Tiled Map Editor – User Manual

URL: <https://doc.mapeditor.org/en/stable/manual/introduction/#>

This document is a Word and pdf version of the Tiled Map Editor (version 1.11.2). I decided to create a copy of the online document in order to a) read it b) follow it, with the purpose of learning the terminology surrounding the use of Tiled (e.g. Layer). - nyguerrillagirl

# Introduction

## About Tiled

**Tiled is a 2D level editor that helps you develop the content of your game. Its primary feature is to edit tile maps of various forms, but it also supports free image placement as well as powerful ways to annotate your level with extra information used by the game. Tiled focuses on general flexibility while trying to stay intuitive.**

In terms of tile maps, it supports straight rectangular tile layers, but also projected isometric, staggered isometric and staggered hexagonal layers. A tileset can be either a single image containing many tiles, or it can be a collection of individual images. In order to support certain depth faking techniques, tiles and layers can be offset by a custom distance and their rendering order can be configured.

The primary tool for editing [tile layers](https://doc.mapeditor.org/en/stable/manual/layers/#tile-layer-introduction) is a stamp brush that allows efficient painting and copying of tile areas. It also supports drawing lines and circles. In addition, there are several selection tools and a tool that does [automatic terrain transitions](https://doc.mapeditor.org/en/stable/manual/terrain/). Finally, it can apply changes based on [pattern-matching](https://doc.mapeditor.org/en/stable/manual/automapping/) to automate parts of your work.

Tiled also supports [object layers](https://doc.mapeditor.org/en/stable/manual/layers/#object-layer-introduction), which traditionally were only for annotating your map with information but more recently they can also be used to place images. You can add rectangle, point, ellipse, polygon, polyline and tile objects. Object placement is not limited to the tile grid and objects can also be scaled or rotated. Object layers offer a lot of flexibility to add almost any information to your level that your game needs.

Other things worth mentioning are the support for adding custom map or tileset formats through plugins, [extending Tiled](https://doc.mapeditor.org/en/stable/manual/scripting/) with JavaScript, the tile stamp memory, [tile animation support](https://doc.mapeditor.org/en/stable/manual/editing-tilesets/#tile-animation-editor) and the [tile collision editor](https://doc.mapeditor.org/en/stable/manual/editing-tilesets/#tile-collision-editor).

## Getting Started

### Setting up a New Project

When launching Tiled for the first time, we are greeted with the following window:

A screenshot of a computer

AI-generated content may be incorrect.

To make all our assets readily accessible from the **Project** view, as well as to be able to quickly switch between multiple projects, it is recommended to first set up a [Tiled project](https://doc.mapeditor.org/en/stable/manual/projects/). This is however an entirely optional step that can be skipped when desired.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1 - Creating a new Project

Choose *File -> New -> New Project…* to create a new project file. It is recommended to save this file in the root of your project. The directory in which you store the project will be automatically added, so that its files are visible in the Project view.

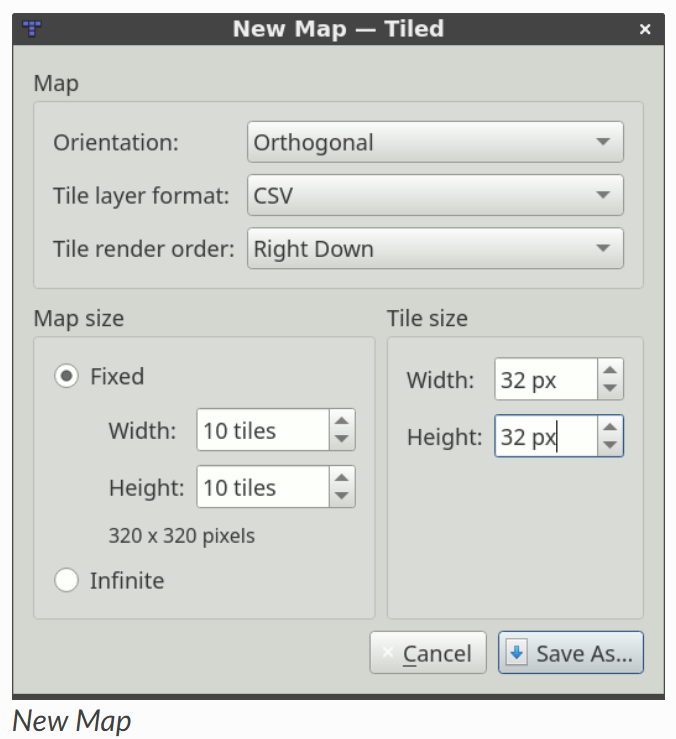
When necessary, you can add additional folders to the project or replace the one added by default. For example, you could choose to add several top-level folders like “tilesets”, “maps”, “templates”, etc. Right-click in the Project view and choose *Add Folder to Project…* to add the relevant folders.

A green and white rectangle with black text

AI-generated content may be incorrect.

### Creating a New Map

To create a new map, choose *File -> New -> New Map…* (Ctrl+N). The following dialog will pop up:



Here, we choose the initial map size, tile size, orientation, tile layer format, tile render order (only supported for *Orthogonal* maps) and whether the map is [infinite](https://doc.mapeditor.org/en/stable/manual/using-infinite-maps/) or not. All of these things can be changed later as needed, so it’s not important to get it all right the first time.

A blue and white text

AI-generated content may be incorrect.

After saving our map, we’ll see the tile grid and an initial tile layer will be added to the map. However, before we can start using any tiles we need to add a tileset. Choose *File -> New -> New Tileset…* to open the New Tileset dialog:

A screenshot of a computer

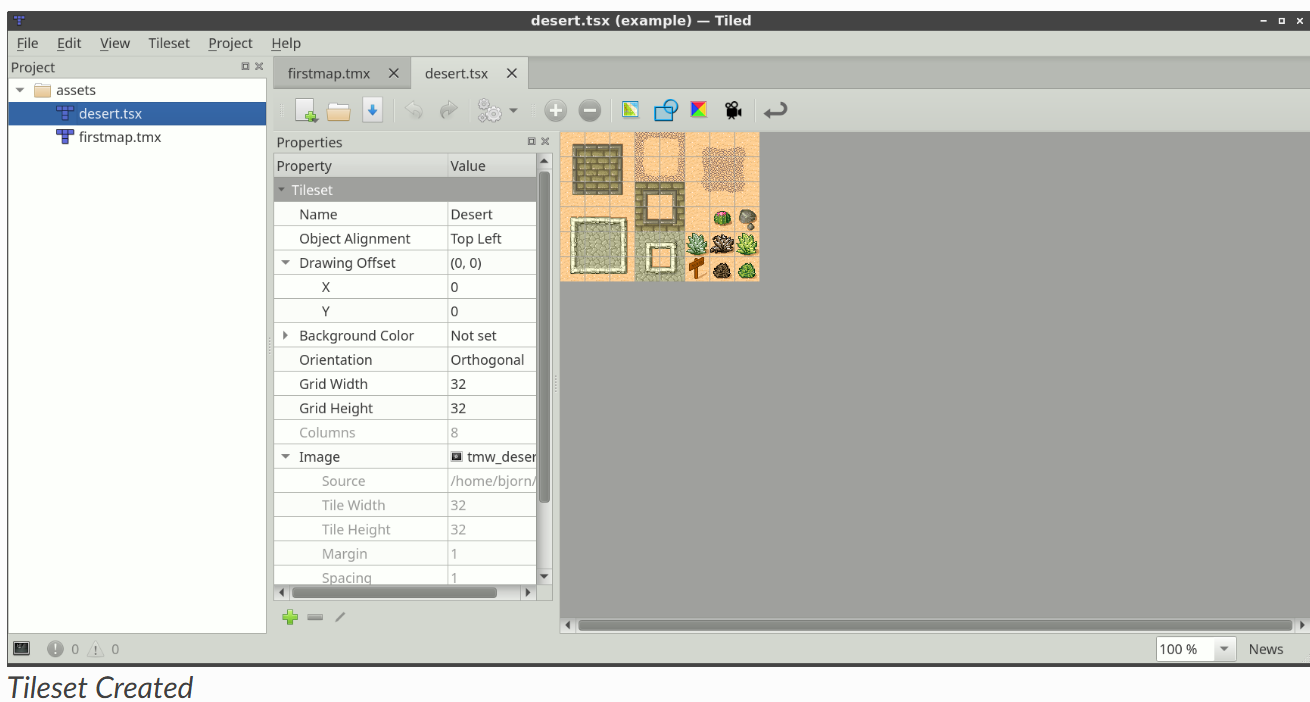
AI-generated content may be incorrect.

