

Package ‘rswipl’

January 15, 2023

Type Package

Title Embed 'SWI'-Prolog'

Version 9.1.3

Date 2023-01-14

Maintainer Matthias Gondan <Matthias.Gondan-Rochon@uibk.ac.at>

Description Interface to 'SWI'-Prolog', <<https://www.swi-prolog.org/>>. This package is normally not loaded directly, please refer to package 'rolog' instead. The purpose of this package is to provide the 'Prolog' runtime on systems that do not have a software installation of 'SWI'-Prolog'.

License FreeBSD

Imports Rcpp (>= 1.0.7)

Depends R (>= 4.2)

URL <https://github.com/mgondan/rswipl>

BugReports <https://github.com/mgondan/rswipl/issues>

LinkingTo Rcpp

RoxygenNote 7.2.1

Encoding UTF-8

SystemRequirements GNU make, CMake (>= 3.10), pandoc, libarchive, libregex, libexpat, liblzma, libzstd, liblz4, libz2, libz, libbcbcrypt

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

NeedsCompilation yes

Author Matthias Gondan [aut, com, cre] (Universität Innsbruck), Jan Wielemaker [ctb, cph] (JW is the lead developer of SWI-Prolog. See <https://www.swi-prolog.org/Contributors.html> for a list of contributors.)

Repository CRAN

Date/Publication 2023-01-15 18:10:02 UTC

R topics documented:

clear	2
query	2
submit	4

Index	5
--------------	----------

clear	<i>Clear current query</i>
-------	----------------------------

Description

Clear current query

Usage

```
clear()
```

Value

TRUE (invisible)

See Also

[query\(\)](#) for opening a query

[submit\(\)](#) for submitting a query

Examples

```
query(call("member", expression(X), list(quote(a), "b", 3L, 4)))
submit() # X = a
submit() # X = "b"
clear()
```

query	<i>Create a query</i>
-------	-----------------------

Description

Create a query

Usage

```
query(
  query = call("member", expression(X), list(quote(a), "b", 3L, 4, TRUE, expression(Y)))
)
```

Arguments

query an R call. The R call consists of symbols, integers and real numbers, character strings, boolean values, expressions, lists, and other calls. Vectors of booleans, integers, floating point numbers, and strings with length $N > 1$ are translated to prolog compounds `!/N`, `%/N`, `#/N` and `$$/N`, respectively.

Details

SWI-Prolog does not allow multiple open queries. If another query is open, it is closed and a warning is shown.

Value

If the creation of the query succeeds, TRUE

See Also

[submit\(\)](#) for submitting a query

[clear\(\)](#) to close the currently open query

Examples

```
query(call("member", expression(X), list(quote(a), "b", 3L, 4, TRUE, expression(Y))))
submit() # X = a
submit() # X = "b"
clear()
```

```
query(call("member", expression(X), list(quote(a), "b", 3L, 4, TRUE, expression(Y),
  NA, NaN, Inf, NULL, function(x) {y <- sin(x); y^2})))
submit() # X = a
submit() # X = "b"
submit() # X = 3L
submit() # X = 4.0
submit() # X = TRUE
submit() # X = expression(Y) or Y = expression(X)
submit() # X = NA
submit() # X = NaN
submit() # X = Inf
submit() # X = NULL
submit() # X = function(x) {y <- sin(x); y^2})
submit() # FALSE (no more results)
submit() # warning that no query is open
```

```
query(call("member", expression(X), list(quote(a), "b", 3L, 4)))
query(call("member", expression(X), list(TRUE, expression(Y)))) # warning that another query is open
clear()
```

`submit`*Submit a query that has been opened with `query()` before.*

Description

Submit a query that has been opened with `query()` before.

Usage

```
submit()
```

Value

If the query fails, FALSE is returned. If the query succeeds, a (possibly empty) list is returned that includes the bindings required to satisfy the query.

See Also

`query()` for a opening a query.

`clear()` for a clearing a query.

Examples

```
query(call("member", expression(X), list(quote(a), "b", 3L, 4, expression(Y))))
submit() # X = 3L
submit() # X = 4.0
submit() # X = TRUE
submit() # X = expression(Y) or Y = expression(X)
submit() # FALSE
submit() # warning that no query is open
```

```
query(call("member", expression(X), list(quote(a), "b", 3L, 4)))
submit() # X = a
submit() # X = "b"
clear()
```

Index

clear, [2](#)
clear(), [3](#), [4](#)

query, [2](#)
query(), [2](#), [4](#)

submit, [4](#)
submit(), [2](#), [3](#)