

Package ‘laminr’

March 10, 2025

Title Client for 'LaminDB'

Version 1.0.0

Description Interact with 'LaminDB'. 'LaminDB' is an open-source data framework for biology. This package allows you to query and download data from 'LaminDB' instances.

License Apache License (>= 2)

URL <https://laminr.lamin.ai>, <https://github.com/laminlabs/laminr>

BugReports <https://github.com/laminlabs/laminr/issues>

Depends R (>= 4.1.0)

Imports cli, reticulate, utils, withr

Suggests anndata, arrow, httr, jsonlite, knitr, nanoparquet, purrr, quarto, R.utils, R6, readr, rlang, rstudioapi, rsvg, s3 (>= 1.1.0), Seurat, testthat (>= 3.0.0), tibble, yaml

VignetteBuilder quarto

Config/testthat/edition 3

Encoding UTF-8

RoxygenNote 7.3.2

NeedsCompilation no

Author Robrecht Cannoodt [aut, cre] (<<https://orcid.org/0000-0003-3641-729X>>),
Luke Zappia [aut] (<<https://orcid.org/0000-0001-7744-8565>>),
Data Intuitive [aut],
Lamin Labs [aut, cph]

Maintainer Robrecht Cannoodt <robrecht@data-intuitive.com>

Repository CRAN

Date/Publication 2025-03-10 14:30:06 UTC

Contents

get_current_lamin_instance	2
get_current_lamin_user	2
import_module	3
install_lamindb	4
lamin_connect	5
lamin_delete	5
lamin_disconnect	6
lamin_init	6
lamin_login	7
lamin_logout	8
lamin_save	8

Index	10
--------------	-----------

get_current_lamin_instance
Get current LaminDB instance

Description

Get the currently connected LaminDB instance

Usage

```
get_current_lamin_instance()
```

Details

This is done via a system call to lamin settings to avoid importing Python lamindb

Value

The slug of the current LaminDB instance, or NULL invisibly if no instance is found

get_current_lamin_user
Get current LaminDB user

Description

Get the currently logged in LaminDB user

Usage

```
get_current_lamin_user()
```

Details

This is done via a system call to `lamin settings` to avoid importing Python `lamindb`

Value

The handle of the current LaminDB user, or NULL invisibly if no user is found

<code>import_module</code>	<i>Import Python modules</i>
----------------------------	------------------------------

Description

This function can be used to import LaminDB Python modules with additional checks and nicer error messages.

Usage

```
import_module(module)
```

Arguments

`module` The name of the Python module to import

Value

An object representing a Python package

Examples

```
## Not run:  
# Import lamindb to start interacting with an instance  
ln <- import_module("lamindb")  
  
# Import other LaminDB modules  
bt <- import_module("bionty")  
wl <- import_module("wetlab")  
cc <- import_module("clincore")  
  
# Import any Python module  
np <- import_module("numpy")  
  
## End(Not run)
```

install_lamindb	<i>Install LaminDB</i>
-----------------	------------------------

Description

Create a Python environment containing **lamindb** or install **lamindb** into an existing environment.

Usage

```
install_lamindb(  
  ...,  
  envname = "r-lamindb",  
  extra_packages = NULL,  
  new_env = identical(envname, "r-lamindb"),  
  use = TRUE  
)
```

Arguments

...	Additional arguments passed to <code>reticulate::py_install()</code>
envname	String giving the name of the environment to install packages into
extra_packages	A vector giving the names of additional Python packages to install
new_env	Whether to remove any existing virtualenv with the same name before creating a new one with the requested packages
use	Whether to attempt use the new environment

Details

See `vignette("setup", package = "laminr")` for further details on setting up a Python environment

Value

NULL, invisibly

Examples

```
## Not run:  
install_lamindb()  
  
# Add additional packages to the environment  
install_lamindb(extra_packages = c("bionty", "wetlab"))  
  
# Install into a different environment  
install_lamindb(envname = "your-env")  
  
## End(Not run)
```

lamin_connect	<i>Connect to a LaminDB instance</i>
---------------	--------------------------------------

Description

Connect to a LaminDB instance by calling `lamin connect` on the command line

Usage

```
lamin_connect(instance)
```

Arguments

instance	Either a slug giving the instance to connect to (<owner>/<name>) or an instance URL (https://lamin.ai/owner/name)
----------	---

Details

Running this will set the LaminDB auto-connect option to True so you auto-connect to instance when importing Python lamindb.

