

Contents

1	about this guide	1
2	some names	1
3	custom properties?	1
3.1	visual manual	2
4	layers?	7
4.1	tile layers	7
4.2	object layers	7
4.2.1	visual manual	8
4.3	object layer lists	8
4.3.1	visual manual	9
5	Tiled special custom properties	15
5.1	collide	15
5.2	kill	15
5.3	teleport_to	15
5.4	visual examples	16
6	Tiled special layers	17
6.1	Spawn	17
6.2	Player	17
6.3	moving*	17
6.4	npc*	18

1 about this guide

This guide is separated into two parts. First general tutorials about Tiled. Then the documentation of the custom magic stuff implemented by dexter to make you able to create complex levels without having to touch program code.

2 some names

- collisions/colliding/... if the player collides with something, he/she can hit it. It's like an obstacle instead of a background wall. The player can't walk through it, would stand on it, etc.

3 custom properties?

Everything you draw/place/... in Tiled has some properties, like position, color, etc. Additionally you can set custom properties that have a type and value you choose.

Custom properties can have one of various data types. The most important are:

- bool: yes/no, true/false, checked box/unchecked box, etc.
- string: normal text

3.1 visual manual

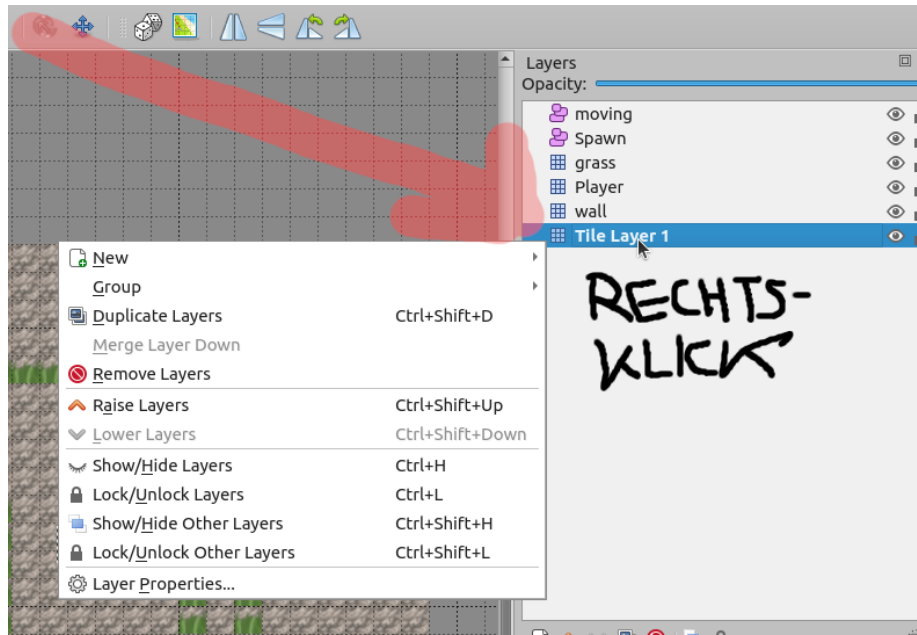


Figure 1: Right click on the layer.

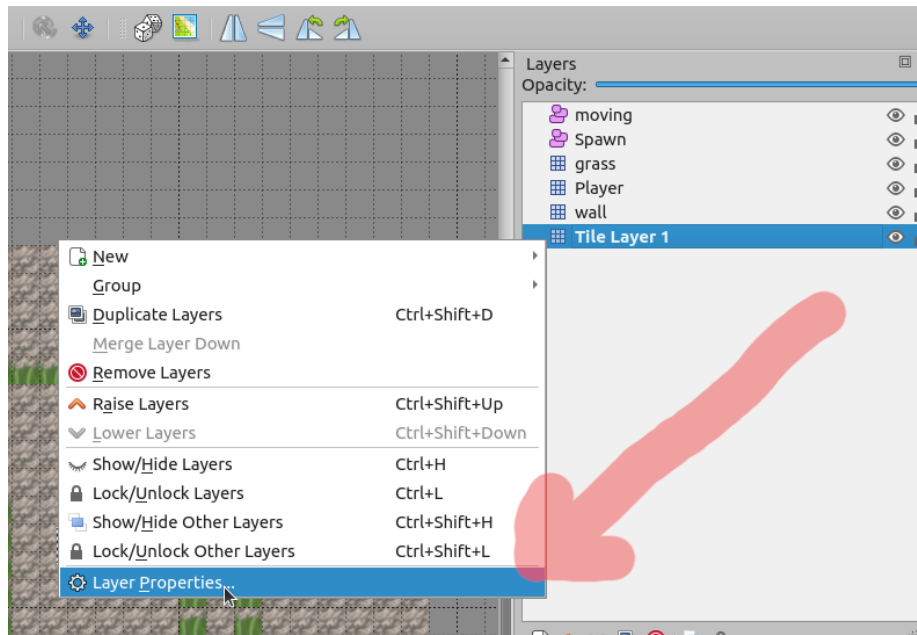


Figure 2: Left click on "Layer Properties..."

