:mod:`pathlib` --- Object-oriented filesystem paths

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

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 5)

Unknown directive type "module".

```
.. module:: pathlib
   :synopsis: Object-oriented filesystem paths
```

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

Unknown directive type "versionadded".

```
.. versionadded:: 3.4
```

Source code: :source:`Lib/pathlib.py`

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 10); *backlink*

Unknown interpreted text role "source".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 12)

Unknown directive type "index".

```
.. index:: single: path; operations
```

This module offers classes representing filesystem paths with semantics appropriate for different operating systems. Path classes are divided between :ref:`pure paths <pure-paths>`, which provide purely computational operations without I/O, and :ref:`concrete paths <concrete-paths>`, which inherit from pure paths but also provide I/O operations.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 16); *backlink*

Unknown interpreted text role "ref".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 16); *backlink*

Unknown interpreted text role "ref".

If you've never used this module before or just aren't sure which class is right for your task, :class:`Path` is most likely what you need. It instantiates a :ref:`concrete path <concrete-paths>` for the platform the code is running on.

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

Unknown interpreted text role "class".

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

Unknown interpreted text role "ref".

Pure paths are useful in some special cases; for example:

1. If you want to manipulate Windows paths on a Unix machine (or vice versa). You cannot instantiate a :class:`WindowsPath` when running on Unix, but you can instantiate :class:`PureWindowsPath`.

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

   Unknown interpreted text role "class".

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

   Unknown interpreted text role "class".

2. You want to make sure that your code only manipulates paths without actually accessing the OS. In this case, instantiating one of the pure classes may be useful since those simply don't have any OS-accessing operations.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 38)

Unknown directive type "seealso".

```
.. seealso::
   :pep:`428`: The pathlib module -- object-oriented filesystem paths.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 41)

Unknown directive type "seealso".

```
.. seealso::
   For low-level path manipulation on strings, you can also use the
   :mod:`os.path` module.
```

Basic use

Importing the main class:

```
>>> from pathlib import Path
```

Listing subdirectories:

```
>>> p = Path('.')
>>> [x for x in p.iterdir() if x.is_dir()]
[PosixPath('.hg'), PosixPath('docs'), PosixPath('dist'),
 PosixPath('__pycache__'), PosixPath('build')]
```

Listing Python source files in this directory tree:

```
>>> list(p.glob('**/*.py'))
[PosixPath('test_pathlib.py'), PosixPath('setup.py'),
 PosixPath('pathlib.py'), PosixPath('docs/conf.py'),
 PosixPath('build/lib/pathlib.py')]
```

Navigating inside a directory tree:

```
>>> p = Path('/etc')
>>> q = p / 'init.d' / 'reboot'
>>> q
PosixPath('/etc/init.d/reboot')
>>> q.resolve()
PosixPath('/etc/rc.d/init.d/halt')
```

Querying path properties:

```
>>> q.exists()
True
>>> q.is_dir()
False
```

Opening a file:

```
>>> with q.open() as f: f.readline()
...
'#!/bin/bash\n'
```

Pure paths

Pure path objects provide path-handling operations which don't actually access a filesystem. There are three ways to access these classes, which we also call *flavours*:

A generic class that represents the system's path flavour (instantiating it creates either a :class:`PurePosixPath` or a :class:`PureWindowsPath`):

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 101); *backlink*

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 101); *backlink*

Unknown interpreted text role "class".

```
>>> PurePath('setup.py')      # Running on a Unix machine
PurePosixPath('setup.py')
```

Each element of *pathsegments* can be either a string representing a path segment, an object implementing the :class:`os.PathLike` interface which returns a string, or another path object:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 107); *backlink*

Unknown interpreted text role "class".

```
>>> PurePath('foo', 'some/path', 'bar')
PurePosixPath('foo/some/path/bar')
>>> PurePath(Path('foo'), Path('bar'))
PurePosixPath('foo/bar')
```

When *pathsegments* is empty, the current directory is assumed:

```
>>> PurePath()
PurePosixPath('.')
```

When several absolute paths are given, the last is taken as an anchor (mimicking :func:`os.path.join`'s behaviour):

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 121); *backlink*

Unknown interpreted text role "func".

```
>>> PurePath('/etc', '/usr', 'lib64')
PurePosixPath('/usr/lib64')
>>> PureWindowsPath('c:/Windows', 'd:bar')
PureWindowsPath('d:bar')
```

However, in a Windows path, changing the local root doesn't discard the previous drive setting:

```
>>> PureWindowsPath('c:/Windows', '/Program Files')
PureWindowsPath('c:/Program Files')
```

Spurious slashes and single dots are collapsed, but double dots (`'..'`) are not, since this would change the meaning of a path in the face of symbolic links:

```
>>> PurePath('foo//bar')
PurePosixPath('foo/bar')
>>> PurePath('foo/./bar')
PurePosixPath('foo/bar')
>>> PurePath('foo/../bar')
PurePosixPath('foo/../bar')
```

(a naÃ¯ve approach would make `PurePosixPath('foo/../bar')` equivalent to `PurePosixPath('bar')`, which is wrong if `foo` is a symbolic link to another directory)

Pure path objects implement the :class:`os.PathLike` interface, allowing them to be used anywhere the interface is accepted.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 150); *backlink*

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 153)

Unknown directive type "versionchanged".

```
.. versionchanged:: 3.6
   Added support for the :class:`os.PathLike` interface.
```

A subclass of :class:`PurePath`, this path flavour represents non-Windows filesystem paths:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 158); *backlink*

Unknown interpreted text role "class".

```
>>> PurePosixPath('/etc')
PurePosixPath('/etc')
```

pathsegments is specified similarly to :class:`PurePath`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 164); *backlink*

Unknown interpreted text role "class".

A subclass of :class:`PurePath`, this path flavour represents Windows filesystem paths:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 168); *backlink*

Unknown interpreted text role "class".