Click the **Browse…** button and select the tmw\_desert\_spacing.png tileset from the examples shipping with Tiled (or use one of your own if you wish). This example tileset uses a tile size of 32x32. It also has a one pixel *margin* around the tiles and a one pixel *spacing* in between the tiles (this is pretty rare actually, usually you should leave these values on 0).

A screenshot of a computer

AI-generated content may be incorrect.

After saving the tileset, Tiled should look as follows:



Since we don’t want to do anything else with the tileset for now, just switch back to the map file:

A screenshot of a computer

AI-generated content may be incorrect.

We’re ready to select some tiles and start painting! But first, let’s have a quick look at the [various layer types](https://doc.mapeditor.org/en/stable/manual/layers/) supported by Tiled.

A screenshot of a computer

AI-generated content may be incorrect.

Tiled Map Editor Tutorial Series: https://gamefromscratch.com/tiled-map-editor-tutorial-series/

# Projects

## What’s in a Project

A Tiled project file primarily defines the list of folders containing the assets belonging to that project. In addition, it provides an anchor for the [session file](https://doc.mapeditor.org/en/stable/manual/projects/#sessions).

Apart from the list of folders, a project currently has the following properties, which can be changed through the *Project -> Project Properties…* dialog.

**Compatibility Version**

The Tiled version to target when saving or exporting files. Can be used to maintain compatibility with earlier versions of Tiled or with [Libraries and Frameworks](https://doc.mapeditor.org/en/stable/reference/support-for-tmx-maps/) that do not yet support certain backwards-incompatible changes.

**Extensions Directory**

A project-specific directory where you can put [Tiled extensions](https://doc.mapeditor.org/en/stable/manual/scripting/). It defaults to simply extensions, so when you have a directory called “extensions” alongside your project file it will be picked up automatically.

The directory is loaded in addition to the global extensions.

**Automapping Rules File**

Refers to an [Automapping](https://doc.mapeditor.org/en/stable/manual/automapping/) rules file, or a single rule map, that should be used for all maps while this project is loaded. It is ignored for maps that have a rules.txt file saved alongside them.

Any types defined in the [Custom Types Editor](https://doc.mapeditor.org/en/stable/manual/custom-properties/#custom-property-types) are also saved in the project.

## Sessions

Each project file gets an associated ***.tiled-session*** file, stored alongside it. The session file should generally not be shared with others and stores your last opened files, part of their last editor state, last used parameters in dialogs, etc.

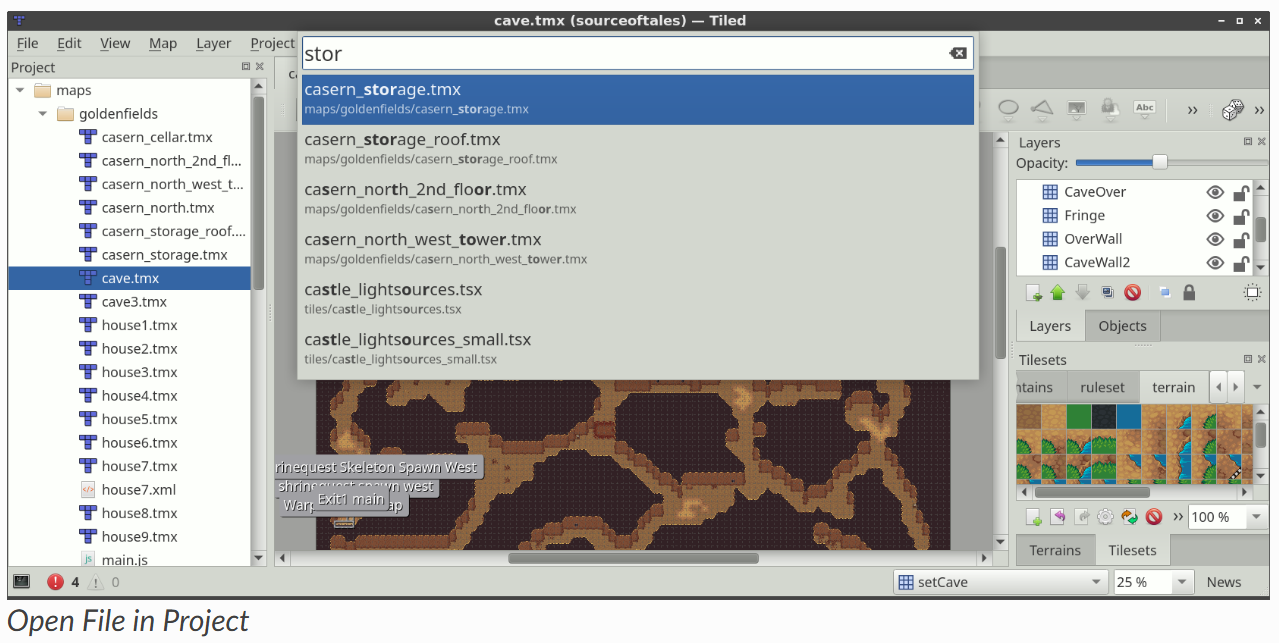
A screenshot of a computer

AI-generated content may be incorrect.

When switching projects Tiled automatically switches to the associated session, so you can easily resume where you left off. When no project is loaded a global session file is used.

## Opening a File in the Project

Another advantage of setting up a project is that you can quickly open any file with a recognized extension located in one of the folders of the project. Use *File -> Open File in Project* (Ctrl+P) to open the file filter and just type the name of the file you’d like to open.



