

# Package ‘sqlstrings’

May 9, 2026

**Title** Map 'SQL' Code to R Lists

**Version** 1.0.0

**Description** Provides a helper function, to bulk read 'SQL' code from separate files and load it into an 'R' list, where the list elements contain the individual statements and queries as strings. This works by annotating the 'SQL' code with a name comment, which also will be the name of the list element.

**License** MIT + file LICENSE

**Encoding** UTF-8

**RoxygenNote** 7.2.1

**Suggests** testthat (>= 3.0.0)

**Config/testthat/edition** 3

**Imports** fs, readr, stringr

**NeedsCompilation** no

**Author** Dejan Prvulovic [aut, cre]

**Maintainer** Dejan Prvulovic <dejan.prv@gmail.com>

**Repository** CRAN

**Date/Publication** 2022-09-09 08:10:05 UTC

## Contents

generate_sql_strings . . . . .	2
<b>Index</b>	<b>3</b>

---

`generate_sql_strings` *Read all files from path and return a list, where the list elements contain the individual 'SQL' statements and queries as strings*

---

### Description

Read all files from path and return a list, where the list elements contain the individual 'SQL' statements and queries as strings

### Usage

```
generate_sql_strings(path = "")
```

### Arguments

`path`                    A path to a folder or a file containing 'SQL' code

### Value

A list with named elements containing 'SQL' statements as strings

### Examples

```
## prepare example file
p <- fs::path(tempdir(), "test-file-001.sql")
fs::file_create(p)
readr::write_lines("
  -- name: select_count
  select count(*) from tab1;
", p)

## sqlstrings
s <- generate_sql_strings(path = p)
s$select_count
```

# Index

`generate_sql_strings`, [2](#)