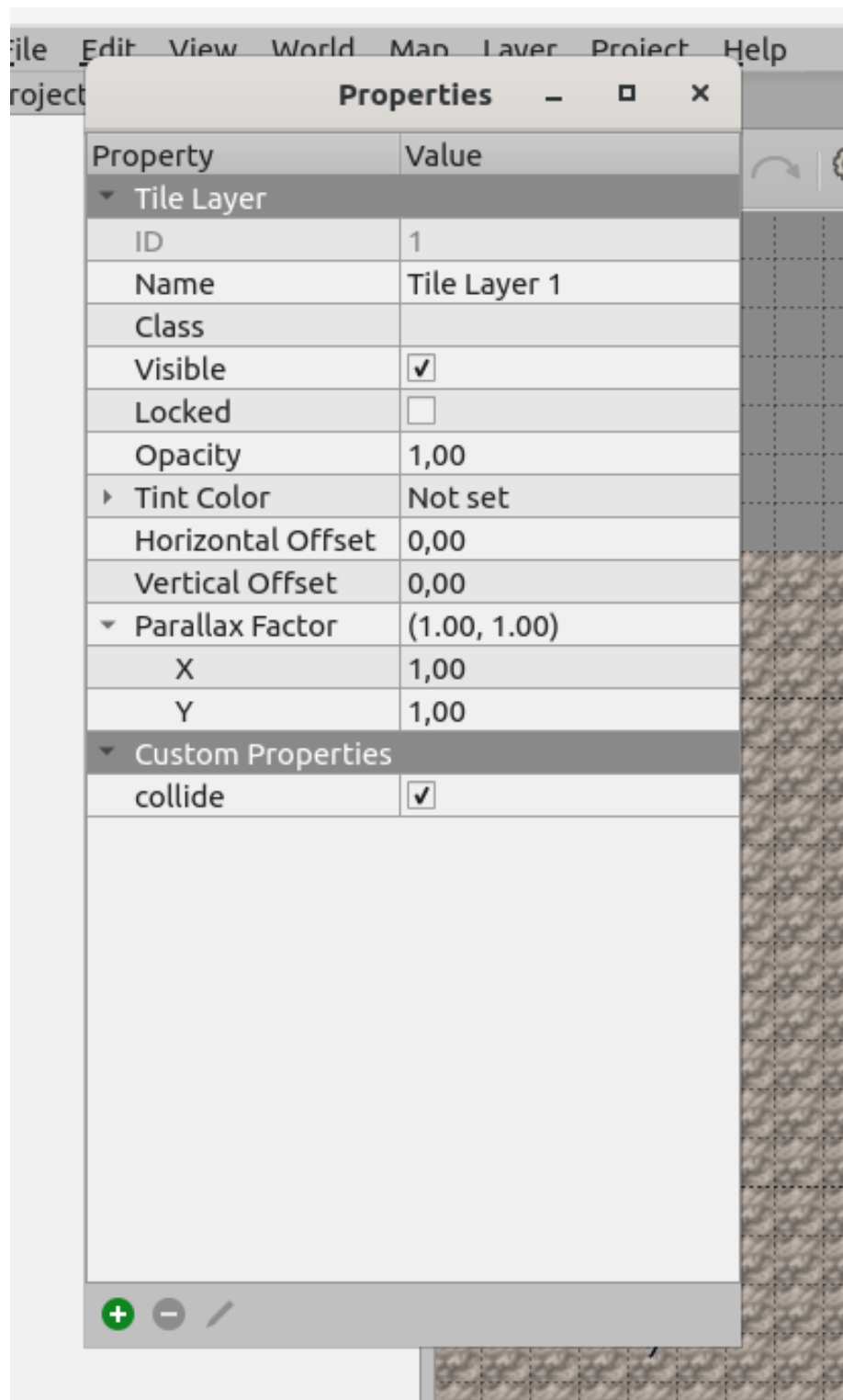
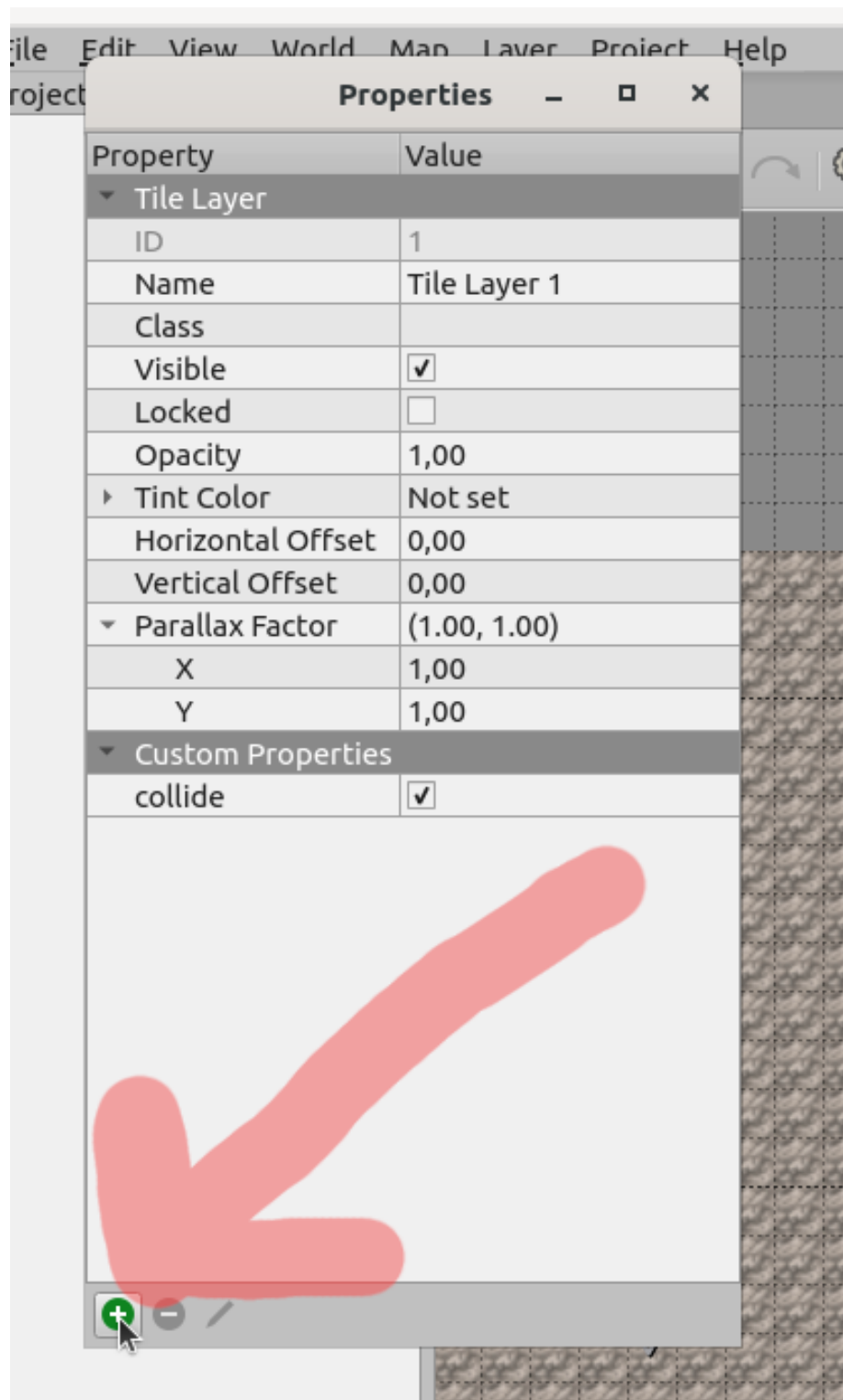


