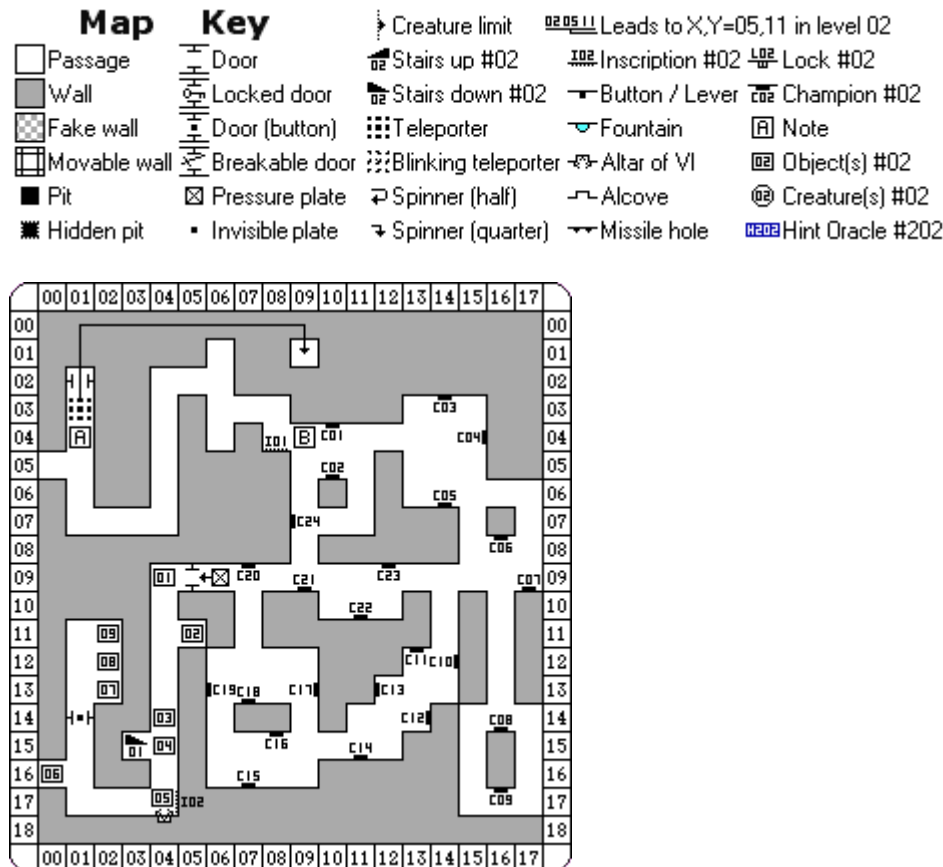


Using Tiled To Create Dungeon Levels

The first level of Dungeon Master from <http://dmweb.free.fr/?q=node/134>

This is listed as Level 00



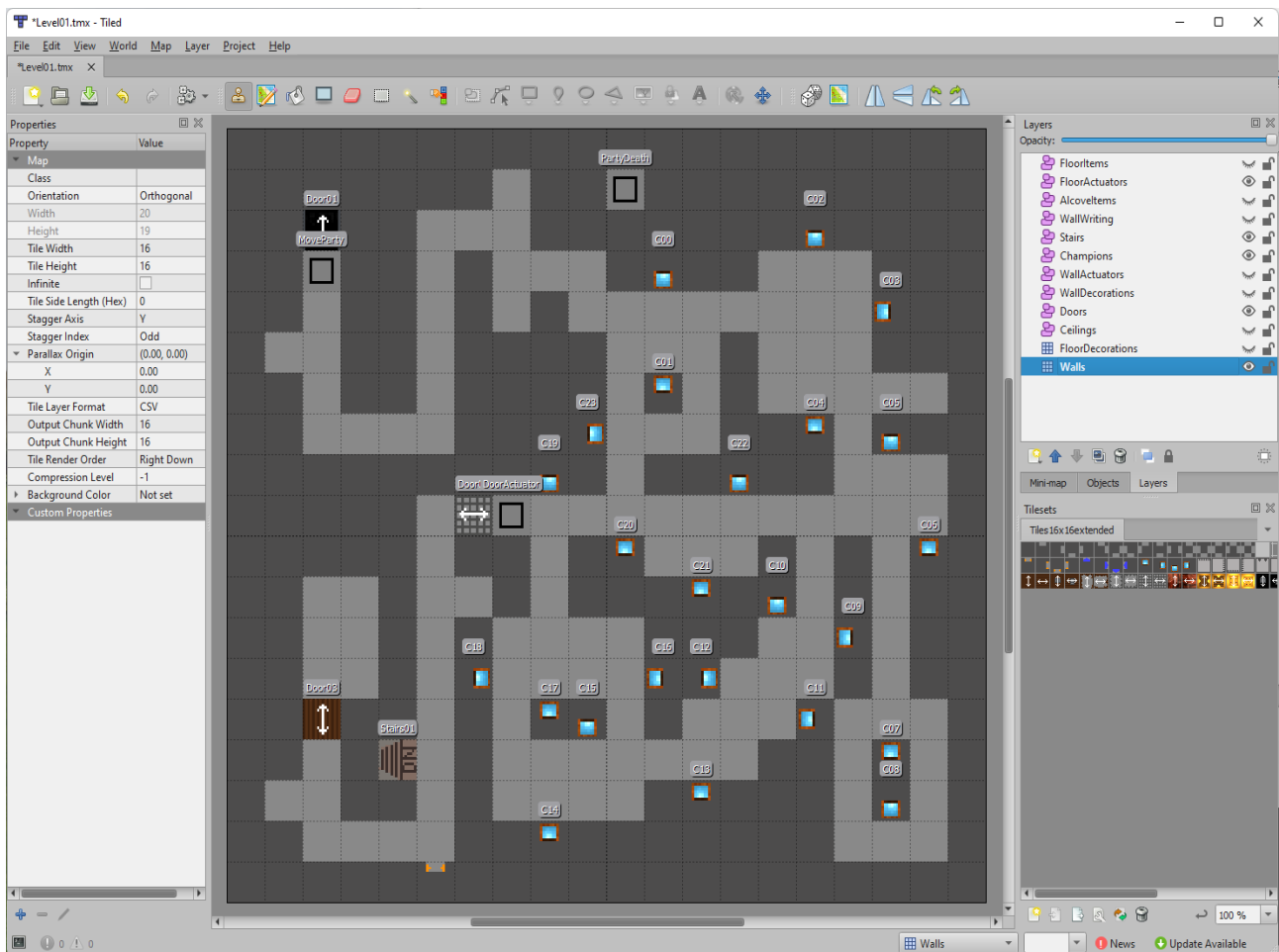
Tiled is usually used to create 2D games by allowing the creator to use a “tileset” of similar sized images to build levels. It can be mis-used for the creation of Dungeon Master levels.

The first thing required is a tileset:



These images are NOT used in the game, they simply represent some of the walls, doors, stairs, alcoves, altars to allow a map to be created that looks credible.

Here is the first level made in Tiled using this tileset:



Note the following:

A continuous outer wall surrounds the map, unlike the downloaded image show above, where the empty cells reach the edge.

