

# Package ‘jot’

August 27, 2022

**Title** Jot Down Values for Later

**Version** 0.0.4

**Description** Reproducible work requires a record of where every statistic originated.

When writing reports, some data is too big to load in the same environment and some statistics take a while to compute.

This package offers a way to keep notes on statistics, simple functions, and small objects.

Notepads can be locked to avoid accidental updates.

Notepads keep track of who added the notes and when the notes were added.

A simple text representation is used to allow for clear version histories.

**License** MIT + file LICENSE

**Encoding** UTF-8

**RoxygenNote** 7.2.1

**Imports** cli, yaml

**URL** <http://christophertkenny.com/jot/>,  
<https://github.com/christopherkenny/jot>

**BugReports** <https://github.com/christopherkenny/jot/issues>

**NeedsCompilation** no

**Author** Christopher T. Kenny [aut, cre]  
(<https://orcid.org/0000-0002-9386-6860>)

**Maintainer** Christopher T. Kenny <[christopherkenny@fas.harvard.edu](mailto:christopherkenny@fas.harvard.edu)>

**Repository** CRAN

**Date/Publication** 2022-08-27 15:00:01 UTC

## R topics documented:

|                        |   |
|------------------------|---|
| get_jot_user . . . . . | 2 |
| get_sys_time . . . . . | 2 |
| jot . . . . .          | 3 |
| jot_erase . . . . .    | 4 |
| jot_list . . . . .     | 4 |

|                       |   |
|-----------------------|---|
| jot_lock . . . . .    | 5 |
| jot_new_pad . . . . . | 6 |
| jot_open . . . . .    | 6 |
| jot_read . . . . .    | 7 |
| jot_skim . . . . .    | 8 |

|              |          |
|--------------|----------|
| <b>Index</b> | <b>9</b> |
|--------------|----------|

---

|              |                         |
|--------------|-------------------------|
| get_jot_user | <i>Get Current User</i> |
|--------------|-------------------------|

---

### Description

Get Current User

### Usage

```
get_jot_user(user = "")
```

### Arguments

|      |  |
|------|--|
| user | user to override get with. Default is '', where it checks options, then system info. |
|------|--|

### Value

character

### Examples

```
get_jot_user()
get_jot_user(NA_character_)
```

---

|              |                                   |
|--------------|-----------------------------------|
| get_sys_time | <i>Get System Time as Integer</i> |
|--------------|-----------------------------------|

---

### Description

Get System Time as Integer

### Usage

```
get_sys_time()
```

### Value

integer

**Examples**

```
get_sys_time()
```

---

jot

*Jot Down a Note*

---

**Description**

Jot Down a Note

**Usage**

```
jot(  
  note,  
  name,  
  overwrite = FALSE,  
  user = get_jot_user(),  
  time = get_sys_time(),  
  pad = jot_active()  
)
```

```
write_jot(  
  note,  
  name,  
  overwrite = FALSE,  
  user = get_jot_user(),  
  time = get_sys_time(),  
  pad = jot_active()  
)
```

**Arguments**

|           |  |
|-----------|--|
| note      | data to save. required.  |
| name      | character, name of data. required.   |
| overwrite | logical, default is FALSE. If a note of the same name exists, should it be over-written? |
| user      | name of user. default is get_jot_user().   |
| time      | system time. default is get_sys_time().  |
| pad       | path to file of pad to write to. default is jot_active().                                |

**Value**

path to pad, invisibly

**Examples**

```
tf <- tempfile()
jot_new_pad(tf)
jot_activate(tf)
jot(3, 'example')
```

---

jot\_erase

*Erase a Note*


---

**Description**

Erase a Note

**Usage**

```
jot_erase(name, pad = jot_active())
```

**Arguments**

|      |   |
|------|---|
| name | character, name of note to erase. required.               |
| pad  | path to file of pad to write to. default is jot_active(). |

**Value**

path to pad, invisibly

**Examples**

```
tf <- tempfile()
jot_new_pad(tf)
jot(3, 'example')
jot_erase('example')
```

---

jot\_list

*List out names of values in a jot notepad*


---

**Description**

List out names of values in a jot notepad

**Usage**

```
jot_list(pad = jot_active())
```

**Arguments**

|     |                         |
|-----|-------------------------|
| pad | path to notepad to open |
|-----|-------------------------|

**Value**

vlist of names of values in notepad

**Examples**

```
tf <- tempfile()
jot_new_pad(tf)
jot_activate(tf)
jot('ex_val', 'ex')
jot_list()
```

---

jot\_lock

*Lock or Unlock a jot Notepad*

---

**Description**

- jot\_lock() locks a notepad so that jot will not write to it.
- jot\_unlock() unlocks a notepad so that jot can write to it.