Figure 3: Now you see the layer properties. Maybe you see it in some area in Tiled instead of a new window. It doesn't matter.



5

Figure 4: To add a custom property, left click on the green + icon.

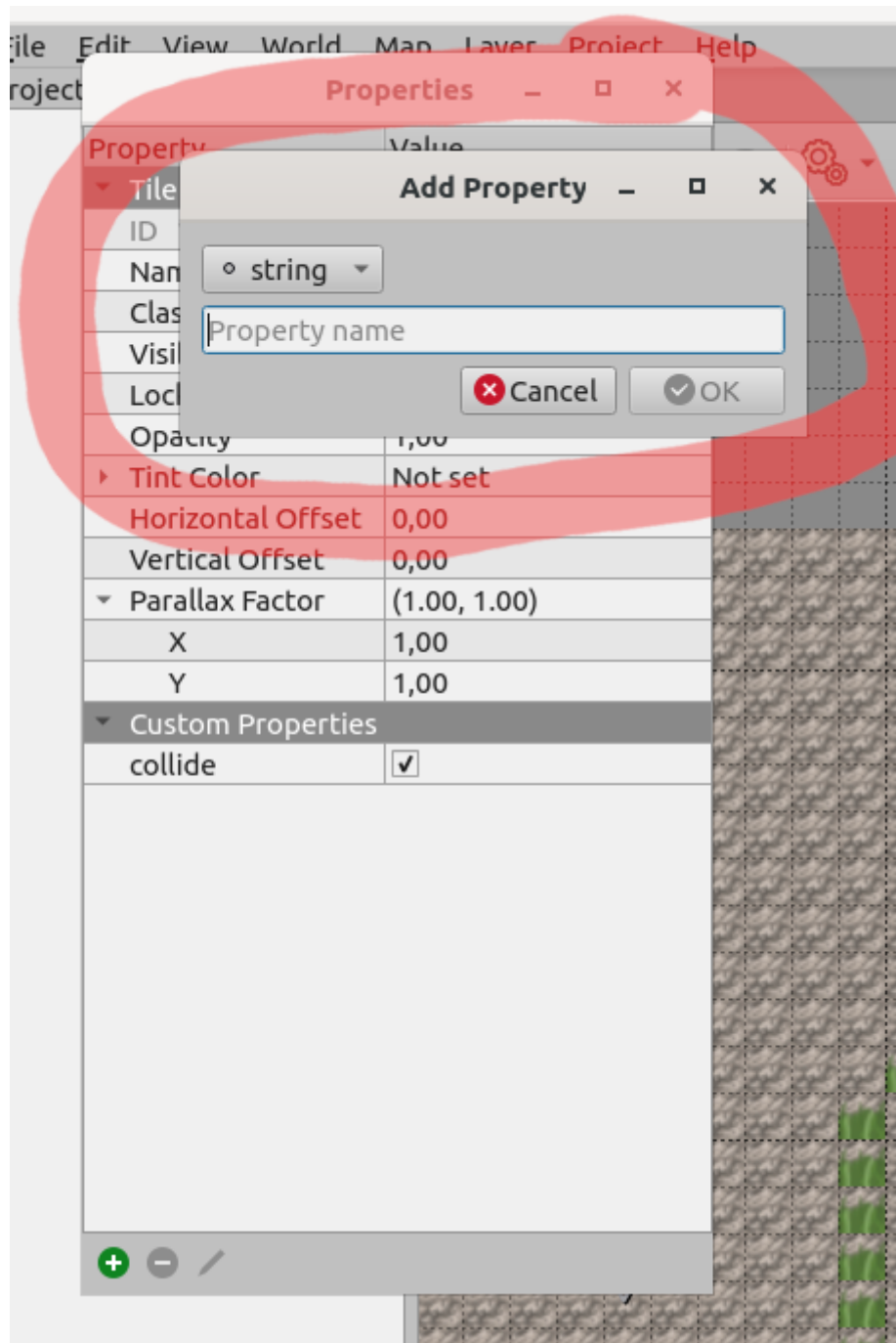


Figure 5: The property adding window opens. You can select the data type (in the image string is selected) and the property name.

