

Package ‘dotenv’

August 27, 2014

Title Load environment variables from .env

Version 1.0

Description Load configuration from a .env file, that is
in the current working directory, into environment variables.

License MIT + file LICENSE

LazyData true

Imports magrittr, falsy

URL <https://github.com/gaborcsardi/dotenv>

BugReports <https://github.com/gaborcsardi/dotenv/issues>

Author “Gabor Csardi” [aut, cre]

Maintainer “Gabor Csardi” <csardi.gabor@gmail.com>

NeedsCompilation no

Repository CRAN

Date/Publication 2014-08-27 23:42:52

R topics documented:

dotenv-package	2
load_dot_env	2

Index	4
--------------	----------

dotenv-package	<i>Load configuration parameters from .env into environment variables</i>
----------------	---

Description

It has become a practice to store configuration parameters related to a project, in a hidden file called `.env`, in the working directory of a project, and then set them as environment variables.

Details

This package loads the variables defined in the `.env` file in the current working directory (as reported by `getwd`), and sets them as environment variables.

This happens automatically when the `dotenv` package is loaded, so the typical use-case is to just put a `library(dotenv)` code at the beginning of your R script.

Alternatively a `dotenv::load_dot_env()` call can be used to load variables from arbitrary files.

The format of the `.env` file is also a valid unix shell file format, so e.g. in bash the environment variables can also be set by running `source .env`.

See [load_dot_env](#) for the exact file format.

See Also

`load_dot_env`

load_dot_env	<i>Load environment variables from the specified file</i>
--------------	---

Description

Load variables defined in the given file, as environment variables.

Usage

```
load_dot_env(file = ".env")
```

Arguments

`file` The name of the file to use.

Details

The file is parsed line by line, and line is expected to have one of the following formats:

```
VARIABLE=value  
VARIABLE2="quoted value"  
VARIABLE3='another quoted variable'  
# Comment line  
export EXPORTED="exported variable"  
export EXPORTED2=another
```

In more details:

- A leading `export` is ignored, to keep the file compatible with Unix shells.
- No whitespace is allowed right before or after the equal sign, again, to promote compatibility with Unix shells.
- No multi-line variables are supported currently. The file is strictly parsed line by line.
- Unlike for Unix shells, unquoted values are *not* terminated by whitespace.
- Comments start with `#`, without any leading whitespace. You cannot mix variable definitions and comments in the same line.
- Empty lines (containing whitespace only) are ignored.

It is suggested to keep the file in a form that is parsed the same way with `dotenv` and `bash` (or other shells).

Examples

```
# Remove, if it exists  
Sys.unsetenv("dotenvexamplefoo")  
Sys.getenv("dotenvexamplefoo")  
  
# Load from a file  
tmp <- tempfile()  
cat("dotenvexamplefoo=bar\n", file = tmp)  
load_dot_env(tmp)  
Sys.getenv("dotenvexamplefoo")  
  
# Clean up  
unlink(tmp)
```

Index

`dotenv-package`, [2](#)

`load_dot_env`, [2](#), [2](#)