**Future Extensions**

There are many ways in which the projects could be made more powerful:

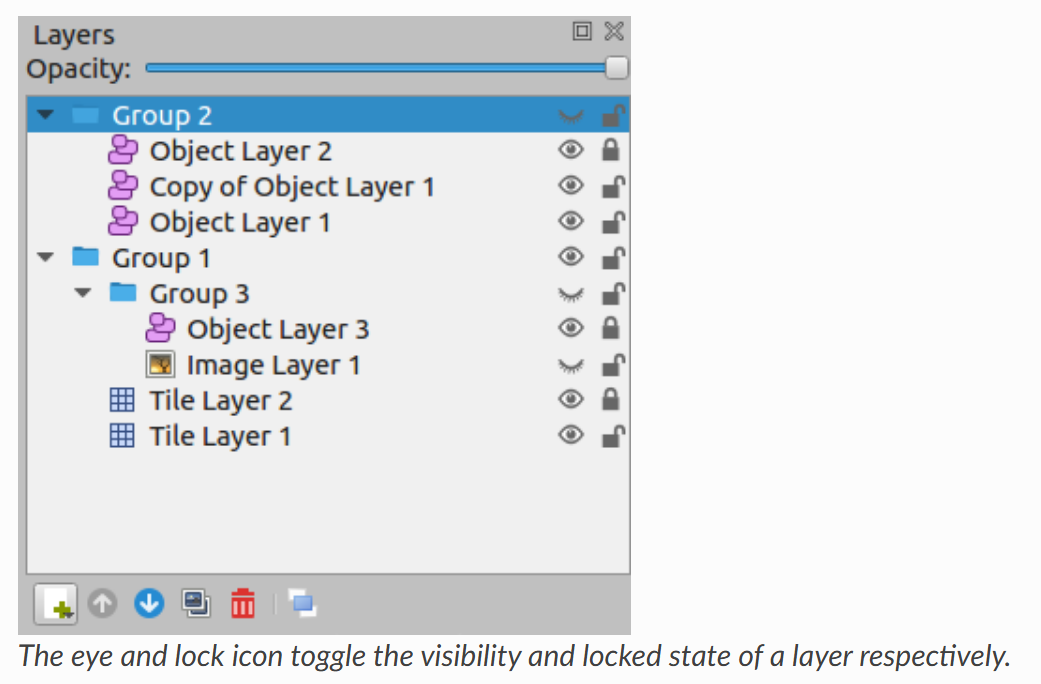
* Make the custom types defined in the project accessible through the [scripting API](https://doc.mapeditor.org/en/stable/manual/scripting/) ([#3419](https://github.com/mapeditor/tiled/issues/3419)).
* Allow turning off features on a per-project basis, to simplify the UI and reduce the chance of accidentally doing something your project doesn’t support.
* Recognizing the various assets in your project, so that selection of images, tilesets and templates can be made more efficient (potentially replacing the system file dialog).

If you like any of these plans, please help me getting around to it faster by [sponsoring Tiled development](https://www.mapeditor.org/donate). The more support I receive the more time I can afford to spend improving Tiled!

# Working with Layers

A Tiled map supports various sorts of content, and this content is organized into various different layers. The most common layers are the [Tile Layer](https://doc.mapeditor.org/en/stable/manual/layers/#tile-layers) and the [Object Layer](https://doc.mapeditor.org/en/stable/manual/layers/#object-layers). There is also an [Image Layer](https://doc.mapeditor.org/en/stable/manual/layers/#image-layers) for including simple foreground or background graphics. The order of the layers determines the rendering order of your content.

Layers can be hidden, made only partially visible and can be locked. Layers also have an offset and a [parallax scrolling factor](https://doc.mapeditor.org/en/stable/manual/layers/#parallax-factor), which can be used to position them independently of each other, for example to fake depth. Finally their contents can be tinted by multiplying with a custom [tint color](https://doc.mapeditor.org/en/stable/manual/layers/#tint-color).



|  |
| --- |
| In **Tiled Map Editor**, layers are used to organize different elements of a map. They allow developers to separate tiles, objects, and images for better control over rendering and gameplay mechanics.  Types of Layers in Tiled:   1. **Tile Layers** – Store tile-based data efficiently, defining the visual layout of the map. 2. **Object Layers** – Contain freely positioned objects like rectangles, polygons, and text, useful for defining collision areas or interactive elements. 3. **Image Layers** – Used for background or foreground images that don’t need to be tiled. 4. **Group Layers** – Help organize multiple layers into a hierarchy, making large maps easier to manage.   Layers can be **hidden, locked, offset**, and even **tinted** to create depth effects. They also support **parallax scrolling**, which allows different layers to move at different speeds for a more dynamic visual experience. |

You use [Group Layers](https://doc.mapeditor.org/en/stable/manual/layers/#group-layers) to organize the layers into a hierarchy. This makes it more comfortable to work with a large amount of layers.

## Layer Types

### Tile Layers

Tile layers provide an efficient way of storing a large area filled with tile data. The data is a simple array of tile references and as such no additional information can be stored for each location. The only extra information stored are a few flags, that allow tile graphics to be flipped vertically, horizontally or anti-diagonally (to support rotation in 90-degree increments).

The information needed to render each tile layer is stored with the map, which specifies the position and rendering order of the tiles based on the orientation and various other properties.

Despite only being able to refer to tiles, tile layers can also be useful for defining various bits of non-graphical information in your level. Collision information can often be conveyed using a special tileset, and any kind of object that does not need custom properties and is always aligned to the grid can also be placed on a tile layer.

### Object Layers

Object layers are useful because they can store many kinds of information that would not fit in a tile layer. Objects can be freely positioned, resized and rotated. They can also have individual custom properties. There are many kinds of objects:

* **Rectangle** - for marking custom rectangular areas
* **Ellipse** - for marking custom ellipse or circular areas
* **Point** - for marking exact locations (since Tiled 1.1)
* **Polygon** - for when a rectangle or ellipse doesn’t cut it (often a collision area)
* **Polyline** - can be a path to follow or a wall to collide with
* **Tile** - for freely placing, scaling and rotating your tile graphics
* **Text** - for custom text or notes (since Tiled 1.0)

All objects can be named, in which case their name will show up in a label above them (by default only for selected objects). Objects can also be given a *class*, which is useful since it can be used to customize the color of their label and the available [custom properties](https://doc.mapeditor.org/en/stable/manual/custom-properties/#custom-classes) for this object. For tile objects, the class can be [inherited from their tile](https://doc.mapeditor.org/en/stable/manual/custom-properties/#tile-property-inheritance).

For most map types, objects are positioned in plain pixels. The only exception to this are isometric maps (not isometric staggered). For isometric maps, it was deemed useful to store their positions in a projected coordinate space. For this, the isometric tiles are assumed to represent projected squares with both sides equal to the *tile height*. If you’re using a different coordinate space for objects in your isometric game, you’ll need to convert these coordinates accordingly.

The object width and height is also mostly stored in pixels. For isometric maps, all shape objects (rectangle, point, ellipse, polygon and polyline) are projected into the same coordinate space described above. This is based on the assumption that these objects are generally used to mark areas on the map.

### Image Layers

Image layers provide a way to quickly include a single image as foreground or background of your map. They currently have limited functionality and you may consider adding the image as a Tileset instead and place it as a [Tile Object](https://doc.mapeditor.org/en/stable/manual/objects/#insert-tile-tool). This way, you gain the ability to freely scale and rotate the image.