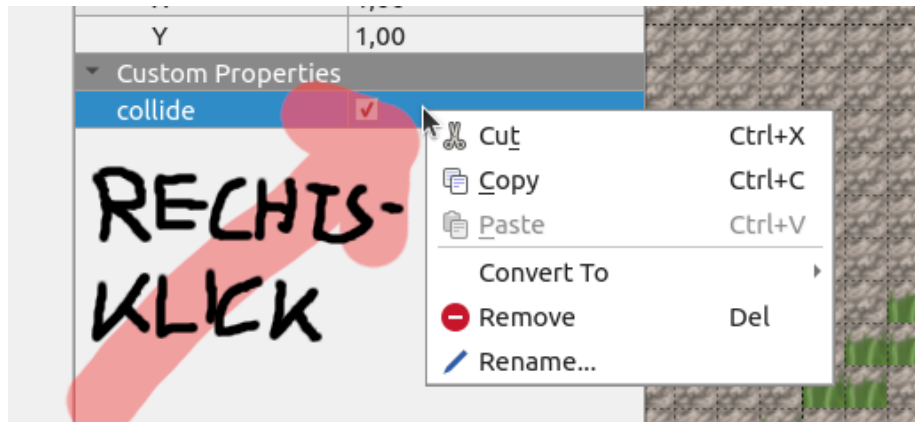


Figure 6: To edit a custom property, right click it.

4 layers?

4.1 tile layers

A tile layer consists of tiles. Tiles are the individual parts of the spritesheet. Tile layers are used for directly visible stuff. Here you're drawing.

4.2 object layers

An object layer consists of objects. Objects can be whatever you want and aren't directly visible. We use them for example for spawn points, platform movement paths, etc.

Objects can be of various types. Currently we only use points. As the name implies, they simply are a point in the map.

4.2.1 visual manual

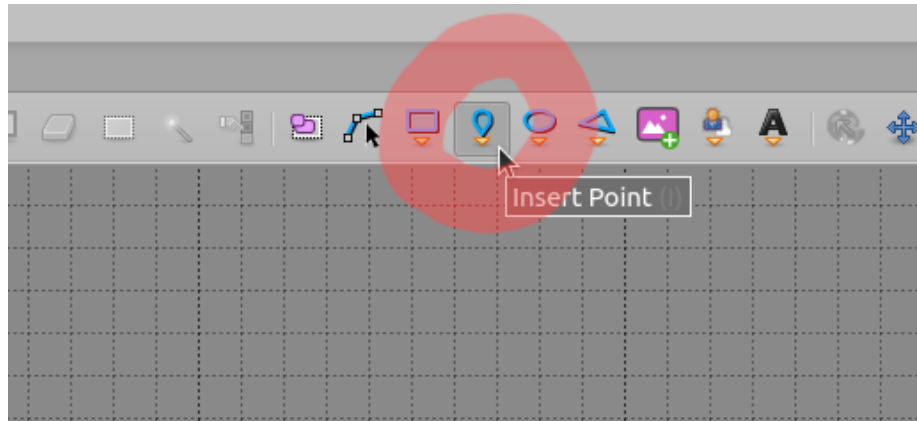


Figure 7: This is the button to select the point adding tool.

4.3 object layer lists

You can see the list of objects of an object layer and edit the individual name and properties of an object.

4.3.1 visual manual

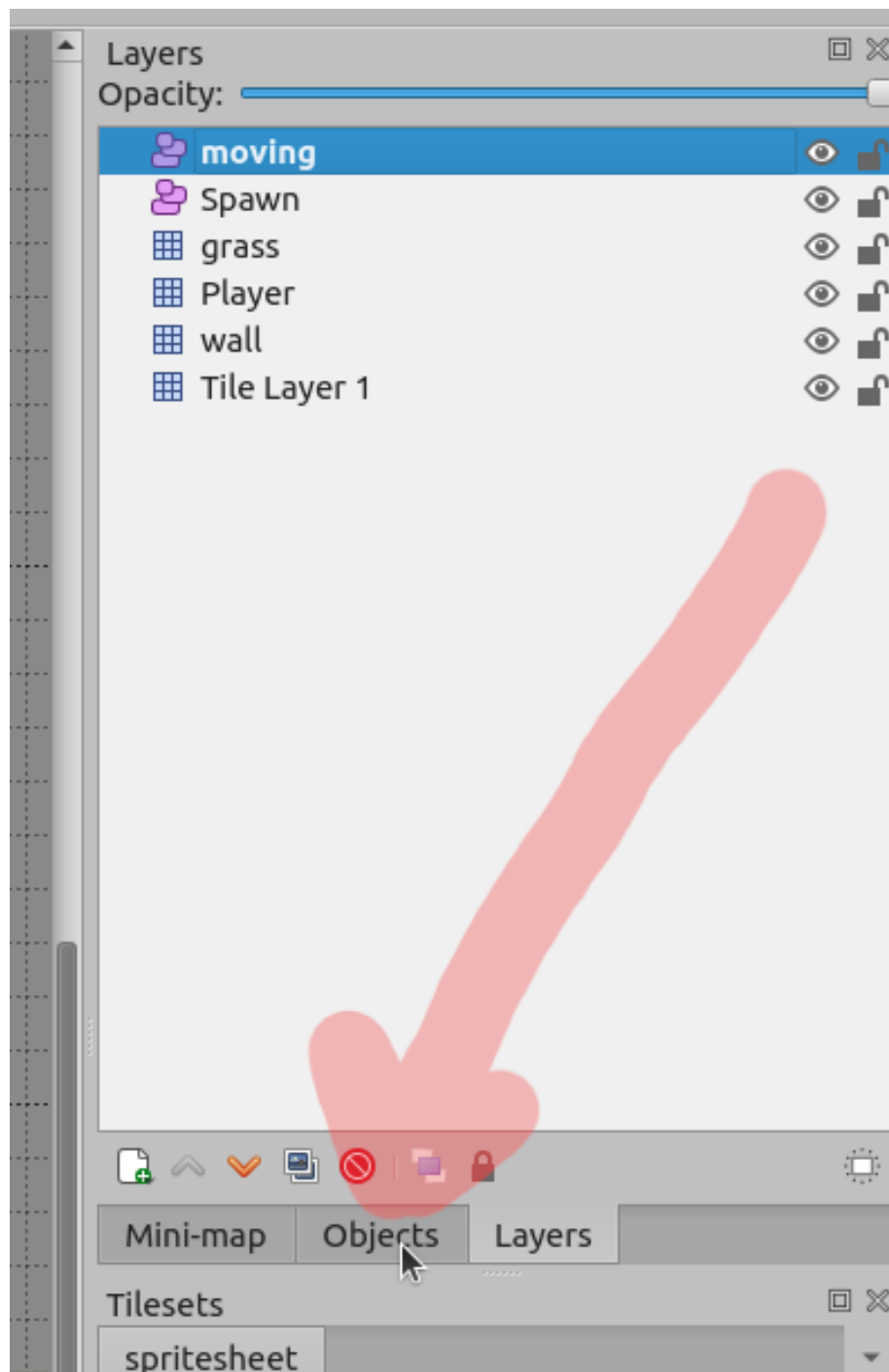


Figure 8: To see the object lists, click on the Objects tab.

