

# Package ‘sqlserverconnect’

May 9, 2026

**Title** Simple Helpers for Connecting to 'SQL Server'

**Version** 0.1.0

**Description** Lightweight helpers for connecting to Microsoft 'SQL Server' using 'DBI', 'odbc', and 'pool'. Provides simple wrappers for building connection arguments, establishing connections, and safely disconnecting.

**License** MIT + file LICENSE

**Encoding** UTF-8

**RoxygenNote** 7.3.3

**Imports** DBI, odbc, pool, cli

**NeedsCompilation** no

**Author** Dave Rosenman [aut, cre]

**Maintainer** Dave Rosenman <dave.rosenman.data@gmail.com>

**Repository** CRAN

**Date/Publication** 2026-02-16 17:30:08 UTC

## Contents

db_connect . . . . .	1
db_connection_args . . . . .	3
db_disconnect . . . . .	4

<b>Index</b>	<b>5</b>
--------------	----------

---

db_connect	<i>Connect to a SQL Server database</i>
------------	---

---

## Description

Connect to a SQL Server database

**Usage**

```
db_connect(
  server,
  database,
  uid = NULL,
  pwd = NULL,
  port = NULL,
  trusted = TRUE,
  driver = "ODBC Driver 17 for SQL Server",
  pool = FALSE,
  quiet = FALSE,
  ...
)
```

**Arguments**

server	SQL Server hostname, IP address, or instance name
database	Database name
uid	Username (ignore/keep as NULL if trusted = TRUE)
pwd	Password (ignore/keep as NULL if trusted = TRUE)
port	Optional port number. If NULL, odbc package handles port resolution
trusted	(logical) If TRUE (default), uses Windows authentication
driver	ODBC driver name (default is "ODBC Driver 17 for SQL Server")
pool	(logical) if TRUE, returns a pooled connection
quiet	(logical) if TRUE, suppresses messages
...	Additional arguments passed to DBI::dbConnect or pool::dbPool

**Value**

A DBI connection or a pool object

**Examples**

```
# Connect to a SQL Server database
conn <- db_connect(
  server = "localhost",
  database = "master",
  quiet = TRUE
)

# Run a simple query
DBI::dbGetQuery(conn, "SELECT name FROM sys.databases")

# Disconnect when finished
db_disconnect(conn)
```

---

db\_connection\_args     *Build SQL Server connection arguments*

---

### Description

Build SQL Server connection arguments

### Usage

```
db_connection_args(  
  server,  
  database,  
  uid = NULL,  
  pwd = NULL,  
  port = NULL,  
  trusted = TRUE,  
  driver = "ODBC Driver 17 for SQL Server"  
)
```

### Arguments

server	SQL Server hostname, IP address, or instance name
database	Database name
uid	Username (ignore/keep as NULL if trusted = TRUE)
pwd	Password (ignore/keep as NULL if trusted = TRUE)
port	Optional port number. If NULL, odbc package handles port resolution
trusted	(logical) If TRUE (default), uses Windows authentication
driver	ODBC driver name (default is "ODBC Driver 17 for SQL Server")

### Value

A named list of arguments suitable for a SQL Server connection string in `DBI::dbConnect()` or `pool::dbPool()`. Used internally by `db_connect()` to construct the argument list.

### Examples

```
# Build arguments using Windows authentication  
db_connection_args(  
  server = "localhost",  
  database = "master"  
)  
  
# Build arguments using SQL authentication  
db_connection_args(  
  server = "localhost",  
  database = "master",
```

```
uid      = "sa",
pwd      = "password",
trusted  = FALSE
)
```

---

**db\_disconnect***Disconnect from a SQL Server connection or pool*

---

**Description**

Disconnect from a SQL Server connection or pool

**Usage**

```
db_disconnect(conn, quiet = FALSE)
```

**Arguments**

conn	A DBI connection or a pool connection object
quiet	(logical) if TRUE, suppresses messages

**Value**

TRUE (invisibly) if disconnected, FALSE otherwise

**Examples**

```
# Establish a connection
conn <- db_connect(
  server  = "localhost",
  database = "master",
  quiet   = TRUE
)

# Disconnect when finished
db_disconnect(conn)

# Disconnecting a NULL connection
db_disconnect(NULL)
```

# Index

db\_connect, 1  
db\_connection\_args, 3  
db\_disconnect, 4