**Getting started**

Install NodeJS – download <https://nodejs.org/dist/v6.11.2/node-v6.11.2-x64.msi>and run as admin.

Install superpowers – download

<https://github.com/superpowers/superpowers-app/releases/download/v4.0.0/superpowers-v4.0.0-win-x64.zip>

and unpack to: C:\Program Files\superpowers

You can create a shortcut to C:\Program Files\superpowers\Superpowers.exe

Install the support libraries -

<https://github.com/superpowers/superpowers-core/releases/download/v4.0.0/superpowers-core-v4.0.0.zip>

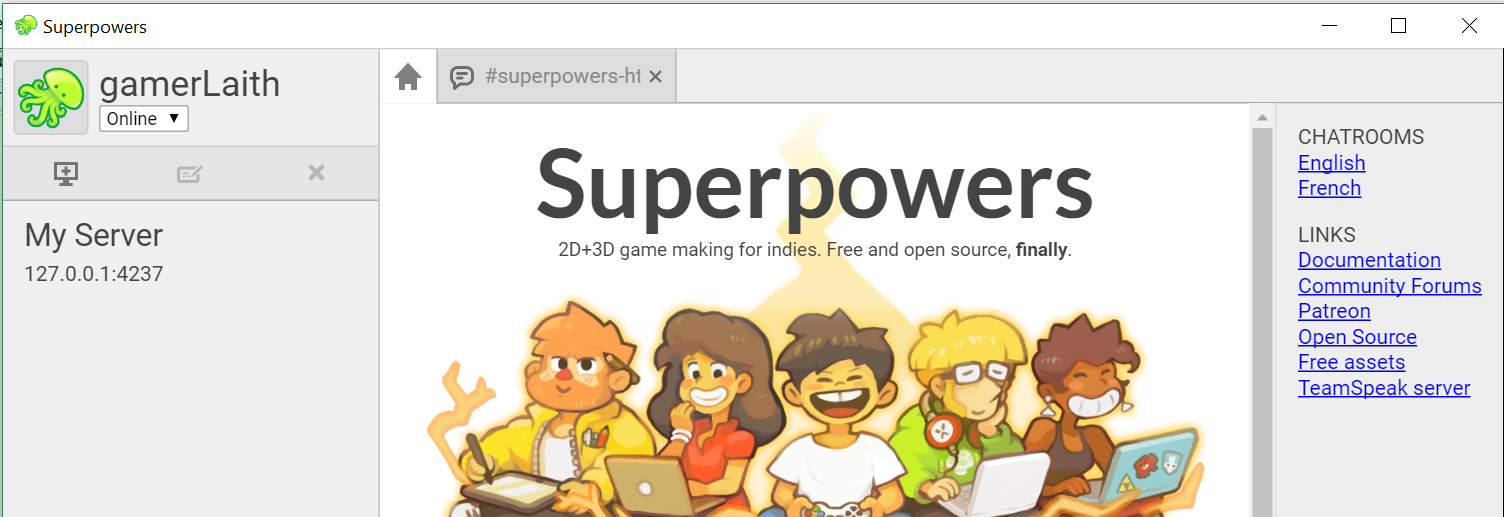
and unpack to: C:\Users\User\AppData\Roaming\Superpowers\core

<https://github.com/superpowers/superpowers-game/releases/download/v4.0.3/superpowers-game-v4.0.3.zip>