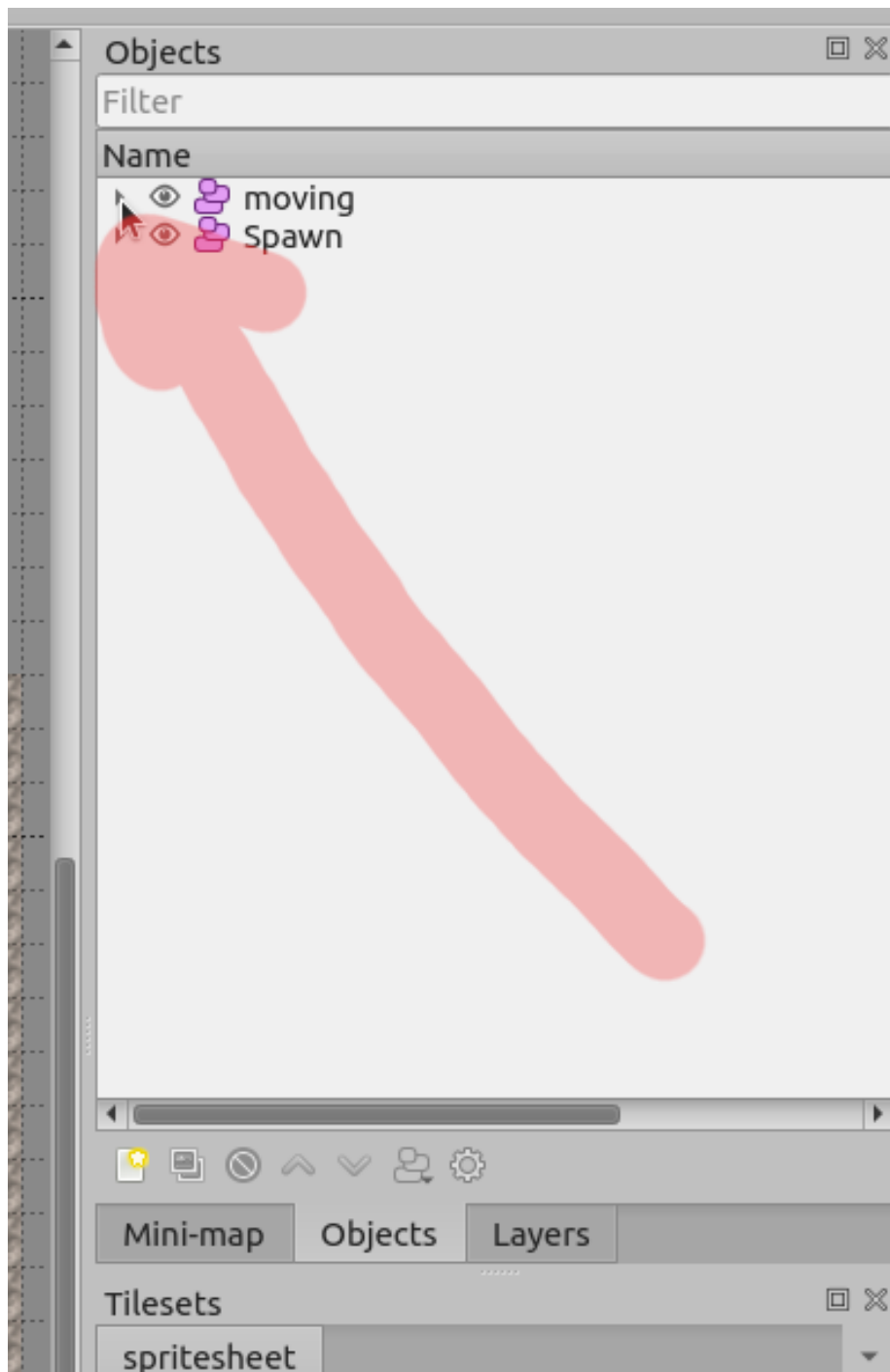


Figure 9: Expand the list of the wanted layer.

