

Package ‘envvar’

May 8, 2026

Title Make Working with Environment Variables Easier and More Consistent

Version 0.1.2

Description A collection of functions that allows for easy and consistent use of environment variables. This includes setting, checking, retrieving, transforming, and validating values stored in environment variables.

License MIT + file LICENSE

URL <https://github.com/briandconnelly/envvar>,
<https://briandconnelly.github.io/envvar/>

BugReports <https://github.com/briandconnelly/envvar/issues>

Imports cli, fs, lubridate, rlang

Suggests httr2, ipaddress, spelling, testthat (>= 3.0.0), uuid, withr

Config/testthat/edition 3

Encoding UTF-8

Language en-US

RoxygenNote 7.3.2

NeedsCompilation no

Author Brian Connelly [aut, cre, cph] (ORCID:
<<https://orcid.org/0000-0002-9948-0379>>)

Maintainer Brian Connelly <bdc@bconnelly.net>

Repository CRAN

Date/Publication 2024-08-17 20:20:02 UTC

Contents

envvar_get	2
envvar_get_file	3
envvar_get_integer	5
envvar_get_list	6

envvar_get_url	7
envvar_get_uuid	8
envvar_is_set	9
envvar_list	9
envvar_set	10

Index	11
--------------	-----------

envvar_get	<i>Get the value of an environment variable</i>
------------	---

Description

envvar_get() returns the value of an environment variable.

envvar_get_oneof() gets the value of an environment variable and ensures that it is within a defined set of potential values.

Usage

```
envvar_get(
  x,
  default = NULL,
  transform = NULL,
  validate = NULL,
  warn_default = TRUE
)
```

```
envvar_get_oneof(
  x,
  choices,
  default = NULL,
  transform = NULL,
  validate = NULL,
  warn_default = TRUE
)
```

Arguments

x	String containing an environment variable name
default	Optional default value if the environment variable is not set
transform	Optional function that applies a transformation to the variable's value
validate	Optional function that checks a value for validity
warn_default	Show a warning if the default value is used (default: TRUE)
choices	A list containing the potential values

Details

envvar_get() and the other type-specific variants follow this workflow:

- The value of the environment variable is retrieved. If that variable is not set and default non-NULL, the default value is used. Otherwise an error is raised.
- Optionally, the value can be transformed by a function specified by the transform argument. Transformation functions return a scalar object.
- Optionally, the value can be validated by a function specified by the validate argument. Validation functions return a logical value indicating whether or not the value matches the given criteria. An error is raised if the validation function does not return TRUE.
- The transformed, validated value is returned.

Value

The value of the given environment variable, if set. This is a string unless a transform function has changed the object's type.

Examples

```
# Get the current user's home directory
envvar_get("HOME")

# Get the current logging level for your app, ensuring it is a known value
envvar_set("LOG_LEVEL" = "DEBUG")
envvar_get_oneof(
  "LOG_LEVEL",
  choices = c("TRACE", "DEBUG", "INFO", "SUCCESS", "WARN", "ERROR", "FATAL")
)
```

envvar_get_file	<i>Environment variables for files and directories</i>
-----------------	--

Description

envvar_get_file() gets a file name from an environment variable

envvar_get_dir() gets a directory name from an environment variable

Usage

```
envvar_get_file(
  x,
  default = NULL,
  create = TRUE,
  check_readable = FALSE,
  check_writable = FALSE,
  transform = NULL,
  warn_default = TRUE,
```

```

    ...
)

envvar_get_dir(
  x,
  default = NULL,
  create = TRUE,
  transform = NULL,
  check_readable = FALSE,
  check_writable = FALSE,
  warn_default = TRUE,
  ...
)

```

Arguments

<code>x</code>	String containing an environment variable name
<code>default</code>	Optional default value if the environment variable is not set
<code>create</code>	Create the file or directory if it does not exist (default: TRUE)
<code>check_readable</code>	Ensure that the file or directory is readable
<code>check_writable</code>	Ensure that the file or directory is writable
<code>transform</code>	Optional function that applies a transformation to the variable's value
<code>warn_default</code>	Show a warning if the default value is used (default: TRUE)
<code>...</code>	Additional arguments passed to fs::file_create for <code>envvar_get_file()</code> or fs::dir_create for <code>envvar_get_dir()</code>