There are 3 doors, marked with the direction they open Black, Grate and Wood

There is a staircase going down

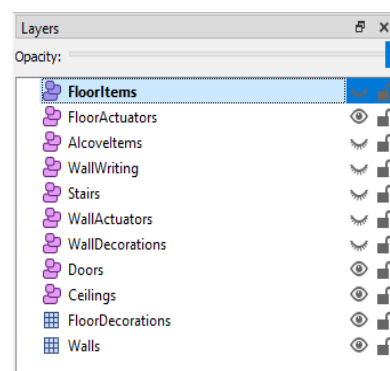
There are Mirrors containing Heros

There are puddle/slime floor decorations

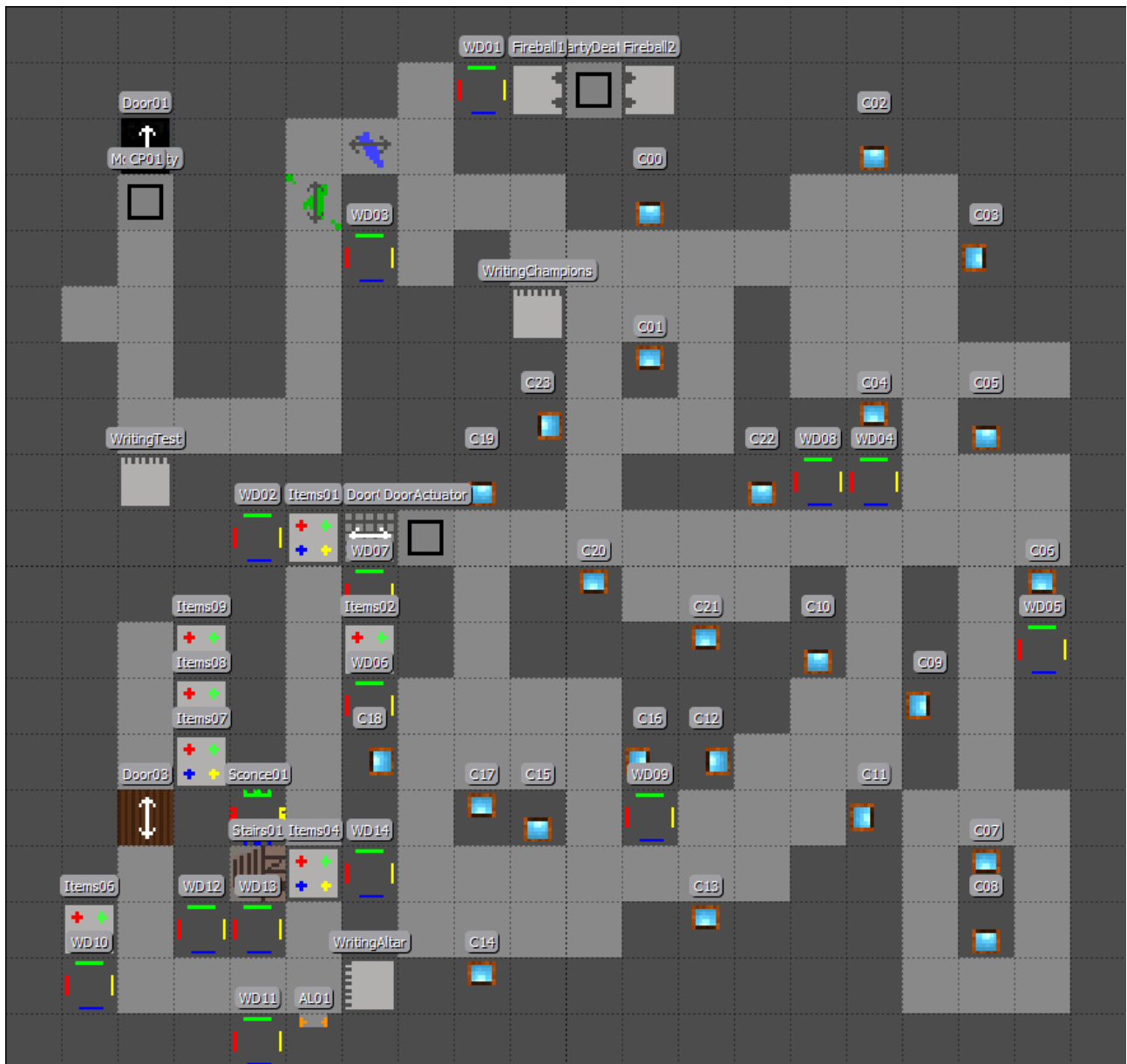
There is an altar of re-birth

Only 5 of the 12 layers have been set to visible.

These include the only 2 Image layers



Most layers are Object Layers. When all layers are visible, this is the same map



This map is saved as Level00.tmx

The .tmx file format is xml compatible and so can be read in C# to obtain information about the level.

Below is a section of the file, showing the grid of “Walls” in csv format with a width = 20 and height = 19

0 represents empty cell and does not have an image from the tileset

1 represents solid wall and is the top left square on the tileset

50 and 52 on the second row represent the 50th and 52nd images: Fireball throwers!

```

<layer id="1" name="Walls" width="20" height="19">
  <data encoding="csv">
1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,
1,1,1,1,1,1,1,0,1,50,0,52,1,1,1,1,1,1,1,1,
1,1,0,1,1,0,0,0,1,1,1,1,1,1,1,43,1,1,1,1,
1,1,0,1,1,0,1,0,0,0,1,43,1,1,0,0,0,1,1,1,
1,1,0,1,1,0,1,0,1,0,0,0,0,0,0,0,0,44,1,1,
1,0,0,1,1,0,1,1,1,1,0,0,0,1,0,0,0,1,1,1,
1,1,0,1,1,0,1,1,1,1,0,41,0,1,0,0,0,0,0,1,
1,1,0,0,0,0,1,1,1,42,0,0,0,1,1,41,0,43,1,1,
1,1,1,1,1,1,1,1,43,1,0,1,1,43,1,1,0,0,0,1,
1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,
1,1,1,1,1,0,1,1,0,1,41,0,0,0,1,0,1,0,41,1,
1,1,0,0,1,0,0,1,0,1,1,1,41,1,43,0,1,0,1,1,
1,1,0,0,1,0,1,0,0,0,0,1,1,1,0,0,44,0,1,1,
1,1,0,0,1,0,42,0,0,0,0,44,42,0,0,0,1,0,1,1,
1,1,0,1,1,0,1,0,41,43,0,1,0,0,0,44,0,0,0,1,
1,1,0,1,27,0,1,0,0,0,0,0,0,0,1,1,0,41,0,1,
1,0,0,1,1,0,1,0,0,0,0,1,41,1,1,1,0,43,0,1,
1,1,0,0,0,0,1,1,41,1,1,1,1,1,1,0,0,0,1,
1,1,1,1,1,33,1,1,1,1,1,1,1,1,1,1,1,1,1,1
</data>
</layer>
<object id="54" name="Door03" gid="65" x="32" y="240" width="16" height="16">
  <properties>
    <property name="Control" value="Button"/>
    <property name="ControlLocation" value=""/>
    <property name="IsBreakable" type="bool" value="false"/>
    <property name="IsToggled" type="bool" value="true"/>
    <property name="OpenState" type="float" value="0"/>
  </properties>
</object>
</objectgroup>
<objectgroup id="16" name="WallDecorations">
  <object id="51" name="WD02" class="Decorations" gid="62" x="64" y="160" width="16"
height="16">
    <properties>
      <property name="E" value="Manacles"/>
      <property name="N" value=""/>
      <property name="S" value=""/>
      <property name="W" value=""/>
    </properties>
  </object>

```

The next few lines are objects from the “Doors” layer.

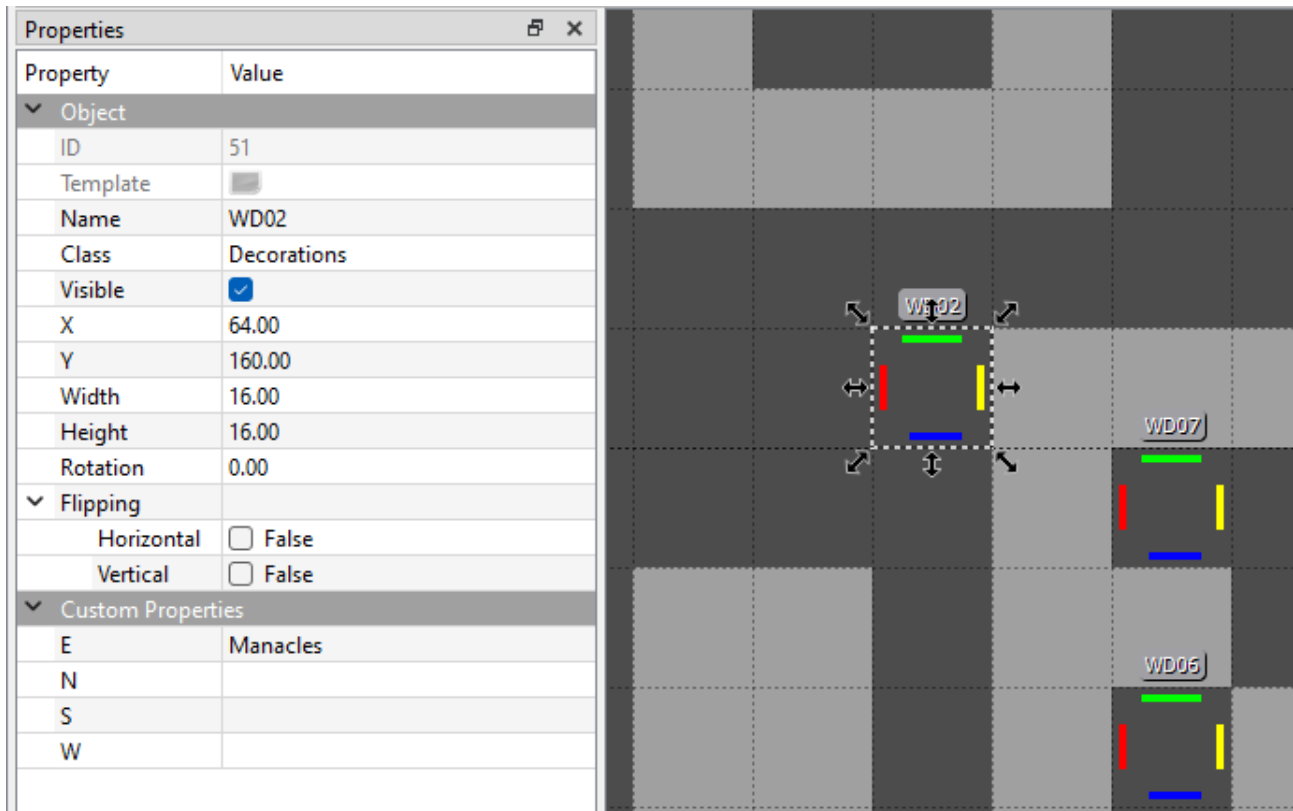
The next few lines are objects from the “WallDecorations” layer.