However, image layers can be repeated along the respective axes through their *Repeat X* and *Repeat Y* properties.

The other advantage of using an image layer is that it avoids selecting / dragging the image while using the Select Objects tool. However, since Tiled 1.1 this can also be achieved by locking the object layer containing the tile object you’d like to avoid interacting with.

### Group Layers

Group layers work like folders and can be used for organizing the layers into a hierarchy. This is mainly useful when your map contains a large amount of layers.

The visibility, opacity, offset, lock and [tint color](https://doc.mapeditor.org/en/stable/manual/layers/#tint-color) of a group layer affects all child layers.

Layers can be easily dragged in and out of groups with the mouse. The Raise Layer / Lower Layer actions also allow moving layers in and out of groups.

## Parallax Scrolling Factor

The parallax scrolling factor determines the amount by which the layer moves in relation to the camera.

By default its value is 1, which means its position on the screen changes at the same rate as the position of the camera (in opposite direction). A lower value makes it move slower, simulating a layer that is further away, whereas a higher value makes it move faster, simulating a layer positioned in between the screen and the camera.

A value of 0 makes the layer not move at all, which can be useful to include some pieces of your ingame UI or to mark its general viewport boundaries.

Negative values make the layer move in opposite direction, though this is rarely useful.

When the parallax scrolling factor is set on a group layer, it applies to all its child layers. The effective parallax scrolling factor of a layer is determined by multiplying the parallax scrolling factor by the scrolling factors of all parent layers.

### Parallax Reference Point

To match not only the scrolling speed but also the positioning of layers, we need to use the same points of reference. In Tiled these are the parallax origin and the center of the view. The parallax origin is stored per map and defaults to (0,0), which is the top-left of the maps bounding box. The distance between these two points is multiplied by the parallax factor to determine the final position on the screen for each layer. For example:

* If the parallax origin is in the center of the view, the distance is (0,0) and none of the parallax factors have any effect. The layers are rendered where they would have been, if parallax was disabled.
* Now, when the map is scrolled right by 10 pixels, the distance between the parallax origin and the center of the view is 10. So a layer with a parallax factor of 0.7 will have moved just 0.7 \* 10 = 7 pixels.

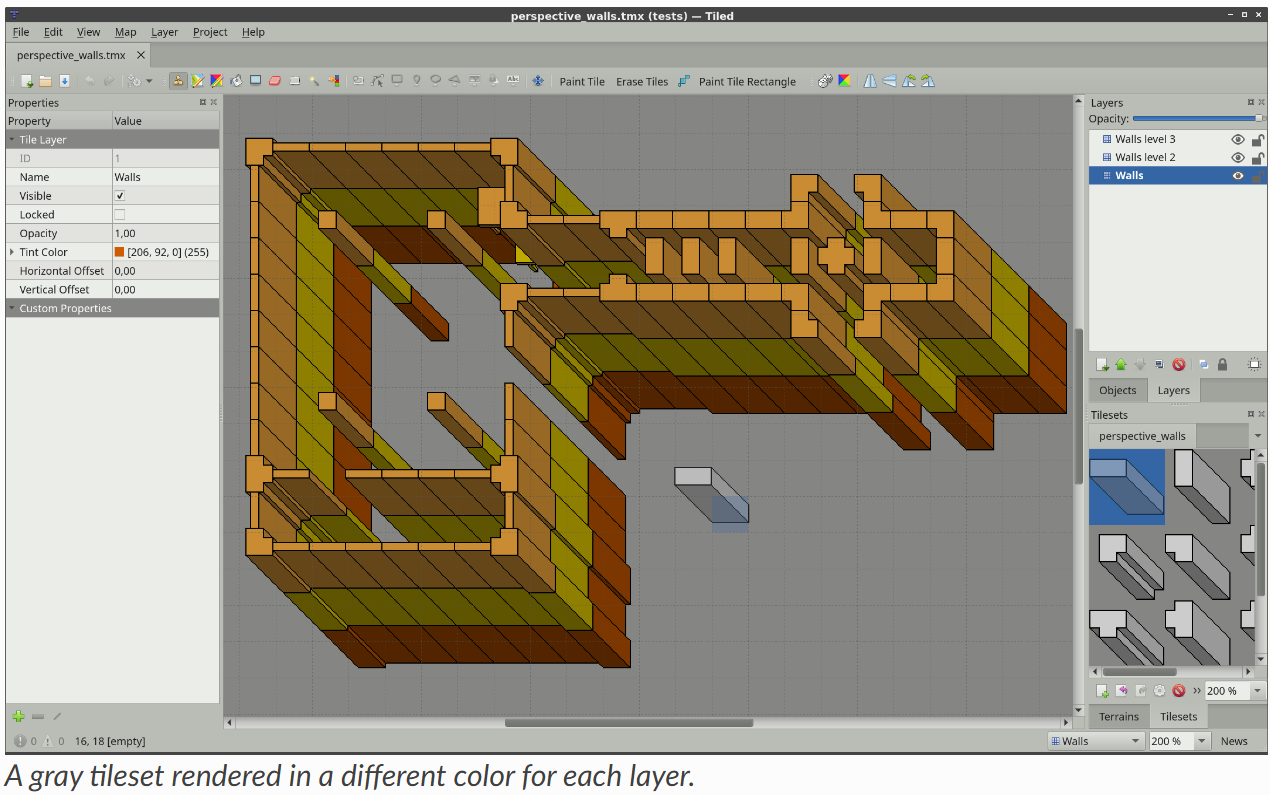
Quite often, a viewport transform is used to scroll the entire map. In this case, one may need to adjust the position of each layer to take its parallax factor into account. Instead of multiplying the distance with the parallax factor directly, we now multiply by 1 - parallaxFactor to get the layer position. For example:

* When the camera moves right by 10 pixels, the layer will have moved 10 pixels to the left (-10), so by positioning the layer at 10 \* (1 - 0.7) = 3, we’re making sure that it only moves 7 pixels to the left.

## Tinting Layers

When you set the *Tint Color* property of a layer, this affects the way images are rendered. This includes tiles, tile objects and the image of an [Image Layer](https://doc.mapeditor.org/en/stable/manual/layers/#image-layers).

Each pixel color value is multiplied by the tint color. This way you can darken or colorize your graphics in various ways without needing to set up separate images for it.



The tint color can also be set on a [Group Layer](https://doc.mapeditor.org/en/stable/manual/layers/#group-layers), in which case it is inherited by all layers in the group.

**Future Extensions**

There are many ways in which the layers can be made more powerful:

* Ability to lock individual objects ([#828](https://github.com/mapeditor/tiled/issues/828)).
* Moving certain map-global properties to the Tile Layer ([#149](https://github.com/mapeditor/tiled/issues/149)). It would be useful if one map could accommodate layers of different tile sizes and maybe even of different orientation.

# Editing Tile Layers

[Tile Layers](https://doc.mapeditor.org/en/stable/manual/layers/#tile-layer-introduction) are what makes Tiled a *tile map editor*. Although not as flexible as [Object Layers](https://doc.mapeditor.org/en/stable/manual/layers/#object-layer-introduction), they provide efficient data storage and good rendering performance as well as efficient content creation. Every new map gets one by default, though feel free to delete it when you’re not going to use it.