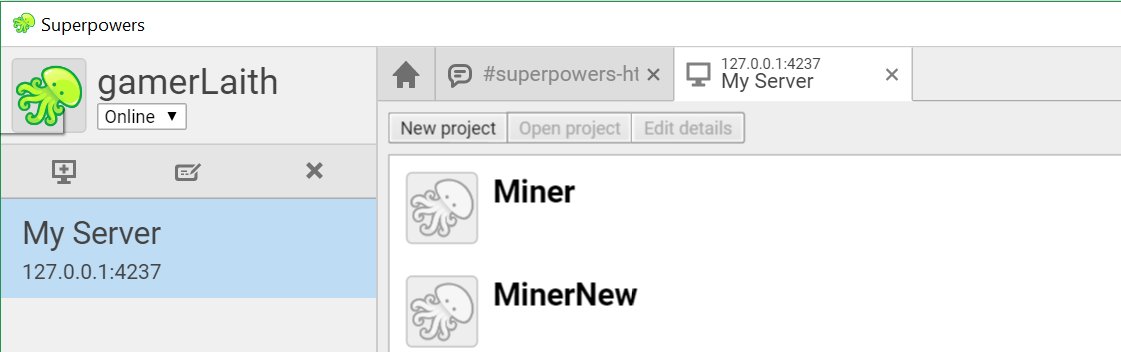
unpack to: C:\Users\User\AppData\Roaming\Superpowers\systems\game

You can then run the IDE from the exe.

On first run, it will ask you for a nickname, this is for the IRC channel and you can change it later. You can put in anything here, and the screen should look similar to the below when running properly:



If you doubleclick on ‘My Server’ it will connect you to the IDE:



Projects are stored in: C:\Users\User\AppData\Roaming\Superpowers\projects