```
>>> PureWindowsPath('c:/Program Files/')
PureWindowsPath('c:/Program Files')
```

pathsegments is specified similarly to :class:`PurePath`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 174); *backlink*

Unknown interpreted text role "class".

Regardless of the system you're running on, you can instantiate all of these classes, since they don't provide any operation that does system calls.

General properties

Paths are immutable and hashable. Paths of a same flavour are comparable and orderable. These properties respect the flavour's case-folding semantics:

```
>>> PurePosixPath('foo') == PurePosixPath('FOO')
False
>>> PureWindowsPath('foo') == PureWindowsPath('FOO')
True
>>> PureWindowsPath('FOO') in { PureWindowsPath('foo') }
True
>>> PureWindowsPath('C:') < PureWindowsPath('d:')
True
```

Paths of a different flavour compare unequal and cannot be ordered:

```
>>> PureWindowsPath('foo') == PurePosixPath('foo')
False
>>> PureWindowsPath('foo') < PurePosixPath('foo')
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: '<' not supported between instances of 'PureWindowsPath' and 'PurePosixPath'
```

Operators

The slash operator helps create child paths, similarly to :func:`os.path.join`:

> **System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 209); *backlink***
>
> Unknown interpreted text role "func".

```
>>> p = PurePath('/etc')
>>> p
PurePosixPath('/etc')
>>> p / 'init.d' / 'apache2'
PurePosixPath('/etc/init.d/apache2')
>>> q = PurePath('bin')
>>> '/usr' / q
PurePosixPath('/usr/bin')
```

A path object can be used anywhere an object implementing :class:`os.PathLike` is accepted:

> **System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 220); *backlink***
>
> Unknown interpreted text role "class".

```
>>> import os
>>> p = PurePath('/etc')
>>> os.fspath(p)
'/etc'
```

The string representation of a path is the raw filesystem path itself (in native form, e.g. with backslashes under Windows), which you can pass to any function taking a file path as a string:

```
>>> p = PurePath('/etc')
>>> str(p)
'/etc'
>>> p = PureWindowsPath('c:/Program Files')
>>> str(p)
'c:\\Program Files'
```

Similarly, calling :class:`bytes` on a path gives the raw filesystem path as a bytes object, as encoded by :func:`os.fsencode`:

> **System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 239); *backlink***
>
> Unknown interpreted text role "class".

> **System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 239); *backlink***
>
> Unknown interpreted text role "func".

```
>>> bytes(p)
b'/etc'
```

Note

Calling :class:`bytes` is only recommended under Unix. Under Windows, the unicode form is the canonical representation of filesystem paths.

> **System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 246); *backlink***
>
> Unknown interpreted text role "class".

Accessing individual parts

To access the individual "parts" (components) of a path, use the following property:

> **System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 256)**
>
> Unknown directive type "data".

```
.. data:: PurePath.parts

   A tuple giving access to the path's various components::

      >>> p = PurePath('/usr/bin/python3')
      >>> p.parts
      ('/', 'usr', 'bin', 'python3')

      >>> p = PureWindowsPath('c:/Program Files/PSF')
      >>> p.parts
      ('c:\\', 'Program Files', 'PSF')

   (note how the drive and local root are regrouped in a single part)
```

Methods and properties

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 274)

Unknown directive type "testsetup".

```
.. testsetup::

   from pathlib import PurePath, PurePosixPath, PureWindowsPath
```

Pure paths provide the following methods and properties:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 280)

Unknown directive type "data".

```
.. data:: PurePath.drive

   A string representing the drive letter or name, if any::

      >>> PureWindowsPath('c:/Program Files/').drive
      'c:'
      >>> PureWindowsPath('/Program Files/').drive
      ''
      >>> PurePosixPath('/etc').drive
      ''

   UNC shares are also considered drives::

      >>> PureWindowsPath('//host/share/foo.txt').drive
      '\\\\host\\share'
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 296)

Unknown directive type "data".

```
.. data:: PurePath.root

   A string representing the (local or global) root, if any::

      >>> PureWindowsPath('c:/Program Files/').root
      '\\'
      >>> PureWindowsPath('c:Program Files/').root
      ''
      >>> PurePosixPath('/etc').root
      '/'

   UNC shares always have a root::

      >>> PureWindowsPath('//host/share').root
      '\\'
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 312)

Unknown directive type "data".

```
.. data:: PurePath.anchor

   The concatenation of the drive and root::

      >>> PureWindowsPath('c:/Program Files/').anchor
      'c:\\'
      >>> PureWindowsPath('c:Program Files/').anchor
      'c:'
      >>> PurePosixPath('/etc').anchor
      '/'
      >>> PureWindowsPath('//host/share').anchor
      '\\\\host\\share\\'
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 326)

Unknown directive type "data".

```
.. data:: PurePath.parents

   An immutable sequence providing access to the logical ancestors of
   the path::

      >>> p = PureWindowsPath('c:/foo/bar/setup.py')
      >>> p.parents[0]
      PureWindowsPath('c:/foo/bar')
      >>> p.parents[1]
      PureWindowsPath('c:/foo')
      >>> p.parents[2]
      PureWindowsPath('c:/')

   .. versionchanged:: 3.10
      The parents sequence now supports :term:`slices <slice>` and negative index values.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 342)

Unknown directive type "data".

```
.. data:: PurePath.parent

   The logical parent of the path::

      >>> p = PurePosixPath('/a/b/c/d')
      >>> p.parent
      PurePosixPath('/a/b/c')

   You cannot go past an anchor, or empty path::

      >>> p = PurePosixPath('/')
      >>> p.parent
      PurePosixPath('/')
      >>> p = PurePosixPath('.')
      >>> p.parent
      PurePosixPath('.')

   .. note::
      This is a purely lexical operation, hence the following behaviour::

         >>> p = PurePosixPath('foo/..')
         >>> p.parent
         PurePosixPath('foo')

      If you want to walk an arbitrary filesystem path upwards, it is
      recommended to first call :meth:`Path.resolve` so as to resolve
      symlinks and eliminate `".."` components.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 371)

Unknown directive type "data".

```
.. data:: PurePath.name

   A string representing the final path component, excluding the drive and
   root, if any::

      >>> PurePosixPath('my/library/setup.py').name
      'setup.py'

   UNC drive names are not considered::

      >>> PureWindowsPath('//some/share/setup.py').name
      'setup.py'
      >>> PureWindowsPath('//some/share').name
      ''
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 387)

Unknown directive type "data".

