

---

# **Todoman Documentation**

***Release 1.8.0***

**Hugo Osvaldo Barrera**

January 20, 2017



<b>1</b>	<b>Features</b>	<b>3</b>
<b>2</b>	<b>Planned Features</b>	<b>5</b>
<b>3</b>	<b>Caveats</b>	<b>7</b>
<b>4</b>	<b>Table of Contents</b>	<b>9</b>
4.1	Installing . . . . .	9
4.2	Configuring . . . . .	9
4.3	Usage . . . . .	10
4.4	Licence . . . . .	12
<b>5</b>	<b>Indices and tables</b>	<b>13</b>



Todoman is a simple, standards-based, cli todo (aka: task) manager. Todos are stored into [icalendar](#) files, which means you can sync them via [CalDAV](#) using, for example, [vdirsyncer](#).

Todoman is now part of the `pimutils` project, and is hosted at [GitHub](#). The original location at [GitLab.com](#) is still kept as a mirror.



---

### Features

---

- Listing, editing and creating todos.
- Todos are read from individual ics files from the configured directory. This matches the `vdir` specification.
- There's support for the most common TODO features for now (summary, description, location, due date and priority) for now.
- Todoman should run on any major operating system.
- Unsupported fields may not be shown but are *never* deleted or altered.





---

### Planned Features

---

- Keep a SQL cache of all the entries, and update this only when the modification time of the ics files changes. This design is inspired (identical, really) to what khal does.
- Interactive mode. Including a list of todos, and easy selection for marking done and editing.
- Support for other icalendar fields which are not implemented yet.
- Customizable output format.

Pull requests and patches are welcome! Please report any issues on the project [issue tracker](#).



---

### Caveats

---

Priority granularity hasn't been completely implemented yet. Icalendar supports priorities 1-9 or none. Todoman supports only none or 1 (highest).

Due dates are generally shown and editable as dates with no time component.

Support for the `percent-completed` attribute is incomplete. Todoman can only mark todos as completed (100%), and will not reflect nor allow editing for values for `percent > 0 ^ percent < 100`.



---

## Table of Contents

---

### 4.1 Installing

If *todoman* is packaged for your OS/distribution, using your system's standard package manager is probably the easiest way to install *khal*:

- ArchLinux ([AUR](#))

#### 4.1.1 Install via PIP

Since *todoman* is written in python, you can use python's package managers, *pip* by executing:

```
pip install todoman
```

or the latest development version by executing:

```
pip install git+git://github.com/pimutils/todoman.git
```

This should also take care of installing all required dependencies.

#### 4.1.2 Manual installation

If *pip* is not available either (this is most unlikely), you'll need to download the source tarball and install via *setup.py*, though this is not a recommended installation method:

```
python3 setup.py install
```

#### 4.1.3 Requirements

Todoman requires python 3.3 or later. Installation of required libraries can be done via *pip*, or your OS's package manager. If you're interested in packaging *todoman*, all dependencies are listed in [requirements.txt](#).

Todoman will not work with python 2. However, keep in mind that python 2 and python 3 can coexist (and most distributions actually ship both).

### 4.2 Configuring

You'll need to configure Todoman before the first usage, using its simple ini-like configuration file.

### 4.2.1 Configuration File

The configuration file should be placed in `$XDG_CONFIG_DIR/todoman/todoman.conf`. `$XDG_CONFIG_DIR` defaults to `~/.config` in most situations, so this will generally be `~/.config/todoman/todoman.conf`.

#### Main section

- `path`: A glob pattern matching the directories where your todos are located.
- `date_format`: The date format used both for displaying dates, and parsing input dates. If this option is not specified the ISO-8601 (`%Y-%m-%d`) format is used.
- `color`: By default todoman will disable colored output if stdout is not a TTY (value `auto`). Set to `never` to disable colored output entirely, or `always` to enable it regardless. This can be overridden with the `--color` option.
- `default_list`: The default list for adding a todo. If you do not specify this option, you must use the `--list / -l` option every time you add a todo.

### 4.2.2 Sample configuration

The below example should serve as a reference. It will read ics files from any directory inside `~/.local/share/calendars/`, and use the ISO-8601 date format (note that this is the default format, so this particular declaration is redundant).

```
[main]
# A glob expression which matches all directories relevant.
path = ~/.local/share/calendars/*
date_format = %Y-%m-%d
default_list = Personal
```

### 4.2.3 Color and displayname

- You can set a color for each task list by creating a `color` file containing a colorcode in the format `#RRGGBB`.
- A file named `displayname` decides how the task list should be named. The default is the directory name.

See also [this discussion about metadata for collections in vdirsyncer](#).

## 4.3 Usage

Todoman usage is [CLI](#) based (thought there are some TUI bits, and the intentions is to also provide a fully [TUI](#)-based interface).

First of all, the classic usage output:

```
$ todo --help
Usage: todo [OPTIONS] COMMAND [ARGS]...

Options:
  --human-time / --no-human-time  Accept informal descriptions such as
                                  "tomorrow" instead of a properly formatted
                                  date.
```

```
--colour, --color TEXT      By default todoman will disable colored
                             output if stdout is not a TTY (value
                             `auto`). Set to `never` to disable colored
                             output entirely, or `always` to enable it
                             regardless.

--version                  Show the version and exit.
--help                     Show this message and exit.
```

#### Commands:

```
copy    Copy tasks to another list.
delete  Delete tasks.
done    Mark a task as done.
edit    Edit the task with id ID.
flush   Delete done tasks.
list    List unfinished tasks.
move    Move tasks to another list.
new     Create a new task with SUMMARY.
show    Show details about a task.
```

The default action is `list`, which outputs all tasks for all calendars, each with a non-permanent unique id:

```
1 [ ] ! 2015-04-30 Close bank account (0%)
2 [ ] !           Send minimimer back for warranty replacement (0%)
3 [X] 2015-03-29 Buy soy milk (100%)
4 [ ]           Fix the iPad's screen (0%)
5 [ ]           Fix the Touchad battery (0%)
```

The columns, in order, are:

- A temporary id.
- Whether the task has been completed or not.
- An `!` indicating it's an urgent task.
- The due date
- The task summary
- The completed percentage

The temporary id is retained by `todoman` until the next time you run the `list` command (remember: it's the default command is no other is specified).

To operate on a todo, the id is what's used to reference it. For example, to edit the *Buy soy milk* task from the example above, the proper command is `todo edit 3`, or `todo undo 3` to un-mark the task as done.

Editing tasks can only be done via the TUI interface for now, and cannot be done via the command line yet.

### 4.3.1 Synchronization

If you want to synchronize your tasks, you'll need something that syncs via CalDAV. `vdirsyncer` is the recommended tool for this.

### 4.3.2 Interactive shell

If you install `click-repl`, `todoman` gets a new command called `repl`, which launches an interactive shell with tab-completion.

## 4.4 Licence

Todoman is licensed under the MIT licence:

```
Copyright (c) 2015-2016, Hugo Osvaldo Barrera <hugo@barrera.io>
```

```
Permission to use, copy, modify, and/or distribute this software for any  
purpose with or without fee is hereby granted, provided that the above  
copyright notice and this permission notice appear in all copies.
```

```
THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH  
REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND  
FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT,  
INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM  
LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR  
OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR  
PERFORMANCE OF THIS SOFTWARE.
```



---

## Indices and tables

---

- `genindex`
- `modindex`
- `search`