Let’s break this down into component parts:

Select the WallDecorations Object layer

Click on “WD02” at grid ref 4, 9 (using the top left as 0,0)

Look at the properties set for this object:

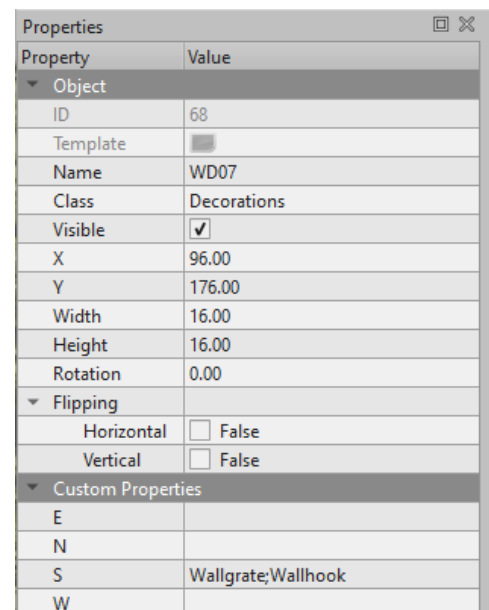


Here you can see the Name WD02, X and Y coordinates and custom Properties E,N,S,W of which the E property has “Manacles”

This means the East wall of the cell has Manacles as a wall decoration (equates to the yellow bar)

If you click on WD07 it has 2 items in it's South wall: Wallgrate;Wallhook

The items are separated by a semicolon.
Compare this with the XML file segment:
All this information is present in the xml file:



```
<object id="68" name="WD07" class="Decorations" gid="62" x="96" y="176" width="16" height="16">
  <properties>
    <property name="E" value=""/>
    <property name="N" value=""/>
    <property name="S" value="Wallgrate;Wallhook"/>
    <property name="W" value=""/>
  </properties>
</object>
```

The xml file is read directly to deal with adding WallDecorations into the level using the information in the XML element

As you can see the combination of the grid of images and specific properties on a per cell basis can provide all the information required to start creating a level in code.

The next thing needed is a way of translating the grid of numbers into something the code can use. For this a text file called TiledToItem.txt is used. Here are the first 10/71 lines:

```
1=Wall:Wall;Wall;Wall;Wall
2=Wall:Alcove;Wall;Wall;Wall
3=Wall:Wall;Alcove;Wall;Wall
4=Wall:Wall;Wall;Alcove;Wall
5=Wall:NESW:Wall;Wall;Wall;Alcove
6=Wall:Alcove;Alcove;Wall;Wall
7=Wall:Wall;Alcove;Alcove;Wall
8=Wall:Wall;Wall;Alcove;Alcove
9=Wall:Alcove;Wall;Wall;Alcove
10=Wall:Alcove;Wall;Alcove;Wall
```



You should be able to see the relationship between tile 1 on the image and the description in the text file:

1	Tile no = 1
Wall:	Wall (not door, pillar, hidden etc)
Wall;Wall;Wall;Wall	WallType = N, E, S, W

When used in code, this file can convert no 1 tile into a cell surrounded by 4 plain walls. If the cell was image 10, then there are alcoves on North/South walls, plain on East/West.

This file is used only for walls, floor decorations (slime/puddle) and Door types. *Update: This file is now included in the database in a table called 'TiledToItem'*

The XML file is read in and used to create a 2D array of cell objects. Each cell has many properties and methods to work with this information.

Creating a new Dungeon Map using Tiled

<https://thorbjorn.itch.io/tiled?download>

Using the methods discussed above this is how to create a map of Dungeon Master level 01

Map from <http://dmweb.free.fr/?q=node/135>

Dungeon Master Level 01 (The maps are listed as starting from level 00)

Mon, 2005-10-24 01:00 — ChristopheF

Map

Map	Key			
 Passage	 Door	 Creature limit	 Leads to X,Y=05,11 in level 02	
 Wall	 Locked door	 Stairs up #02	 Inscription #02	 Lock #02
 Fake wall	 Door (button)	 Stairs down #02	 Button / Lever	 Champion #02
 Movable wall	 Breakable door	 Teleporter	 Fountain	 Note
 Pit	 Pressure plate	 Blinking teleporter	 Altar of VI	 Object(s) #02
 Hidden pit	 Invisible plate	 Spinner (half)	 Alcove	 Creature(s) #02
		 Spinner (quarter)	 Missile hole	 Hint Oracle #202

Items

- 01 (06,14) Club
- 02 (06,17) Corn, Apple, Scroll ("Small details can hide great rewards"), Dagger, Falchion
- 03 (00,16) Gold Key
- 04 (01,23) Topaz Key
- 05 (07,23) Leather Boots, Gold Key
- 06 (08,19) Elven Doublet, Dagger, Arrow, Torch (Charges=15), Emerald Key
- 07 (04,30) Rock
- 08 (15,24) Iron Key, Ghi Trousers
- 09 (14,21) Solid Key
- 10 (13,22) Falchion
- 11 (14,22) Torch (Charges=15), Leather Boots
- 12 (08,17) Gold Key
- 13 (13,19) Gold Key
- 14 (25,14) Boulder
- 15 (30,15) Gold Key, Apple
- 16 (17,25) Apple
- 17 (29,25) Rock
- 18 (31,20) Iron Key
- 19 (31,24) Key of B
- 20 (19,25) Scroll ("Casting Vi into a flask creates a serum that heals wounds"), Empty flask (3), Leather pants, Scroll ("Casting Vi Bro into a flask creates a serum for curing poison"), Drumstick
- 21 (16,27) Apple, Cheese (2), Torch (Charges=15)
- 22 (13,28) Gold Key, Throwing Star
- 23 (30,33) Copper Coin
- 24 (30,39) Torch (Charges=15)
- 25 (27,43) Chest [Scroll ("Drink these to gain magical defense"), Ya potion (2)]
- 26 (12,44) Throwing Star
- 27 (13,43) Water (Charges=3)
- 28 (14,42) Apple
- 29 (16,42) Falchion
- 30 (15,45) Magical Box (Blue)
- 31 (05,32) Buckler
- 32 (02,45) Chest [Drumstick, Corn, Cheese, Bread (2), Magical Box (Blue), Scroll ("Des Ven will conjure a poison spell"), Empty Flask]
- 33 (06,43) Torch (Charges=15)

Inscriptions

- I01 (29,16) "Step inside take a ride"
- I02 (22,22) "This wall says nothing"
- I03 (13,30) "To close pit leave a valuable on floor"

I04 (28,41) "None shall pass"

I05 (18,40) "This fountain accepts one wish."