Examples

```
## Not run:  
lamin_connect("laminlabs/cellxgene")  
  
## End(Not run)
```

lamin_delete	<i>LaminDB delete</i>
--------------	-----------------------

Description

Delete a LaminDB entity. Currently only supports instances.

Usage

```
lamin_delete(instance, force = FALSE)
```

Arguments

instance	Identifier for the instance to delete (e.g. "owner/name")
force	Whether to force deletion without asking for confirmation

Examples

```
## Not run:
lamin_init("to-delete")
lamin_delete("to-delete")

## End(Not run)
```

lamin_disconnect	<i>Disconnect from a LaminDB instance</i>
------------------	---

Description

Disconnect from the current LaminDB instance by calling `lamin connect` on the command line

Usage

```
lamin_disconnect()
```

Examples

```
## Not run:
lamin_disconnect()

## End(Not run)
```

lamin_init	<i>Initialise LaminDB</i>
------------	---------------------------

Description

Initialise a new LaminDB instance

Usage

```
lamin_init(storage, name = NULL, db = NULL, modules = NULL)

lamin_init_temp(
  name = "laminr-temp",
  db = NULL,
  modules = NULL,
  add_timestamp = TRUE,
  envir = parent.frame()
)
```

Arguments

storage	A local directory, AWS S3 bucket or Google Cloud Storage bucket
name	A name for the instance
db	A Postgres database connection URL, use NULL for SQLite.
modules	A vector of modules to include (e.g. "bionty")
add_timestamp	Whether to append a timestamp to name to make it unique
envir	An environment passed to <code>withr::defer()</code>

Details

For `lamin_init_temp()`, a time stamp is appended to name (if `add_timestamp = TRUE`) and then a new instance is initialised with `lamin_init()` using a temporary directory. A `lamin_delete()` call is registered as an exit handler with `withr::defer()` to clean up the instance when `envir` finishes.

The `lamin_init_temp()` function is mostly for internal use and in most cases users will want `lamin_init()`.

Examples

```
## Not run:
lamin_init("mydata", modules = c("bionty", "wetlab"))

## End(Not run)
```

lamin_login	<i>Log into LaminDB</i>
-------------	-------------------------

Description

Log in as a LaminDB user

Usage

```
lamin_login(user = NULL, api_key = NULL)
```

Arguments

user	Handle for the user to login as
api_key	API key for a user

Details

Depending on the input, one of these commands will be run (in this order):

1. If user is set then `lamin login <user>`
2. Else if `api_key` is set then set the `LAMIN_API_KEY` environment variable temporarily with `withr::with_envvar()` and run `lamin login`
3. Else if there is a stored user handle run `lamin login <handle>`
4. Else if the `LAMIN_API_KEY` environment variable is set run `lamin login`

Otherwise, exit with an error

<code>lamin_logout</code>	<i>Log out of LaminDB</i>
---------------------------	---------------------------

Description

Log out of LaminDB

Usage

```
lamin_logout()
```

<code>lamin_save</code>	<i>Save to a LaminDB instance</i>
-------------------------	-----------------------------------

Description

Save a file or folder to a LaminDB instance by calling `lamin save` on the command line

Usage

```
lamin_save(filepath, key = NULL, description = NULL, registry = NULL)
```

Arguments

<code>filepath</code>	Path to the file or folder to save
<code>key</code>	The key for the saved item
<code>description</code>	The description for the saved item
<code>registry</code>	The registry for the saved item

Details

See `lamin save --help` for details of what database entries are created for different file types

Examples

```
## Not run:  
my_file <- tempfile()  
lamin_save(my_file)  
  
## End(Not run)
```

Index

`get_current_lamin_instance`, 2
`get_current_lamin_user`, 2

`import_module`, 3
`install_lamindb`, 4

`lamin_connect`, 5
`lamin_delete`, 5
`lamin_delete()`, 7
`lamin_disconnect`, 6
`lamin_init`, 6
`lamin_init()`, 7
`lamin_init_temp(lamin_init)`, 6
`lamin_init_temp()`, 7
`lamin_login`, 7
`lamin_logout`, 8
`lamin_save`, 8

`withr::defer()`, 7