

2to3 - Automated Python 2 to 3 code translation

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 6)

Unknown directive type "sectionauthor".

```
.. sectionauthor:: Benjamin Peterson <benjamin@python.org>
```

2to3 is a Python program that reads Python 2.x source code and applies a series of *fixers* to transform it into valid Python 3.x code. The standard library contains a rich set of fixers that will handle almost all code. 2to3 supporting library `:mod:`lib2to3`` is, however, a flexible and generic library, so it is possible to write your own fixers for 2to3.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 8); [backlink](#)

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 14)

Unknown directive type "deprecated-removed".

```
.. deprecated-removed:: 3.11 3.13
   The ``lib2to3`` module was marked pending for deprecation in Python 3.9
   (raising :exc:`PendingDeprecationWarning` on import) and fully deprecated
   in Python 3.11 (raising :exc:`DeprecationWarning`). The ``2to3`` tool is
   part of that. It will be removed in Python 3.13.
```

Using 2to3

2to3 will usually be installed with the Python interpreter as a script. It is also located in the `:file:`Tools/scripts`` directory of the Python root.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 25); [backlink](#)

Unknown interpreted text role "file".

2to3's basic arguments are a list of files or directories to transform. The directories are recursively traversed for Python sources. Here is a sample Python 2.x source file, `:file:`example.py``:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 31); [backlink](#)

Unknown interpreted text role "file".

```
def greet(name):
    print "Hello, {}!".format(name)
print "What's your name?"
name = raw_input()
greet(name)
```

It can be converted to Python 3.x code via 2to3 on the command line:

```
$ 2to3 example.py
```

A diff against the original source file is printed. 2to3 can also write the needed modifications right back to the source file. (A backup of the original file is made unless `:option:`!-n`` is also given.) Writing the changes back is enabled with the `:option:`!-w`` flag:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 45); [backlink](#)

Unknown interpreted text role "option".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 45); [backlink](#)

Unknown interpreted text role "option".

```
$ 2to3 -w example.py
```

After transformation, `:file:`example.py`` looks like this:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 54); [backlink](#)

Unknown interpreted text role "file".

```
def greet(name):
    print("Hello, {0}!".format(name))
print("What's your name?")
name = input()
greet(name)
```

Comments and exact indentation are preserved throughout the translation process.

By default, 2to3 runs a set of `:ref:`predefined fixers <2to3-fixers>``. The `:option:`!-f`` flag lists all available fixers. An explicit set of fixers to run can be given with `:option:`!-f``. Likewise the `:option:`!-x`` explicitly disables a fixer. The following example runs only the `imports` and `has_key` fixers:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 64); [backlink](#)

Unknown interpreted text role "ref".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 64); [backlink](#)

Unknown interpreted text role "option".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 64); [backlink](#)

Unknown interpreted text role "option".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 64); [backlink](#)

Unknown interpreted text role "option".

```
$ 2to3 -f imports -f has_key example.py
```

This command runs every fixer except the `apply` fixer:

```
$ 2to3 -x apply example.py
```

Some fixers are *explicit*, meaning they aren't run by default and must be listed on the command line to be run. Here, in addition to the default fixers, the `idioms` fixer is run:

```
$ 2to3 -f all -f idioms example.py
```

Notice how passing `all` enables all default fixers.

Sometimes 2to3 will find a place in your source code that needs to be changed, but 2to3 cannot fix automatically. In this case, 2to3 will print a warning beneath the diff for a file. You should address the warning in order to have compliant 3.x code.

2to3 can also refactor doctests. To enable this mode, use the `:option:`!-d`` flag. Note that *only* doctests will be refactored. This also doesn't require the module to be valid Python. For example, doctest like examples in a reST document could also be refactored with this option.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 94); [backlink](#)

Unknown interpreted text role "option".

The `:option:'!-v'` option enables output of more information on the translation process.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 99); [backlink](#)

Unknown interpreted text role "option".

Since some print statements can be parsed as function calls or statements, 2to3 cannot always read files containing the print function. When 2to3 detects the presence of the `from __future__ import print_function` compiler directive, it modifies its internal grammar to interpret `:func:'print'` as a function. This change can also be enabled manually with the `:option:'!-p'` flag. Use `:option:'!-p'` to run fixers on code that already has had its print statements converted. Also `:option:'!-e'` can be used to make `:func:'exec'` a function.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 102); [backlink](#)

Unknown interpreted text role "func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 102); [backlink](#)

Unknown interpreted text role "option".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 102); [backlink](#)

Unknown interpreted text role "option".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 102); [backlink](#)

Unknown interpreted text role "option".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 102); [backlink](#)

Unknown interpreted text role "func".