## Stamp Brush

A screenshot of a computer

AI-generated content may be incorrect.

Shortcut: B stock-tool-clone

The primary tool for editing tile layers is the Stamp Brush. It can be used to paint single tiles as well as larger “stamps”, which is where it gets its name from. Using the right mouse button, it can also quickly capture tile stamps from the currently active layer. A tile stamp is commonly created by selecting one or more tiles in the Tilesets view.

The Stamp Brush has some extra features:

* While holding Shift, click any two points to draw a line between them.
* While holding Ctrl+Shift, click any two points two draw a circle or ellipse centered on the first point.
* Activate the *Random Mode* using the dice button on the Tool Options toolbar to have the Stamp Brush paint with random tiles from the tile stamp. The probability of each tile depends on how often it occurred on the tile stamp, as well as the probability set on each tile in the *Tileset Editor*.
* Activate the *Terrain Fill Mode* using the Terrain tile terrain button on the tool bar to have the Stamp Brush paint using random terrain tiles. This makes adjacent tiles match edge and corner terrains to be placed. Terrain tiles are described in detail in [Using Terrains](https://doc.mapeditor.org/en/stable/manual/terrain/).
* In combination with the *Tile Stamps* view, it can also place randomly from a set of predefined tile stamps. This can be more useful than the *Random Mode*, which randomly places individual tiles.
* You can flip the current tile stamp horizontally/vertically by using X and Y respectively. You can also rotate left/right by using Z and Shift+Z respectively. These actions can also be triggered from the Tool Options tool bar.

## Terrain Brush

A close up of a button

AI-generated content may be incorrect.

Shortcut: T terrain-edit

The Terrain Brush allows for efficient editing with certain types of terrain transitions (corner-based, edge-based or a combination). Setting it up requires associating terrain information with your tiles, which is described in detail in [Using Terrains](https://doc.mapeditor.org/en/stable/manual/terrain/).

Similarly to the [Stamp Brush](https://doc.mapeditor.org/en/stable/manual/editing-tile-layers/#stamp-tool), you can draw lines by holding Shift. When holding Ctrl, the size of the edited area is increased to cover an entire tile rather than just one corner or edge.

**Since Tiled 1.0**

When holding Alt, the editing operations are also applied at a 180 degree rotation. This is especially useful when editing strategic maps where two sides need to have equal opportunities. The modifier works well in combination with either Shift for drawing lines or Ctrl for increasing the edited area.

## Bucket Fill Tool

A screenshot of a computer

AI-generated content may be incorrect.

Shortcut: F stock-tool-bucket-fill

The Bucket Fill Tool provides a quick way of filling empty areas or areas covered with the same tiles. The currently active tile stamp will be repeated in the filled area. It can also be used in combination with the *Random Mode*, or *Terrain Fill Mode*.

When holding Shift, the tool fills the currently selected area regardless of its contents. This is useful for filling custom areas that have been selected previously using one or more [Selection Tools](https://doc.mapeditor.org/en/stable/manual/editing-tile-layers/#tile-selection-tools).

You can also flip and rotate the current stamp as described for the [Stamp Brush](https://doc.mapeditor.org/en/stable/manual/editing-tile-layers/#stamp-tool).

## Shape Fill Tool

A screenshot of a computer

AI-generated content may be incorrect.

Shortcut: P rectangle-fill

This tool provides a quick way to fill rectangles or ellipses with a certain tile or pattern.

* Holding Shift fills an exact square or circle.

**Since Tiled 1.10.2**

* Holding Alt draws the rectangle or ellipse centered around the starting location.

You can also flip and rotate the current stamp as described for the [Stamp Brush](https://doc.mapeditor.org/en/stable/manual/editing-tile-layers/#stamp-tool).

## Eraser

A screenshot of a computer

AI-generated content may be incorrect.

**Eraser**[**ℑ**](https://doc.mapeditor.org/en/stable/manual/editing-tile-layers/#eraser)

Shortcut: E stock-tool-eraser

A simple eraser tool. Left click erases single tiles and right click can be used to quickly erase rectangular areas.

**Since Tiled 1.5**

* Holding Shift erases on all layers.

## Selection Tools

A screenshot of a computer game

AI-generated content may be incorrect.

There are various tile selection tools that all work in similar fashion:

* stock-tool-rect-select **Rectangular Select** allows selection of rectangular areas (shortcut: R)
* stock-tool-fuzzy-select-22 **Magic Wand** allows selection of connected areas filled with the same tile (shortcut: W)
* stock-tool-by-color-select **Select Same Tile** allows selection of same-tiles across the entire layer (shortcut: S)

By default, each of these tools replaces the currently selected area. The following modifiers can be used to change this behavior:

* Holding Shift expands the current selection with the new area
* Holding Ctrl subtracts the new area from the current selection
* Holding Ctrl and Shift selects the intersection of the new area with the current selection

You can also lock into one of these modes (Add, Subtract or Intersect) by clicking on one of the tool buttons in the Tool Options toolbar.

## Managing Tile Stamps

It can often be useful to store the current tile stamp somewhere to use it again later. The following shortcuts work for this purpose:

* Ctrl + 1-9 - Store current tile stamp. When no tile drawing tool is selected, tries to capture the current tile selection (similar to Ctrl + C).
* 1-9 - Recall the stamp stored at this location (similar to Ctrl + V)

Tile stamps can also be stored by name and extended with variations using the *Tile Stamps* view.

# Working with Objects

Using objects you can add a great deal of information to your map for use in your game. They can replace tedious alternatives like hardcoding coordinates (like spawn points) in your source code or maintaining additional data files for storing gameplay elements.

By using *tile objects*, objects of various types can be made easy to recognize or they can be used for purely graphical purposes. In some cases they can replace the use of tile layers entirely, as demonstrated by the “Sticker Knight” example shipping with Tiled.

All objects can have [custom properties](https://doc.mapeditor.org/en/stable/manual/custom-properties/), which can also be used to create [connections between objects](https://doc.mapeditor.org/en/stable/manual/objects/#connecting-objects).

To start using objects, add an [Object Layer](https://doc.mapeditor.org/en/stable/manual/layers/#object-layer-introduction) to your map.

## Placement Tools

Each type of object has its own placement tool.

**Since Tiled 1.2**

A preview is shown of the object you’re about to place when you hover over the map. While placing an object, you can press Escape or right-click to cancel placement of the object. Press Escape again to switch to the [Select Objects](https://doc.mapeditor.org/en/stable/manual/objects/#select-objects-tool) tool.

### Insert Rectangle

Shortcut: R

The rectangle was the first type of object supported by Tiled, which is why objects are rectangles by default in the [TMX Map Format](https://doc.mapeditor.org/en/stable/reference/tmx-map-format/). They are useful for marking rectangular areas and assigning custom properties to them. They are also often used for specifying collision boxes.

Place a rectangle by clicking-and-dragging in any direction. Holding Shift makes it square and holding Ctrl snaps its size to the tile size.