Locks

L01 (00,16) Gold Key

L02 (03,19) Topaz Key

L03 (05,19) Gold Key

L04 (11,27) Emerald Key

L05 (02,27) Iron Key

L06 (05,27) Solid Key

L07 (10,14) Gold Key

L08 (22,19) Gold Key

L09 (28,18) Gold Key

L10 (30,21) Iron Key

L11 (24,24) Key of B

L12 (21,35) Gold Key

L13 (18,40) Copper Coin

Creatures

01 (06,16) Mummy

02 (04,27) Screamer

03 (11,19) Screamer

04 (13,18) Screamer (4)

05 (22,19) Screamer (2)

06 (27,16) Mummy

07 (31,23) Screamer

08 (24,27) Mummy

09 (16,30) Screamer (2)

10 (27,34) Mummy (3)

11 (20,43) Screamer (4)

12 (08,38) Screamer (4) [Torch (Charges=15), Empty Flask]

13 (05,38) Screamer [Throwing Star]

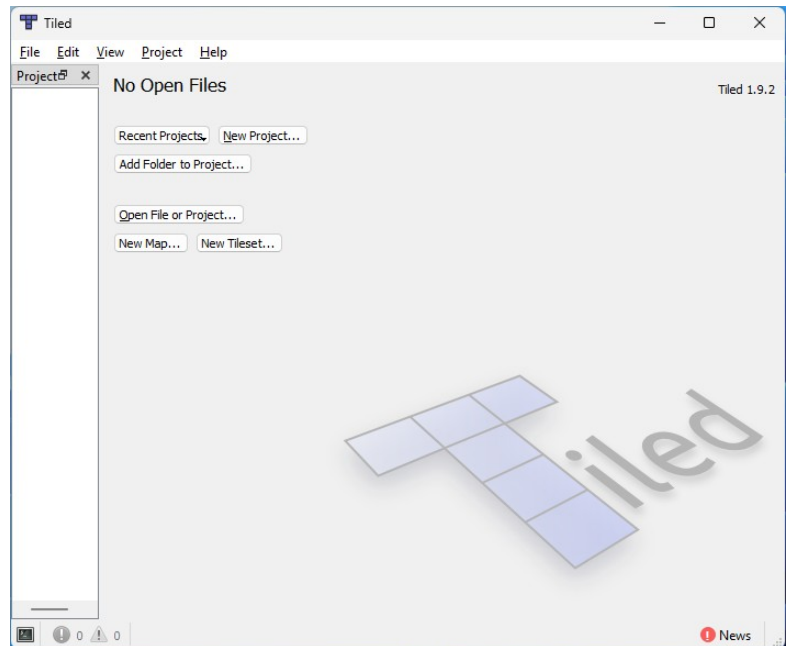
14 (03,32) Mummy (4)

15 (01,33) Mummy (2)

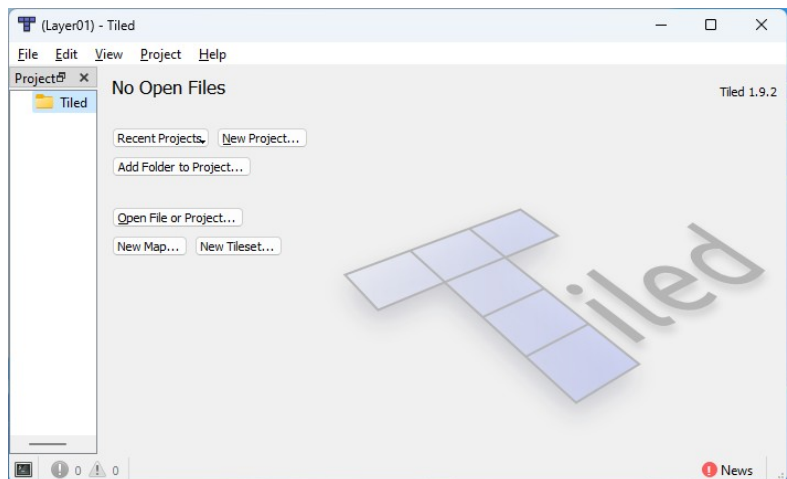
Start Tiled:

Click New Project.

Save in your Assets/Tiled folder as Layer01



Click New Map



The settings can be left as seen

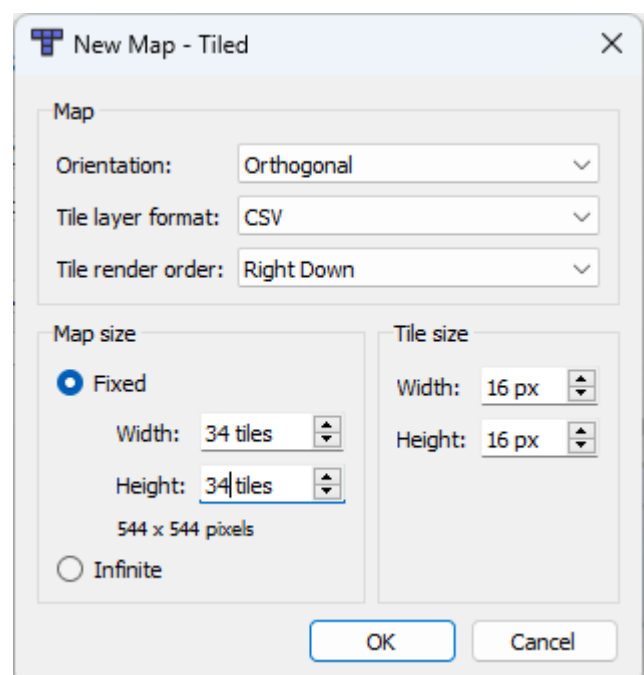
Change the Tile Size width to 16px
Change the Tile Size height to 16px

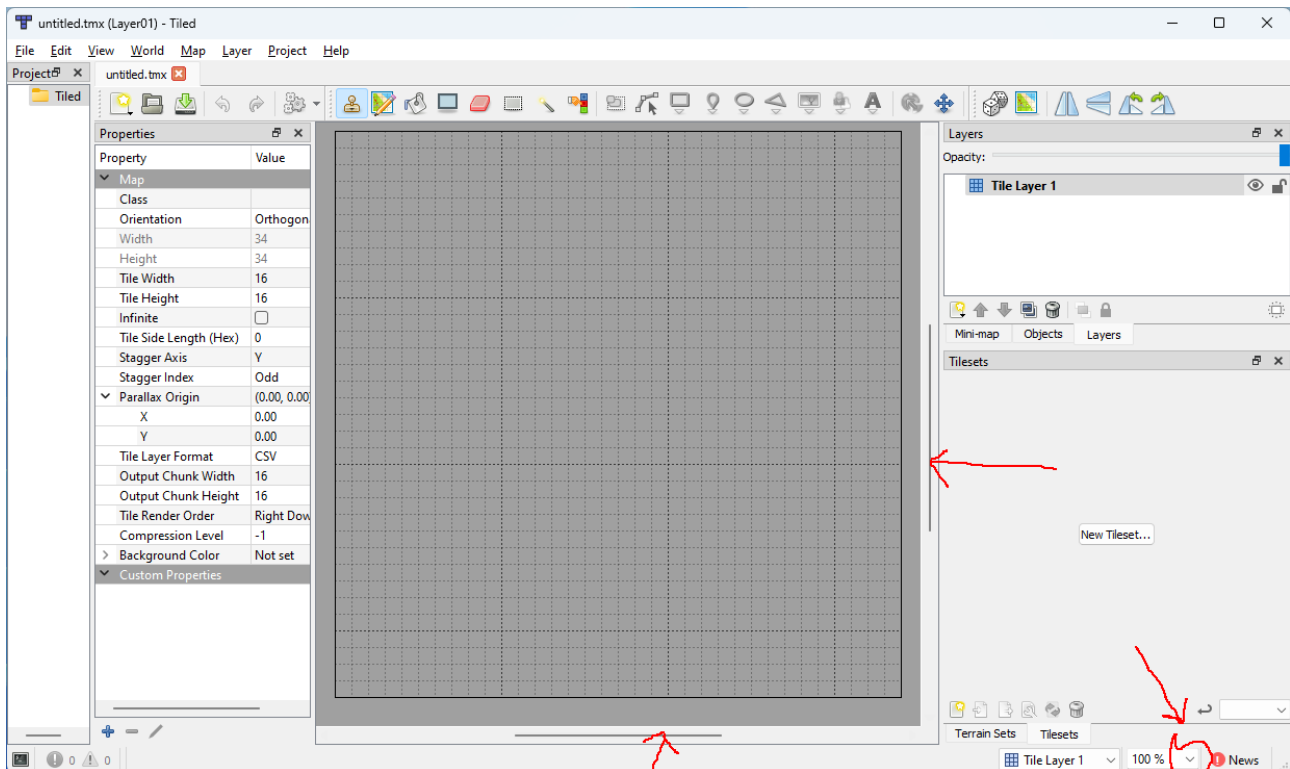
Count the number of squares from the downloaded map and add 2 to width and height to compensate for surrounding the dungeon with solid tiles. (with map 01 it is 34 x 34)

Change the Map size to Width 34 and Height 34

Click OK.

You may not see your map yet...