The `:option:'!-o'` or `:option:'!--output-dir'` option allows specification of an alternate directory for processed output files to be written to. The `:option:'!-n'` flag is required when using this as backup files do not make sense when not overwriting the input files.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 110); [backlink](#)

Unknown interpreted text role "option".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 110); [backlink](#)

Unknown interpreted text role "option".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 110); [backlink](#)

Unknown interpreted text role "option".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 115)

Unknown directive type "versionadded".

```
.. versionadded:: 3.2.3
   The :option:'!-o' option was added.
```

The `:option:'!-W'` or `:option:'!--write-unchanged-files'` flag tells 2to3 to always write output files even if no changes were required to

the file. This is most useful with `:option:'!-o'` so that an entire Python source tree is copied with translation from one directory to another. This option implies the `:option:'!-w'` flag as it would not make sense otherwise.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 118); [backlink](#)

Unknown interpreted text role "option".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 118); [backlink](#)

Unknown interpreted text role "option".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 118); [backlink](#)

Unknown interpreted text role "option".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 118); [backlink](#)

Unknown interpreted text role "option".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 124)

Unknown directive type "versionadded".

```
.. versionadded:: 3.2.3
   The :option:'!-W' flag was added.
```

The `:option:'!--add-suffix'` option specifies a string to append to all output filenames. The `:option:'!-n'` flag is required when specifying this as backups are not necessary when writing to different filenames. Example:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 127); [backlink](#)

Unknown interpreted text role "option".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 127); [backlink](#)

Unknown interpreted text role "option".

```
$ 2to3 -n -W --add-suffix=3 example.py
```

Will cause a converted file named `example.py3` to be written.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 137)

Unknown directive type "versionadded".

```
.. versionadded:: 3.2.3
   The :option:'!--add-suffix' option was added.
```

To translate an entire project from one directory tree to another use:

```
$ 2to3 --output-dir=python3-version/mycode -W -n python2-version/mycode
```

Fixers

Each step of transforming code is encapsulated in a fixer. The command `2to3 -l` lists them. As [ref](#) documented above `<2to3-using>`, each can be turned on and off individually. They are described here in more detail.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 152); [backlink](#)

Unknown interpreted text role "ref".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 157)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: apply
```

Removes usage of `:func:`apply``. For example ``apply(function, *args, **kwargs)`` is converted to ``function(*args, **kwargs)``.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 162)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: asserts
```

Replaces deprecated `:mod:`unittest`` method names with the correct ones.

From	To
<code>`failUnlessEqual(a, b)`</code>	<code>:meth:`assertEqual(a, b)</code> <unittest.TestCase.assertEqual>
<code>`assertEquals(a, b)`</code>	<code>:meth:`assertEqual(a, b)</code> <unittest.TestCase.assertEqual>
<code>`failIfEqual(a, b)`</code>	<code>:meth:`assertNotEqual(a, b)</code> <unittest.TestCase.assertNotEqual>
<code>`assertNotEquals(a, b)`</code>	<code>:meth:`assertNotEqual(a, b)</code> <unittest.TestCase.assertNotEqual>
<code>`failUnless(a)`</code>	<code>:meth:`assertTrue(a)</code> <unittest.TestCase.assertTrue>
<code>`assert_(a)`</code>	<code>:meth:`assertTrue(a)</code> <unittest.TestCase.assertTrue>
<code>`failIf(a)`</code>	<code>:meth:`assertFalse(a)</code> <unittest.TestCase.assertFalse>
<code>`failUnlessRaises(exc, cal)`</code>	<code>:meth:`assertRaises(exc, cal)</code> <unittest.TestCase.assertRaises>
<code>`failUnlessAlmostEqual(a, b)`</code>	<code>:meth:`assertAlmostEqual(a, b)</code> <unittest.TestCase.assertAlmostEqual>
<code>`assertAlmostEquals(a, b)`</code>	<code>:meth:`assertAlmostEqual(a, b)</code> <unittest.TestCase.assertAlmostEqual>
<code>`failIfAlmostEqual(a, b)`</code>	<code>:meth:`assertNotAlmostEqual(a, b)</code> <unittest.TestCase.assertNotAlmostEqual>
<code>`assertNotAlmostEquals(a, b)`</code>	<code>:meth:`assertNotAlmostEqual(a, b)</code> <unittest.TestCase.assertNotAlmostEqual>

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 195)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: basestring
```

Converts `:class:`basestring`` to `:class:`str``.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 199)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: buffer
```