```
.. data:: PurePath.suffix

   The file extension of the final component, if any::

      >>> PurePosixPath('my/library/setup.py').suffix
      '.py'
      >>> PurePosixPath('my/library.tar.gz').suffix
      '.gz'
      >>> PurePosixPath('my/library').suffix
      ''
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 399)

Unknown directive type "data".

```
.. data:: PurePath.suffixes

   A list of the path's file extensions::

      >>> PurePosixPath('my/library.tar.gar').suffixes
      ['.tar', '.gar']
      >>> PurePosixPath('my/library.tar.gz').suffixes
      ['.tar', '.gz']
      >>> PurePosixPath('my/library').suffixes
      []
```

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

Unknown directive type "data".

```
.. data:: PurePath.stem

   The final path component, without its suffix::

      >>> PurePosixPath('my/library.tar.gz').stem
      'library.tar'
```

```
>>> PurePosixPath('my/library.tar').stem
'library'
>>> PurePosixPath('my/library').stem
'library'
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 423)

Unknown directive type "method".

```
.. method:: PurePath.as_posix()

   Return a string representation of the path with forward slashes (``/``)::

      >>> p = PureWindowsPath('c:\\windows')
      >>> str(p)
      'c:\\windows'
      >>> p.as_posix()
      'c:/windows'
```

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

Unknown directive type "method".

```
.. method:: PurePath.as_uri()

   Represent the path as a ``file`` URI.  :exc:`ValueError` is raised if
   the path isn't absolute.

      >>> p = PurePosixPath('/etc/passwd')
      >>> p.as_uri()
      'file:///etc/passwd'
      >>> p = PureWindowsPath('c:/Windows')
      >>> p.as_uri()
      'file:///c:/Windows'
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 447)

Unknown directive type "method".

```
.. method:: PurePath.is_absolute()

   Return whether the path is absolute or not.  A path is considered absolute
   if it has both a root and (if the flavour allows) a drive::

      >>> PurePosixPath('/a/b').is_absolute()
      True
      >>> PurePosixPath('a/b').is_absolute()
      False

      >>> PureWindowsPath('c:/a/b').is_absolute()
      True
      >>> PureWindowsPath('/a/b').is_absolute()
      False
      >>> PureWindowsPath('c:').is_absolute()
      False
      >>> PureWindowsPath('//some/share').is_absolute()
      True
```

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

Unknown directive type "method".

```
.. method:: PurePath.is_relative_to(*other)

   Return whether or not this path is relative to the *other* path.

      >>> p = PurePath('/etc/passwd')
      >>> p.is_relative_to('/etc')
      True
      >>> p.is_relative_to('/usr')
      False

   .. versionadded:: 3.9
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 480)

Unknown directive type "method".

```
.. method:: PurePath.is_reserved()

   With :class:`PureWindowsPath`, return ``True`` if the path is considered
   reserved under Windows, ``False`` otherwise.  With :class:`PurePosixPath`,
   ``False`` is always returned.

      >>> PureWindowsPath('nul').is_reserved()
      True
      >>> PurePosixPath('nul').is_reserved()
      False
```

```
   File system calls on reserved paths can fail mysteriously or have
   unintended effects.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 495)

Unknown directive type "method".

```
.. method:: PurePath.joinpath(*other)

   Calling this method is equivalent to combining the path with each of
   the *other* arguments in turn::

      >>> PurePosixPath('/etc').joinpath('passwd')
      PurePosixPath('/etc/passwd')
      >>> PurePosixPath('/etc').joinpath(PurePosixPath('passwd'))
      PurePosixPath('/etc/passwd')
      >>> PurePosixPath('/etc').joinpath('init.d', 'apache2')
      PurePosixPath('/etc/init.d/apache2')
      >>> PureWindowsPath('c:').joinpath('/Program Files')
      PureWindowsPath('c:/Program Files')
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 510)

Unknown directive type "method".

```
.. method:: PurePath.match(pattern)

   Match this path against the provided glob-style pattern.  Return ``True``
   if matching is successful, ``False`` otherwise.

   If *pattern* is relative, the path can be either relative or absolute,
   and matching is done from the right::

      >>> PurePath('a/b.py').match('*.py')
      True
      >>> PurePath('/a/b/c.py').match('b/*.py')
      True
      >>> PurePath('/a/b/c.py').match('a/*.py')
      False

   If *pattern* is absolute, the path must be absolute, and the whole path
   must match::

      >>> PurePath('/a.py').match('/*.py')
      True
      >>> PurePath('a/b.py').match('/*.py')
      False

   As with other methods, case-sensitivity follows platform defaults::

      >>> PurePosixPath('b.py').match('*.PY')
      False
      >>> PureWindowsPath('b.py').match('*.PY')
      True
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 541)

Unknown directive type "method".

```
.. method:: PurePath.relative_to(*other)

   Compute a version of this path relative to the path represented by
   *other*.  If it's impossible, ValueError is raised::

      >>> p = PurePosixPath('/etc/passwd')
      >>> p.relative_to('/')
      PurePosixPath('etc/passwd')
      >>> p.relative_to('/etc')
      PurePosixPath('passwd')
      >>> p.relative_to('/usr')
      Traceback (most recent call last):
        File "<stdin>", line 1, in <module>
        File "pathlib.py", line 694, in relative_to
          .format(str(self), str(formatted)))
      ValueError: '/etc/passwd' is not in the subpath of '/usr' OR one path is relative and the other absolute.

   NOTE: This function is part of :class:`PurePath` and works with strings. It does not check or access the underly
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 561)

Unknown directive type "method".

