

# Package ‘languageserver’

September 27, 2019

**Type** Package

**Title** Language Server Protocol

**Version** 0.3.1

**Date** 2019-9-27

**Author** Randy Lai

**Maintainer** Randy Lai <randy.cs.lai@gmail.com>

**Description** An implementation of the Language Server Protocol for R. The Language Server protocol is used by an editor client to integrate features like auto completion. See <<https://microsoft.github.io/language-server-protocol>> for details.

**URL** <https://github.com/REditorSupport/languageserver>

**License** MIT + file LICENSE

**Depends** R (>= 3.4.0)

**Imports** callr (>= 3.0.0), collections, desc, jsonlite, lintr (>= 1.0.3), R6, repr, readr, stringr, styler (>= 1.0.2), tools, utils

**Suggests** processx, knitr, magrittr, purrr, fs, withr, testthat

**Encoding** UTF-8

**LazyData** true

**NeedsCompilation** yes

**ByteCompile** yes

**RoxygenNote** 6.1.1

**Repository** CRAN

**Date/Publication** 2019-09-27 19:10:02 UTC

## R topics documented:

languageserver-package . . . . .	3
arg_completion . . . . .	3
check_scope . . . . .	4

code_action_params . . . . .	4
code_lens_params . . . . .	5
completion_params . . . . .	5
completion_reply . . . . .	6
content_backward_search . . . . .	6
DefinitionCache . . . . .	7
diagnose_file . . . . .	7
diagnostics . . . . .	8
did_change_configuration_params . . . . .	8
did_change_text_document_params . . . . .	9
did_close_text_document_params . . . . .	9
did_open_text_document_params . . . . .	10
did_save_text_document_params . . . . .	10
document_formatting_params . . . . .	11
document_link_params . . . . .	11
document_on_type_formatting_params . . . . .	12
document_range_formatting_params . . . . .	12
document_symbol_params . . . . .	13
document_uri . . . . .	13
formatting_reply . . . . .	13
hover_reply . . . . .	14
is_rmarkdown . . . . .	14
LanguageServer . . . . .	15
location . . . . .	15
Message . . . . .	15
Namespace . . . . .	16
Notification . . . . .	16
on_initialize . . . . .	17
package_completion . . . . .	17
parse_document . . . . .	18
path_from_uri . . . . .	18
position . . . . .	19
range . . . . .	19
range_formatting_reply . . . . .	20
reference_params . . . . .	20
rename_params . . . . .	21
Request . . . . .	21
Response . . . . .	21
ResponseErrorMessage . . . . .	22
run . . . . .	22
ServerCapabilities . . . . .	23
signature_reply . . . . .	23
style_file . . . . .	24
style_text . . . . .	24
symbol_information . . . . .	25
text_document_position_params . . . . .	25
text_edit . . . . .	26
will_save_text_document_params . . . . .	26

*languageserver-package* 3

Workspace . . . . .	27
workspace_completion . . . . .	28
workspace_sync . . . . .	28

**Index** 29

---

languageserver-package  
*languageserver: Language Server Protocol*

---

### Description

An implementation of the Language Server Protocol for R. The Language Server protocol is used by an editor client to integrate features like auto completion. See <<https://microsoft.github.io/language-server-protocol>> for details.

### Details

An implementation of the Language Server Protocol for R

### See Also

Useful links:

- <https://github.com/REditorSupport/languageserver>

---

arg\_completion      *complete a function argument*

---

### Description

complete a function argument

### Usage

```
arg_completion(workspace, token, funct, package = NULL)
```

### Arguments

workspace	a <a href="#">Workspace</a> object
token	a character, the start of the argument to identify
funct	a character, the function name
package	a character, the optional package name

### Value

a list of candidates

---

check_scope	<i>check if a token is in a R code block in an Rmarkdown file</i>
-------------	---

---

### Description

In an RMarkdown document, tokens can be either inside an R code block or in the text. This function will return FALSE if the token is in the text and TRUE if it is in a code block. For R scripts, it always returns TRUE.