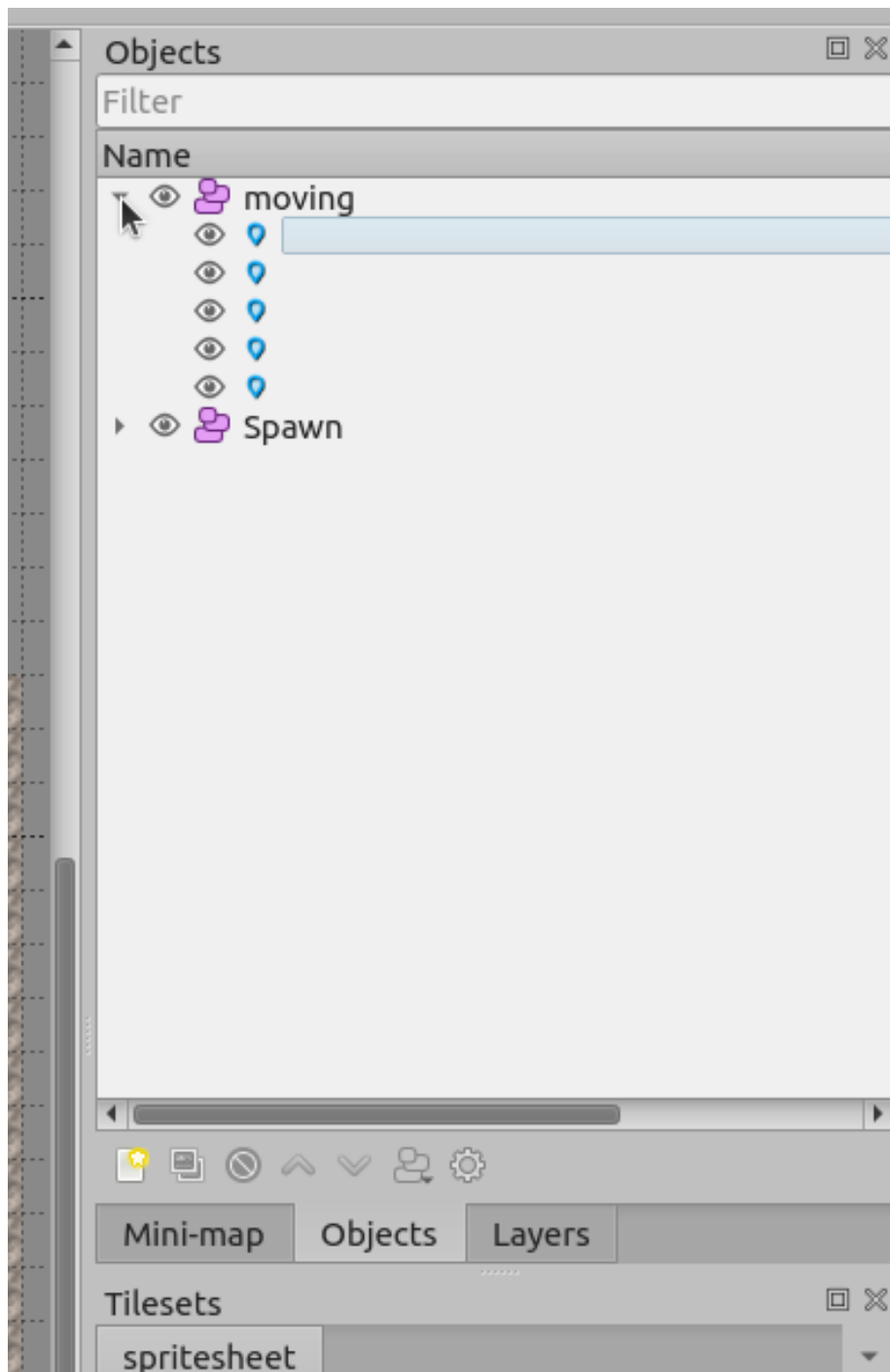


Figure 10: Now you see every object that's in the layer. In the image the layer 'moving' has five points.