**Usage**

```
jot_lock(pad = jot_active())
```

```
jot_unlock(pad = jot_active())
```

**Arguments**

pad                    path to file to open

**Value**

path to pad, invisibly

**Examples**

```
tf <- tempfile()
jot_new_pad(tf)
jot_activate(tf)
jot_lock()
jot_unlock()
```

---

|             |                               |
|-------------|-------------------------------|
| jot_new_pad | <i>Create new jot Notepad</i> |
|-------------|-------------------------------|

---

**Description**

Create new jot Notepad

**Usage**

```
jot_new_pad(pad = ".jot.yaml", title)
```

```
jot_new(pad = ".jot.yaml", title)
```

**Arguments**

|       |   |
|-------|---|
| pad   | path to file to store new notepad.                                    |
| title | what to call the notepad. If not supplied, uses the name of the file. |

**Value**

path to file, invisibly

**Examples**

```
tf <- tempfile()
jot_new_pad(tf)
```

---

|          |                                    |
|----------|------------------------------------|
| jot_open | <i>Open and Close jot Notepads</i> |
|----------|------------------------------------|

---

**Description**

- `jot_open()` and `jot_activate()` opens an existing notepad and sets it to active.
- `jot_current()` and `jot_active()` return the path to the current notepad.
- `jot_close()` and `jot_deactivate()` closes the open notepad, if there is one.

**Usage**

```
jot_open(pad)
```

```
jot_activate(pad)
```

```
jot_active()
```

```
jot_current()
```

jot\_close()

jot\_deactivate()

### Arguments

pad                    path to file to open

### Value

path to pad, invisibly

### Examples

```
tf <- tempfile()
jot_new_pad(tf)
jot_open(tf)
jot_active()
jot_current()
jot_close()
jot_active()
```

---

|          |                           |
|----------|---------------------------|
| jot_read | <i>Read a jot notepad</i> |
|----------|---------------------------|

---

### Description

Read a jot notepad

### Usage

```
jot_read(name, pad = jot_active())
```

```
read_jot(name, pad = jot_active())
```

### Arguments

name                    name of noteto read

pad                    path to file to open

### Value

value specified or all values

**Examples**

```
tf <- tempfile()
jot_new_pad(tf)
jot_activate(tf)
jot(note = 'ex_note', name = 'ex_name')
jot_read()
jot_read('ex_name')
```

---

jot\_skim

*Read in all values from a jot notepad*

---

**Description**

Read in all values from a jot notepad

**Usage**

```
jot_skim(pad = jot_active())
```

**Arguments**

pad                    path to file to open

**Value**

value specified or list of all values

**Examples**

```
tf <- tempfile()
jot_new_pad(tf)
jot_activate(tf)
jot('ex_val', 'ex')
jot_skim()
```



# Index

## \* helpers

- get\_jot\_user, 2
- get\_sys\_time, 2

## \* lock

- jot\_lock, 5

## \* pad

- jot\_new\_pad, 6
- jot\_open, 6

## \* read

- jot\_list, 4
- jot\_read, 7
- jot\_skim, 8

## \* write

- jot, 3
- jot\_erase, 4

get\_jot\_user, 2

get\_sys\_time, 2

jot, 3

jot\_activate (jot\_open), 6

jot\_active (jot\_open), 6

jot\_close (jot\_open), 6

jot\_current (jot\_open), 6

jot\_deactivate (jot\_open), 6

jot\_erase, 4

jot\_list, 4

jot\_lock, 5

jot\_new (jot\_new\_pad), 6

jot\_new\_pad, 6

jot\_open, 6

jot\_read, 7

jot\_skim, 8

jot\_unlock (jot\_lock), 5

read\_jot (jot\_read), 7

write\_jot (jot), 3