

Package: rscontract (via r-universe)

June 27, 2024

Title Generic implementation of the 'RStudio' connections contract

Version 0.1.2

Description Provides a generic implementation of the 'RStudio' connection contract to make it easier for database connections, and other type of connections, opened via R packages integrate with the connections pane inside the 'RStudio' interactive development environment (IDE).

License GPL-3

Suggests testthat (>= 2.1.0), covr

Encoding UTF-8

LazyData true

RoxygenNote 7.0.2

URL <https://github.com/rstudio/rscontract>

BugReports <https://github.com/rstudio/rscontract/issues>

Language en-US

Repository <https://rstudio.r-universe.dev>

RemoteUrl <https://github.com/rstudio/rscontract>

RemoteRef HEAD

RemoteSha 209a6a6ded09a434821e632a0efe820ad4f74d87

Contents

as_rscontract	2
rscontract_close	2
rscontract_ide	3
rscontract_open	4
rscontract_spec	5
rscontract_update	6
sample_catalog	7

Index	8
--------------	----------

as_rscontract	<i>Coerces object into an RStudio connection contract</i>
---------------	---

Description

Coerces object into an RStudio connection contract

Usage

```
as_rscontract(x)
```

Arguments

x Object that will be coerced

Value

An 'rs_contract_ide' class object

Examples

```
as_rscontract(sample_catalog())
```

rscontract_close	<i>Close a connection</i>
------------------	---------------------------

Description

Close a connection

Usage

```
rscontract_close(host = "", type = "")
```

Arguments

host Host name of the connection. Optional, defaults to empty
type Type of connection. Optional, defaults to empty

Value

Returns a NULL object. If using the RStudio IDE, it will attempt to close the connection identified by the 'host' and 'type' arguments

Examples

```
rscontract_open(rscontract_spec())
rscontract_close("spec_host", "spec_type")
```

rscontract_ide

Mirrors the RStudio IDE connection contract arguments

Description

Creates an object that contains all of the necessary information to create a connection contract. This function is meant as a lower level integration with the connections pane.

Usage

```
rscontract_ide(
  connectionObject = NULL,
  type = "",
  host = "",
  icon = NULL,
  displayName = "",
  connectCode = "",
  disconnect = function() { },
  previewObject = function() { },
  listObjectTypes = default_types(),
  listObjects = function() { },
  listColumns = function() { },
  actions = NULL
)
```

Arguments

connectionObject	The connection object. Default to NULL.
type	Type of the connection.
host	Name of the host
icon	Path to the connection's icon. Defaults to NULL.
displayName	The connection's name.
connectCode	The text of the connection code.
disconnect	Function to use to disconnect. Default to function().
previewObject	Function to run when the preview table icon is clicked on. Default to function().
listObjectTypes	Function that provides the structure of the connection. The default function will work properly, it is going to be rare when it needs to be changed.

listObjects	Function to run to get the catalogs, schema, tables or views based what has been expanded on. Default to function().
listColumns	Function to run that pull the field information. Default to function().
actions	Additional buttons to add to the connection pane. Defaults to NULL.

Value

An 'rs_contract_ide' class object

Examples

```
str(rscontract_ide())
```

rscontract_open	<i>Opens a connection</i>
-----------------	---------------------------

Description

Opens a connection

Usage

```
rscontract_open(x)
```

Arguments

x An rscontract_ide object, or an object coercible to an rscontract_ide

Value

Returns a NULL object. If using the RStudio IDE, it will attempt to open the connection, and set the identification using the 'host' and 'type' argument values

Examples

```
rscontract_open(rscontract_spec())
rscontract_close("spec_host", "spec_type")
```

rscontract_spec	<i>A flexible API that can be converted to an RStudio Connection Contract</i>
-----------------	---

Description

A flexible API that can be converted to an RStudio Connection Contract

Usage

```
rscontract_spec(
  connection_object = NULL,
  type = "spec_type",
  host = "spec_host",
  icon = NULL,
  name = "",
  connect_script = "library(connections)\n[Place your code here]",
  disconnect_code = "function() rscontract_close('spec_host', 'spec_type')",
  preview_code = "function(){}",
  catalog_list = "sample_catalog()",
  object_types = "default_types()",
  object_list = NULL,
  object_columns = NULL,
  actions = NULL
)
```

Arguments

connection_object	The connection object. Default to NULL.
type	Type of the connection.
host	Name of the host
icon	Path to the connection's icon. Defaults to NULL.
name	The connection's name.
connect_script	The text of the connection code.
disconnect_code	Function to use to disconnect. Default to function().
preview_code	Function to run when the preview table icon is clicked on. Default to function().
catalog_list	Hierarchical list of the catalogs, schema, table/view and fields.
object_types	Function that provides the structure of the connection. The default function will work properly, it is going to be rare when it needs to be changed.
object_list	Function to run to get the catalogs, schema, tables or views based what has been expanded on. Defaults to NULL.
object_columns	Function to run that pull the field information. Default to NULL
actions	Additional buttons to add to the connection pane. Defaults to NULL.

Value

Returns an 'rscontract_spec' class object

Examples

```
str(rscontract_spec())
```

rscontract_update	<i>Refreshes a connection</i>
-------------------	-------------------------------

Description

Refreshes a connection

Usage

```
rscontract_update(host = "", type = "", hint = "")
```

Arguments

host	Connection's Host name
type	The connection's type
hint	Optional argument passed to the Contract

Value

Returns a NULL object. If using the RStudio IDE, it will attempt to refresh the connection identified by the 'host' and 'type' arguments

Examples

```
con <- rscontract_open(rscontract_spec())  
rscontract_update(con)  
rscontract_close(con)
```

sample_catalog	<i>A example of how a connection hierarchy structure</i>
----------------	--

Description

A example of how a connection hierarchy structure

Usage

```
sample_catalog()
```

Value

A 'list' object containing all of the needed information to open a very simple connection contract

Examples

```
str(sample_catalog())
```

Index

`as_rscontract`, [2](#)

`rscontract_close`, [2](#)

`rscontract_ide`, [3](#)

`rscontract_open`, [4](#)

`rscontract_spec`, [5](#)

`rscontract_update`, [6](#)

`sample_catalog`, [7](#)