### Usage

```
check_scope(uri, document, position)
```

### Arguments

uri	a character, the path to a file as defined by <a href="#">RFC 3986</a>
document	a character, the contents of the document
position	a <a href="#">position</a> object

### Value

a logical

---

code_action_params	<i>parameters for code action requests</i>
--------------------	--

---

### Description

parameters for code action requests

### Usage

```
code_action_params(uri, range, context = NULL)
```

### Arguments

uri	a character, the path to a file as defined by <a href="#">RFC 3986</a>
range	a <a href="#">range</a> object
context	a named list

---

code\_lens\_params      *parameters for code lens requests*

---

**Description**

parameters for code lens requests

**Usage**

```
code_lens_params(uri)
```

**Arguments**

uri                  a character, the path to a file as defined by [RFC 3986](#)

---

completion\_params      *parameters for completion requests*

---

**Description**

parameters for completion requests

**Usage**

```
completion_params(uri, position, context = NULL)
```

**Arguments**

uri                  a character, the path to a file as defined by [RFC 3986](#)

position             a [position](#) object

context              a named list

---

completion\_reply      *the response to a textDocument/completion request*

---

**Description**

the response to a textDocument/completion request

**Usage**

```
completion_reply(id, uri, workspace, document, position)
```

**Arguments**

id	a numeric, the id of the process that started the server
uri	a character, the path to a file as defined by <a href="#">RFC 3986</a>
workspace	a <a href="#">Workspace</a> object, the current workspace
document	a character, the contents of the document
position	a <a href="#">position</a> object

**Value**

a [Response](#) object

---

content\_backward\_search  
*search backwards in a document content for a specific character*

---

**Description**

search backwards in a document content for a specific character

**Usage**

```
content_backward_search(content, row, column, char,  
  skip_empty_line = TRUE)
```

**Arguments**

content	a character vector
row	an 0-indexed integer
column	an 0-indexed integer
char	a single character
skip_empty_line	a logical

**Value**

a tuple of positive integers, the row and column position of the character if found, otherwise (-1, -1)

---

DefinitionCache	<i>A data structure to hold function definition locations</i>
-----------------	---

---

**Description**

The key reason for using this rather than a list is that this also cleans up when functions are removed from files.

**Usage**

DefinitionCache

**Format**

An object of class R6ClassGenerator of length 24.

**Methods**

- `get(funcnt)`: get the location of a definition, or NULL if it is not found
- `get_funcs_for_uri(uri)`: get the locations of all the definitions in uri
- `filter(pattern)`: get the locations of all the definitions matching pattern
- `update(uri, ranges)`: create locations of functions in uri from ranges, and remove functions no longer in uri

---

<code>diagnose_file</code>	<i>run diagnostic on a file</i>
----------------------------	---------------------------------

---

**Description**

Lint and diagnose problems in a file.

**Usage**

`diagnose_file(path)`

**Arguments**

`path` a character, the path to a file

diagnostics

*diagnostics*

---

### Description

Diagnose problems in files after linting.

### Usage

diagnostic\_range(result)

diagnostic\_severity(result)

diagnostic\_from\_lint(result)

### Arguments

result            the result from `lintr::lint()`

---

did\_change\_configuration\_params

*parameters for workspace/didChangeConfiguration notifications*

---

### Description

parameters for workspace/didChangeConfiguration notifications

### Usage

did\_change\_configuration\_params(settings)

### Arguments

settings            a named list



---

*did\_change\_text\_document\_params*  
*parameters for didChange notifications*

---

### **Description**

parameters for didChange notifications

### **Usage**

`did_change_text_document_params(uri, changes)`

### **Arguments**

`uri` a character, the path to a file as defined by [RFC 3986](#)  
`changes` a list of `text_document_content_change_event`

---

*did\_close\_text\_document\_params*  
*parameters for didClose notifications*

---

### **Description**

parameters for didClose notifications

### **Usage**

`did_close_text_document_params(uri)`

### **Arguments**

`uri` a character, the path to a file as defined by [RFC 3986](#)