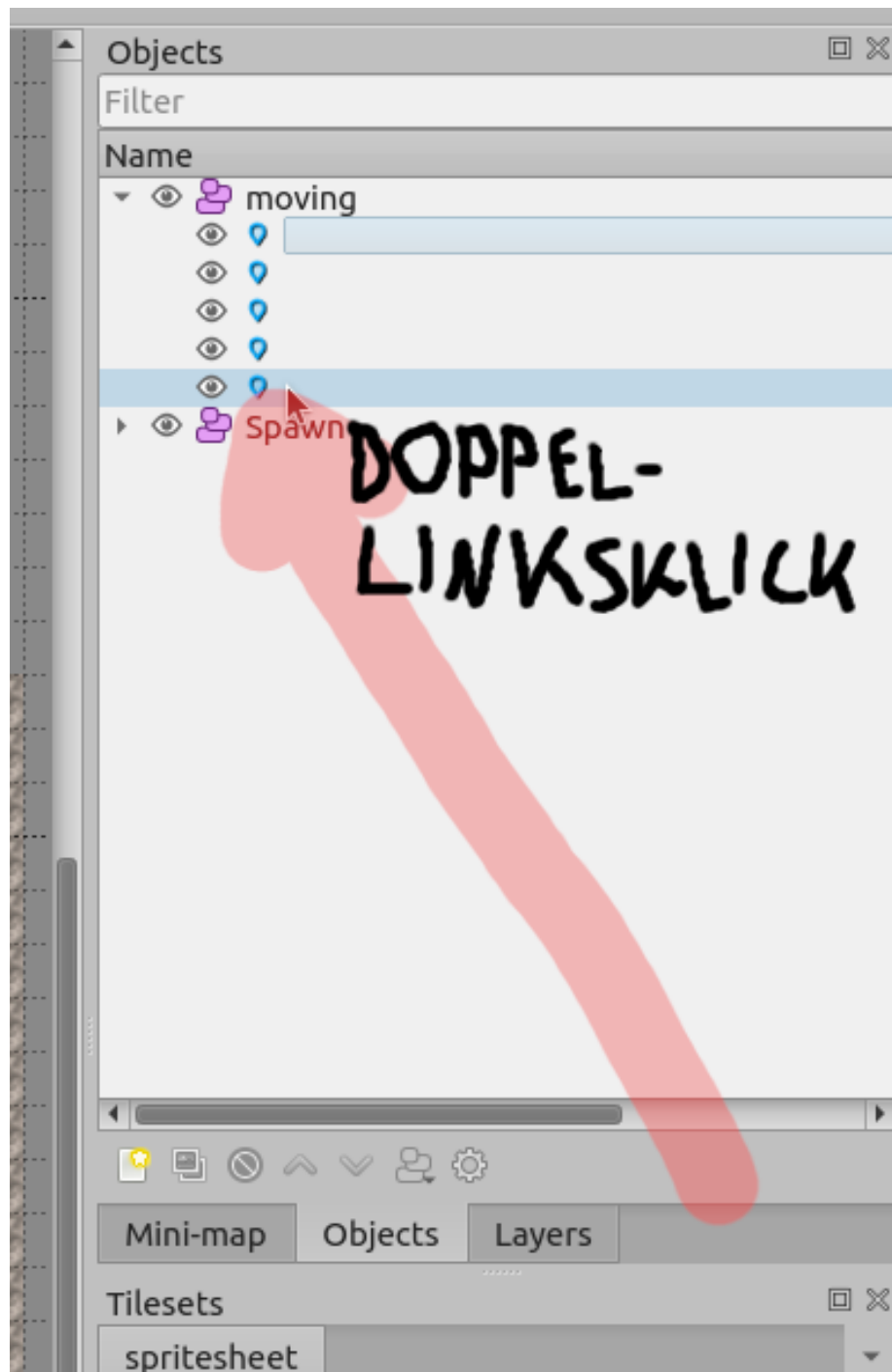


Figure 11: To edit the name of an object, double left click it.

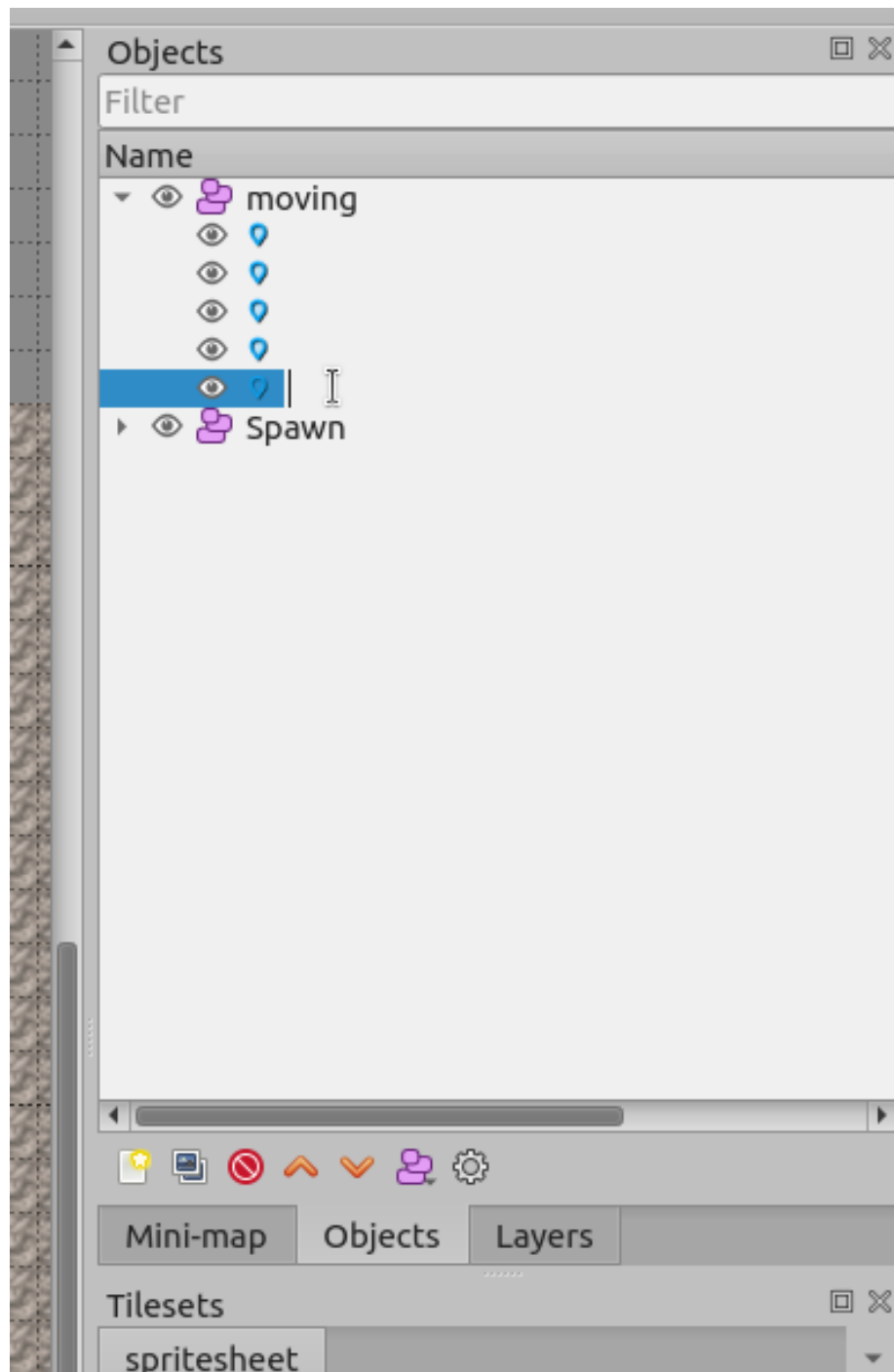


Figure 12: Then type whatever you want.

