
modulereport Documentation

Release 0.3.2

Bertil Kronlund

January 07, 2017

1	User's Guide	3
1.1	Installation	3
1.2	Quickstart	3
1.3	<code>modulereport(1)</code> man page	5
1.4	Development	6
1.5	Roadmap	7
1.6	History	7

[Github](#) | [PyPI](#)

Lists the modules imported by python script

1.1 Installation

1.1.1 Do I need to install pip to install modulereport?

pip is installed if you're using Python binaries downloaded from python.org.

1.1.2 Using Linux Package Managers

pip may also be installed by your Package Manager:

```
$ which pip
/usr/bin/pip
```

1.1.3 Install modulereport

With pip:

```
$ [sudo] pip install modulereport
```

1.1.4 Uninstalling modulereport

Uninstall modulereport like so:

```
$ [sudo] pip uninstall modulereport
```

1.2 Quickstart

List all imported modules in a given python script:

```
$ modulereport setup.py
-----
Full report:
-----

Name                               File
```

```
-----
m __future__      /usr/lib/python3.4/__future__.py
m __main__        setup.py
m _ast
m _bisect
m _bootlocale     /usr/lib/python3.4/_bootlocale.py
m _bz2            /usr/lib/python3.4/lib-dynload/_bz2.cpython-34m-x86_64-linux-gnu.so
m _codecs
m _collections
m _collections_abc /usr/lib/python3.4/_collections_abc.py
m _compat_pickle  /usr/lib/python3.4/_compat_pickle.py
m _ctypes         /usr/lib/python3.4/lib-dynload/_ctypes.cpython-34m-x86_64-linux-gnu.so
...
...
m warnings        /usr/lib/python3.4/warnings.py
m weakref         /usr/lib/python3.4/weakref.py
m webbrowser      /usr/lib/python3.4/webbrowser.py
P xml             /usr/lib/python3.4/xml/__init__.py
P xml.parsers     /usr/lib/python3.4/xml/parsers/__init__.py
m xml.parsers.expat /usr/lib/python3.4/xml/parsers/expat.py
P xmlrpc          /usr/lib/python3.4/xmlrpc/__init__.py
m xmlrpc.client   /usr/lib/python3.4/xmlrpc/client.py
m zipfile         /usr/lib/python3.4/zipfile.py
m zipimport
m zlib

Missing modules:
? _dummy_threading imported from dummy_threading
? _frozen_importlib imported from importlib
? _sysconfigdata_dm imported from _sysconfigdata
? apport_python_hook imported from sitecustomize
...
...
```

1.2.1 Show help:

```
modulereport --help
usage: modulereport [-h] [-s, --skipreport] [-l, --loaded] [-m, --missing]
                  [-V]
                  pathname

positional arguments:
  pathname              path to python file to analyze for imports

optional arguments:
  -h, --help            show this help message and exit
  -s, --skipreport      skip list of all modules
  -l, --loaded          show loaded modules
  -m, --missing         show missing modules
  -V,                  show program's version number and exit
```

1.2.2 Reference:

Modulereporter use *modulefinder.ModuleFinder* from Python 3 standard library. Source code for [Lib/modulefinder.py](#).

1.3 modulereport(1) man page

1.3.1 SYNOPSIS

```
modulereport [options]

modulereport --help
```

1.3.2 DESCRIPTION

Determine the set of modules imported by a script. Given path can be a list of directories to search for modules or a package.

1.3.3 OPTIONS

To see all options available in your installation, run:

```
modulereport --help
```

All options available as of modulereport v0.3.2:

```
positional arguments:
  pathname             path to python file to analyze for imports

optional arguments:
  -h, --help           show this help message and exit
  -s, --skipreport     skip list of all modules
  -l, --loaded         show loaded modules
  -m, --missing        show missing modules
  -V,                  show program's version number and exit
```

1.3.4 EXAMPLES

Show usage:

```
modulereport --help
```

Show program version:

```
modulereport -V
```

1.3.5 SEE ALSO

Module Reporter Homepage: <https://github.com/berrak/modulereport>

modulereport documentation: <https://modulereport.readthedocs.io>

1.3.6 BUGS

Please report all bugs to <https://github.com/berrak/modulereport/issues/>

1.4 Development

1.4.1 Upload to PyPI

A configuration file is required to upload to PyPI with the Makefile targets, *pypi-test* and *pypi*.

Typically add this to `~/.pypirc`:

```
[distutils]
index-servers=
    pypi
    test

[test]
repository = https://testpypi.python.org/pypi
username = <pypitesting-user-name>
password = <pypitesting-password>

[pypi]
repository = https://pypi.python.org/pypi
username = <pypi-user-name>
password = <pypi-password>
```

Ensure that the uploading tool *twine* is installed like so:

```
[sudo] pip install -U twine
```

1.4.2 Pull Requests

- Submit Pull Requests against the *master* branch.
- Provide a good description of what you’re doing and why.
- Provide tests that cover your changes and try to run the tests locally first.

Example. Assuming you set up GitHub account, forked modulereport repository from <https://github.com/berrak/modulereport> to your own page via web interface, and your fork is located at <https://github.com/<your-github-user-name>/modulereport>

```
$ git clone git@github.com:modulereport/modulereport.git
$ cd modulereport
# ...
$ git diff
$ git add <modified>
$ git status
$ git commit
```

You may reference relevant issues in commit messages (like #113) to make GitHub link issues and commits together, and with phrase like “fixes #113” you can even close relevant issues automatically. Now push the changes to your fork:

```
$ git push git@github.com:<your-github-user-name>/modulereport.git
```

Open Pull Requests page at <https://github.com/<your-github-user-name>/modulereport/pulls> and click “New pull request”. That’s it.

1.4.3 Running tests

Ways to run the tests locally:

```
$ make lint          # ensure code follow best practices
$ make test          # runs all unittests
$ make coverage      # runs coverage on code
$ make report        # makes a nice html page of coverage result
```

Lint (flake8) may complain for great many details, but `make test` will not run without clean code.

It can be configured to ignore certain codes in `setup.cfg` configuration file:

```
[flake8]
# it's not a bug, ignore:
# H101: Use TODO(NAME)
# H301: one import per line
ignore = H101,H301
```

1.4.4 Getting involved

The Module Reporter welcomes help in the following ways:

- Making Pull Requests for code, tests, or docs.
- Commenting on open issues and pull requests.

1.5 Roadmap

https://github.com/berrak/modulereport/milestones?direction=desc&sort=due_date&state=open

1.6 History

Patch release for documentation only.

0.3.2 (2017-01-07)

- Update developers docs.

Initial commit.

0.3.0 (2017-01-07)

- First Beta-release on PyPI.