Value

A string containing the path to a file or directory

Examples

```

# Get a file path and make sure it exists
envvar_set("MY_DATA" = "data.parquet")
envvar_get_file("MY_DATA")

# Cleanup
file.remove("data.parquet")

envvar_set("MY_DATA_DIR" = "data")
envvar_get_dir("MY_DATA_DIR")

# Cleanup
unlink("data", recursive = TRUE)

```

envvar_get_integer *Get the value of an environment variable as a particular type*

Description

envvar_get_integer() reads environment variables with integer values (e.g., 15, 0, -1)
 envvar_get_numeric() reads environment variables with numeric values (e.g., 100.12, -31)
 envvar_get_logical() reads environment variables with logical values (e.g., TRUE, 1, "T")
 envvar_get_version reads environment variables with semantic version numbers (e.g., 4.3.1)
 envvar_get_date() reads environment variables with date values (e.g., "2023-01-02")
 envvar_get_datetime() reads environment variables with date-time values (e.g., "2023-01-02 01:23:45 UTC" or 1697037804)

Usage

```
envvar_get_integer(x, default = NULL, validate = NULL, warn_default = TRUE)
envvar_get_numeric(x, default = NULL, validate = NULL, warn_default = TRUE)
envvar_get_logical(x, default = NULL, validate = NULL, warn_default = TRUE)
envvar_get_version(x, default = NULL, validate = NULL, warn_default = TRUE)
envvar_get_date(x, default = NULL, validate = NULL, warn_default = TRUE, ...)

envvar_get_datetime(
  x,
  default = NULL,
  validate = NULL,
  warn_default = TRUE,
  ...
)
```

Arguments

x	String containing an environment variable name
default	Optional default value if the environment variable is not set
validate	Optional function that checks a value for validity
warn_default	Show a warning if the default value is used (default: TRUE)
...	Additional arguments passed to lubridate::as_date for envvar_get_date() or lubridate::as_datetime for envvar_get_datetime()

Value

A scalar object

Examples

```

# Get and use an integer value
envvar_set("MYNUMBER" = 12)
envvar_get_integer("MYNUMBER") + 5

# Get and use a numeric value
envvar_set("MYNUMBER" = 12.34)
envvar_get_numeric("MYNUMBER") + 5

# Check a logical value
isTRUE(envvar_get_logical("RSTUDIO_CLI_HYPERLINKS", default = FALSE))

# Get an IP address value and ensure that it is IPv4
envvar_set("MY_VER" = "4.3.1")
envvar_get_version("MY_VER", validate = \(x) x > as.numeric_version("4.3"))
# Get a date and make sure it's in the past
envvar_set("LAUNCH_DATE" = "2023-03-03")
envvar_get_date("LAUNCH_DATE", validate = \(x) x < Sys.Date())

```

envvar_get_list

Environment variables containing lists

Description

envvar_get_list() gets lists from environment variables. At the moment, only unnamed lists are supported.

envvar_get_csv() and envvar_get_psv() are an easy way to use envvar_get_list() with comma or pipe separators.

Usage

```

envvar_get_list(
  x,
  pattern = ":",
  default = NULL,
  validate = NULL,
  warn_default = TRUE,
  ...
)

```

```
envvar_get_csv(x, default = NA, validate = NULL, warn_default = TRUE)
```

```
envvar_get_psv(x, default = NA, validate = NULL, warn_default = TRUE)
```

Arguments

x String containing an environment variable name

pattern	String specifying the pattern used to separate elements in the list.
default	Optional default value if the environment variable is not set
validate	Optional function that checks a value for validity
warn_default	Show a warning if the default value is used (default: TRUE)
...	Additional arguments passed to strsplit