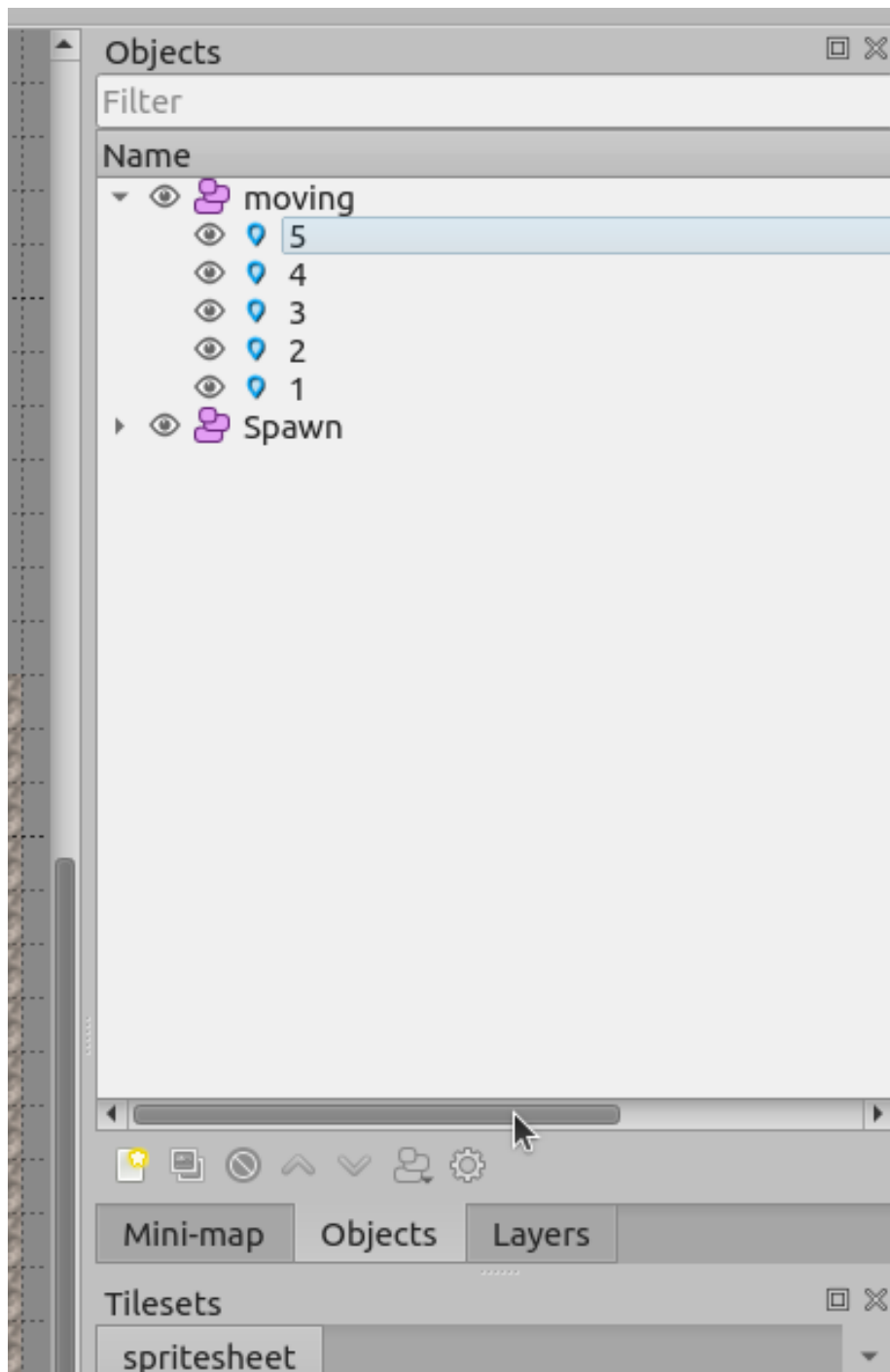


Figure 13: In the image the five points were numbered 1 to 5.

5 Tiled special custom properties

Until now all special custom properties are used on layers and NOT an individual tiles. For the collide property this has serious impact on file sizes and using this style everywhere makes the usage easier.

5.1 collide

- Name: collide
- Type: bool
- Meant for: Tile layers

All tiles of the tile layer will collide with the player.

5.2 kill

- Name: kill
- Type: bool
- Meant for: any layer

Everything in this layer will kill the player on contact. Tiles, enemies, etc.

5.3 teleport_to

- Name: teleport_to
- Type: string
- Meant for: any layer

When the player hits anything in this layer, he/she gets teleported to the map with the name set in the teleport_to property.

5.4 visual examples

X	1,00
Y	1,00
▼ Custom Properties	
collide	<input checked="" type="checkbox"/>

Figure 14: This layer has the collide property set to true. The player will collide with every tile of it.

▼ Parallax Factor	(1.00, 1.00)
X	1,00
Y	1,00
▼ Custom Properties	
kill	<input checked="" type="checkbox"/>

Figure 15: This layer has the kill property set to true. The player will die when hitting something of this layer. This includes tiles and NPCs that got spawned because of the layer. (until now these two things are the only killing ones)

▼ Parallax Factor	(1.00, 1.00)
X	1,00
Y	1,00
▼ Custom Properties	
teleport_to	secret_chamber_map

Figure 16: This layer has the teleport_to property set to "secret_chamber_map". If the player hits anything in this layer, he/she will get teleported to the map that's saved in the file named "secret_chamber_map".

6 Tiled special layers

6.1 Spawn

- Name: Spawn
- Type: Object layer

Contains the players spawn. The first object of the layer is selected and used as the spawn position.

6.2 Player

- Name: Player
- Type: Tilelayer

Contains nothing. The layer at which the player will be displayed. Layers above will be displayed above the player and layers underneath will be displayed behind the player.

6.3 moving*

- Name: moving*write whatever you want here, only the "moving" prefix is important*
- Type: Object layer

Contains two or more points. The points mark the path which the moving platform will follow. The object layer itself has the following parameters (still work in progress):

- platform look
- platform size
- platform speed

In the Tiled Objects view, the points have to be numbered. The movement path just loops through the points going from lowest to biggest. E.g. with the points named 1, 2, 3 and 4, the platform will first go to 1, then 2, then 3, then 4, then 1, then 2, then 3, then 4, then 1 and so on.

6.4 npc*

- Name: *npcwrite whatever you want here, only the "npc" prefix is important*
- Type: object layer

Contains points. Each point needs the custom property 'type' which determines the type of NPC. It can be one of the following:

- work in progress...

Points with an invalid type (including no type) will be ignored. A point only determines the spawn location of an NPC. After spawning they behave on their own.

To set enemies, i.e. NPCs that kill on contact, use the custom 'kill' property.