so unpack this demo project here and restart the IDE to see it show up as MinerNew. Doubleclick it to edit the project.

**Demo Project Layout**

All the project assets are displayed in the leftmost tab, note the Game folder holds details of the global game bits eg. the main scene, the camera for the scene etc. There are 5 main types of asset:

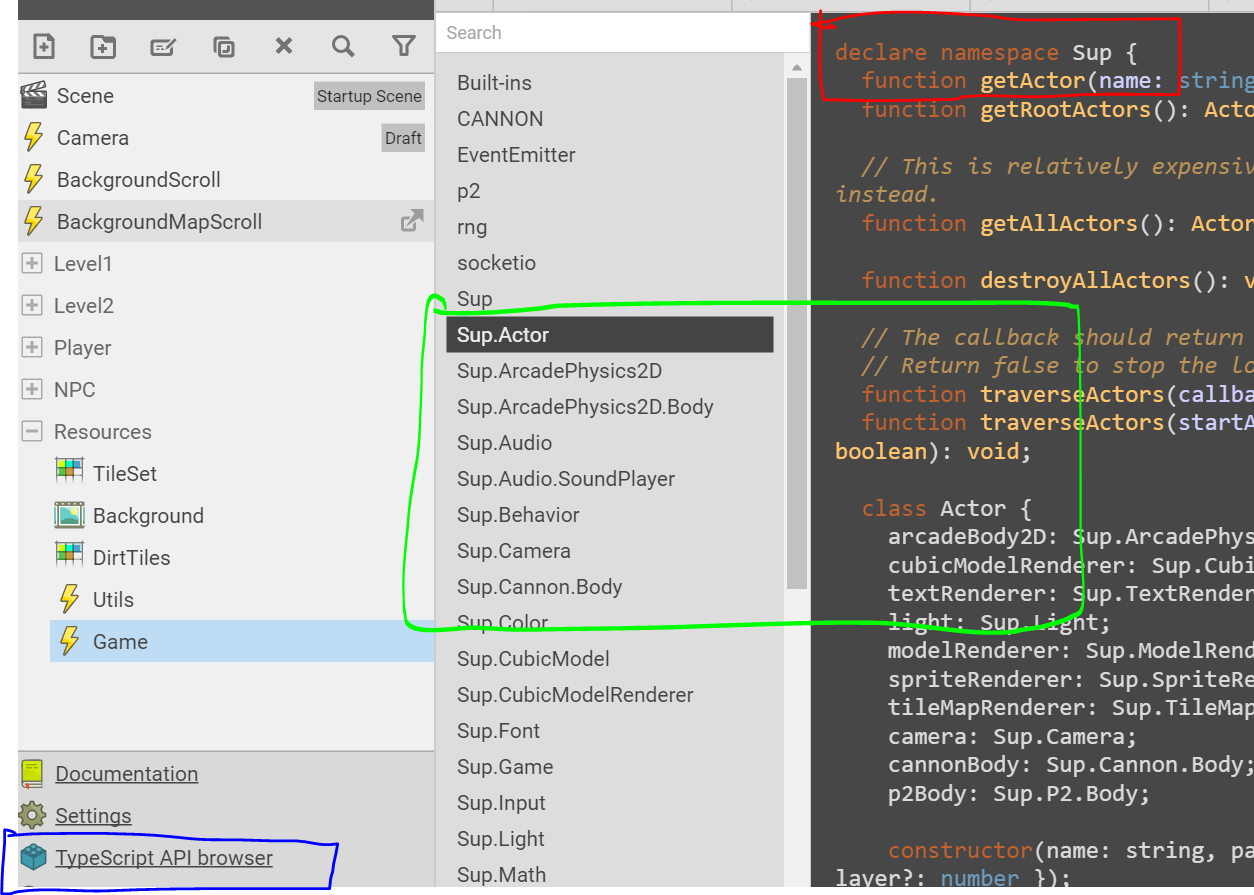
The scene, which describes the game world.