```
.. method:: PurePath.with_name(name)

   Return a new path with the :attr:`name` changed.  If the original path
   doesn't have a name, ValueError is raised::

      >>> p = PureWindowsPath('c:/Downloads/pathlib.tar.gz')
      >>> p.with_name('setup.py')
      PureWindowsPath('c:/Downloads/setup.py')
      >>> p = PureWindowsPath('c:/')
      >>> p.with_name('setup.py')
      Traceback (most recent call last):
```

```
  File "<stdin>", line 1, in <module>
  File "/home/antoine/cpython/default/Lib/pathlib.py", line 751, in with_name
    raise ValueError("%r has an empty name" % (self,))
ValueError: PureWindowsPath('c:/') has an empty name
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 578)

Unknown directive type "method".

```
.. method:: PurePath.with_stem(stem)

   Return a new path with the :attr:`stem` changed.  If the original path
   doesn't have a name, ValueError is raised::

      >>> p = PureWindowsPath('c:/Downloads/draft.txt')
      >>> p.with_stem('final')
      PureWindowsPath('c:/Downloads/final.txt')
      >>> p = PureWindowsPath('c:/Downloads/pathlib.tar.gz')
      >>> p.with_stem('lib')
      PureWindowsPath('c:/Downloads/lib.gz')
      >>> p = PureWindowsPath('c:/')
      >>> p.with_stem('')
      Traceback (most recent call last):
        File "<stdin>", line 1, in <module>
        File "/home/antoine/cpython/default/Lib/pathlib.py", line 861, in with_stem
          return self.with_name(stem + self.suffix)
        File "/home/antoine/cpython/default/Lib/pathlib.py", line 851, in with_name
          raise ValueError("%r has an empty name" % (self,))
      ValueError: PureWindowsPath('c:/') has an empty name

   .. versionadded:: 3.9
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 602)

Unknown directive type "method".

```
.. method:: PurePath.with_suffix(suffix)

   Return a new path with the :attr:`suffix` changed.  If the original path
   doesn't have a suffix, the new *suffix* is appended instead.  If the
   *suffix* is an empty string, the original suffix is removed::

      >>> p = PureWindowsPath('c:/Downloads/pathlib.tar.gz')
      >>> p.with_suffix('.bz2')
      PureWindowsPath('c:/Downloads/pathlib.tar.bz2')
      >>> p = PureWindowsPath('README')
      >>> p.with_suffix('.txt')
      PureWindowsPath('README.txt')
      >>> p = PureWindowsPath('README.txt')
      >>> p.with_suffix('')
      PureWindowsPath('README')
```

Concrete paths

Concrete paths are subclasses of the pure path classes. In addition to operations provided by the latter, they also provide methods to do system calls on path objects. There are three ways to instantiate concrete paths:

A subclass of :class:`PurePath`, this class represents concrete paths of the system's path flavour (instantiating it creates either a :class:`PosixPath` or a :class:`WindowsPath`):

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 631); *backlink*

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 631); *backlink*

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 631); *backlink*

Unknown interpreted text role "class".

```
>>> Path('setup.py')
PosixPath('setup.py')
```

pathsegments is specified similarly to :class:`PurePath`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 638); *backlink*

Unknown interpreted text role "class".

A subclass of :class:`Path` and :class:`PurePosixPath`, this class represents concrete non-Windows filesystem paths:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 642); *backlink*

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 642); backlink

Unknown interpreted text role "class".

```
>>> PosixPath('/etc')
PosixPath('/etc')
```

pathsegments is specified similarly to :class:`PurePath`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 648); backlink

Unknown interpreted text role "class".

A subclass of :class:`Path` and :class:`PureWindowsPath`, this class represents concrete Windows filesystem paths:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 652); backlink

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 652); backlink

Unknown interpreted text role "class".

```
>>> WindowsPath('c:/Program Files/')
WindowsPath('c:/Program Files')
```

pathsegments is specified similarly to :class:`PurePath`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 658); backlink

Unknown interpreted text role "class".

You can only instantiate the class flavour that corresponds to your system (allowing system calls on non-compatible path flavours could lead to bugs or failures in your application):

```
>>> import os
>>> os.name
'posix'
>>> Path('setup.py')
PosixPath('setup.py')
>>> PosixPath('setup.py')
PosixPath('setup.py')
>>> WindowsPath('setup.py')
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  File "pathlib.py", line 798, in __new__
    % (cls.__name__,))
NotImplementedError: cannot instantiate 'WindowsPath' on your system
```

Methods

Concrete paths provide the following methods in addition to pure paths methods. Many of these methods can raise an :exc:`OSError` if a system call fails (for example because the path doesn't exist).

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 682); backlink

Unknown interpreted text role "exc".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 686)

Unknown directive type "versionchanged".

```
.. versionchanged:: 3.8

   :meth:`~Path.exists()`, :meth:`~Path.is_dir()`, :meth:`~Path.is_file()`,
   :meth:`~Path.is_mount()`, :meth:`~Path.is_symlink()`,
   :meth:`~Path.is_block_device()`, :meth:`~Path.is_char_device()`,
   :meth:`~Path.is_fifo()`, :meth:`~Path.is_socket()` now return ``False``
   instead of raising an exception for paths that contain characters
   unrepresentable at the OS level.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 696)

Unknown directive type "classmethod".

```
.. classmethod:: Path.cwd()

   Return a new path object representing the current directory (as returned
   by :func:`os.getcwd`)::

      >>> Path.cwd()
      PosixPath('/home/antoine/pathlib')
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 705)

Unknown directive type "classmethod".

```
.. classmethod:: Path.home()

   Return a new path object representing the user's home directory (as
   returned by :func:`os.path.expanduser` with ``~`` construct). If the home
   directory can't be resolved, :exc:`RuntimeError` is raised.

   ::

      >>> Path.home()
      PosixPath('/home/antoine')

   .. versionadded:: 3.5
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 719)

Unknown directive type "method".

```
.. method:: Path.stat(*, follow_symlinks=True)

   Return a :class:`os.stat_result` object containing information about this path, like :func:`os.stat`.
   The result is looked up at each call to this method.

   This method normally follows symlinks; to stat a symlink add the argument
   ``follow_symlinks=False``, or use :meth:`~Path.lstat`.

   ::

      >>> p = Path('setup.py')
      >>> p.stat().st_size
      956
      >>> p.stat().st_mtime
      1327883547.852554

   .. versionchanged:: 3.10
      The *follow_symlinks* parameter was added.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 738)

Unknown directive type "method".