---

*did\_open\_text\_document\_params*  
*parameters for didOpen notifications*

---

**Description**

parameters for didOpen notifications

**Usage**

`did_open_text_document_params(uri)`

**Arguments**

`uri` a character, the path to a file as defined by [RFC 3986](#)

---

*did\_save\_text\_document\_params*  
*parameters for didSave notifications*

---

**Description**

parameters for didSave notifications

**Usage**

`did_save_text_document_params(uri, text)`

**Arguments**

`uri` a character, the path to a file as defined by [RFC 3986](#)

`text` a character

---

document\_formatting\_params  
*parameters for document formatting requests*

---

**Description**

parameters for document formatting requests

**Usage**

document\_formatting\_params(uri, options)

**Arguments**

uri                    a character, the path to a file as defined by [RFC 3986](#)  
options                a named list

---

document\_link\_params    *parameters for document link requests*

---

**Description**

parameters for document link requests

**Usage**

document\_link\_params(uri)

**Arguments**

uri                    a character, the path to a file as defined by [RFC 3986](#)

document\_on\_type\_formatting\_params

*parameters for document on type formatting requests*

---

### **Description**

parameters for document on type formatting requests

### **Usage**

document\_on\_type\_formatting\_params(uri, position, character, options)

### **Arguments**

uri	a character, the path to a file as defined by <a href="#">RFC 3986</a>
position	a <a href="#">position</a> object
character	a single character
options	a named list

---

document\_range\_formatting\_params

*parameters for document range formatting requests*

---

### **Description**

parameters for document range formatting requests

### **Usage**

document\_range\_formatting\_params(uri, range, options)

### **Arguments**

uri	a character, the path to a file as defined by <a href="#">RFC 3986</a>
range	a <a href="#">range</a> object
options	a named list

---

document\_symbol\_params  
*parameters for document symbol requests*

---

**Description**

parameters for document symbol requests

**Usage**

document\_symbol\_params(uri)

**Arguments**

uri                    a character, the path to a file as defined by [RFC 3986](#)

---

document\_uri            *a document URI*

---

**Description**

A Uniform Resource Identifier as defined by [RFC 3986](#).

**Usage**

document\_uri(uri)

**Arguments**

uri                    a character

---

formatting\_reply        *format a document*

---

**Description**

format a document

**Usage**

formatting\_reply(id, uri, document, options)

**Arguments**

id	a numeric, the id of the process that started the server
uri	a character, the path to a file as defined by <a href="#">RFC 3986</a>
document	a character, the contents of the document
options	a named list of options, with a <code>tabSize</code> parameter

---

hover_reply	<i>the response to a <code>textDocument/hover</code> Request</i>
-------------	--

---

**Description**

When hovering on a symbol, if it is a function, return its help text if it exists in the current [Workspace](#).

**Usage**

```
hover_reply(id, uri, workspace, document, position)
```

**Arguments**

id	a numeric, the id of the process that started the server
uri	a character, the path to a file as defined by <a href="#">RFC 3986</a>
workspace	a <a href="#">Workspace</a> object, the current workspace
document	a character, the contents of the document
position	a <a href="#">position</a> object

**Value**

a [Response](#) object

---

is_rmarkdown	<i>check if a file is an <code>RMarkdown</code> file</i>
--------------	--

---

**Description**

check if a file is an `RMarkdown` file

**Usage**

```
is_rmarkdown(uri)
```

**Arguments**

uri	a character, the path to a file as defined by <a href="#">RFC 3986</a>
-----	--

**Value**

a logical

---

LanguageServer	<i>the language server</i>
----------------	----------------------------

---

**Description**

Describe the language server and how it interacts with clients.

**Usage**

LanguageServer

**Format**

An object of class R6ClassGenerator of length 24.

---

location	<i>a location inside a resource</i>
----------	-------------------------------------

---

**Description**

Represents a location inside a resource, such as a line inside a text file.

**Usage**

location(uri, range)

**Arguments**

uri	a character, the path to a file as defined by <a href="#">RFC 3986</a>
range	a <a href="#">range</a>

---

Message	<i>base Message class</i>
---------	---------------------------

---

**Description**

base Message class

**Usage**

Message

**Format**

An object of class R6ClassGenerator of length 24.