Rectangle objects have their origin in the top-left. However, if the rectangle is empty (width and height are both 0), it is rendered as a small square around its position. This is mainly to keep it visible and selectable.

### Insert Point

Shortcut: I

Points are the simplest objects you can place on a map. They only represent a location, and cannot be resized or rotated. Simply click on the map to position a point object.

### Insert Ellipse

Shortcut: C

Ellipses work the same way as [rectangles](https://doc.mapeditor.org/en/stable/manual/objects/#insert-rectangle), except that they are rendered as an ellipse. Useful for when your area or collision shape needs to represent a circle or ellipse.

### Insert Polygon

Shortcut: P

Polygons are the most flexible way of defining the shape of an area. They are most commonly used for defining collision shapes.

When placing a polygon, the first click determines the location of the object as well as the location of the first point of the polygon. Subsequent clicks are used to add additional points to the polygon. Polygons needs to have at least three points. Click the first point again to finish creating the polygon. You can press Escape to cancel the creation of the polygon.

When you want to change a polygon after it has been placed, you need to use the [Edit Polygons](https://doc.mapeditor.org/en/stable/manual/objects/#edit-polygons-tool) tool.

#### Polylines

Polylines are created by not closing a polygon. Right-click or press Enter while creating a polygon to finish it as a polyline.

Polylines are rendered as a line and require only two points. While they can represent collision walls, they are also often used to represent paths to be followed.

**Since Tiled 1.2**

You can extend an existing polyline at either end when it is selected, by clicking on the displayed dots. It is also possible to finish the polyline by connecting it to either end of another existing polyline object. The other polyline object needs to be selected as well, since the interactive dots only show on selected polylines.

The [Edit Polygons](https://doc.mapeditor.org/en/stable/manual/objects/#edit-polygons-tool) tool is used to edit polylines as well.

### Insert Tile

Shortcut: T

Tiles can be inserted as objects to have full flexibility in placing, scaling and rotating the tile image on your map. Like all objects, tile objects can also have custom properties associated with them. This makes them useful for placement of recognizable interactive objects that need special information, like a chest with defined contents or an NPC with defined script.

To place a tile object, first select the tile you want to place in the *Tilesets* view. Then use the Left mouse button on the map to start placing the object, move to position it and release to finish placing the object.

**Since Tiled 1.0**

To change the tile used by existing tile objects, select all the objects you want to change using the [Select Objects](https://doc.mapeditor.org/en/stable/manual/objects/#select-objects-tool) tool and then right-click on a tile in the *Tilesets* view, and choose *Replace Tile of Selected Objects*.

**Since Tiled 1.4**

You can customize the alignment of tile objects using the *Object Alignment* property on the *Tileset*. For compatibility reasons this property is set to *Unspecified* by default, in which case tile objects are bottom-left aligned in all orientations except on *Isometric* maps, where they are bottom-center aligned. Setting this property to *Top Left* makes the alignment of tile objects consistent with that of [rectangle objects](https://doc.mapeditor.org/en/stable/manual/objects/#insert-rectangle-tool).

### Insert Template

Shortcut: V

Can be used to quickly insert multiple instances of the template selected in the Templates view. See [Creating Template Instances](https://doc.mapeditor.org/en/stable/manual/using-templates/#creating-template-instances).

### Insert Text

Shortcut: X

Text objects can be used to add arbitrary multi-line text to your maps. You can configure various font properties and the wrapping / clipping area, making them useful for both quick notes as well as text used in the game.

## Select Objects

Shortcut: S

When you’re not inserting new objects, you’re generally using the Select Objects tool. It packs a lot of functionality, which is outlined below.

### Selecting and Deselecting

You can select objects by clicking them or by dragging a rectangular lasso, selecting any object that intersect with its area. By holding Shift or Ctrl while clicking, you can add/remove single objects to/from the selection. Press Escape to deselect all objects.

When pressing and dragging on an object, this object is selected and moved. When this prevents you from starting a rectangular selection, you can hold Shift to force the selection rectangle.

**Since Tiled 1.0**

By default you interact with the top-most object. When you need to select an object below another object, first select the higher object and then hold Alt while clicking at the same location to select lower objects. You can also hold Alt while opening the context menu to get a list of all objects at the clicked location, so you may directly select the desired object.

**Since Tiled 1.2**

You can quickly switch to the [Edit Polygons](https://doc.mapeditor.org/en/stable/manual/objects/#edit-polygons-tool) tool by double-clicking on the polygon or polyline you want to edit.

### Moving

You can simply drag any single object, or drag already selected objects by dragging any one of them. Hold Ctrl to toggle snapping to the tile grid.

Hold Alt to force a move operation on the currently selected objects, regardless of where you click on the map. This is useful when the selected objects are small or covered by other objects.

The selected objects can also be moved with the arrow keys. By default this moves the objects pixel by pixel. Hold Shift while using the arrow keys to move the objects by distance of one tile.

### Resizing

You can use the resize handles to resize one or more selected objects. Hold Ctrl to keep the aspect ratio of the object and/or Shift to place the resize origin in the center.

Note that you can only change width and height independently when resizing a single object. When having multiple objects selected, the aspect ratio is constant because there would be no way to make that work for rotated objects without full support for transformations.

### Rotating

To rotate, click any selected object to change the resize handles into rotation handles. Before rotating, you can drag the rotation origin to another position if necessary. Hold Shift to rotate in 15-degree increments. Click any selected object again to go back to resize mode.

You can also rotate the selected objects in 90-degree steps by pressing Z or Shift + Z.

### Changing Stacking Order

If the active [Object Layer](https://doc.mapeditor.org/en/stable/manual/layers/#object-layer-introduction) has its Drawing Order property set to Manual (the default is Top Down), you can control the stacking order of the selected objects within their object layer using the following keys:

* PgUp - Raise selected objects
* PgDown - Lower selected objects
* Home - Move selected objects to Top
* End - Move selected objects to Bottom

You can also find these actions in the context menu. When you have multiple Object Layers, the context menu also contains actions to move the selected objects to another layer.

### Flipping Objects

You can flip the selected objects horizontally by pressing X or vertically by pressing Y. For tile objects, this also flips their images.

## Edit Polygons

Shortcut: E

Polygons and polylines have their own editing needs and as such are covered by a separate tool, which allows selecting and moving around their nodes. You can select and move the nodes of multiple polygons at the same time. Click a segment to select the nodes at both ends. Press Escape to deselect all nodes, or to switch back to the [Select Objects](https://doc.mapeditor.org/en/stable/manual/objects/#select-objects-tool) tool.

Nodes can be deleted by selecting them and choosing “Delete Nodes” from the context menu. The Delete key can also be used to delete the selected nodes, or the selected objects if no nodes are selected.

When you have selected multiple consecutive nodes of the same polygon, you can join them together by choosing “Join Nodes” from the context menu. You can also split the segments in between the nodes by choosing “Split Segments”. Alternatively, you can simply double-click a segment to split it at that location.

You can also delete a segment when two consecutive nodes are selected in a polygon by choosing “Delete Segment” in the context menu. This will convert a polygon into a polyline, or turn one polyline object in two polyline objects.

