



# tabnanny — Indentation validator

**Purpose:** Scan Python source code looking for suspicious indentation.

Consistent use of indentation is important in a language like Python, where white-space is significant. The tabnanny module provides a scanner to report on ambiguous use of indentation.

## Running from the Command Line

The simplest way to use tabnanny is to run it from the command line, passing the names of files to check. If you pass directory names, the directories are scanned recursively to find .py files to check.

Running tabnanny across the PyMOTW source code exposed one old module with tabs instead of spaces.

```
$ python3 -m tabnanny .
./source/queue/fetch_podcasts.py 65 "    \t\tparsed_url = \
urlparse(enclosure['url'])\n"
```

Line 65 of fetch\_podcasts.py had two tabs instead of 8 spaces. This wasn't apparent in a text editor, which was configured to with tabstops set to 4 spaces, so visually there was no difference.

```
for enclosure in entry.get('enclosures', []):
    parsed_url = urlparse(enclosure['url'])
    message('queuing {}'.format(
        parsed_url.path.rpartition('/')[0]))
    enclosure_queue.put(enclosure['url'])
```

Correcting line 65 and running tabnanny again showed another error on line 66. One last problem showed up on line 67.

If you want to scan files, but not see the details about the error, use the -q option to suppress all information except the filename.

```
$ python3 -m tabnanny -q .
./source/queue/fetch_podcasts.py
```

To see more information about the files being scanned, use the -v option.

```
'source/queue/': listing directory
'source/queue/fetch_podcasts.py': *** Line 65: trouble in tab
city! ***
offending line: "    \t\tparsed_url = urlparse(enclosure['url'])
\n"
indent not greater e.g. at tab sizes 1, 2
'source/queue/queue_fifo.py': Clean bill of health.
'source/queue/queue_lifo.py': Clean bill of health.
'source/queue/queue_priority.py': Clean bill of health.
```

### Note

Running these examples against the PyMOTW source files won't report the same errors, since the issues have been fixed.

### See also

- [Standard library documentation for tabnanny](#)
- tokenize – Lexical scanner for Python source code.
- [flake8](#) – modular source code checker
- [pycodestyle](#) – Python style guide checker
- [pylint](#) – python code static checker

Quick Links

[Running from the Command Line](#)

*This page was last updated 2016-12-31.*

Navigation

- [timeit](#) — Time the execution of small bits of Python code.
- [compileall](#) — Byte-compile Source Files



[Get the book](#)

*The output from all the example programs from PyMOTW-3 has been generated with Python 3.7.1, unless otherwise noted. Some of the features described here may not be available in earlier versions of Python.*

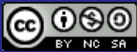
Looking for [examples for Python 2?](#)

This Site

- Module Index
- Index



© Copyright 2019, Doug Hellmann



Other Writing

- Blog
- The Python Standard Library By Example