---

Namespace	<i>A data structure for package namespaces</i>
-----------	--

---

**Description**

A Namespace is initialized with a package name and builds the list of objects defined in the package namespace.

**Usage**

Namespace

**Format**

An object of class R6ClassGenerator of length 24.

**Fields**

objname a character, an object name

funct a character, a function name

**Methods**

- exists(objname): returns true if objname can be found in the Namespace
- get\_signature(funct): return the signature of funct
- get\_formals(funct): return the `base::formals()` of funct

---

Notification	<i>Notification Message class</i>
--------------	-----------------------------------

---

**Description**

Notification Message class

**Usage**

Notification

**Format**

An object of class R6ClassGenerator of length 24.



---

on_initialize	<i>initialize handler</i>
---------------	---------------------------

---

**Description**

Handler to the [initialize Request](#).

**Usage**

```
on_initialize(self, id, params)
```

**Arguments**

self	a <a href="#">LanguageServer</a> object
id	a numeric, the id of the process that started the server
params	a named list, the <a href="#">initialize Request</a> options @keywords internal

---

package_completion	<i>complete a package name</i>
--------------------	--------------------------------

---

**Description**

complete a package name

**Usage**

```
package_completion(token)
```

**Arguments**

token	a character, the start of the package name to identify
-------	--

**Value**

a list of candidates

---

<code>parse_document</code>	<i>parse a document</i>
-----------------------------	-------------------------

---

**Description**

Build the list of called packages, functions, variables, formals and signatures in the document in order to add them to the current [Workspace](#).

**Usage**

```
parse_document(path)
```

**Arguments**

<code>path</code>	a character, the path to the document
-------------------	---------------------------------------

---

<code>path_from_uri</code>	<i>paths and uris</i>
----------------------------	-----------------------

---

**Description**

paths and uris

**Usage**

```
path_from_uri(uri)
```

```
path_to_uri(path)
```

**Arguments**

<code>uri</code>	a character, the path to a file in URI format
------------------	---

<code>path</code>	a character, the path to a file
-------------------	---------------------------------

---

position	<i>a position in a text document</i>
----------	--------------------------------------

---

**Description**

Position in a text document expressed as zero-based line and zero-based character offset.

**Usage**

```
position(line, character)
```

**Arguments**

line	an integer
character	an integer

---

range	<i>a range in a text document</i>
-------	-----------------------------------

---

**Description**

A range in a text document expressed as start and end [positions](#). A range is comparable to a selection in an editor.

**Usage**

```
range(start, end)
```

**Arguments**

start	a <a href="#">position</a>
end	a <a href="#">position</a>

range\_formatting\_reply

*format a part of a document*

---

### Description

format a part of a document

### Usage

```
range_formatting_reply(id, uri, document, range, options)
```

### Arguments

id	a numeric, the id of the process that started the server
uri	a character, the path to a file as defined by <a href="#">RFC 3986</a>
document	a character, the contents of the document
range	a <a href="#">range</a> , the part of the document to format
options	a named list of options, with a <code>tabSize</code> parameter

---

reference\_params

*parameters for reference requests*

---

### Description

parameters for reference requests

### Usage

```
reference_params(uri, position, context = NULL)
```

### Arguments

uri	a character, the path to a file as defined by <a href="#">RFC 3986</a>
position	a <a href="#">position</a> object
context	a named list

---

rename_params	<i>parameters for rename requests</i>
---------------	---------------------------------------

---

**Description**

parameters for rename requests

**Usage**

```
rename_params(uri, position, new_name)
```

**Arguments**

uri	a character, the path to a file as defined by <a href="#">RFC 3986</a>
position	a <a href="#">position</a> object
new_name	a character

---

Request	<i>Request Message class</i>
---------	------------------------------

---

**Description**

Describe a request between the client and the server.

**Usage**

Request

**Format**

An object of class R6ClassGenerator of length 24.

