

# Package ‘whoami’

October 12, 2022

**Title** Username, Full Name, Email Address, 'GitHub' Username of the Current User

**Version** 1.3.0

**Description** Look up the username and full name of the current user, the current user's email address and 'GitHub' username, using various sources of system and configuration information.

**License** MIT + file LICENSE

**URL** <https://github.com/r-lib/whoami#readme>

**BugReports** <https://github.com/r-lib/whoami/issues>

**Imports** htr, jsonlite, utils

**Suggests** covr, mockery, testthat, withr

**Encoding** UTF-8

**RoxygenNote** 6.1.1

**NeedsCompilation** no

**Author** Gábor Csárdi [aut, cre],  
Maëlle Salmon [ctb]

**Maintainer** Gábor Csárdi <csardi.gabor@gmail.com>

**Repository** CRAN

**Date/Publication** 2019-03-19 12:00:03 UTC

## R topics documented:

email_address . . . . .	2
fullname . . . . .	2
gh_username . . . . .	3
username . . . . .	4
whoami . . . . .	5

<b>Index</b>	<b>6</b>
--------------	----------

email\_address            *Email address of the current user*

---

**Description**

If uses the EMAIL environment variable, if set. Otherwise it tries to find it in the user's global git configuration.

**Usage**

```
email_address(fallback = NULL)
```

**Arguments**

fallback            If not NULL then this value is returned if the email address cannot be found, instead of triggering an error.

**Value**

Email address on success. Otherwise an error is thrown.

**See Also**

Other user names: [fullname](#), [gh\\_username](#), [username](#), [whoami](#)

**Examples**

```
## Not run:  
email_address()  
  
## End(Not run)
```

---

fullname            *Full name of the current user*

---

**Description**

Uses the FULLNAME environment variable, if set. Otherwise tries system full names and the git configuration as well.

**Usage**

```
fullname(fallback = NULL)
```

**Arguments**

fallback            If not NULL then this value is returned if the full name of the user cannot be found, instead of triggering an error.

**Value**

The full name of the current user.

**See Also**

Other user names: [email\\_address](#), [gh\\_username](#), [username](#), [whoami](#)

**Examples**

```
## Not run:  
fullname()  
  
## End(Not run)
```

---

gh\_username

*Find the current user's GitHub username*

---

**Description**

Uses the GITHUB\_USERNAME global variable or searches on GitHub, for the user's email address, see [email\\_address](#).

**Usage**

```
gh_username(token = NULL, fallback = NULL)
```

**Arguments**

token            GitHub token to use. By default it uses the GITHUB\_TOKEN environment variable, if set. If unset, uses the GITHUB\_PAT environment variable, if set.

fallback        If not NULL then this value is returned if the GitHub username cannot be found, instead of triggering an error.

**Details**

This function caches the username in the current R session, and if the email address of the user is unchanged, it does not perform another GitHub query.

**Value**

GitHub username, or an error is thrown if it cannot be found.

**See Also**

Other user names: [email\\_address](#), [fullname](#), [username](#), [whoami](#)

**Examples**

```
## Not run:  
gh_username()  
  
## End(Not run)
```

---

username	<i>User name of the current user</i>
----------	--------------------------------------

---

**Description**

Tries the LOGNAME, USER, LNAME, USERNAME environment variables first. Then it tries the `id` command on Unix-like platforms and `whoami` on Windows.

**Usage**

```
username(fallback = NULL)
```

**Arguments**

fallback	If not NULL then this value is returned if the username cannot be found, instead of triggering an error.
----------	--

**Value**

The user name of the current user.

**See Also**

Other user names: [email\\_address](#), [fullname](#), [gh\\_username](#), [whoami](#)

**Examples**

```
## Not run:  
username()  
  
## End(Not run)
```

---

whoami	<i>User name and full name of the current user</i>
--------	--

---

## Description

Calls [username](#) and [fullname](#).

## Usage

```
whoami()
```

## Details

For the username it tries the LOGNAME, USER, LNAME and USERNAME environment variables first. If these are all unset, or set to an empty string, then it tries running `id` on Unix-like systems and `whoami` on Windows.

For the full name of the user, it queries the system services and also tries the user's global git configuration. On Windows, it tries finding the global git configuration in `Sys.getenv("USERPROFILE")` if it doesn't find it in `Sys.getenv("HOME")` (often "Documents").

For the email address it uses the user's global git configuration. It tries finding the global git configuration in `Sys.getenv("USERPROFILE")` if it doesn't find it in `Sys.getenv("HOME")`.

For the GitHub username it uses the `GITHUB_USERNAME` environment variable then it tries searching on GitHub for the user's email address.

## Value

A named character vector with entries: `username`, `fullname`, `email_address`, `gh_username`.

## See Also

Other user names: [email\\_address](#), [fullname](#), [gh\\_username](#), [username](#)

## Examples

```
## Not run:  
whoami()  
  
## End(Not run)
```

# Index

## \* **user names**

email\_address, 2

fullname, 2

gh\_username, 3

username, 4

whoami, 5

email\_address, 2, 3–5

fullname, 2, 2, 4, 5

gh\_username, 2, 3, 3, 4, 5

username, 2–4, 4, 5

whoami, 2–4, 5