Converts `:class:`buffer`` to `:class:`memoryview``. This fixer is optional because the `:class:`memoryview`` API is similar but not exactly the same as that of `:class:`buffer``.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 205)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: dict
```

Fixes dictionary iteration methods. `:meth:`dict.iteritems`` is converted to `:meth:`dict.items``, `:meth:`dict.iterkeys`` to `:meth:`dict.keys``, and `:meth:`dict.itervalues`` to `:meth:`dict.values``. Similarly, `:meth:`dict.viewitems``, `:meth:`dict.viewkeys`` and `:meth:`dict.viewvalues`` are converted respectively to `:meth:`dict.items``, `:meth:`dict.keys`` and `:meth:`dict.values``. It also wraps existing usages of `:meth:`dict.items``, `:meth:`dict.keys``, and `:meth:`dict.values`` in a call to `:class:`list``.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 215)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: except
```

Converts ```except X, T``` to ```except X as T```.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 219)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: exec
```

Converts the ```exec``` statement to the `:func:`exec`` function.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 223)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: execfile
```

Removes usage of `:func:`execfile``. The argument to `:func:`execfile`` is wrapped in calls to `:func:`open``, `:func:`compile``, and `:func:`exec``.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 228)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: exitfunc
```

Changes assignment of `:attr:`sys.exitfunc`` to use of the `:mod:`atexit`` module.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 233)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: filter
```

Wraps `:func:`filter`` usage in a `:class:`list`` call.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 237)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: funcattrs
```

Fixes function attributes that have been renamed. For example, `my_function.func_closure` is converted to `my_function.__closure__`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main [Doc] [library]2to3.rst, line 242)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: future
```

Removes `from __future__ import new_feature` statements.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main [Doc] [library]2to3.rst, line 246)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: getcwdu
```

Renames `:func:`os.getcwdu`` to `:func:`os.getcwd``.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main [Doc] [library]2to3.rst, line 250)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: has_key
```

Changes `dict.has_key(key)` to `key in dict`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main [Doc] [library]2to3.rst, line 254)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: idioms
```

This optional fixer performs several transformations that make Python code more idiomatic. Type comparisons like `type(x) is SomeClass` and `type(x) == SomeClass` are converted to `isinstance(x, SomeClass)`. `while 1` becomes `while True`. This fixer also tries to make use of `:func:`sorted`` in appropriate places. For example, this block ::

```
L = list(some_iterable)
L.sort()
```

is changed to ::

```
L = sorted(some_iterable)
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main [Doc] [library]2to3.rst, line 269)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: import
```

Detects sibling imports and converts them to relative imports.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\cpython-main [Doc] [library]2to3.rst, line 273)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: imports
```

Handles module renames in the standard library.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 277)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: imports2
```

Handles other modules renames in the standard library. It is separate from the :2to3fixer:`imports` fixer only because of technical limitations.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 282)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: input
```

Converts ``input(prompt)`` to ``eval(input(prompt))``.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 286)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: intern
```

Converts :func:`intern` to :func:`sys.intern`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 290)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: isinstance
```

Fixes duplicate types in the second argument of :func:`isinstance`. For example, ``isinstance(x, (int, int))`` is converted to ``isinstance(x, int)`` and ``isinstance(x, (int, float, int))`` is converted to ``isinstance(x, (int, float))``.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 297)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: itertools_imports
```

Removes imports of :func:`itertools.ifilter`, :func:`itertools.izip`, and :func:`itertools.imap`. Imports of :func:`itertools.ifilterfalse` are also changed to :func:`itertools.filterfalse`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 303)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: itertools
```

Changes usage of :func:`itertools.ifilter`, :func:`itertools.izip`, and :func:`itertools.imap` to their built-in equivalents. :func:`itertools.ifilterfalse` is changed to :func:`itertools.filterfalse`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 309)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: long
```


Renames `:class:`long`` to `:class:`int``.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 313)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: map
```

Wraps `:func:`map`` in a `:class:`list`` call. It also changes ``map(None, x)`` to ``list(x)``. Using ``from future_builtins import map`` disables this fixer.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 319)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: metaclass
```

Converts the old metaclass syntax (``__metaclass__ = Meta`` in the class body) to the new (``class X(metaclass=Meta)``).

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 324)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: methodattrs
```

Fixes old method attribute names. For example, ``meth.im_func`` is converted to ``meth.__func__``.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 329)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: ne
```

Converts the old not-equal syntax, ``<>``, to ``!=``.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 333)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: next
```

Converts the use of iterator's `:meth:`~iterator.next`` methods to the `:func:`next`` function. It also renames `:meth:`next`` methods to `:meth:`~iterator.__next__``.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 339)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: nonzero
```

Renames definitions of methods called `:meth:`~__nonzero__`` to `:meth:`~object.__bool__``.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 344)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: numliterals
```

