Package: svHttp (via r-universe)

October 13, 2026

Type Package

Version 1.0.5

Title 'SciViews::R' - HTTP Server

Description A simple HTTP server allows to connect GUI clients to R.

Maintainer Philippe Grosjean <phgrosjean@sciviews.org>

Depends R (>= 2.11.0)

Imports svMisc (>= 0.9-68), tools (>= 2.11.0), utils (>= 2.11.0)

Suggests curl (>= 6.2.2), svSocket (>= 1.1.5), covr (>= 3.5.0), knitr (>= 1.42), rmarkdown (>= 2.21), spelling (>= 2.2.1), testthat (>= 3.0.0)

License GPL-2

URL https://github.com/SciViews/svHttp, https://www.sciviews.org/svHttp/, https://sciviews.r-universe.dev/svHttp

BugReports https://github.com/SciViews/svHttp/issues

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

VignetteBuilder knitr

Encoding UTF-8

Language en-US

ByteCompile yes

Config/testthat/edition 3

Repository https://sciviews.r-universe.dev

Date/Publication 2025-08-29 06:20:17 UTC

RemoteUrl https://github.com/SciViews/svHttp

RemoteRef HEAD

RemoteSha a82eb901d3593bca9a2fc7dfd1f055180714db50

Contents

svHttp-package 2
http_server_clients 3
http_server_name 3
http_server_port 4
par_http_server 5
start_http_server 5
stop_http_server 6

Index **8**

`svHttp-package` *'SciViews::R' - HTTP Server*

Description

A SciViews R server using HTTP R help server and JSONP for communication.

Details

See `start_http_server()` for further implementation details.

Author(s)

Maintainer: Philippe Grosjean <phgrosjean@sciviews.org> (ORCID)

See Also

Useful links:

- https://github.com/SciViews/svHttp
- https://www.sciviews.org/svHttp/
- https://sciviews.r-universe.dev/svHttp
- Report bugs at https://github.com/SciViews/svHttp/issues

http_server_clients *Get list of all names of clients that already connected to the http server*

Description

Get list of all names of clients that already connected to the http server

Usage

```
http_server_clients()

HttpClientsNames()
```

Value

A character vector with the name of currently connected clients.

See Also

`start_http_server()` for a complete example.

Examples

```
library(svHttp)
http_server_clients()
```

http_server_name *Get or change the name of the HTTP server*

Description

Get or change the name of the HTTP server

Usage

```
http_server_name(name)

HttpServerName(name)
```

Arguments

`name`	the name given to the SciViews server. By default, it is `R`.

Value

A character vector with the name of the HTTP server.

See Also

start_http_server() for a complete example.

Examples

```
http_server_name()
```

`http_server_port` *Get or change the port of the HTTP server*

Description

Get or change the port of the HTTP server

Usage

```
http_server_port(port)

HttpServerPort(port)
```

Arguments

port	port on which the server should run (both help and SciViews). By default, it is port 8888. Note that this server runs only locally and can only serve requests from 127.0.0.1 (because communication is not crypted).

Value

A number with the port of the HTTP server.

See Also

start_http_server() for a complete example.

Examples

```
http_server_port()
```

par_http_server *Get or change http server options*

Description

Get or change http server options

Usage

```
par_http_server(client, ...)

parHttp(client, ...)
```

Arguments

`client`	the name of one client. A client that does not identify itself is named `default`.
`...`	named arguments specifying options to set or change.

Value

An environment that contains the whole configuration is returned invisibly.

Note

Possible named arguments (with their default values) are: `prompt = ":> "` for the server prompt, `continue = ":+ "` for the continuation prompt when multiline instructions are send, `code = ""` for current partial code in multiline mode, `last = ""` for a string to add at the end of each evaluation, `echo = FALSE` to echo commands at the R console or terminal, `multiline = TRUE` to allow multiline mode, `bare = TRUE` a bare mode that inactivates all other options (the server is always started in bare mode).

See Also

`start_http_server()` for a complete example.

start_http_server *(Re)start an HTTP server in R*

Description

Turn the default R help HTTP server into a RJSONp SciViews server (while still serving help pages, of course).

Usage

```
start_http_server(port = http_server_port(), name = http_server_name())

startHttpServer(port = http_server_port(), name = http_server_name())
```

Arguments

port	port on which the server should run (both help and SciViews). By default, it is port 8888. Note that this server runs only locally and can only serve requests from 127.0.0.1 (because communication is not crypted).
name	the name given to the SciViews server. By default, it is R.

Value

An integer indicating the port used.

See Also

svSocket::start_socket_server()

Examples

```
## Not run:
library(svHttp)
# Try to start the HTTP server on default port with default name
res <- try(start_http_server(), silent = TRUE)
if (!inherits(res, "try-error")) {
  # Get the port
  http_server_port()

  # Get the name
  http_server_name()

  # Get the list of clients... empty, unless you connect a client in between
  http_server_clients()

}
# Stop the server now
 stop_http_server()

## End(Not run)
```

---

stop_http_server *Stop the SciViews and R HTTP server and eliminate all tracks*

Description

Stop the SciViews and R HTTP server and eliminate all tracks

Usage

```
stop_http_server(remove.clients = FALSE)

stopHttpSever(remove.clients = FALSE)
```

Arguments

`remove.clients`	do we remove also persistent data for the clients, `FALSE` by default.

Value

A number with the port of the HTTP server.

See Also

`start_http_server()` for a complete example.

Index

∗ **IO**
    http_server_clients, 3
    http_server_name, 3
    http_server_port, 4
    par_http_server, 5
    start_http_server, 5
    stop_http_server, 6
∗ **Interprocess communication**
    http_server_clients, 3
    http_server_name, 3
    http_server_port, 4
    par_http_server, 5
    start_http_server, 5
    stop_http_server, 6

http_server_clients, 3
http_server_name, 3
http_server_port, 4
HttpClientsNames *(*http_server_clients*)*, 3
HttpServerName *(*http_server_name*)*, 3
HttpServerPort *(*http_server_port*)*, 4

par_http_server, 5
parHttp *(*par_http_server*)*, 5

start_http_server, 5
start_http_server(), *2–5*, *7*
startHttpServer *(*start_http_server*)*, 5
stop_http_server, 6
stopHttpSever *(*stop_http_server*)*, 6
svHttp *(*svHttp-package*)*, 2
svHttp-package, 2
svSocket::start_socket_server(), *6*