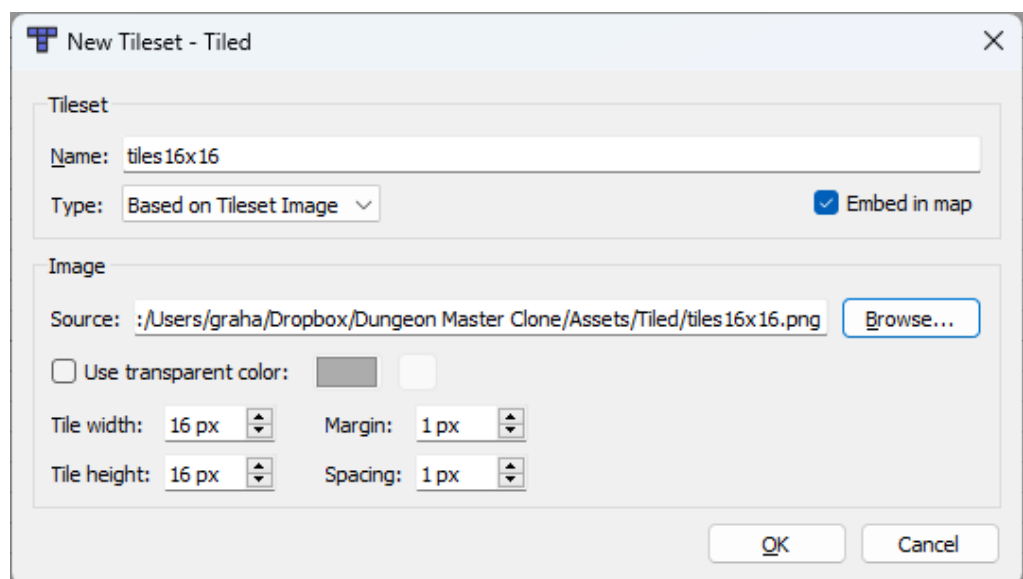


Change the zoom to 100%

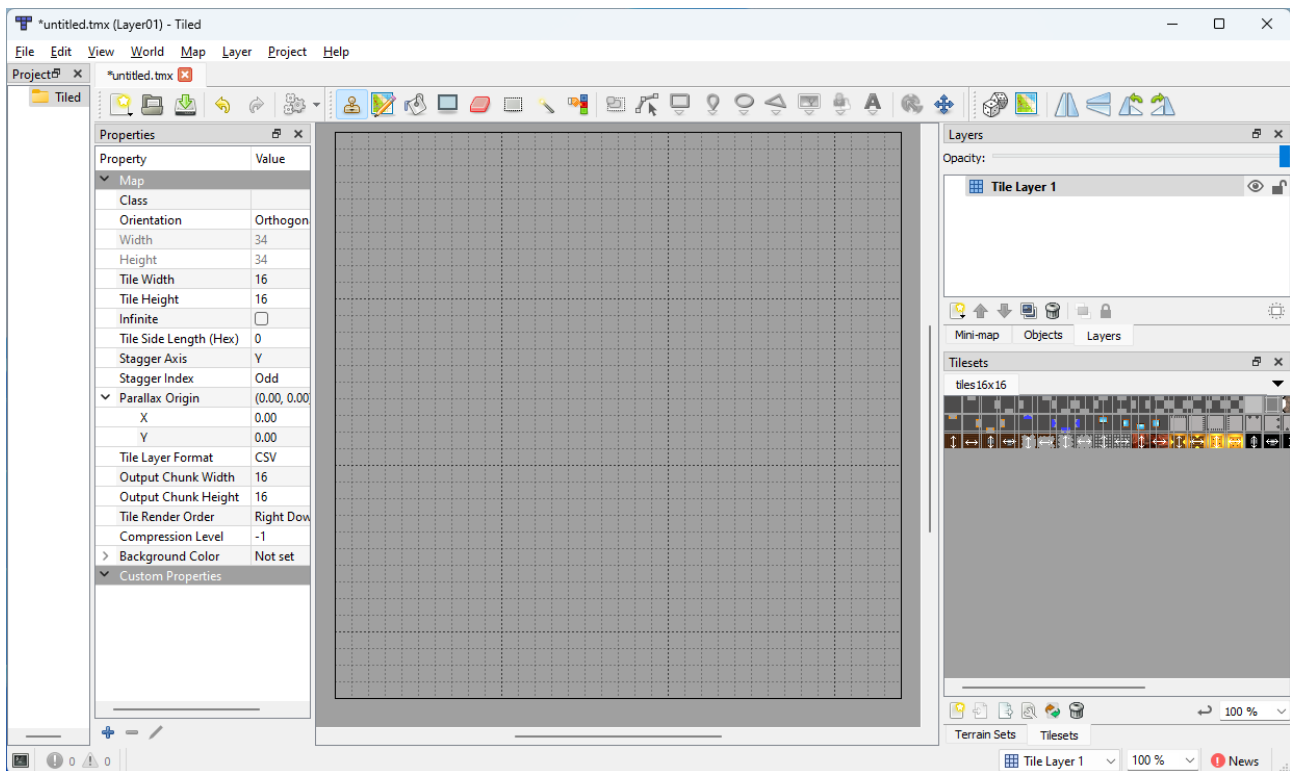
Use the sliders to adjust the position so you can see the entire map.

Click on New Tileset in the middle of the lower right panel

Click the Browse button and load the file tiles 16x16.png (Download from <https://github.com/Inksaver/MonogameDungeonMaster/blob/main/Assets/Tiled/tiles16x16.png>)



Adjust settings as displayed here. Click OK



Menu → File → Save → Level01.tmx

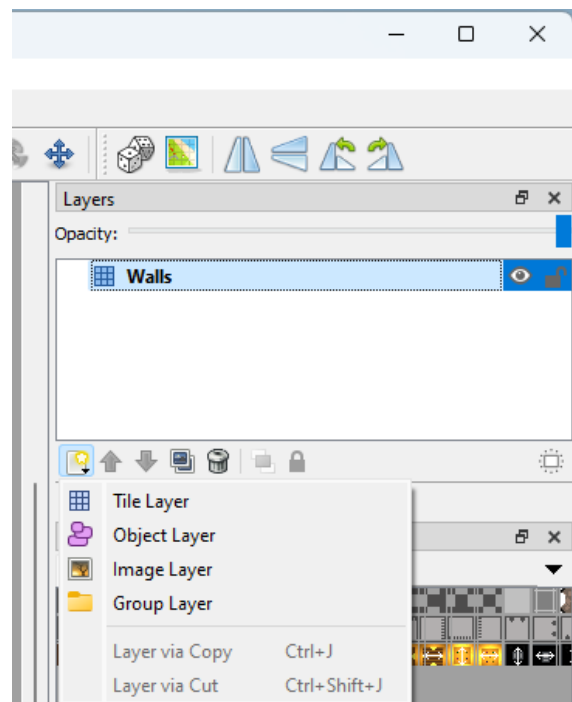
(The file Level00.tmx shown at the start of this document has been copied over to the same folder, so now both of them show in the Project Explore on the left side.)

Double-Click “Tile Layer “ in the Layers panel and rename it to “Walls”

Click on the New Layer Icon and select “Tile Layer”

Rename it FloorDecorations (camelCase)

It is essential you name the layers exactly as shown with the correct spelling and case as there are at least 2 C# programs that will be looking for these properties.



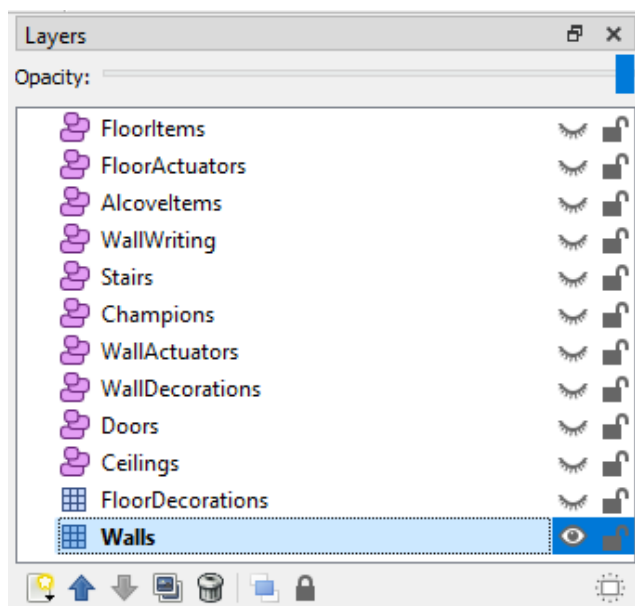
Create another layer, but this time choose **Object Layer**. Name it **FloorItems**

Create a total of 12 layers (including those already created) as follows:

Layer Name	Layer Type
FloorItems	Object
FloorActuators	Object
AlcoveItems	Object
WallWriting	Object
Stairs	Object
Champions	Object
WallActuators	Object
WallDecorations	Object
Doors	Object
Ceilings	Object
FloorDecorations	Tile
Walls	Tile

Use the up/down arrows to put them in the order above.

Close all the “eyes” except for Walls, and make sure that layer is selected:



Adjust the Tilesset zoom to 300%.

