Package ‘littler’

December 17, 2023

Type Package

Title R at the Command-Line via ‘r’

Version 0.3.19

Date 2023-12-17

Author Dirk Eddelbuettel and Jeff Horner (2006-2008)

Maintainer Dirk Eddelbuettel <edd@debian.org>

Description A scripting and command-line front-end is provided by ‘r’ (aka 'littler') as a lightweight binary wrapper around the GNU R language and environment for statistical computing and graphics. While R can be used in batch mode, the r binary adds full support for both 'shebang'-style scripting (i.e. using a hash-mark-exclamation-path expression as the first line in scripts) as well as command-line use in standard Unix pipelines. In other words, r provides the R language without the environment.

URL https://github.com/eddelbuettel/littler,
    https://dirk.eddelbuettel.com/code/littler.html,
    https://eddelbuettel.github.io/littler/

BugReports https://github.com/eddelbuettel/littler/issues

License GPL (>= 2)

OS_type unix

SystemRequirements libR

Suggests simplermarkdown, docopt, rcmdcheck, foghorn

VignetteBuilder simplermarkdown

RoxygenNote 5.0.1

NeedsCompilation yes

Repository CRAN

Date/Publication 2023-12-17 22:30:10 UTC
Description

The \textit{r} binary provides a convenient and powerful front-end. By embedding R, it permits four distinct ways to leverage the power of R at the shell prompt: scripting, filename execution, piping and direct expression evaluation.

Details

The \textit{r} front-end was written with four distinct usage modes in mind.

First, it allows to write so-called 'shebang' scripts starting with \texttt{#!/usr/bin/env r}. These 'shebang' scripts are perfectly suited for automation and execution via e.g. via \texttt{cron}.

Second, we can use \texttt{r somefile.R} to quickly execute the name R source file. This is useful as \texttt{r} is both easy to type—and quicker to start than either R itself, or its scripting tool \texttt{Rscript}, while still loading the \texttt{methods} package.

Third, \texttt{r} can be used in 'pipes' which are very common in Unix. A simple and trivial example is \texttt{echo 'cat(2+2)' | r} illustrating that the standard output of one program can be used as the standard input of another program.

Fourth, \texttt{r} can be used as a calculator by supplying expressions after the \texttt{-e} or \texttt{--eval} options.

Value

Common with other shell tools and programs, \texttt{r} returns its exit code where a value of zero indicates success.

Note

On OS X one may have to link the binary to, say, \texttt{lr} instead. As OS X insists that files named \texttt{R} and \texttt{r} are the same, we cannot use the latter.

Author(s)

Jeff Horner and Dirk Eddelbuettel wrote \texttt{littler} from 2006 to today, with contributions from several others.

Dirk Eddelbuettel \texttt{<edd@debian.org>} is the maintainer.
Examples

```r
## Not run:
#!/usr/bin/env r       ## for use in scripts
other input | r        ## for use in pipes
r somefile.R          ## for running files
r -e 'expr'           ## for evaluating expressions
r --help              ## to show a quick synopsis

## End(Not run)
```

---

**r**  
*Return Path to r Binary*

---

**Description**

Return the path of the install r binary.

**Usage**

```r
r(usecat = FALSE)
```

**Arguments**

- `usecat`  
  Optional toggle to request output to stdout (useful in Makefiles)

**Details**

The test for Windows is of course superfluous as we have no binary for Windows. Maybe one day...

**Value**

The path is returned as character variable. If the usecat option is set the character variable is displayed via `cat` instead.

**Author(s)**

Dirk Eddelbuettel
Index

* package
  * littler, 2

cat, 3
littler, 2
r, 3