---

Response	<i>Response Message class</i>
----------	-------------------------------

---

**Description**

Message sent as the result of a [Request](#)

**Usage**

Response

**Format**

An object of class R6ClassGenerator of length 24.

---

ResponseErrorMessage    *Response Error Message class*

---

### Description

Message sent as the result of a [Request](#) in case of an error.

### Usage

ResponseErrorMessage

### Format

An object of class R6ClassGenerator of length 24.

---

run                            *Run the R language server*

---

### Description

Run the R language server

### Usage

```
run(debug = FALSE, host = "localhost", port = NULL)
```

### Arguments

debug	set TRUE to show debug information in stderr; or it could be a character string specifying the log file
host	the hostname used to create the tcp server, not used when port is NULL
port	the port used to create the tcp server. If NULL, use stdio instead.

### Examples

```
## Not run:
# to use stdio
languageserver::run()

# to use tcp server
languageserver::run(port = 8888)

## End(Not run)
```

---

ServerCapabilities      *Server capabilities*

---

### Description

List all supported capabilities of the language server, as defined in the [Language Server Protocol specification](#).

### Details

Currently the capabilities are:

- textDocumentSync
- hoverProvider
- completionProvider
- signatureHelpProvider
- definitionProvider
- documentSymbolProvider
- workspaceSymbolProvider
- documentFormattingProvider
- documentRangeFormattingProvider

---

signature\_reply      *the response to a textDocument/signatureHelp Request*

---

### Description

If the symbol at the current position is a function, return its arguments (as with [base::args\(\)](#)).

### Usage

```
signature_reply(id, uri, workspace, document, position)
```

### Arguments

id	a numeric, the id of the process that started the server
uri	a character, the path to a file as defined by <a href="#">RFC 3986</a>
workspace	a <a href="#">Workspace</a> object, the current workspace
document	a character, the contents of the document
position	a <a href="#">position</a> object

### Value

a [Response](#) object

---

style_file	<i>style a file</i>
------------	---------------------

---

**Description**

This functions formats a document using `styler::style_file()` with the `styler::tidyverse_style()` style.

**Usage**

```
style_file(path, options)
```

**Arguments**

path	file path
options	a named list of options, with a <code>tabSize</code> parameter

---

style_text	<i>edit code style</i>
------------	------------------------

---

**Description**

This functions formats a list of text using `styler::style_text()` with the `styler::tidyverse_style()` style.

**Usage**

```
style_text(text, options, scope = "tokens", indentation = "")
```

**Arguments**

text	a vector of text
options	a named list of options, with a <code>tabSize</code> parameter
scope	the scope type used in <code>styler::tidyverse_style</code>
indentation	amount of whitespaces put at the begining of each line



---

symbol_information	<i>non-hierarchical symbol information</i>
--------------------	--

---

**Description**

non-hierarchical symbol information

**Usage**

```
symbol_information(name, kind, location)
```

**Arguments**

name	a character
kind	an integer
location	a <a href="#">location</a>

---

text_document_position_params	<i>a text document and a position inside that document</i>
-------------------------------	--

---

**Description**

a text document and a position inside that document

**Usage**

```
text_document_position_params(uri, position)
```

**Arguments**

uri	a character, the path to a file as defined by <a href="#">RFC 3986</a>
position	a <a href="#">position</a> object

text\_edit                    *a textual edit applicable to a text document*

---

**Description**

a textual edit applicable to a text document

**Usage**

```
text_edit(range, new_text)
```

**Arguments**

range                    a [range](#), the part of the document to replace  
new\_text                a character, the text to replace

---

will\_save\_text\_document\_params  
*parameters for willSave notifications*

---

**Description**

parameters for willSave notifications

**Usage**

```
will_save_text_document_params(uri, reason)
```

**Arguments**

uri                    a character, the path to a file as defined by [RFC 3986](#)  
reason                an integer, see `TextDocumentSaveReason`

---

Workspace

*A data structure for a session workspace*

---

### Description

A Workspace is initialized at the start of a session, when the language server is started. Its goal is to contain the Namespaces of the packages that are loaded during the session for quick reference.

