

# Package: languageserversetup (via r-universe)

June 7, 2024

**Type** Package

**Title** Automated Setup and Auto Run for R Language Server

**Version** 0.1.2.900

**Maintainer** Jozef Hajnala <jozef.hajnala@gmail.com>

**Description** Allows to install the R 'languageserver' with all dependencies into a separate library and use that independent installation automatically when R is instantiated as a language server process. Useful for making language server seamless to use without running into package version conflicts.

**License** AGPL-3

**Suggests** tinytest

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.1.1

**SystemRequirements** procps (linux), ps (macos), wmic (windows)

**Roxygen** list(markdown = TRUE)

**URL** <https://github.com/jozefhajnala/languageserversetup>

**BugReports** <https://github.com/jozefhajnala/languageserversetup/issues>

**Language** en-US

**Repository** <https://jozefhajnala.r-universe.dev>

**RemoteUrl** <https://github.com/jozefhajnala/languageserversetup>

**RemoteRef** HEAD

**RemoteSha** 3daf44a77cdc6ca38d183db7e7cf489a7e4d15b6

## Contents

|   |   |
|---|---|
| languageserver_add_to_rprofile . . . . .      | 2 |
| languageserver_detect . . . . .               | 3 |
| languageserver_install . . . . .              | 3 |
| languageserver_remove_from_rprofile . . . . . | 4 |
| languageserver_startup . . . . .              | 5 |

---

languageserver\_add\_to\_rprofile  
*Add language server initialization to .Rprofile*

---

### Description

Add language server initialization to .Rprofile

### Usage

```
languageserver_add_to_rprofile(
  rlsLib = getOption("langserver_library"),
  rprofilePath = locate_rprofile(),
  confirmBeforeChanging = TRUE,
  code = append_code(rlsLib = rlsLib)
)
```

### Arguments

|                       |   |
|-----------------------|---|
| rlsLib                | character(1), path to the library where the languageserver package will be installed.   |
| rprofilePath          | character(1), path to the file where to add the initialization code, or NULL. By default, adds the code to a .Rprofile file in the home directory of the current user. Please refer to ?Startup for more details around .Rprofile files.<br><br>Notably, if the R_PROFILE_USER environment variable is set, the .Rprofile located in the home directory is ignored, therefore we may want to place the initialization code into the file specified by that variable using the rprofilePath argument in that case. |
| confirmBeforeChanging | logical(1), if TRUE, asks for user confirmation before changing the file. For non-interactive use, FALSE will skip the confirmation.  |
| code                  | character(), the code to be added to the file. Defaults to the value of append_code().  |

### Value

side-effects

---

languageserver\_detect *Detect whether a process relates to the R Language Server*

---

**Description**

Detect whether a process relates to the R Language Server

**Usage**

```
languageserver_detect(pid, os, langServerProcessPatt, checkParents = TRUE)
```

**Arguments**

|                       |  |
|-----------------------|--|
| pid                   | integer(1), id of the process to investigate, usually retrieved by <code>Sys.getpid</code>   |
| os                    | character(1), name of the OS, usually retrieved as the "sysname" element of <code>Sys.info</code> , all lowercase.   |
| langServerProcessPatt | character(1), pattern to recognize the process created by languageserver.  |
| checkParents          | logical(1), if TRUE, parent processes are also checked in case when pid is not the R Language Server process. This is needed as the linting processes are created with callr as sub-processes of the main Language Server process. |

**Value**

logical(1), TRUE if the process with pid (or, optionally, any of its parents) is detected as the R Language Server process. Otherwise FALSE.

---

languageserver\_install

*Install the languageserver package to a separate library*

---

**Description**

Install the languageserver package to a separate library

**Usage**

```
languageserver_install(  
  rlsLib = getOption("langserver_library"),  
  strictLibrary = TRUE,  
  fullReinstall = TRUE,  
  fromGitHub = TRUE,  
  confirmBeforeInstall = TRUE,  
  dryRun = FALSE,  
  ref = "master",
```

```

    Ncpus = max(1L, getOption("Ncpus")),
    ...
)

