
repo-helper-devenv

Release 0.6.1

Create virtual environments with repo-helper.

Dominic Davis-Foster

Apr 18, 2023

Contents

1	Installation	1
1.1	from PyPI	1
1.2	from GitHub	1
2	Usage	3
2.1	repo-helper devenv	3
3	Public API	5
3.1	install_requirements	5
3.2	mkdevenv	5
3.3	read_pyvenv	6
4	Downloading source code	7
4.1	Building from source	8
5	License	9
	Python Module Index	11
	Index	13

Installation

1.1 from PyPI

```
$ python3 -m pip install repo_helper_devenv --user
```

1.2 from GitHub

```
$ python3 -m pip install git+https://github.com/repo-helper/repo_helper_devenv@master --user
```


Usage

2.1 repo-helper devenv

Create a virtualenv.

```
repo-helper devenv [OPTIONS] [DEST]
```

Options

- U, --upgrade**
Upgrade all specified packages to the newest available version.
- version**
Show the version and exit.
- colour, --no-colour**
Whether to use coloured output.
- v, --verbose**
Show verbose output.

Arguments

DEST
Optional argument. Default 'venv'

DEST is relative to the repository root.

Changed in version 0.4.0: Added the `-U / --upgrade` option.

Public API

Create virtual environments with `repo-helper`.

Functions:

<code>install_requirements(session, requirements_file)</code>	require-	Install requirements into a virtualenv.
<code>mkdevenv(repo_dir[, venv_dir, verbosity, ...])</code>		Create a “devenv”.
<code>read_pyvenv(venv_dir)</code>		Reads the <code>pyvenv.cfg</code> for the given virtualenv, and returns a <code>key: value</code> mapping of its contents.

`install_requirements` (*session*, *requirements_file*, *verbosity=1*, *, *upgrade=False*)

Install requirements into a virtualenv.

Parameters

- **`session`** (`Session`)
- **`requirements_file`** (`Union[str, Path, PathLike]`)
- **`verbosity`** (`int`) – The verbosity of the function. 0 = quiet, 2 = very verbose. Default 1.
- **`upgrade`** (`bool`) – Whether to upgrade all specified packages to the newest available version. Default `False`.

Changed in version 0.4.0: Added the `upgrade` keyword-only argument.

Deprecated since version 0.5.0: This will be removed in 1.0.0.

`mkdevenv` (*repo_dir*, *venv_dir='venv'*, *verbosity=1*, *, *upgrade=False*)

Create a “devenv”.

New in version 0.3.2.

Parameters

- **`repo_dir`** (`Union[str, Path, PathLike]`) – The root of the repository to create the devenv for.
- **`venv_dir`** (`Union[str, Path, PathLike]`) – The directory to create the devenv in, relative to `repo_dir`. Default `'venv'`.
- **`verbosity`** (`int`) – The verbosity of the function. 0 = quiet, 2 = very verbose. Default 1.
- **`upgrade`** (`bool`) – Whether to upgrade all specified packages to the newest available version. Default `False`.

Return type `int`

Changed in version 0.4.0: Added the `upgrade` keyword-only argument.

read_pyvenv (*venv_dir*)

Reads the `pyvenv.cfg` for the given virtualenv, and returns a `key: value` mapping of its contents.

New in version 0.3.2.

Parameters `venv_dir` (`Union[str, Path, PathLike]`)

Return type `Dict[str, str]`

Deprecated since version 0.4.0: This will be removed in 1.0.0. Use ‘`shippinglabel.read_pyvenv`’ instead.

Downloading source code

The `repo_helper_devenv` source code is available on GitHub, and can be accessed from the following URL:
`https://github.com/repo-helper/repo_helper_devenv`

If you have `git` installed, you can clone the repository with the following command:

```
$ git clone https://github.com/repo-helper/repo_helper_devenv
```

```
Cloning into 'repo_helper_devenv'...
remote: Enumerating objects: 47, done.
remote: Counting objects: 100% (47/47), done.
remote: Compressing objects: 100% (41/41), done.
remote: Total 173 (delta 16), reused 17 (delta 6), pack-reused 126
Receiving objects: 100% (173/173), 126.56 KiB | 678.00 KiB/s, done.
Resolving deltas: 100% (66/66), done.
```

Alternatively, the code can be downloaded in a ‘zip’ file by clicking:

Clone or download → Download Zip

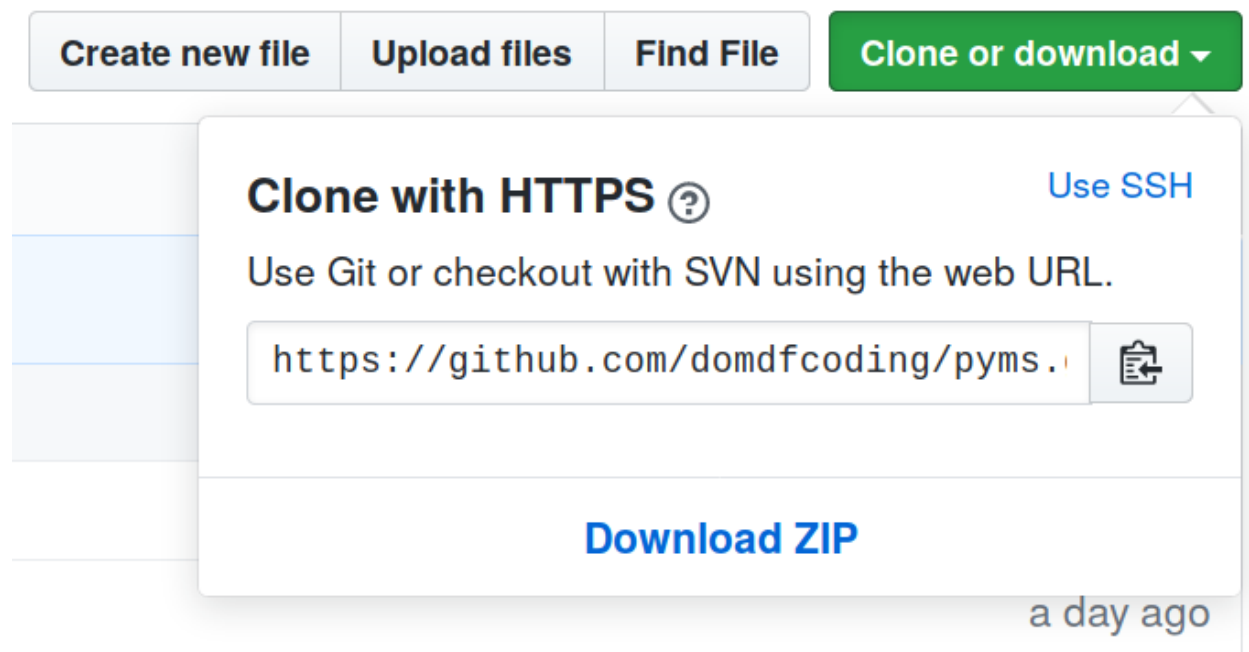


Fig. 1: Downloading a ‘zip’ file of the source code

4.1 Building from source

The recommended way to build `repo_helper_devenv` is to use `tox`:

```
$ tox -e build
```

The source and wheel distributions will be in the directory `dist`.

If you wish, you may also use `pep517.build` or another **PEP 517**-compatible build tool.

License

repo_helper_devenv is licensed under the [MIT License](#)

A short and simple permissive license with conditions only requiring preservation of copyright and license notices. Licensed works, modifications, and larger works may be distributed under different terms and without source code.

Permissions

- Commercial use – The licensed material and derivatives may be used for commercial purposes.
- Modification – The licensed material may be modified.
- Distribution – The licensed material may be distributed.
- Private use – The licensed material may be used and modified in private.

Conditions

- License and copyright notice – A copy of the license and copyright notice must be included with the licensed material.

Limitations

- Liability – This license includes a limitation of liability.
- Warranty – This license explicitly states that it does NOT provide any warranty.

[See more information on choosealicense.com](#) ⇒

```
Copyright (c) 2020 Dominic Davis-Foster
```

```
Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:
```

```
The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.
```

```
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR
OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE
OR OTHER DEALINGS IN THE SOFTWARE.
```


Python Module Index

r

`repo_helper_devenv`, [5](#)

Symbols

-U
 repo-helper-devenv command line
 option, 3

--colour
 repo-helper-devenv command line
 option, 3

--no-colour
 repo-helper-devenv command line
 option, 3

--upgrade
 repo-helper-devenv command line
 option, 3

--verbose
 repo-helper-devenv command line
 option, 3

--version
 repo-helper-devenv command line
 option, 3

-v
 repo-helper-devenv command line
 option, 3

D

DEST
 repo-helper-devenv command line
 option, 3

I

install_requirements() (*in module
 repo_helper_devenv*), 5

M

MIT License, 9

mkdevenv() (*in module repo_helper_devenv*), 5

module
 repo_helper_devenv, 5

P

Python Enhancement Proposals
 PEP 517, 8

R

read_pyvenv() (*in module repo_helper_devenv*), 5

repo_helper_devenv
 module, 5

repo-helper-devenv command line option
 -U, 3

 --colour, 3

 --no-colour, 3

 --upgrade, 3

 --verbose, 3

 --version, 3

 -v, 3

 DEST, 3