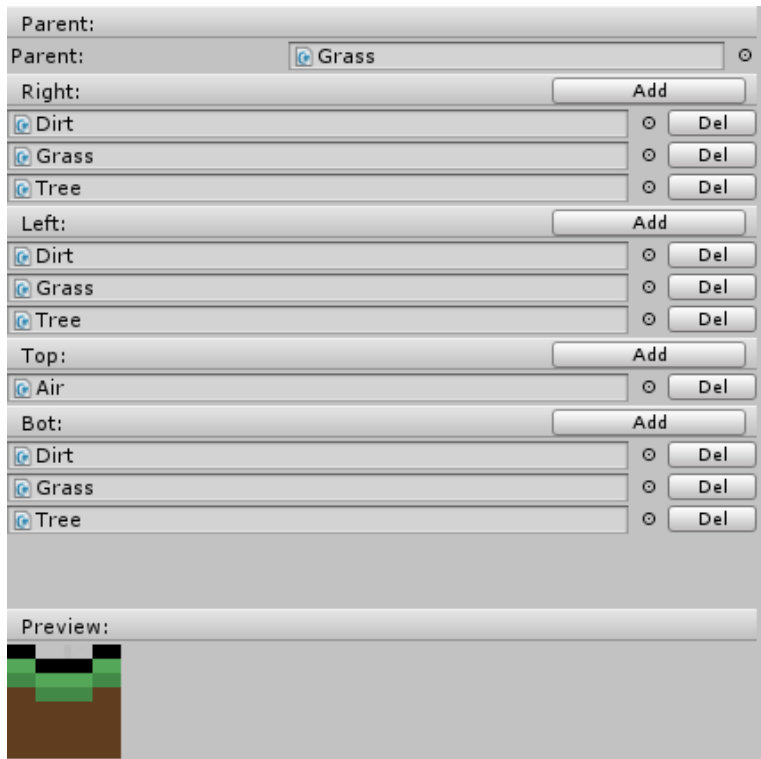
Now here is one of the conditions where this tile will show:

As you can see it looks much better with the marching squares used in world. In the picture above there is also a conditions for a dirt too.

CUSTOM WORLD

To customize your own world you will need to create a new script and inherit from a world class. This way you can override the OnChunkCreated(Chunk chunk). Which is almost all you need to customize your world.

This method is called when the chunk is created for the first time(it is not called if the chunk is loaded from the save file).

Here you need to loop through each tile and set your tile depending for example on noise which is also included in the system.

```csharp
using UnityEngine;
using System.Collections;
using MarchingSquares;

public class CustomWorld : World {

    protected override void OnChunkCreated(Chunk chunk)
    {

        //SETS ALL TILES IN THE CHUNK TO TILE WITH ARRAY INDEX OF ZERO


        for (int x = 0; x < World.CHUNK_SIZE; x++)
        {
            for (int y = 0; y < World.CHUNK_SIZE; y++)
            {
                chunk.SetTileLocal(x, y, 0);
            }
        }
    }
}
```

To save or load the world use SaveManager that every world has.

At the moment every world is saved on quit but not Loaded at the game start.