```
.. method:: Path.chmod(mode, *, follow_symlinks=True)

   Change the file mode and permissions, like :func:`os.chmod`.

   This method normally follows symlinks. Some Unix flavours support changing
   permissions on the symlink itself; on these platforms you may add the
   argument ``follow_symlinks=False``, or use :meth:`~Path.lchmod`.

   ::

      >>> p = Path('setup.py')
      >>> p.stat().st_mode
      33277
      >>> p.chmod(0o444)
      >>> p.stat().st_mode
      33060

   .. versionchanged:: 3.10
      The *follow_symlinks* parameter was added.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 758)

Unknown directive type "method".

```
.. method:: Path.exists()

   Whether the path points to an existing file or directory::

      >>> Path('.').exists()
      True
      >>> Path('setup.py').exists()
      True
      >>> Path('/etc').exists()
      True
      >>> Path('nonexistentfile').exists()
      False

   .. note::
      If the path points to a symlink, :meth:`exists` returns whether the
      symlink *points to* an existing file or directory.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 776)

Unknown directive type "method".

```
.. method:: Path.expanduser()

   Return a new path with expanded ``~`` and ``~user`` constructs,
   as returned by :meth:`os.path.expanduser`. If a home directory can't be
```

```
   resolved, :exc:`RuntimeError` is raised.

   ::

      >>> p = PosixPath('~/films/Monty Python')
      >>> p.expanduser()
      PosixPath('/home/eric/films/Monty Python')

   .. versionadded:: 3.5
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 791)

Unknown directive type "method".

```
.. method:: Path.glob(pattern)

   Glob the given relative *pattern* in the directory represented by this path,
   yielding all matching files (of any kind)::

      >>> sorted(Path('.').glob('*.py'))
      [PosixPath('pathlib.py'), PosixPath('setup.py'), PosixPath('test_pathlib.py')]
      >>> sorted(Path('.').glob('*/*.py'))
      [PosixPath('docs/conf.py')]

   Patterns are the same as for :mod:`fnmatch`, with the addition of "``**``"
   which means "this directory and all subdirectories, recursively".  In other
   words, it enables recursive globbing::

      >>> sorted(Path('.').glob('**/*.py'))
      [PosixPath('build/lib/pathlib.py'),
       PosixPath('docs/conf.py'),
       PosixPath('pathlib.py'),
       PosixPath('setup.py'),
       PosixPath('test_pathlib.py')]

   .. note::
      Using the "``**``" pattern in large directory trees may consume
      an inordinate amount of time.

   .. audit-event:: pathlib.Path.glob self,pattern pathlib.Path.glob
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 819)

Unknown directive type "method".

```
.. method:: Path.group()

   Return the name of the group owning the file.  :exc:`KeyError` is raised
   if the file's gid isn't found in the system database.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 825)

Unknown directive type "method".

```
.. method:: Path.is_dir()

   Return ``True`` if the path points to a directory (or a symbolic link
   pointing to a directory), ``False`` if it points to another kind of file.

   ``False`` is also returned if the path doesn't exist or is a broken symlink;
   other errors (such as permission errors) are propagated.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 834)

Unknown directive type "method".

```
.. method:: Path.is_file()

   Return ``True`` if the path points to a regular file (or a symbolic link
   pointing to a regular file), ``False`` if it points to another kind of file.

   ``False`` is also returned if the path doesn't exist or is a broken symlink;
   other errors (such as permission errors) are propagated.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 843)

Unknown directive type "method".

```
.. method:: Path.is_mount()

   Return ``True`` if the path is a :dfn:`mount point`: a point in a
   file system where a different file system has been mounted.  On POSIX, the
   function checks whether *path*'s parent, :file:`path/..`, is on a different
   device than *path*, or whether :file:`path/..` and *path* point to the same
   i-node on the same device --- this should detect mount points for all Unix
   and POSIX variants.  Not implemented on Windows.

   .. versionadded:: 3.7
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 855)

Unknown directive type "method".

```
.. method:: Path.is_symlink()

   Return ``True`` if the path points to a symbolic link, ``False`` otherwise.

   ``False`` is also returned if the path doesn't exist; other errors (such
   as permission errors) are propagated.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 863)

Unknown directive type "method".

```
.. method:: Path.is_socket()

   Return ``True`` if the path points to a Unix socket (or a symbolic link
   pointing to a Unix socket), ``False`` if it points to another kind of file.

   ``False`` is also returned if the path doesn't exist or is a broken symlink;
   other errors (such as permission errors) are propagated.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 872)

Unknown directive type "method".

```
.. method:: Path.is_fifo()

   Return ``True`` if the path points to a FIFO (or a symbolic link
   pointing to a FIFO), ``False`` if it points to another kind of file.

   ``False`` is also returned if the path doesn't exist or is a broken symlink;
   other errors (such as permission errors) are propagated.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 881)

Unknown directive type "method".

```
.. method:: Path.is_block_device()

   Return ``True`` if the path points to a block device (or a symbolic link
   pointing to a block device), ``False`` if it points to another kind of file.

   ``False`` is also returned if the path doesn't exist or is a broken symlink;
   other errors (such as permission errors) are propagated.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 890)

Unknown directive type "method".

```
.. method:: Path.is_char_device()

   Return ``True`` if the path points to a character device (or a symbolic link
   pointing to a character device), ``False`` if it points to another kind of file.

   ``False`` is also returned if the path doesn't exist or is a broken symlink;
   other errors (such as permission errors) are propagated.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 899)

Unknown directive type "method".

```
.. method:: Path.iterdir()

   When the path points to a directory, yield path objects of the directory
   contents::

      >>> p = Path('docs')
      >>> for child in p.iterdir(): child
      ...
      PosixPath('docs/conf.py')
      PosixPath('docs/_templates')
      PosixPath('docs/make.bat')
      PosixPath('docs/index.rst')
      PosixPath('docs/_build')
      PosixPath('docs/_static')
      PosixPath('docs/Makefile')

   The children are yielded in arbitrary order, and the special entries
   ``'.'`` and ``'..'`` are not included.  If a file is removed from or added
   to the directory after creating the iterator, whether a path object for
   that file be included is unspecified.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 920)

Unknown directive type "method".