Converts octal literals into the new syntax.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 348)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: operator
```

Converts calls to various functions in the :mod:`operator` module to other, but equivalent, function calls. When needed, the appropriate ``import`` statements are added, e.g. ``import collections.abc``. The following mapping are made:

From	To
``operator.isCallable(obj)``	``callable(obj)``
``operator.sequenceIncludes(obj)``	``operator.contains(obj)``
``operator.isSequenceType(obj)``	``isinstance(obj, collections.abc.Sequence)``
``operator.isMappingType(obj)``	``isinstance(obj, collections.abc.Mapping)``
``operator.isNumberType(obj)``	``isinstance(obj, numbers.Number)``
``operator.repeat(obj, n)``	``operator.mul(obj, n)``
``operator.irepeat(obj, n)``	``operator.imul(obj, n)``

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 367)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: paren
```

Add extra parenthesis where they are required in list comprehensions. For example, ``[x for x in 1, 2]`` becomes ``[x for x in (1, 2)]``.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 372)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: print
```

Converts the ``print`` statement to the :func:`print` function.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 376)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: raise
```

Converts ``raise E, V`` to ``raise E(V)`` and ``raise E, V, T`` to ``raise E(V).with_traceback(T)``. If ``E`` is a tuple, the translation will be incorrect because substituting tuples for exceptions has been removed in 3.0.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 382)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: raw_input
```

Converts :func:`raw_input` to :func:`input`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 386)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: reduce
```

Handles the move of `:func:`reduce`` to `:func:`functools.reduce``.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 390)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: reload
```

Converts `:func:`reload`` to `:func:`importlib.reload``.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 394)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: renames
```

Changes `:data:`sys.maxint`` to `:data:`sys.maxsize``.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 398)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: repr
```

Replaces backtick `repr` with the `:func:`repr`` function.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 402)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: set_literal
```

Replaces use of the `:class:`set`` constructor with set literals. This fixer is optional.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 407)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: standarderror
```

Renames `:exc:`StandardError`` to `:exc:`Exception``.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 411)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: sys_exc
```

Changes the deprecated `:data:`sys.exc_value``, `:data:`sys.exc_type``, `:data:`sys.exc_traceback`` to use `:func:`sys.exc_info``.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 416)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: throw
```

Fixes the API change in generator's :meth:`throw` method.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 420)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: tuple_params
```

Removes implicit tuple parameter unpacking. This fixer inserts temporary variables.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 425)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: types
```

Fixes code broken from the removal of some members in the :mod:`types` module.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 430)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: unicode
```

Renames :class:`unicode` to :class:`str`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 434)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: urllib
```

Handles the rename of :mod:`urllib` and :mod:`urllib2` to the :mod:`urllib` package.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 439)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: ws_comma
```

Removes excess whitespace from comma separated items. This fixer is optional.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 444)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: xrange
```

Renames :func:`xrange` to :func:`range` and wraps existing :func:`range` calls with :class:`list`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 449)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: xreadlines

Changes ``for x in file.xreadlines()`` to ``for x in file``.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 453)

Unknown directive type "2to3fixer".

```
.. 2to3fixer:: zip

Wraps :func:`zip` usage in a :class:`list` call. This is disabled when
``from future_builtins import zip`` appears.
```

:mod:`lib2to3` - 2to3's library

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 459); [backlink](#)

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 462)

Unknown directive type "module".

```
.. module:: lib2to3
   :synopsis: The 2to3 library
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 465)

Unknown directive type "moduleauthor".

```
.. moduleauthor:: Guido van Rossum
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 466)

Unknown directive type "moduleauthor".

```
.. moduleauthor:: Collin Winter
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 467)

Unknown directive type "moduleauthor".

```
.. moduleauthor:: Benjamin Peterson <benjamin@python.org>
```

Source code: `:source:`Lib/lib2to3/``

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 469); [backlink](#)

Unknown interpreted text role "source".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main] [Doc] [library]2to3.rst, line 473)

Unknown directive type "deprecated-removed".

```
.. deprecated-removed:: 3.11 3.13
```

Python 3.9 switched to a PEG parser (see :pep:`617`) while lib2to3 is using a less flexible LL(1) parser. Python 3.10 includes new language syntax that is not parsable by lib2to3's LL(1) parser (see :pep:`634`). The ``lib2to3`` module was marked pending for deprecation in Python 3.9 (raising :exc:`PendingDeprecationWarning` on import) and fully deprecated in Python 3.11 (raising :exc:`DeprecationWarning`). It will be removed from the standard library in Python 3.13. Consider third-party alternatives such as ``LibCST`_` or ``parso`_`.

Note

The `.mod:lib2to3` API should be considered unstable and may change drastically in the future.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]2to3.rst, line 485); [backlink](#)

Unknown interpreted text role "mod".