

Package: spud (via r-universe)

July 2, 2026

Title Sharepoint upload & download
Version 0.1.7
Description Enables uploading, downloading and listing of files in sharepoint.
Depends R (>= 3.5.0)
License MIT + file LICENSE
Encoding UTF-8
LazyData true
Suggests mockery, testthat, tibble (>= 3.0.0), withr
Imports getPass, glue, httr, jsonlite, tools, xml2
Roxygen list(markdown = TRUE)
RoxygenNote 7.1.1
Config/pak/sysreqs libxml2-dev libssl-dev
Repository <https://ncov-ic.r-universe.dev>
Date/Publication 2020-11-04 16:37:04 UTC
RemoteUrl <https://github.com/reside-ic/spud>
RemoteRef master
RemoteSha 046f6fd0198daa62bf9ae5b3c4fad1a0925d87ae

Contents

sharepoint	2
sharepoint_download	3
sharepoint_folder	4

Index	7
--------------	----------

sharepoint

Create sharepoint connection for downloading data.

Description

Create sharepoint connection for downloading data.

Create sharepoint connection for downloading data.

Public fields

`client` A low-level sharepoint client object, which can be used to interact directly with the sharepoint API. This object mostly handles authentication, etc.

Methods

Public methods:

- [sharepoint\\$new\(\)](#)
- [sharepoint\\$download\(\)](#)
- [sharepoint\\$folder\(\)](#)

Method `new()`: Create sharepoint object for downloading data from sharepoint

Usage:

```
sharepoint$new(sharepoint_url, auth = NULL)
```

Arguments:

`sharepoint_url` Root URL of sharepoint site to download from

`auth` Authentication data passed to the client

Returns: A new sharepoint object

Method `download()`: Download data from sharepoint

Usage:

```
sharepoint$download(  
  sharepoint_path,  
  dest = NULL,  
  progress = FALSE,  
  overwrite = FALSE  
)
```

Arguments:

`sharepoint_path` Path to the resource within sharepoint

`dest` Path to save downloaded data to

`progress` Display a progress bar during download?

`overwrite` Overwrite existing files?

Returns: Path to saved data

Method `folder()`: Create a folder object representing a sharepoint folder, with which one can list, download and upload files. See [sharepoint_folder](#) for more details.

Usage:

```
sharepoint$folder(site, path, verify = FALSE)
```

Arguments:

`site` The name of the sharepoint site (most likely a short string)

`path` Relative path within that shared site. It seems that "Shared Documents" is a common path that most likely represents a "Documents" collection when viewed in the sharepoint web interface.

`verify` Logical, indicating if the site/path combination is valid (slower but safer).

sharepoint_download *Download a dataset from sharepoint*

Description

Download a dataset from sharepoint

Usage

```
sharepoint_download(
  sharepoint_url,
  sharepoint_path,
  dest = NULL,
  progress = FALSE,
  overwrite = FALSE
)
```

Arguments

`sharepoint_url` The base URL of sharepoint e.g. <https://imperiallondon.sharepoint.com>

`sharepoint_path`

The path to the dataset you want to download - this should include any subsites in the url and should be of the form `sites/nested/subsites/docs/path/to/document` e.g. if you want to get the file at `Data/shape files/example.geojson` from the site `groupA` which is in site `facultyA` the full path would be `sites/facultyA/groupA/docs/Data/shape files/example.geojson` You should be able to get this if you locate the data you want to download in a browser and click menu on the RHS of the file name which appears on hover -> Copy link and manually edit to get the file path. See vignette for more details.

`dest` Path to location you want to save the data. The default save location is a tempfile with the same file extension as the downloaded file.

`progress` If TRUE then HTTP requests will print a progress bar

`overwrite` if TRUE then the `dest` will be overwritten if it exists (otherwise it an error will be thrown)

Value

Path to downloaded data

sharepoint_folder *Interact with sharepoint folders and their files.*

Description

Interact with sharepoint folders and their files.

Interact with sharepoint folders and their files.

Public fields

site Name of the sharepoint site (readonly)

path Path of the folder (readonly)

Methods**Public methods:**

- `sharepoint_folder$new()`
- `sharepoint_folder$files()`
- `sharepoint_folder$folders()`
- `sharepoint_folder$list()`
- `sharepoint_folder$delete()`
- `sharepoint_folder$parent()`
- `sharepoint_folder$folder()`
- `sharepoint_folder$create()`
- `sharepoint_folder$download()`
- `sharepoint_folder$upload()`

Method `new()`: Create `sharepoint_folder` object to enable listing, creating downloading and uploading files & folders

Usage:

```
sharepoint_folder$new(client, site, path, verify = FALSE)
```

Arguments:

`client` A low-level sharepoint client object, which can be used to interact directly with the sharepoint API.

`site` The name of the sharepoint site (most likely a short string)

`path` Relative path within that shared site. It seems that "Shared Documents" is a common path that most likely represents a "Documents" collection when viewed in the sharepoint web interface.

`verify` Logical, indicating if the site/path combination is valid (slower but safer).

Method files(): List all files within the folder

Usage:

```
sharepoint_folder$files(path = NULL)
```

Arguments:

path Folder relative to this folder, uses this folder if NULL

Method folders(): List all folders within the folder

Usage:

```
sharepoint_folder$folders(path = NULL)
```

Arguments:

path Folder relative to this folder, uses this folder if NULL

Method list(): List all folders and files within the folder; this is a convenience wrapper around the files and folders methods.

Usage:

```
sharepoint_folder$list(path = NULL)
```

Arguments:

path Folder relative to this folder, uses this folder if NULL

Method delete(): Delete a folder. Be extremely careful as you could use this to delete an entire sharepoint. Deleted files are sent to the recycle bin, so can be restored with relative ease, but it will still be alarming. There is a mechanism to prevent accidental deletion by declaring a file that exists within the folder.

Usage:

```
sharepoint_folder$delete(path, check)
```

Arguments:

path The path to delete. Use NULL to delete the current folder.

check A file (not folder) that exists directly within path, used as a method to verify that you really do want to delete this folder (to prevent things like accidental deletion of the entire sharepoint, for example).

Method parent(): Create an object referring to the parent folder

Usage:

```
sharepoint_folder$parent(verify = FALSE)
```

Arguments:

verify Verify that the folder exists (which it must really here)

Method folder(): Create an object referring to a child folder

Usage:

```
sharepoint_folder$folder(path, verify = FALSE)
```

Arguments:

path The name of the folder, relative to this folder

verify Verify that the folder exists (which it must really here)

Method create(): Create a folder on sharepoint

Usage:

```
sharepoint_folder$create(path)
```

Arguments:

path Folder relative to this folder

Method download(): Download a file from a folder

Usage:

```
sharepoint_folder$download(  
  path,  
  dest = NULL,  
  progress = FALSE,  
  overwrite = FALSE  
)
```

Arguments:

path The name of the path to download, relative to this folder

dest Path to save downloaded data to. If NULL then a temporary file with the same file extension as path is used. If coderaw() (or any other raw value) then the raw bytes will be returned.

progress Display htr's progress bar?

overwrite Overwrite the file if it exists?

Method upload(): Upload a file into a folder

Usage:

```
sharepoint_folder$upload(path, dest = NULL, progress = FALSE)
```

Arguments:

path The name of the path to upload, absolute, or relative to R's working directory.

dest Remote path save downloaded data to, relative to this folder. If NULL then the basename of the file is used.

progress Display htr's progress bar?

Index

sharepoint, [2](#)

sharepoint_download, [3](#)

sharepoint_folder, [3](#), [4](#)