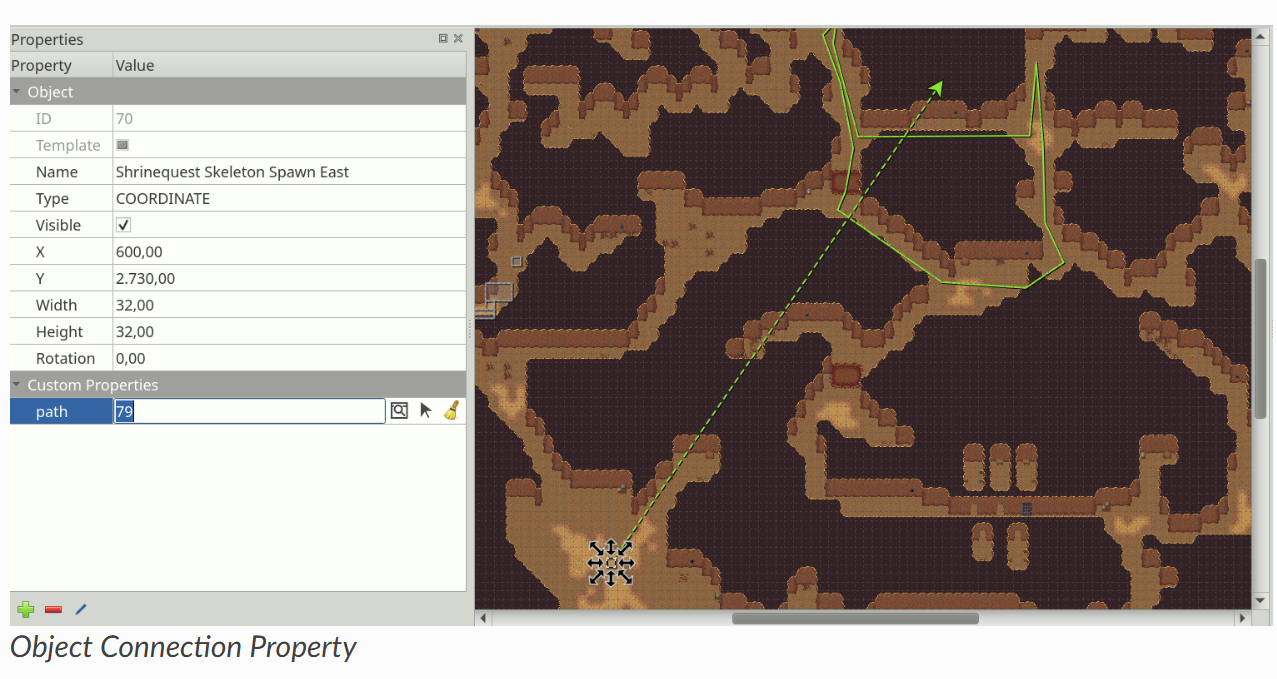
**Since Tiled 1.2**

It is possible to extend a polyline at either end, either by right-clicking those nodes and choosing “Extend Polyline”, or by switching to the [Insert Polygon](https://doc.mapeditor.org/en/stable/manual/objects/#insert-polygon-tool) tool and clicking on either end of an already selected polyline.

## Connecting Objects

It can often be useful to connect one object with another, like when a switch should open a certain door or an NPC should follow a certain path. To do this, add a custom property of type object to the source object. This property can then be set to the desired target object in several ways.

Make sure the property value is selected, as seen on the following screenshot:



Then, you can set the connection by either:

* Typing in the ID of the target object.
* Clicking the icon with the window and magnifier, to open a dialog where you can filter all objects on the map to find your target object.
* Clicking the arrow icon and then clicking an object on the map to set it as the target object.

As shown on the screenshot above, any connections between objects are rendered as arrows, taking the color of their target object (defined as part of the [object class](https://doc.mapeditor.org/en/stable/manual/custom-properties/#custom-classes) or by the color of the object layer). You can toggle the display of these arrows using *View -> Show Object References*.

If you’d like to get to the target object, but it’s very far away, you can jump there by right-clicking the property and selecting *Go to Object*.

**Future Extensions**

Here are some ideas about improvements that could be made to the above tools:

* Some improvements could still be made to the support for editing polygons and polylines, like allowing to rotate and scale the selected nodes ([#1487](https://github.com/mapeditor/tiled/issues/1487)).
* The tools could put short usage instructions in the status bar, to help new users without requiring them to carefully read the manual ([#1855](https://github.com/mapeditor/tiled/issues/1855)).

# Editing Tilesets

To edit a tileset it needs to be opened explicitly for editing. External tilesets can be opened via the *File* menu, but in general the quickest way to edit the tileset when it is already open in the *Tilesets* view is to click the small *Edit Tileset* button in the tool bar below the tileset.

## Two Types of Tileset

A tileset is a collection of tiles. Tiled currently supports two types of tilesets, which are chosen when creating a new tileset:

**Based on Tileset Image**

This tileset defines a fixed size for all tiles and the image from which these tiles are supposed to be cut. In addition it supports a margin around the tiles and a spacing between the tiles, which allows for using tileset images that either happen to have space between or around their tiles or those that have extruded the border pixels of each tile to avoid color bleeding.

**Collection of Images**

In this type of tileset each tile refers to its own image file. It is useful when the tiles aren’t the same size, or when the packing of tiles into a texture is done later on.

Regardless of the type of tileset, you can associate a lot of meta- information with it and its tiles. Some of this information can be for use in your game, like [collision information](https://doc.mapeditor.org/en/stable/manual/editing-tilesets/#tile-collision-editor) and [animations](https://doc.mapeditor.org/en/stable/manual/editing-tilesets/#tile-animation-editor). Other information is primarily meant for certain editing tools.

A screenshot of a computer

AI-generated content may be incorrect.

## Tileset Properties

You can access the tileset properties by using the menu action *Tileset > Tileset Properties*.

**Name**

The name of the tileset. Used to identify the tileset in the *Tilesets* view when editing a map.

**Since Tiled 1.4**

**Object Alignment**

The alignment to use for [tile objects](https://doc.mapeditor.org/en/stable/manual/objects/#insert-tile-tool) referring to tiles from this tileset. This affects the placement of the tile relative to the position of the object (the origin) and is also the location around which the rotation is applied.

Possible values are: *Unspecified* (the default), *Top Left*, *Top*, *Top Right*, *Left*, *Center*, *Right*, *Bottom Left*, *Bottom* and *Bottom Right*. When unspecified, tile object alignment is generally *Bottom Left*, except for Isometric maps where it is *Bottom*.

**Drawing Offset**

A drawing offset in pixels, applied when rendering any tile from the tileset (as part of tile layers or as tile objects). This is can be useful to make your tiles align to the grid.

**Background Color**

A background color for the tileset, which can be set in case the default dark-gray background is not suitable for your tiles.

**Since Tiled 1.0**

**Orientation**

When the tileset contains isometric tiles, you can set this to *Isometric*. This value, along with the **Grid Width** and **Grid Height** properties, is taken into account by overlays rendered on top of the tiles. This helps for example when specifying [Terrain Information](https://doc.mapeditor.org/en/stable/manual/editing-tilesets/#terrain-information). It also affects the orientation used by the [Tile Collision Editor](https://doc.mapeditor.org/en/stable/manual/editing-tilesets/#tile-collision-editor).