Select the top left tile: (Plain dark grey)

Use it draw round all edges of the map.

Follow the downloaded DM map to fill in all solid walls.

Do not add any walls with alcoves or altars at this stage

Correcting an error in the tileset:

Note in the tileset as shown here, there are now only 2 rows of tiles showing.

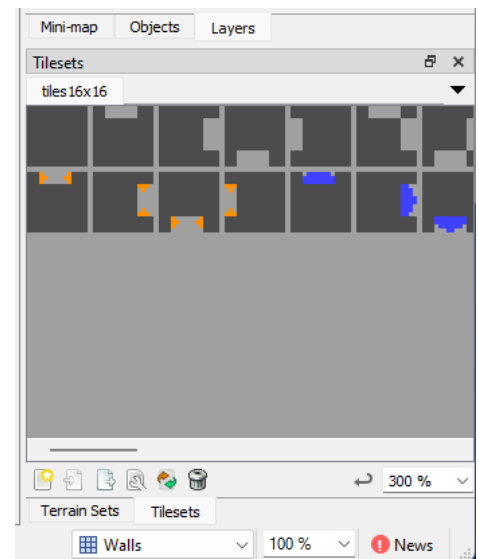
This occurred after all the layers were added.

The tileset had margin set to 1, should be 0!

Luckily there is a fix:

Save again and close Tiled.

Open the Level01.tmx file in a text editor



Edit the line as shown below:

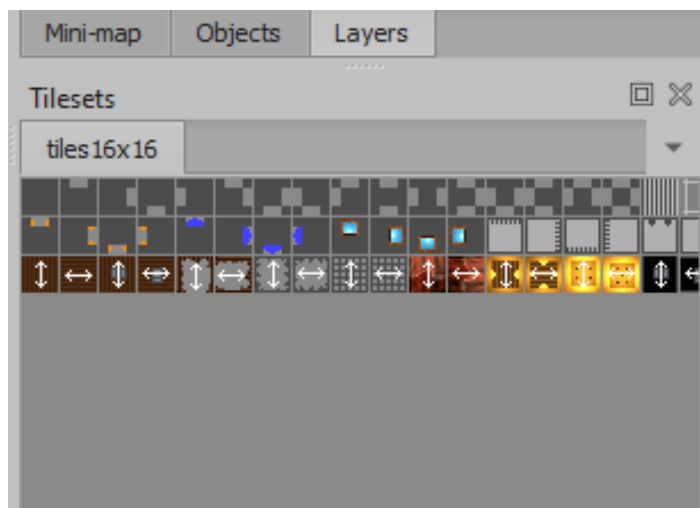
```
<tileset firstgid="1" name="tiles16x16" tilewidth="16" tileheight="16" spacing="1" margin="1" tilecount="62"
columns="31">
  <image source="tiles16x16.png" width="543" height="50"/>
</tileset>
```

Delete margin="1" and change tilecount="96" and columns="32"

```
<tileset firstgid="1" name="tiles16x16" tilewidth="16" tileheight="16" spacing="1" tilecount="96"
columns="32">
  <image source="tiles16x16.png" width="543" height="50"/>
</tileset>
```

Save and re-open Tiled:

All tiles are restored!



Here is a list of the meanings of all the tiles on the tileset 16x16.png:

Top row

1 = Solid wall normal (Image Layer)
2, 3, 4, 5 = Alcove on n, e, s, w (Image Layer)
6, 7, 8, 9, 10, 11 = Alcoves n+e, e+s, s+w, w+n, n+s, e+w (Image Layer)
12, 13, 14, 15, 16 = Alcoves n+e+s, e+s+w, s+w+n, w+n+e, n+e+s+w (Image Layer)
17 = Fake wall (Image Layer)
18 = Moveable wall (Image Layer)
19 = Ceiling Pit (Object Layer)
20, 21, 22, 23 = Stairs up n, e, s, w (Object Layer)
24, 25, 26, 27 = Stairs down n, e, s, w (Object Layer)
28 = Teleporter
29 = Flux cage
30, 31, 32 = Floor pit, Hidden Floor pit, Floorswitch (Object Layer)

Centre Row

33, 34, 35, 36 = altar on n, e, s, w
37, 38, 39, 40 = fountain on n, e, s, w
41, 42, 43, 44 = mirror on n, e, s, w (Object Layer)
45, 46, 47, 48 = writing on n, e, s, w (Object Layer)
49, 50, 51, 52 = WallActuator (launcher fireball/weapon on n, e, s, w) (Object Layer)
53, 54 = Slime, Puddle
55, 56, 57 = WallActuators (Sconce), Floor Items, WallDecorations (Object Layer)
58, 59, 60, 61 = spin 90 deg (Object Layer)
62, 63 = spin 180deg E ↔ W, spin 180 deg N ↔ S (Object Layer)

Bottom Row

65, 66 = wood door n/s, e/w (Doors Layer)
67, 68 = wood dr+window n/s, e/w (Doors Layer)
69, 70 = broken door n/s, e/w (Doors Layer)
71, 72 = broken grate n/s, e/w (Doors Layer)
73, 74 = complete grate n/s, e/w (Doors Layer)
75, 76 = red door n/s, e/w (Object Layer)
77, 78 = decorated wood door n/s, e/w (Doors Layer)
79, 80 = decorated metal door n/s, e/w (Doors Layer)
81, 82 = black door+grate n/s, e/w (Doors Layer)
83, 84 = black door n/s, e/w (Doors Layer)
85, 86 = glass door n/s (Doors Layer)
87, 88 = Stone pillar, Brick pillar (Image Layer)



Walls - Tile No 1



This is a Tile Layer, so will produce a grid containing the index of the solid wall tile in the .tmx file.

Place anywhere a solid wall is present that does not have an Altar or Alcove in it

FloorDecorations - Tiles: 53, 54



This is the only other Tile Layer. There are 2 tiles that can be placed here representing Slime or Puddle.

Ceilings - Tile:19



Place this wherever there is a ceiling pit above.
As it is an Object Layer it needs to have a name

eg CP01.

It needs 2 custom properties:
The level above it.

The coordinates of the level above where it
appears as a floor pit.

Use -1 if there is no level above

Properties

Property	Value
Object	
ID	78
Template	
Name	CP01
Class	
Visible	<input checked="" type="checkbox"/>
X	32.00
Y	64.00
Width	16.00
Height	16.00
Rotation	0.00
Flipping	
Horizontal	<input type="checkbox"/> False
Vertical	<input type="checkbox"/> False
Custom Properties	
FromCoordinate	-1,-1
FromLevel	-1

A map view showing a grid of tiles. A ceiling pit tile (CP01) is placed on a dark gray tile. The tile is labeled 'CP01' and has a small 'C' icon. The map view shows a grid of tiles with different shades of gray and black.

Doors – Tiles: 65 – 86



Also an Object Layer. Choose the door that matches the required type and direction of movement (NS or EW)

Give it a name eg Door01

Custom properties are also needed:

Control (string)

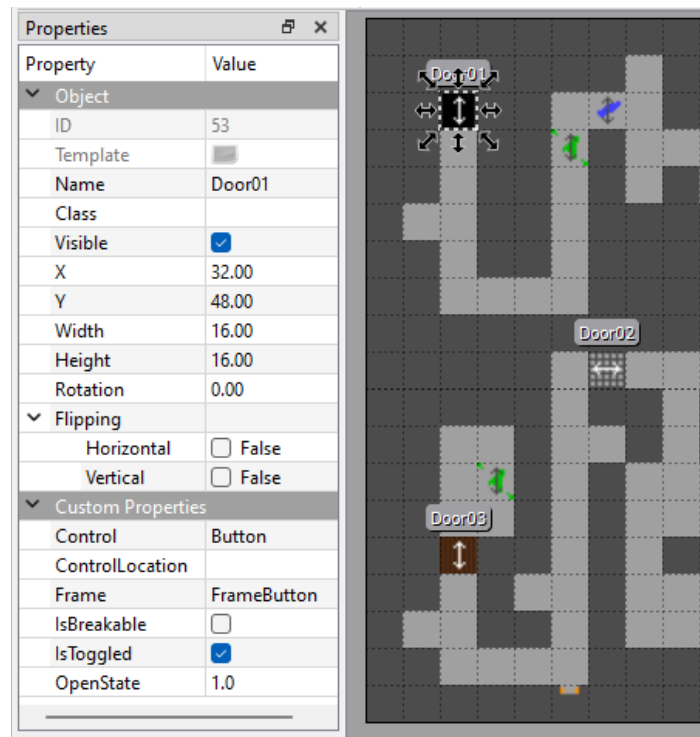
ControlLocation (string)

Frame (string)

IsBreakable (bool)

IsToggled (bool)

OpenState (float)



Property Name	dataType	Acceptable values
Control	string	Button, Remote, Key
ControlLocation	string	Leave Blank, -1,-1 for specific code, x,y coordinates of the map
Frame	string	Frame, FrameButton, FrameKey
IsBreakable	bool	Checked, Un-Checked: True = door can be broken
IsToggled	bool	Checked, Un-Checked :True = door can be opened and closed
OpenState	float	0 = open, 1 = closed, 0.5 = half open eg 1.0

WallDecorations – Tiles: 57



Name: WD02

Custom Properties:

E
N
S
W

All are string values

Use the object name attached to the relevant wall.

