

# Package ‘pathlib’

February 22, 2019

**Title** OO Path Manipulation in R

**Version** 0.1.0

**Description** An OO Interface for path manipulation, emulating pythons  
`pathlib`.

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** true

**Imports** R6, magrittr, glue (>= 1.3.0), logging, rlang, purrr

**RoxygenNote** 6.1.1

**Suggests** testthat, knitr, rmarkdown

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Matthew Strasiotto [aut, cre]

**Maintainer** Matthew Strasiotto <matthew.strasiotto@gmail.com>

**Repository** CRAN

**Date/Publication** 2019-02-22 14:40:06 UTC

## R topics documented:

Path . . . . .	2
Path\$. . . . .	2
Path\$. . . . .	3
Path\$dir . . . . .	3
Path\$join . . . . .	4
Path\$name . . . . .	4
Path\$new . . . . .	5
Path\$parent . . . . .	5
Path\$show . . . . .	6
%  % . . . . .	6
<b>Index</b>	<b>8</b>

---

Path	<i>OO Path Navigator</i>
------	--------------------------

---

**Description**

OO Path Navigator

**Usage**

Path

**Format**

An object of class R6ClassGenerator of length 24.

**See Also**

Other Path: [Path\\$..](#), [Path\\$.](#), [Path\\$dir](#), [Path\\$join](#), [Path\\$name](#), [Path\\$new](#), [Path\\$parent](#), [Path\\$show](#), [%/%](#)

---

Path\$.	<i>Get the elements in the directory</i>
---------	--

---

**Description**

Returns a named list of Path elements in a directory

**See Also**

Other Path: [Path\\$..](#), [Path\\$dir](#), [Path\\$join](#), [Path\\$name](#), [Path\\$new](#), [Path\\$parent](#), [Path\\$show](#), [Path, %/%](#)

---

Path\$. .                      *Get the parent of the current Path*

---

**Description**

Get the parent of the current Path

**Value**

a new Path object, corresponding to the parent of the calling Path

**See Also**

Other Path: [Path\\$.](#), [Path\\$dir](#), [Path\\$join](#), [Path\\$name](#), [Path\\$new](#), [Path\\$parent](#), [Path\\$show](#), [Path,%//%](#)

**Examples**

```
path <- Path$new("root/parent/path")
print(glue::glue("path   : {path$show}",
                "parent : {path$parent$show}",
                "..      : {path$..$show}",
                .sep = "\n"))
```

---

Path\$dir                      *Get the elements in the directory*

---

**Description**

Returns a named list of Path elements in a directory

**See Also**

Other Path: [Path\\$. .](#), [Path\\$.](#), [Path\\$join](#), [Path\\$name](#), [Path\\$new](#), [Path\\$parent](#), [Path\\$show](#), [Path,%//%](#)

---

Path\$join                      *Join two path components*

---

### Description

Given two path components, join them and return a new Path object

### Arguments

other                      the path component to join on the RHS. Can be any object coercible to character, including Path

### Value

A new Path object resulting from the joined paths

### See Also

Other Path: [Path\\$..](#), [Path\\$.](#), [Path\\$dir](#), [Path\\$name](#), [Path\\$new](#), [Path\\$parent](#), [Path\\$show](#), [Path,%//%](#)

### Examples

```
{
  path <- Path$new("some/path")
  path$join("other/path")
}
```

---

Path\$name                      *Get the name of the last element in the path (Incl. extension)*

---

### Description

Get the name of the last element in the path (Incl. extension)

### Value

the string corresponding to the path's name

### See Also

Other Path: [Path\\$..](#), [Path\\$.](#), [Path\\$dir](#), [Path\\$join](#), [Path\\$new](#), [Path\\$parent](#), [Path\\$show](#), [Path,%//%](#)

### Examples

```
path <- Path$new("path/to/some/element")
path$name
```

---

Path\$new	<i>Initialize the Path object</i>
-----------	-----------------------------------

---

**Description**

Initialize a Path object using a string

**Arguments**

path                    the path string to Path should refer to

**Value**

a new Path object, corresponding to the supplied string

**See Also**

Other Path: [Path\\$..](#), [Path\\$.](#), [Path\\$dir](#), [Path\\$join](#), [Path\\$name](#), [Path\\$parent](#), [Path\\$show](#), [Path,%//%](#)

**Examples**

```
Path$new("path/components/as/string")
```

---

Path\$parent	<i>Get the parent of the current Path</i>
--------------	---

---

**Description**

Get the parent of the current Path

**Value**

a new Path object, corresponding to the parent of the calling Path

**See Also**

Other Path: [Path\\$..](#), [Path\\$.](#), [Path\\$dir](#), [Path\\$join](#), [Path\\$name](#), [Path\\$new](#), [Path\\$show](#), [Path,%//%](#)

**Examples**

```
path <- Path$new("root/parent/path")
print(glue::glue("path   : {path$show}",
                 "parent : {path$parent$show}",
                 "..      : {path$..$show}",
                 .sep = "\n"))
```

---

Path\$show	<i>Show the entire path as a string</i>
------------	---

---

**Description**

Returns the path as a string

**Value**

the path as a string

**See Also**

Other Path: [Path\\$. .](#), [Path\\$.](#), [Path\\$dir](#), [Path\\$join](#), [Path\\$name](#), [Path\\$new](#), [Path\\$parent](#), [Path](#), [%//%](#)

---

<a href="#">%//%</a>	<i>Join two paths</i>
----------------------	-----------------------

---

**Description**

Join two path components in the "natural" way, as you would expect to using a / operator. lhs and rhs may be any object that implements the as.character S3 method, including Path

**Usage**

```
lhs %//% rhs
```

**Arguments**

lhs	the path component(s) to be joined on lhs
rhs	path components(s) to be joined on rhs

**Value**

a new Path object (or vector of) corresponding to the joined elements

**See Also**

Other Path: [Path\\$. .](#), [Path\\$.](#), [Path\\$dir](#), [Path\\$join](#), [Path\\$name](#), [Path\\$new](#), [Path\\$parent](#), [Path\\$show](#), [Path](#)

### Examples

```
path <- Path$new("root/parent/name")
child <- path %//% "child"
print(glue::glue("path : {path$show}",
                "child: {child$show}",
                .sep = "\n"))
```

# Index

## \*Topic **datasets**

Path, 2  
%//%, 2-6, 6

Path, 2, 2, 3-6  
Path\$. , 2, 2, 3-6  
Path\$. . , 2, 3, 3, 4-6  
Path\$dir, 2, 3, 3, 4-6  
Path\$join, 2-4, 4, 5, 6  
Path\$name, 2-4, 4, 5, 6  
Path\$new, 2-5, 5, 6  
Path\$parent, 2-5, 5, 6  
Path\$show, 2-6, 6