**Columns**

This is a read-only property for tilesets based on a tileset image, but for image collection tilesets you can control the number of columns used when displaying the tileset here.

**Image**

This property only exists for tilesets based on a tileset image. Selecting the value field will show an *Edit…* button, allowing you to change the parameters relevant to cutting the tiles from the image.

Of course, as with most data types in Tiled, you can also associate [Custom Properties](https://doc.mapeditor.org/en/stable/manual/custom-properties/) with the tileset.

## Tile Properties

**ID**

The ID of the tile in the tileset (read-only)

**Since Tiled 1.0**

**Class**

This property refers to custom classes defined in the [Custom Types Editor](https://doc.mapeditor.org/en/stable/manual/custom-properties/#custom-property-types). See the section about [Typed Tiles](https://doc.mapeditor.org/en/stable/manual/custom-properties/#typed-tiles) for more information.

**Width and Height**

The size of the tile (read-only)

**Probability**

Represents a relative probability that this tile will get chosen out of multiple options. This value is used in *Random Mode* and by the [Terrain Brush](https://doc.mapeditor.org/en/stable/manual/editing-tile-layers/#terrain-tool).

**Image**

Only relevant for tiles that are part of image collection tilesets, this shows the image file of the tile and allows you to change it.

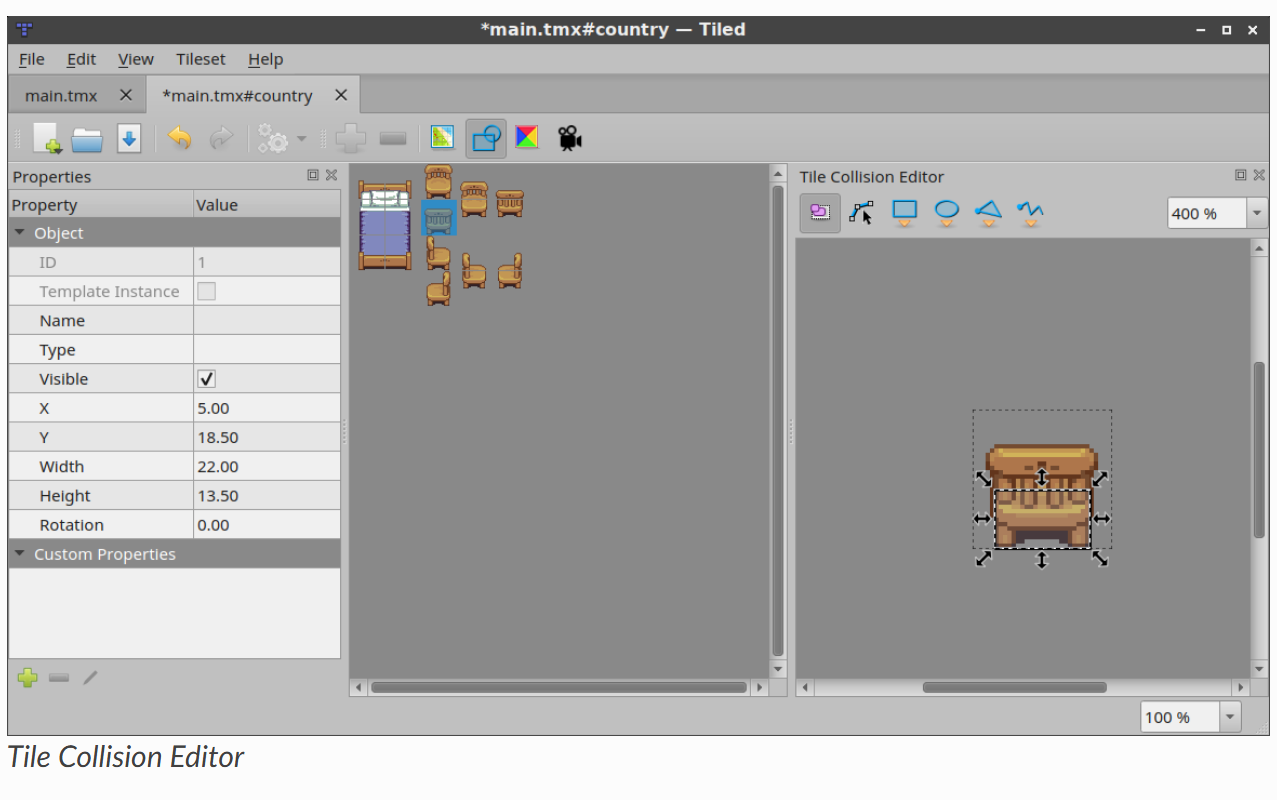
## Terrain Information

Terrain information can be added to a tileset to enable the use of the [Terrain Brush](https://doc.mapeditor.org/en/stable/manual/editing-tile-layers/#terrain-tool). See the section about [defining terrain information](https://doc.mapeditor.org/en/stable/manual/terrain/#define-terrain-information).

## Tile Collision Editor

The tile collision editor is available by clicking the *Tile Collision Editor* [tile-collision-editor-icon](https://doc.mapeditor.org/en/stable/_images/tile-collision-editor.png) button on the tool bar. This will open a view where you can create and edit shapes on the tile. You can also associate custom properties with each shape.

Usually these shapes define collision information for a certain sprite or for a tile representing level geometry, but of course you could also use them to add certain hot-spots to your sprites like for particle emitters or the source of gunshots.



To be able to easily check whether your tiles have the right collision shapes set up, they can be rendered on the map. To enable this, check *Show Tile Collision Shapes* in the *View* menu. The collision shapes are rendered for both tile layers and tile objects.



## Tile Animation Editor

The tile animation editor allows defining a single linear looping animation with each tile by referring to other tiles in the tileset as its frames. Open it by clicking the *Tile Animation Editor* tile-animation-editor-icon button.

Tile animations can be live-previewed in Tiled, which is useful for getting a feeling of what it would look like in-game. The preview can be turned on or off via *View > Show Tile Animations*.

A screenshot of a computer

AI-generated content may be incorrect.

The following steps allow to add or edit a tile animation:

* Select the tile in the main Tiled window. This will make the *Tile Animation Editor* window show the (initially empty) animation associated with that tile, along with all other tiles from the tileset.
* Drag tiles from the tileset view in the Tile Animation Editor into the list on the left to add animation frames. You can drag multiple tiles at the same time. Each new frame gets a default duration of 100 ms (or other value when set using the **Frame Duration** field at the top).
* Double-click on the duration of a frame to change it.
* Drag frames around in the list to reorder them.

A preview of the animation shows in the bottom left corner.

You can change the duration of multiple frames at once by selecting them, changing the value in the **Frame Duration** field and then clicking **Apply**.

# Custom Properties

## Adding Properties

## Custom Types

### Custom Enums

### Custom Classes

## Tile Property Inheritance

### Typed Tiles

# Using Templates

# Using Terrains

# Using Infinite Maps

# Working with Worlds

# Using Commands

# Automapping

# Export Formats

# Keyboard Shortcuts

# User Preferences

# Scripting