Eg E(ast) Manacles

Properties

Property	Value
Object	
ID	51
Template	
Name	WD02
Class	Decorations
Visible	<input checked="" type="checkbox"/>
X	64.00
Y	160.00
Width	16.00
Height	16.00
Rotation	0.00
Flipping	
Horizontal	<input type="checkbox"/> False
Vertical	<input type="checkbox"/> False
Custom Properties	
E	Manacles
N	
S	
W	

WallActuators – Tiles: 49, 50, 51, 52, 55



Name: Sconce01, Fireball1

Custom Properties:

E
N
S
W

Eg for Sconce
Name Sconce01

S SconceT:7 (T = Torch present, brightness 7/7) 0-7
Or
S SconceE (Empty Sconce)

Name: Fireball01
Properties
W Fireball









Champions – Tiles: 41, 42, 43, 44



Name: C00 Important MUST be named C00, C01 C24

No class or Properties needed

Stairs – Tiles: 20 – 27

Symbol	Direction	Symbol	Direction
	Stairs facing North, Going UP		Stairs facing North, Going DOWN
	Stairs facing East, Going UP		Stairs facing East, Going DOWN
	Stairs facing South, Going UP		Stairs facing South, Going DOWN
	Stairs facing West, Going UP		Stairs facing West, Going DOWN

Name: Stairs01

Custom Properties (string)

TargetCell 4,15 (Coordinates of exit on other level)
TargetLevel 01 (Destination level)

WallWriting – Tiles: 45, 46, 47, 48




Name: WallWriting01


See Property N for example

Max 4 lines of text

Separate with ;

Properties

Property	Value
▼ Object	
ID	81
Template	
Name	WritingTest
Class	
Visible	<input checked="" type="checkbox"/>
X	32.00
Y	144.00
Width	16.00
Height	16.00
Rotation	0.00
▼ Flipping	
Horizontal	<input type="checkbox"/> False
Vertical	<input type="checkbox"/> False
▼ Custom Properties	
E	
N	This is line 1;This is line 2;This is line 3;this is line 4
S	
W	



AlcoveItems – Tiles: 2-16, (Alcove) 33-36 (Altar)



Name: AL01 eg

Custom Properties

E eg Scroll:Text=New lives/for old bones (/ is line separator)

N

S eg Apple;Dagger

W

Use ; to searate items

FloorActuators – Tiles: 32



Name: MoveParty (example)

Custom Properties

Property Name	dataType	Valid Values
Action	string	Teleport, Activate
IsToggled	bool	Checked / Un-checked (once only, or continuous use)
IsVisible	bool	Checked / Un-checked (visibility)
ObjectRequired	string	(Blank) , Firestaff (Item required for action)
PartySize	int	1, 2, 3, 4 (Min size for action)
Target	string	10,1 (Coordinates of action item eg door) 9,1;11,1 (2 targets)

FloorItems – Tiles: 56

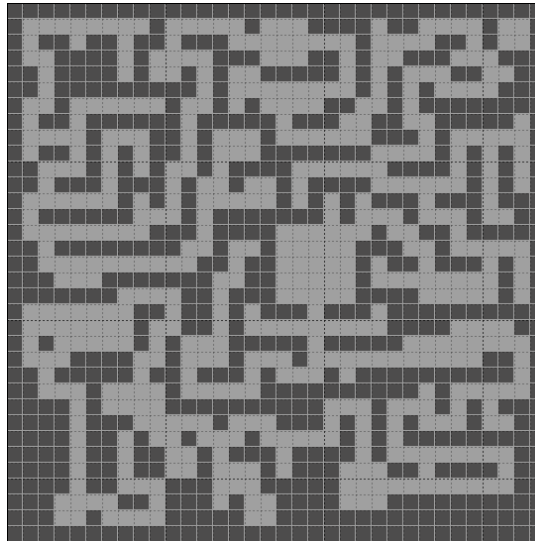


Custom Properties

Property Name	DataType	Valid Values
NE	string	
NW	string	
SE	string	Scroll:Text=Invoke FUL for a magic torch
SW	string	Waterskin:3;Apple

Separate item names with ;

The map with all plain walls:



Next place any alcoves, altars or fountains on the Walls layer.

Adding Items to Object Layers

As an example, add a door to the map:

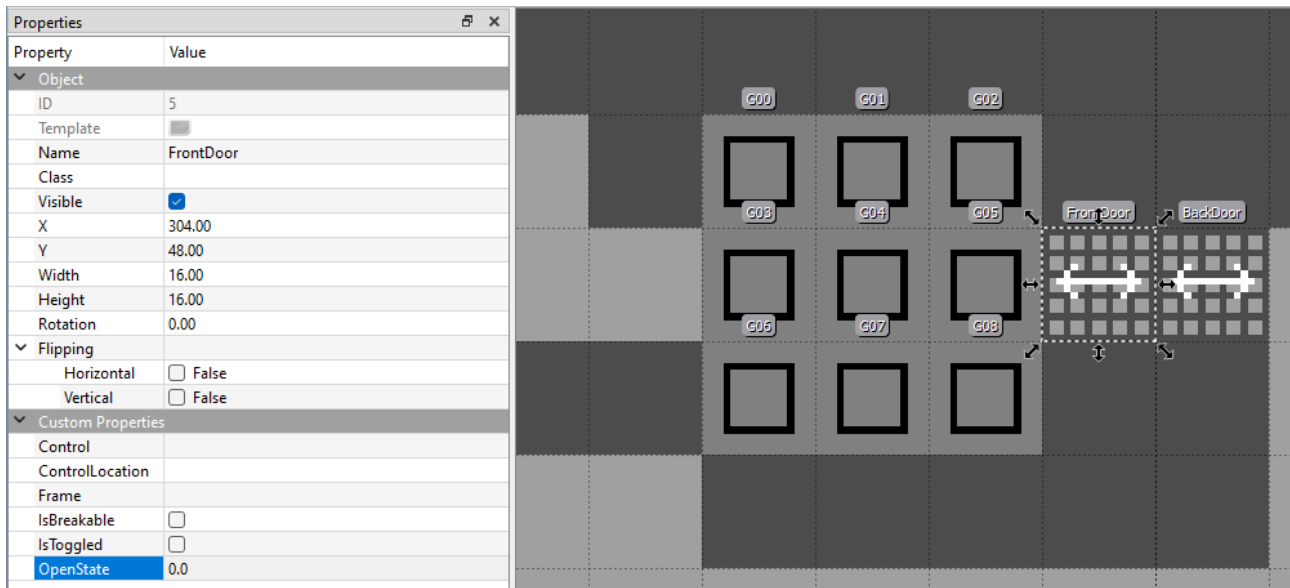
1. Select the Doors Layer.
2. Open the “eye” to make the layer visible.
3. Select a door from the tileset. The first one should do it.
4. Click on the square 0,4 (coords same as screen, starting top left).
5. Change Name and complete the Custom Properties:

Property	Value
▼ Object	
ID	2
Template	
Name	D01
Class	
Visible	<input checked="" type="checkbox"/>
X	32.30
Y	80.15
Width	16.00
Height	16.00
Rotation	0.00
▼ Flipping	
Horizontal	<input type="checkbox"/> False
Vertical	<input type="checkbox"/> False
▼ Custom Properties	
Control	Key
ControlLocation	1,4
Frame	FrameKey
IsBreakable	<input type="checkbox"/>
IsToggled	<input checked="" type="checkbox"/>
OpenState	1.0