```

### Arguments

|                                   |  |
|-----------------------------------|--|
| <code>rlsLib</code>               | character(1), path to the library where the languageserver package will be installed.  |
| <code>strictLibrary</code>        | logical(1), if TRUE, all the dependencies of languageserver will be installed into <code>rlsLib</code> , otherwise only those that are needed but not present in other libraries in <code>.libPaths()</code> will be installed.  |
| <code>fullReinstall</code>        | logical(1). If TRUE, <code>rlsLib</code> will be recursively removed to re-install all the packages cleanly.   |
| <code>fromGitHub</code>           | logical(1), if TRUE, will use the remotes package to install the current development version from GitHub. Otherwise <code>install.packages()</code> is used to install the languageserver package from CRAN.   |
| <code>confirmBeforeInstall</code> | logical(1) if TRUE, will ask the user to confirm the steps before installation. For non-interactive use, FALSE will skip the confirmation.   |
| <code>dryRun</code>               | logical(1), if TRUE, most actions will only be reported, not taken - nothing will be removed, created or installed.  |
| <code>ref</code>                  | character(1), when <code>fromGitHub</code> is TRUE, <code>ref</code> is the desired git reference. Usually a commit, tag, or branch name.  |
| <code>Ncpus</code>                | integer(1), number of the number of parallel processes to use for a parallel install of more than one source package. Values greater than one are supported if the make command specified by <code>Sys.getenv("MAKE", "make")</code> accepts argument <code>-k -j Ncpus</code> . |
| <code>...</code>                  | further arguments passed to <code>install.packages()</code> in case <code>fromGitHub</code> is set to FALSE.   |

### Value

side-effects

### See Also

[utils::install.packages\(\)](#)

---

languageserver\_remove\_from\_rprofile

*Remove language server initialization from .Rprofile*

---

### Description

Remove language server initialization from `.Rprofile`

**Usage**

```
languageserver_remove_from_rprofile(
  rlsLib = getOption("langserver_library"),
  rprofilePath = locate_rprofile(),
  code = append_code(rlsLib = rlsLib),
  confirmBeforeChanging = TRUE
)
```

**Arguments**

|                                    |  |
|------------------------------------|--|
| <code>rlsLib</code>                | character(1), path to the library where the languageserver package will be installed.  |
| <code>rprofilePath</code>          | character(1), path to the file where to add the initialization code, or NULL. By default, adds the code to a .Rprofile file in the home directory of the current user. Please refer to ?Startup for more details around .Rprofile files.<br>Notably, if the R_PROFILE_USER environment variable is set, the .Rprofile located in the home directory is ignored, therefore we may want to place the initialization code into the file specified by that variable using the <code>rprofilePath</code> argument in that case. |
| <code>code</code>                  | character(), the code to be added to the file. Defaults to the value of <code>append_code()</code> .   |
| <code>confirmBeforeChanging</code> | logical(1), if TRUE, asks for user confirmation before changing the file. For non-interactive use, FALSE will skip the confirmation.   |

**Value**

side-effects

---

languageserver\_startup

*Prepare language server to be started*

---

**Description**

Prepare language server to be started

**Usage**

```
languageserver_startup(
  rlsLib = getOption("langserver_library"),
  langServerProcessPatt = getOption("langserver_processPatt"),
  strictLibrary = TRUE,
  os = tolower(Sys.info()[["sysname"]]),
  pid = Sys.getpid()
)
```

**Arguments**

|                                    |   |
|------------------------------------|---|
| <code>rlsLib</code>                | character(1), path to the library where the languageserver package will be installed.   |
| <code>langServerProcessPatt</code> | character(1), pattern to recognize the process created by languageserver.   |
| <code>strictLibrary</code>         | logical(1), if TRUE, all the dependencies of languageserver will be installed into <code>rlsLib</code> , otherwise only those that are needed but not present in other libraries in <code>.libPaths()</code> will be installed. |
| <code>os</code>                    | character(1), name of the OS, usually retrieved as the "sysname" element of <code>Sys.info</code> , all lowercase.  |
| <code>pid</code>                   | integer(1), id of the process to investigate, usually retrieved by <code>Sys.getpid</code>  |

**Value**

side-effects

# Index

languageserver\_add\_to\_rprofile, 2  
languageserver\_detect, 3  
languageserver\_install, 3  
languageserver\_remove\_from\_rprofile, 4  
languageserver\_startup, 5  
  
utils::install.packages(), 4