```
.. method:: Path.lchmod(mode)

   Like :meth:`Path.chmod` but, if the path points to a symbolic link, the
   symbolic link's mode is changed rather than its target's.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 926)

Unknown directive type "method".

```
.. method:: Path.lstat()

   Like :meth:`Path.stat` but, if the path points to a symbolic link, return
   the symbolic link's information rather than its target's.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 932)

Unknown directive type "method".

```
.. method:: Path.mkdir(mode=0o777, parents=False, exist_ok=False)

   Create a new directory at this given path.  If *mode* is given, it is
   combined with the process' ``umask`` value to determine the file mode
   and access flags.  If the path already exists, :exc:`FileExistsError`
   is raised.

   If *parents* is true, any missing parents of this path are created
   as needed; they are created with the default permissions without taking
   *mode* into account (mimicking the POSIX ``mkdir -p`` command).

   If *parents* is false (the default), a missing parent raises
   :exc:`FileNotFoundError`.

   If *exist_ok* is false (the default), :exc:`FileExistsError` is
   raised if the target directory already exists.

   If *exist_ok* is true, :exc:`FileExistsError` exceptions will be
   ignored (same behavior as the POSIX ``mkdir -p`` command), but only if the
   last path component is not an existing non-directory file.

   .. versionchanged:: 3.5
      The *exist_ok* parameter was added.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 957)

Unknown directive type "method".

```
.. method:: Path.open(mode='r', buffering=-1, encoding=None, errors=None, newline=None)

   Open the file pointed to by the path, like the built-in :func:`open`
   function does::

      >>> p = Path('setup.py')
      >>> with p.open() as f:
      ...     f.readline()
      ...
      '#!/usr/bin/env python3\n'
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 969)

Unknown directive type "method".

```
.. method:: Path.owner()

   Return the name of the user owning the file.  :exc:`KeyError` is raised
   if the file's uid isn't found in the system database.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 975)

Unknown directive type "method".

```
.. method:: Path.read_bytes()

   Return the binary contents of the pointed-to file as a bytes object::

      >>> p = Path('my_binary_file')
      >>> p.write_bytes(b'Binary file contents')
      20
      >>> p.read_bytes()
      b'Binary file contents'

   .. versionadded:: 3.5
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 988)

Unknown directive type "method".

```
.. method:: Path.read_text(encoding=None, errors=None)

   Return the decoded contents of the pointed-to file as a string::

      >>> p = Path('my_text_file')
      >>> p.write_text('Text file contents')
      18
      >>> p.read_text()
      'Text file contents'

   The file is opened and then closed. The optional parameters have the same
   meaning as in :func:`open`.

   .. versionadded:: 3.5
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1004)

Unknown directive type "method".

```
.. method:: Path.readlink()

   Return the path to which the symbolic link points (as returned by
   :func:`os.readlink`)::

      >>> p = Path('mylink')
      >>> p.symlink_to('setup.py')
      >>> p.readlink()
      PosixPath('setup.py')

   .. versionadded:: 3.9
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1017)

Unknown directive type "method".

```
.. method:: Path.rename(target)

   Rename this file or directory to the given *target*, and return a new Path
   instance pointing to *target*.  On Unix, if *target* exists and is a file,
   it will be replaced silently if the user has permission.  *target* can be
   either a string or another path object::

      >>> p = Path('foo')
      >>> p.open('w').write('some text')
      9
      >>> target = Path('bar')
      >>> p.rename(target)
      PosixPath('bar')
      >>> target.open().read()
      'some text'

   The target path may be absolute or relative. Relative paths are interpreted
   relative to the current working directory, *not* the directory of the Path
   object.

   .. versionchanged:: 3.8
      Added return value, return the new Path instance.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1041)

Unknown directive type "method".

```
.. method:: Path.replace(target)

   Rename this file or directory to the given *target*, and return a new Path
   instance pointing to *target*.  If *target* points to an existing file or
   directory, it will be unconditionally replaced.

   The target path may be absolute or relative. Relative paths are interpreted
   relative to the current working directory, *not* the directory of the Path
   object.

   .. versionchanged:: 3.8
      Added return value, return the new Path instance.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1055)

Unknown directive type "method".

```
.. method:: Path.absolute()

   Make the path absolute, without normalization or resolving symlinks.
   Returns a new path object::

      >>> p = Path('tests')
      >>> p
      PosixPath('tests')
```

```
   >>> p.absolute()
   PosixPath('/home/antoine/pathlib/tests')
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1067)

Unknown directive type "method".

```
.. method:: Path.resolve(strict=False)

   Make the path absolute, resolving any symlinks.  A new path object is
   returned::

      >>> p = Path()
      >>> p
      PosixPath('.')
      >>> p.resolve()
      PosixPath('/home/antoine/pathlib')

   "``..``" components are also eliminated (this is the only method to do so)::

      >>> p = Path('docs/../setup.py')
      >>> p.resolve()
      PosixPath('/home/antoine/pathlib/setup.py')

   If the path doesn't exist and *strict* is ``True``, :exc:`FileNotFoundError`
   is raised.  If *strict* is ``False``, the path is resolved as far as possible
   and any remainder is appended without checking whether it exists.  If an
   infinite loop is encountered along the resolution path, :exc:`RuntimeError`
   is raised.

   .. versionadded:: 3.6
      The *strict* argument (pre-3.6 behavior is strict).
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1093)

Unknown directive type "method".

```
.. method:: Path.rglob(pattern)

   This is like calling :func:`Path.glob` with "``**/``" added in front of the
   given relative *pattern*::

      >>> sorted(Path().rglob("*.py"))
      [PosixPath('build/lib/pathlib.py'),
       PosixPath('docs/conf.py'),
       PosixPath('pathlib.py'),
       PosixPath('setup.py'),
       PosixPath('test_pathlib.py')]

   .. audit-event:: pathlib.Path.rglob self,pattern pathlib.Path.rglob
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1108)

Unknown directive type "method".

```
.. method:: Path.rmdir()

   Remove this directory.  The directory must be empty.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1113)

Unknown directive type "method".