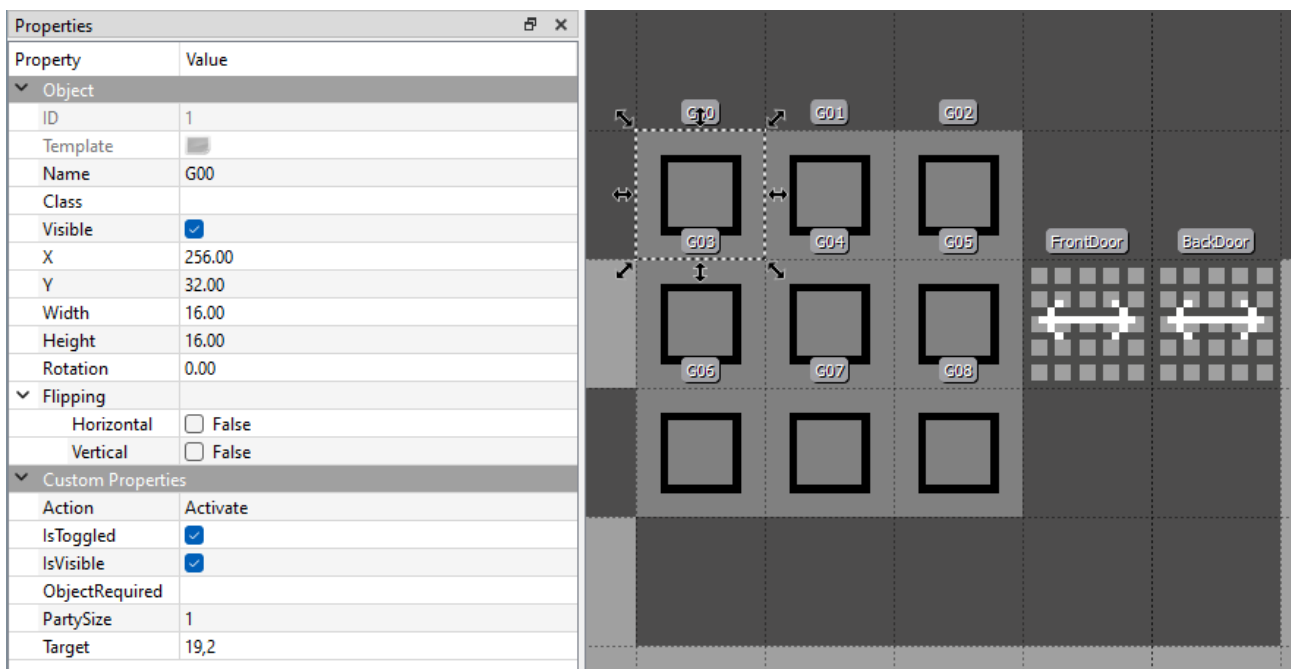
Use the pages above to add different items to the map.

Example of complex FloorPlate / Double Door puzzle

1. Click on the Doors Layer
2. Select a Grate type door
3. Place on 19,3 and 20,3 Names: FrontDoor and BackDoor
4. Complete the properties on FrontDoor only as shown below
5. Select The FloorActuators layer
6. Place FloorPlates at 16,1 17,1 and 18,1 G00, G01, G02 (Only set Name at this stage)
7. Place FloorPlates at 16,2 17,2 and 18,2 G03, G04, G05 (Only set Name at this stage)
8. Place FloorPlates at 16,3 17,3 and 18,3 G06, G07, G08 (Only set Name at this stage)



Select G00 and complete the properties:



Save the file and Close it. You are going to edit the .tmx directly and reload it!

Open Level01.tmx in a text editor.
Check the code below:

The properties of FrontDoor highlighted below in blue can be copied and pasted under the single line making up "BackDoor"

```
<object id="5" name="FrontDoor" gid="74" x="304" y="48" width="16" height="16">
  <properties>
    <property name="Control" value=""/>
    <property name="ControlLocation" value=""/>
    <property name="Frame" value="Frame"/>
    <property name="IsBreakable" type="bool" value="false"/>
    <property name="IsToggled" type="bool" value="false"/>
    <property name="OpenState" type="float" value="0"/>
  </properties>
</object>
<object id="6" name="BackDoor" gid="74" x="320" y="48" width="16" height="16"/>
<object id="7" name="D03" gid="66" x="192.194" y="32.0492" width="16" height="16">
```

Change the values of the properties so it now looks like this. Don't forget to delete the / highlighted above:

```
<object id="5" name="FrontDoor" gid="74" x="304" y="48" width="16" height="16">
  <properties>
    <property name="Control" value="Remote"/>
    <property name="ControlLocation" value=""/>
    <property name="Frame" value="Frame"/>
    <property name="IsBreakable" type="bool" value="false"/>
    <property name="IsToggled" type="bool" value="true"/>
    <property name="OpenState" type="float" value="1"/>
  </properties>
</object>
<object id="6" name="BackDoor" gid="74" x="320" y="48" width="16" height="16">
  <properties>
    <property name="Control" value="Remote"/>
    <property name="ControlLocation" value=""/>
    <property name="Frame" value="Frame"/>
    <property name="IsBreakable" type="bool" value="false"/>
    <property name="IsToggled" type="bool" value="true"/>
    <property name="OpenState" type="float" value="1"/>
  </properties>
</object>
<object id="7" name="D03" gid="66" x="192.194" y="32.0492" width="16" height="16">
```

Save and restart Tiled.

The doors FrontDoor and BackDoor now have identical properties.

The same thing can be done with the FloorPlates:

```
<objectgroup id="4" name="FloorActuators">
  <object id="1" name="G00" gid="32" x="256" y="32" width="16" height="16">
    <properties>
      <property name="Action" value="Activate"/>
      <property name="IsToggled" type="bool" value="true"/>
      <property name="IsVisible" type="bool" value="true"/>
      <property name="ObjectRequired" value=""/>
      <property name="PartySize" type="int" value="1"/>
      <property name="Target" value="19,2"/>
    </properties>
  </object>
  <object id="10" name="G01" gid="32" x="272" y="32" width="16" height="16"/>
  <object id="11" name="G02" gid="32" x="288" y="32" width="16" height="16"/>
  <object id="12" name="G03" gid="32" x="256" y="48" width="16" height="16"/>
  <object id="13" name="G04" gid="32" x="272" y="48" width="16" height="16"/>
  <object id="14" name="G05" gid="32" x="288" y="48" width="16" height="16"/>
  <object id="15" name="G06" gid="32" x="256" y="64" width="16" height="16"/>
  <object id="16" name="G07" gid="32" x="272" y="64" width="16" height="16"/>
  <object id="17" name="G08" gid="32" x="288" y="64" width="16" height="16"/>
</objectgroup>
```

Delete the / highlighted in RED

Copy the yellow highlighted code

Paste the code in between the lines containing “G01” to “G08” repeatedly:

```
<object id="10" name="G01" gid="32" x="272" y="32" width="16" height="16">
  <properties>
    <property name="Action" value="Activate"/>
    <property name="IsToggled" type="bool" value="true"/>
    <property name="IsVisible" type="bool" value="true"/>
    <property name="ObjectRequired" value=""/>
    <property name="PartySize" type="int" value="1"/>
    <property name="Target" value="19,2"/>
  </properties>
</object>
<object id="11" name="G02" gid="32" x="288" y="32" width="16" height="16">
  <properties>
    <property name="Action" value="Activate"/>
    <property name="IsToggled" type="bool" value="true"/>
    <property name="IsVisible" type="bool" value="true"/>
    <property name="ObjectRequired" value=""/>
    <property name="PartySize" type="int" value="1"/>
    <property name="Target" value="19,2"/>
  </properties>
</object>
<object id="12" name="G03" gid="32" x="256" y="48" width="16" height="16">
  <properties>
    <property name="Action" value="Activate"/>
    <property name="IsToggled" type="bool" value="true"/>
    <property name="IsVisible" type="bool" value="true"/>
    <property name="ObjectRequired" value=""/>
    <property name="PartySize" type="int" value="1"/>
    <property name="Target" value="19,2"/>
  </properties>
</object>
```

You get the idea. No need to show all of them.

Finally change the value of the “Target” highlighted in green to the following values:

FloorPlate Name	Target Value (Insert these coordinates)	Target Door
G00	19,2	FrontDoor
G01	20,2	BackDoor
G02	19,2	FrontDoor
G03	19,2	FrontDoor
G04	20,2	BackDoor
G05	19,2	FrontDoor
G06	19,2	FrontDoor
G07	20,2	BackDoor
G08	19,2	FrontDoor

This demonstrates a rapid method for filling in properties on multiple objects.

The logic behind this puzzle:

Stepping on G03 as you enter the puzzle toggles FrontDoor closed → open

Stepping on G04 toggles BackDoor closed → open

Stepping on G05 toggles FrontDoor open → closed

The floorplates in the rows above and below operate identically.

To solve the puzzle:

1. Step onto G03 → FrontDoor opens
2. Step BACK then step onto G03 again → FrontDoor closes
3. Step onto G04 → BackDoor opens
4. Step onto G05 → FrontDoor opens
5. Walk through both open doors!
6. The upper and lower rows are put there to confuse the player