
Tyled
Release 0.1.dev2

Kurtis Fields

Nov 28, 2022

CONTENTS:

1	Installation	3
1.1	From PyPI	3
1.2	From GitHub	3
2	Quickstart	5
2.1	Create a Tyled Project File	5
2.2	Generate the Tileset	5
3	Commands	7
3.1	Bake	7
4	Projects	9
4.1	Options	9
5	Tilesets	11
5.1	Collection Tileset	11
5.2	Orthogonal Tileset	11
5.3	Atlas Tileset	12
6	tyled	13
6.1	tyled package	13
7	Indices and tables	15

CLI utility for generating [Tiled](#) Tilesets

[Source](#)

[Package](#)

**CHAPTER
ONE**

INSTALLATION

1.1 From PyPI

1.1.1 TLDR - do this at your own risk

```
pip install tyled
```

1.1.2 Recommended - pipX

If you don't already have it installed go to <https://pypi.org/project/pipx/> for instructions

```
pipx install tyled
```

1.2 From GitHub

Clone the repository

```
git clone https://github.com/kfields/tyled.git
```

Navigate to the new directory which contains the repository

```
cd tyled
```

Create a Python 3 virtual environment called *env*

```
python3 -m venv env
```

Activate the environment

```
source env/bin/activate
```

Install required packages

```
pip install -r requirements.txt
```


QUICKSTART

2.1 Create a Tyled Project File

```
name = 'mytileset'  
type = 'orthogonal'  
  
firstgid = 1  
source = 'platformer/tiles'  
tilewidth = 128  
tileheight = 128  
spacing = 0  
margin = 0  
columns = 12
```

2.2 Generate the Tileset

```
tyled bake mytileset.toml
```

This example will generate mytileset.tsx and mytileset.png in the current working directory

COMMANDS

3.1 Bake

```
tyled bake mytileset.toml [--save/--no-save] [--show/--no-show] [--rotation/--no-  
→rotation]
```

3.1.1 Options

Default options are `--save`, `--no-show`, `--no-rotation`

**CHAPTER
FOUR**

PROJECTS

Tyled Projects are defined using TOML files.

All Projects must at least have a name and type

```
name = 'mytileset'  
type = 'collection'
```

4.1 Options

Options may be defined within the project file. Any options defined here will override the command line options

```
[options]  
rotation = true
```


TILESETS

Tyled currently supports three different kinds of Tilesets:

5.1 Collection Tileset

A Collection Tileset is composed of tiles with images stored in separate files

```
name = 'mytileset'  
type = 'collection'  
  
firstgid = 1  
source = 'sticker-knight/map'
```

This example will generate mytileset.tsx in the current working directory

5.2 Orthogonal Tileset

An Orthogonal Tileset is composed of tiles that have the same dimensions in one image file

```
name = 'mytileset'  
type = 'orthogonal'  
  
firstgid = 1  
source = 'platormer/tiles'  
tilewidth = 128  
tileheight = 128  
spacing = 0  
margin = 0  
columns = 12
```

This example will generate mytileset.tsx and mytileset.png in the current working directory

5.3 Atlas Tileset

An Atlas Tileset is composed of tiles that do not have the same dimensions in one image file

```
name = 'mytileset'  
type = 'atlas'  
  
firstgid = 0  
source = 'sticker-knight/map'  
width = 1024  
height = 1024  
spacing = 0  
margin = 0
```

This example will generate mytileset.tsx and mytileset.png in the current working directory

5.3.1 Options

If you want a more compact spritesheet and your game engine supports it use the following:

```
[options]  
rotation = true
```

6.1 tyled package

6.1.1 Subpackages

`tyled.command` package

Module contents

`tyled.tileset` package

Subpackages

`tyled.tileset.atlas` package

Submodules

`tyled.tileset.atlas.atlas` module

Module contents

`tyled.tileset.collection` package

Submodules

`tyled.tileset.collection.collection` module

Module contents

`tyled.tileset.orthogonal` package

Submodules

`tyled.tileset.orthogonal.orthogonal` module

Module contents

Submodules

[tyled.tileset.factory module](#)

[tyled.tileset.tileset module](#)

Module contents

6.1.2 Submodules

[6.1.3 tyled.tile module](#)

[6.1.4 tyled.version module](#)

6.1.5 Module contents

CHAPTER
SEVEN

INDICES AND TABLES

- genindex
- modindex
- search