```
.. method:: Path.samefile(other_path)

   Return whether this path points to the same file as *other_path*, which
   can be either a Path object, or a string.  The semantics are similar
   to :func:`os.path.samefile` and :func:`os.path.samestat`.

   An :exc:`OSError` can be raised if either file cannot be accessed for some
   reason.

   ::

      >>> p = Path('spam')
      >>> q = Path('eggs')
      >>> p.samefile(q)
      False
      >>> p.samefile('spam')
      True

   .. versionadded:: 3.5
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1134)

Unknown directive type "method".

```
.. method:: Path.symlink_to(target, target_is_directory=False)

   Make this path a symbolic link to *target*.  Under Windows,
```

```
   *target_is_directory* must be true (default ``False``) if the link's target
   is a directory.  Under POSIX, *target_is_directory*'s value is ignored.

   ::

      >>> p = Path('mylink')
      >>> p.symlink_to('setup.py')
      >>> p.resolve()
      PosixPath('/home/antoine/pathlib/setup.py')
      >>> p.stat().st_size
      956
      >>> p.lstat().st_size
      8

   .. note::
      The order of arguments (link, target) is the reverse
      of :func:`os.symlink`'s.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1155)

Unknown directive type "method".

```
.. method:: Path.hardlink_to(target)

   Make this path a hard link to the same file as *target*.

   .. note::
      The order of arguments (link, target) is the reverse
      of :func:`os.link`'s.

   .. versionadded:: 3.10
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1165)

Unknown directive type "method".

```
.. method:: Path.link_to(target)

   Make *target* a hard link to this path.

   .. warning::

      This function does not make this path a hard link to *target*, despite
      the implication of the function and argument names. The argument order
      (target, link) is the reverse of :func:`Path.symlink_to` and
      :func:`Path.hardlink_to`, but matches that of :func:`os.link`.

   .. versionadded:: 3.8

   .. deprecated:: 3.10

      This method is deprecated in favor of :meth:`Path.hardlink_to`, as the
      argument order of :meth:`Path.link_to`  does not match that of
      :meth:`Path.symlink_to`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1185)

Unknown directive type "method".

```
.. method:: Path.touch(mode=0o666, exist_ok=True)

   Create a file at this given path.  If *mode* is given, it is combined
   with the process' ``umask`` value to determine the file mode and access
   flags.  If the file already exists, the function succeeds if *exist_ok*
   is true (and its modification time is updated to the current time),
   otherwise :exc:`FileExistsError` is raised.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1194)

Unknown directive type "method".

```
.. method:: Path.unlink(missing_ok=False)

   Remove this file or symbolic link.  If the path points to a directory,
   use :func:`Path.rmdir` instead.

   If *missing_ok* is false (the default), :exc:`FileNotFoundError` is
   raised if the path does not exist.

   If *missing_ok* is true, :exc:`FileNotFoundError` exceptions will be
   ignored (same behavior as the POSIX ``rm -f`` command).

   .. versionchanged:: 3.8
      The *missing_ok* parameter was added.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1209)

Unknown directive type "method".

```
.. method:: Path.write_bytes(data)
```

```
Open the file pointed to in bytes mode, write *data* to it, and close the
file::

   >>> p = Path('my_binary_file')
   >>> p.write_bytes(b'Binary file contents')
   20
   >>> p.read_bytes()
   b'Binary file contents'

An existing file of the same name is overwritten.

.. versionadded:: 3.5
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1225)

Unknown directive type "method".

```
.. method:: Path.write_text(data, encoding=None, errors=None, newline=None)

   Open the file pointed to in text mode, write *data* to it, and close the
   file::

      >>> p = Path('my_text_file')
      >>> p.write_text('Text file contents')
      18
      >>> p.read_text()
      'Text file contents'

   An existing file of the same name is overwritten. The optional parameters
   have the same meaning as in :func:`open`.

   .. versionadded:: 3.5

   .. versionchanged:: 3.10
      The *newline* parameter was added.
```

Correspondence to tools in the :mod:`os` module

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1244); *backlink*

Unknown interpreted text role "mod".

Below is a table mapping various :mod:`os` functions to their corresponding :class:`PurePath`/:class:`Path` equivalent.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1247); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1247); *backlink*

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1247); *backlink*

Unknown interpreted text role "class".

Note

Not all pairs of functions/methods below are equivalent. Some of them, despite having some overlapping use-cases, have different semantics. They include :func:`os.path.abspath` and :meth:`Path.absolute`, :func:`os.path.relpath` and :meth:`PurePath.relative_to`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1252); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1252); *backlink*

Unknown interpreted text role "meth".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1252); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1252); *backlink*

Unknown interpreted text role "meth".

:mod:`os` and :mod:`os.path`	:mod:`pathlib`
System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1259); *backlink* Unknown interpreted text role "mod". System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1259); *backlink* Unknown interpreted text role "mod".	System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1259); *backlink* Unknown interpreted text role "mod".
:func:`os.path.abspath` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1261); *backlink* Unknown interpreted text role "func".	:meth:`Path.absolute` [1] System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1261); *backlink* Unknown interpreted text role "meth".
:func:`os.path.realpath` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1262); *backlink* Unknown interpreted text role "func".	:meth:`Path.resolve` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1262); *backlink* Unknown interpreted text role "meth".
:func:`os.chmod` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1263); *backlink* Unknown interpreted text role "func".	:meth:`Path.chmod` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1263); *backlink* Unknown interpreted text role "meth".
:func:`os.mkdir` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1264); *backlink* Unknown interpreted text role "func".	:meth:`Path.mkdir` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1264); *backlink* Unknown interpreted text role "meth".
:func:`os.makedirs` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1265); *backlink* Unknown interpreted text role "func".	:meth:`Path.mkdir` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1265); *backlink* Unknown interpreted text role "meth".