### Usage

Workspace

### Format

An object of class R6ClassGenerator of length 24.

### Fields

pkgname a character, a package name

object a character, an object name

funct a character, a function name

topic a character, a help topic name

parse\_result ?

### Methods

- `load_package(pkgname)`: add a new Namespace for pkgname if necessary and if possible
- `guess_package(object)`: returns the most recently loaded package in which object can be found
- `get_namespace(pkgname)`: return the [Namespace](#) of pkgname
- `get_signature(funct, pkgname = NULL)`: return the signature of funct
- `get_formals(funct, pkgname = NULL)`: return the `base::formals()` of funct
- `get_help(topic, pkgname = NULL)`: return the help text of topic
- `load_to_global(parse_result)`:

---

workspace\_completion    *complete any object in the workspace*

---

### Description

complete any object in the workspace

### Usage

```
workspace_completion(workspace, token, package = NULL,
  exported_only = TRUE)
```

### Arguments

workspace	a <a href="#">Workspace</a> object
token	a character, the object to identify
package	a character
exported_only	a boolean

### Value

a list of candidates

---

workspace\_sync            *Determine workspace information for a given file*

---

### Description

internal use only

### Usage

```
workspace_sync(uri, temp_file = NULL, run_lintr = TRUE,
  parse = FALSE)
```

### Arguments

uri	the file uri
temp_file	the file to lint, determine from uri if NULL
run_lintr	set FALSE to disable lintr diagnostics
parse	set FALSE to disable parsing file

# Index

## \*Topic **datasets**

- DefinitionCache, 7
  - LanguageServer, 15
  - Message, 15
  - Namespace, 16
  - Notification, 16
  - Request, 21
  - Response, 21
  - ResponseErrorMessage, 22
  - Workspace, 27
- arg\_completion, 3
- base::args(), 23
- base::formals(), 16, 27
- check\_scope, 4
- code\_action\_params, 4
- code\_lens\_params, 5
- completion\_params, 5
- completion\_reply, 6
- content\_backward\_search, 6
- DefinitionCache, 7
- diagnose\_file, 7
- diagnostic\_from\_lint (diagnostics), 8
- diagnostic\_range (diagnostics), 8
- diagnostic\_severity (diagnostics), 8
- diagnostics, 8
- did\_change\_configuration\_params, 8
- did\_change\_text\_document\_params, 9
- did\_close\_text\_document\_params, 9
- did\_open\_text\_document\_params, 10
- did\_save\_text\_document\_params, 10
- document\_formatting\_params, 11
- document\_link\_params, 11
- document\_on\_type\_formatting\_params, 12
- document\_range\_formatting\_params, 12
- document\_symbol\_params, 13
- document\_uri, 13
- formatting\_reply, 13
- hover\_reply, 14
- is\_rmarkdown, 14
- LanguageServer, 15, 17
- languageserver  
(languageserver-package), 3
- languageserver-package, 3
- lintr::lint(), 8
- location, 15, 25
- Message, 15
- Namespace, 16, 27
- Notification, 16
- on\_initialize, 17
- package\_completion, 17
- parse\_document, 18
- path\_from\_uri, 18
- path\_to\_uri (path\_from\_uri), 18
- position, 4–6, 12, 14, 19, 19, 20, 21, 23, 25
- range, 4, 12, 15, 19, 20, 26
- range\_formatting\_reply, 20
- reference\_params, 20
- rename\_params, 21
- Request, 17, 21, 21, 22
- Response, 6, 14, 21, 23
- ResponseErrorMessage, 22
- run, 22
- ServerCapabilities, 23
- signature\_reply, 23
- style\_file, 24
- style\_text, 24
- styler::style\_file(), 24
- styler::style\_text(), 24

`styler::tidyverse_style()`, 24  
`symbol_information`, 25

`text_document_position_params`, 25  
`text_edit`, 26

`will_save_text_document_params`, 26  
Workspace, 3, 6, 14, 18, 23, 27, 28  
`workspace_completion`, 28  
`workspace_sync`, 28