Behaviors (yes note US spelling), which are shown with a lightning bolt. They are script bits of functionality which are attached to assets.

Sprites, these are used in the game for the player, baddies etc.

Tilemaps, backgrounds made of repeatable tiles from a tileset. This demo has two tilemap layers for parallax.

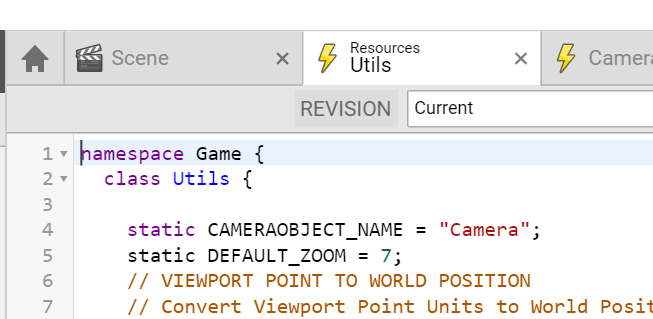
Tilesets.

The Superpowers API documentation can be found from the Typescript API browser (blue box).

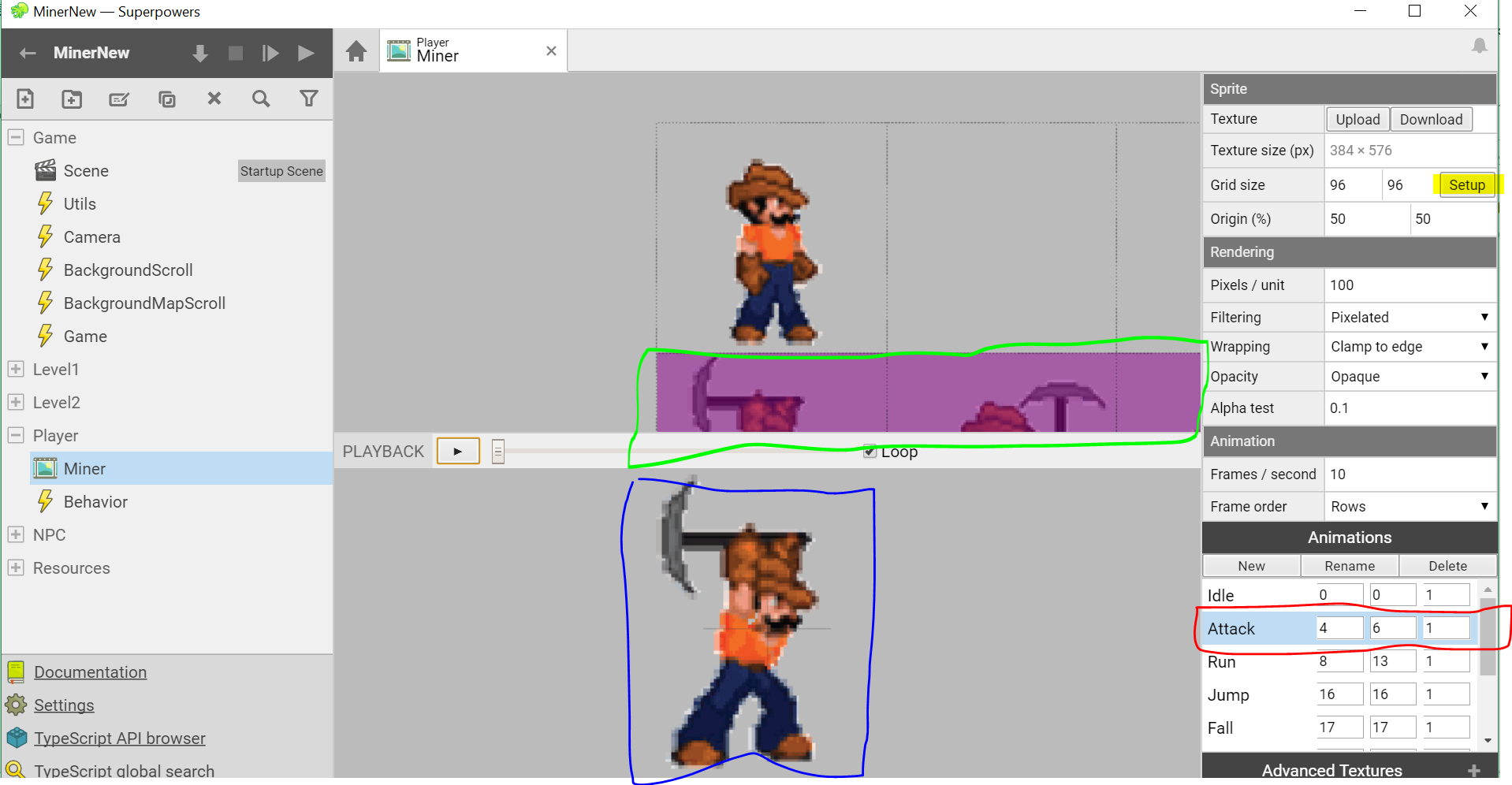
The Namespace for all the Superpowers functions is Sup. (red box).

The Classes are addressed by namespace.class eg. Sup.Actor. in the screenshot.

Namespace for all the game helper code is in Game.