:mod:`os` and :mod:`os.path`	:mod:`pathlib`
System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1259); *backlink* Unknown interpreted text role "mod". System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1259); *backlink* Unknown interpreted text role "mod".	System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1259); *backlink* Unknown interpreted text role "mod".
:func:`os.rename`	:meth:`Path.rename`
System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1266); *backlink* Unknown interpreted text role "func".	System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1266); *backlink* Unknown interpreted text role "meth".
:func:`os.replace`	:meth:`Path.replace`
System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1267); *backlink* Unknown interpreted text role "func".	System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1267); *backlink* Unknown interpreted text role "meth".
:func:`os.rmdir`	:meth:`Path.rmdir`
System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1268); *backlink* Unknown interpreted text role "func".	System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1268); *backlink* Unknown interpreted text role "meth".
:func:`os.remove`, :func:`os.unlink`	:meth:`Path.unlink`
System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1269); *backlink* Unknown interpreted text role "func". System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1269); *backlink* Unknown interpreted text role "func".	System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1269); *backlink* Unknown interpreted text role "meth".
:func:`os.getcwd`	:func:`Path.cwd`
System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1270); *backlink* Unknown interpreted text role "func".	System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1270); *backlink* Unknown interpreted text role "func".

:mod:`os` and :mod:`os.path`	:mod:`pathlib`
System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1259); *backlink* Unknown interpreted text role "mod". System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1259); *backlink* Unknown interpreted text role "mod".	System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1259); *backlink* Unknown interpreted text role "mod".
:func:`os.path.exists`	:meth:`Path.exists`
System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1271); *backlink* Unknown interpreted text role "func".	System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1271); *backlink* Unknown interpreted text role "meth".
:func:`os.path.expanduser`	:meth:`Path.expanduser` and :meth:`Path.home`
System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1272); *backlink* Unknown interpreted text role "func".	System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1272); *backlink* Unknown interpreted text role "meth". System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1272); *backlink* Unknown interpreted text role "meth".
:func:`os.listdir`	:meth:`Path.iterdir`
System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1274); *backlink* Unknown interpreted text role "func".	System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1274); *backlink* Unknown interpreted text role "meth".
:func:`os.path.isdir`	:meth:`Path.is_dir`
System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1275); *backlink* Unknown interpreted text role "func".	System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1275); *backlink* Unknown interpreted text role "meth".

:mod:`os` and :mod:`os.path`	:mod:`pathlib`
System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1259); *backlink* Unknown interpreted text role "mod".	System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1259); *backlink* Unknown interpreted text role "mod".
System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1259); *backlink* Unknown interpreted text role "mod".	
:func:`os.path.isfile`	:meth:`Path.is_file`
System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1276); *backlink* Unknown interpreted text role "func".	System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1276); *backlink* Unknown interpreted text role "meth".
:func:`os.path.islink`	:meth:`Path.is_symlink`
System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1277); *backlink* Unknown interpreted text role "func".	System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1277); *backlink* Unknown interpreted text role "meth".
:func:`os.link`	:meth:`Path.hardlink_to`
System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1278); *backlink* Unknown interpreted text role "func".	System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1278); *backlink* Unknown interpreted text role "meth".
:func:`os.symlink`	:meth:`Path.symlink_to`
System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1279); *backlink* Unknown interpreted text role "func".	System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1279); *backlink* Unknown interpreted text role "meth".
:func:`os.readlink`	:meth:`Path.readlink`
System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1280); *backlink* Unknown interpreted text role "func".	System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1280); *backlink* Unknown interpreted text role "meth".

:mod:`os` and :mod:`os.path`	:mod:`pathlib`
System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1259); *backlink* Unknown interpreted text role "mod". System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1259); *backlink* Unknown interpreted text role "mod".	System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1259); *backlink* Unknown interpreted text role "mod".
:func:`os.path.relpath` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1281); *backlink* Unknown interpreted text role "func".	:meth:`Path.relative_to` [2] System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1281); *backlink* Unknown interpreted text role "meth".
:func:`os.stat` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1282); *backlink* Unknown interpreted text role "func".	:meth:`Path.stat`, :meth:`Path.owner`, :meth:`Path.group` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1282); *backlink* Unknown interpreted text role "meth". System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1282); *backlink* Unknown interpreted text role "meth". System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1282); *backlink* Unknown interpreted text role "meth".
:func:`os.path.isabs` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1285); *backlink* Unknown interpreted text role "func".	:meth:`PurePath.is_absolute` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1285); *backlink* Unknown interpreted text role "meth".
:func:`os.path.join` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1286); *backlink* Unknown interpreted text role "func".	:func:`PurePath.joinpath` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1286); *backlink* Unknown interpreted text role "func".

:mod:`os` and :mod:`os.path`	:mod:`pathlib`
System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1259); *backlink* Unknown interpreted text role "mod". System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1259); *backlink* Unknown interpreted text role "mod".	System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1259); *backlink* Unknown interpreted text role "mod".
:func:`os.path.basename` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1287); *backlink* Unknown interpreted text role "func".	:data:`PurePath.name` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1287); *backlink* Unknown interpreted text role "data".
:func:`os.path.dirname` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1288); *backlink* Unknown interpreted text role "func".	:data:`PurePath.parent` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1288); *backlink* Unknown interpreted text role "data".
:func:`os.path.samefile` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1289); *backlink* Unknown interpreted text role "func".	:meth:`Path.samefile` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1289); *backlink* Unknown interpreted text role "meth".
:func:`os.path.splitext` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1290); *backlink* Unknown interpreted text role "func".	:data:`PurePath.suffix` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1290); *backlink* Unknown interpreted text role "data".

Footnotes

[1] :func:`os.path.abspath` normalizes the resulting path, which may change its meaning in the presence of symlinks, while :meth:`Path.absolute` does not.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1294); *backlink*
Unknown interpreted text role "func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1294); *backlink*
Unknown interpreted text role "meth".

[2] :meth:`Path.relative_to` requires `self` to be the subpath of the argument, but :func:`os.path.relpath` does not.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1295); *backlink*

Unknown interpreted text role "meth".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]pathlib.rst, line 1295); *backlink*

Unknown interpreted text role "func".