Value

A list

Examples

```
# Get the value of `PATH`, creating a list with elements for each directory
envvar_get_list("PATH")

# Parse an list separated by `|`
envvar_set("ROOMMATES" = "nandor|laszlo|nadja|guillermo|colin")
envvar_get_psv("ROOMMATES")
```

envvar_get_url *Environment variables for internet and network-related values*

Description

envvar_get_url() gets a URL value from an environment variable and parses it with [httr2::url_parse](#).
 envvar_get_ipaddress() gets an IP address value from an environment variable

Usage

```
envvar_get_url(x, default = NULL, validate = NULL, warn_default = TRUE)

envvar_get_ipaddress(x, default = NULL, validate = NULL, warn_default = TRUE)
```

Arguments

x	String containing an environment variable name
default	Optional default value if the environment variable is not set
validate	Optional function that checks a value for validity
warn_default	Show a warning if the default value is used (default: TRUE)

Value

envvar_get_url() returns a URL: an S3 list with class `httr2_url` and elements `scheme`, `hostname`, `port`, `path`, `fragment`, `query`, `username`, `password`, where applicable.
 envvar_get_ipaddress() returns an `ip_address` vector

Examples

```
# Get a URL value and ensure that it is https
envvar_set("TEST_URL" = "https://google.com:80/?a=1&b=2")
envvar_get_url("TEST_URL", validate = \(x) x$scheme == "https")

# Get an IP address value and ensure that it is IPv4
envvar_set("TEST_HOST" = "192.168.1.15")
envvar_get_ipaddress("TEST_HOST", validate = ipaddress::is_ipv4)
```

envvar_get_uuid *Environment variables for UUIDs*

Description

envvar_get_uuid() gets a UUID from an environment variable

Usage

```
envvar_get_uuid(x, default = NULL, validate = NULL, warn_default = TRUE)
```

Arguments

x	String containing an environment variable name
default	Optional default value if the environment variable is not set
validate	Optional function that checks a value for validity
warn_default	Show a warning if the default value is used (default: TRUE)

Value

An object of the class "UUID" representing a vector of UUIDs.

Examples

```
# Get a file path and make sure it exists
envvar_set("DEMO_GUID" = "d647f20f-c44c-4914-8255-9eca97150d4c")
envvar_get_uuid("DEMO_GUID")
```

envvar_is_set	<i>Check whether an environment variable is set</i>
---------------	---

Description

envvar_is_set() checks whether a given environment variable is set.

Usage

```
envvar_is_set(x)
```

Arguments

x String with the name of environment variable

Value

A logical value

Examples

```
envvar_is_set("HOME")
```

envvar_list	<i>Get a list of environment variables</i>
-------------	--

Description

envvar_list() returns a named list of the defined environment variables

Usage

```
envvar_list()
```

Value

A named list

Examples

```
envvar_list()
```

`envvar_set`*Set, unset, and check environment variables*

Description

`envvar_set()` sets one or more environment variables.

`envvar_unset()` unsets an environment variable.

Usage

```
envvar_set(...)
```

```
envvar_unset(x)
```

Arguments

... Named arguments containing an environment variable and its value

x String with the name of environment variable

Value

No return value, called for side effects

Examples

```
envvar_set(DEBUG = 1)
envvar_unset("DEBUG")
```

Index

envvar_get, 2
envvar_get_csv (envvar_get_list), 6
envvar_get_date (envvar_get_integer), 5
envvar_get_datetime
 (envvar_get_integer), 5
envvar_get_dir (envvar_get_file), 3
envvar_get_file, 3
envvar_get_integer, 5
envvar_get_ipaddress (envvar_get_url), 7
envvar_get_list, 6
envvar_get_logical
 (envvar_get_integer), 5
envvar_get_numeric
 (envvar_get_integer), 5
envvar_get_oneof (envvar_get), 2
envvar_get_psv (envvar_get_list), 6
envvar_get_url, 7
envvar_get_uuid, 8
envvar_get_version
 (envvar_get_integer), 5
envvar_is_set, 9
envvar_list, 9
envvar_set, 10
envvar_unset (envvar_set), 10

fs::dir_create, 4
fs::file_create, 4

httr2::url_parse, 7

lubridate::as_date, 5
lubridate::as_datetime, 5

strsplit, 7