**The Sprite Editor**

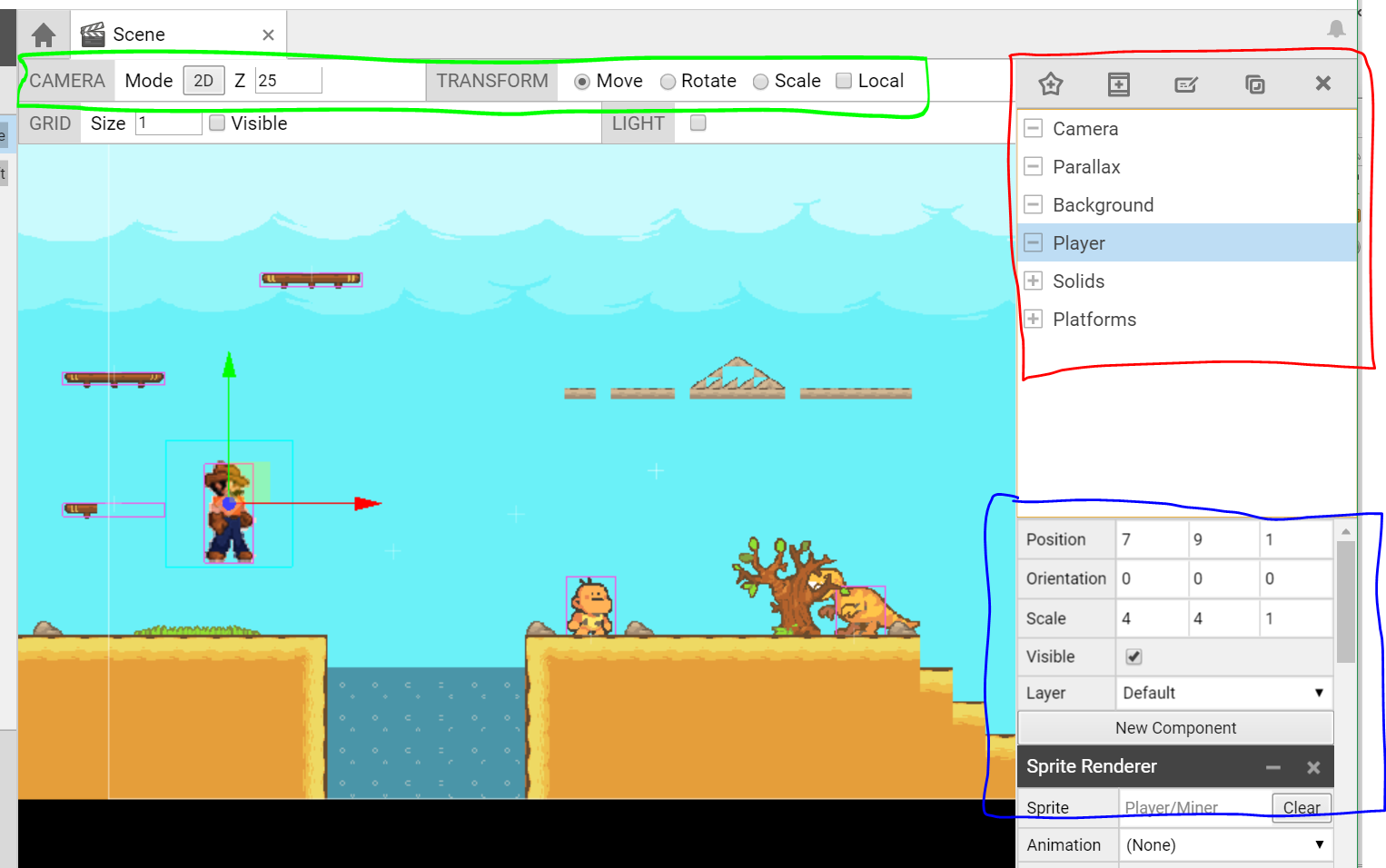


**The Scene Editor**

The scene Entities are shown in the red box. The selected entity is shown in the scene with the orientation arrows, you can drag the arrows to move the entity around. The entity position etc. of the selected entity is shown in the blue box.

The camera controls allow you to control what the orientation arrows do (see green box) eg. in the screenshot they are set to move the entity in 2D.

You can attach components to each entity to make them display, and do things. In the screenshot, the player entity is rendered by a Sprite renderer component, which uses the Miner sprite in the Player folder (see blue box).



**The Tilemap Editor**

Editing the layers:

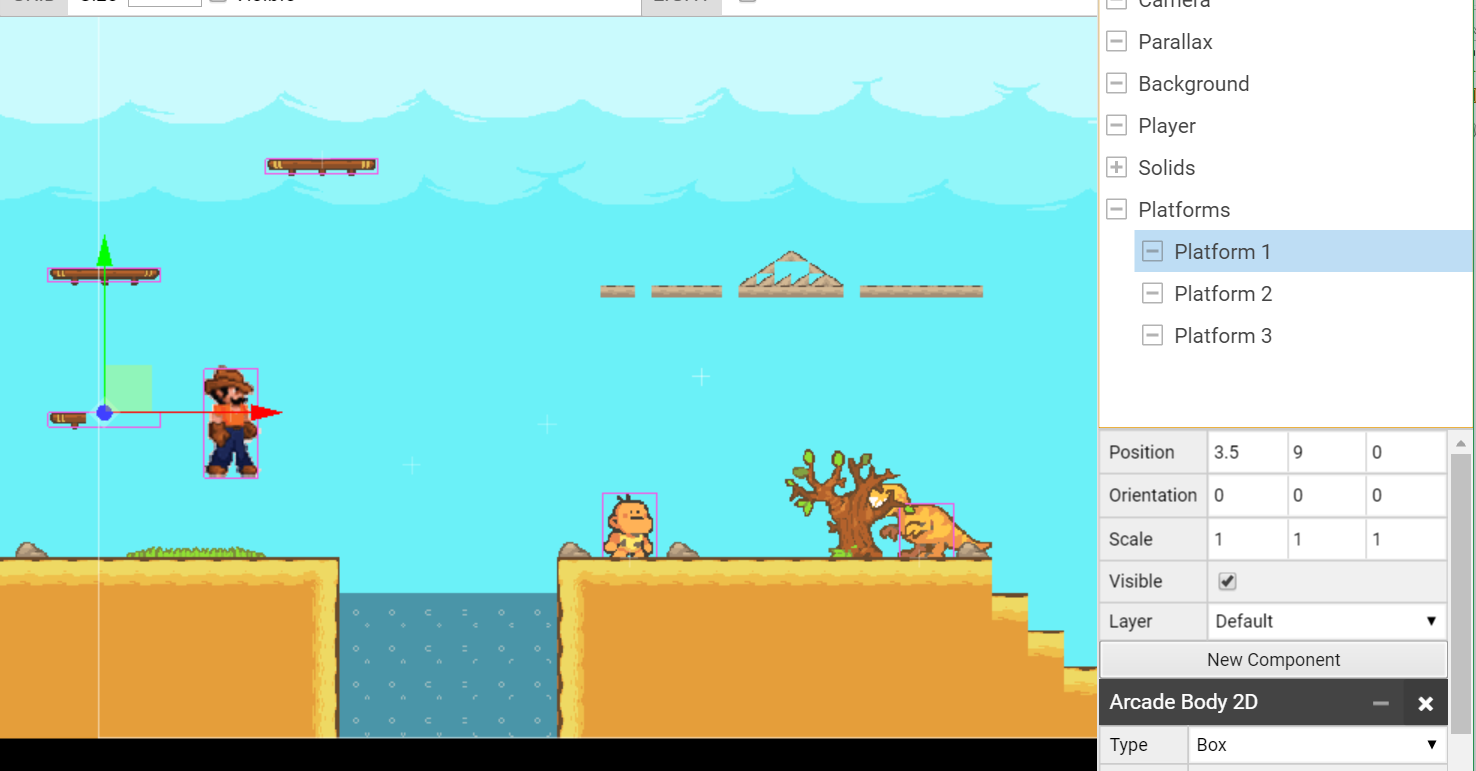
You edit the layer that is highlighted in blue (see blue box).

Scrolling the tileset box (see red box) is done by holding the mouse scrollwheel down and moving.

Use the eraser to remove tiles – anything else may introduce hidden blocks you can collide with.



You can check the state of the game world via the scene component – this is updated in realtime from say the tilemap component:



In this shot, the platform image is from the tilemap, and only the image layer - there is no collision layer as the player would not be able to pass up through the platform.

The platform collision box is a separate entity attached to an arcadeBody2D component.

The map parallax layers need to use the same tileset ie. one tileset per scene (see the triangular platform in the screenshot).

Remember, the green and red orientation arrows denote which entity you’re editing.

The T-rex occlusion is controlled by the z-axis position.