

NAME

gsettings - GSettings configuration tool

SYNOPSIS

gsettings get *SCHEMA* [:*PATH*] *KEY*

gsettings monitor *SCHEMA* [:*PATH*] [*KEY*]

gsettings writable *SCHEMA* [:*PATH*] *KEY*

gsettings range *SCHEMA* [:*PATH*] *KEY*

gsettings describe *SCHEMA* [:*PATH*] *KEY*

gsettings set *SCHEMA* [:*PATH*] *KEY* *VALUE*

gsettings reset *SCHEMA* [:*PATH*] *KEY*

gsettings reset-recursively *SCHEMA* [:*PATH*]

gsettings list-schemas [--print-paths]

gsettings list-relocatable-schemas

gsettings list-keys *SCHEMA* [:*PATH*]

gsettings list-children *SCHEMA* [:*PATH*]

gsettings list-recursively [*SCHEMA* [:*PATH*]]

gsettings help [*COMMAND*]

DESCRIPTION

gsettings offers a simple commandline interface to **GSettings**. It lets you get, set or monitor an individual key for changes.

The *SCHEMA* and *KEY* arguments are required for most commands to specify the schema id and the name of the key to operate on. The schema id may optionally have a *:PATH* suffix. Specifying the path is only needed if the schema does not have a fixed path.

When setting a key, you also need specify a *VALUE*. The format for the value is that of a serialized **GVariant**, so e.g. a string must include explicit quotes: `"'foo'"`. This format is also used when printing out values.

Note that `gsettings` needs a D-Bus session bus connection to write changes to the `dconf` database.

COMMANDS

get

Gets the value of *KEY*. The value is printed out as a serialized **GVariant**.

monitor

Monitors *KEY* for changes and prints the changed values. If no *KEY* is specified, all keys in the schema are monitored. Monitoring will continue until the process is terminated.

writable

Finds out whether *KEY* is writable.

range

Queries the range of valid values for *KEY*.

describe

Queries the description of valid values for *KEY*.

set

Sets the value of *KEY* to *VALUE*. The value is specified as a serialized **GVariant**.

reset

Resets *KEY* to its default value.

reset-recursively

Reset all keys under the given *SCHEMA*.

list-schemas

Lists the installed, non-relocatable schemas. See **list-relocatable-schemas** if you are interested in relocatable schemas. If `[--print-paths]` is given, the path where each schema is mapped is also printed.

list-relocatable-schemas

Lists the installed, relocatable schemas. See **list-schemas** if you are interested in non-relocatable schemas.

list-keys

Lists the keys in *SCHEMA*.

list-children

Lists the children of *SCHEMA*.

list-recursively

Lists keys and values, recursively. If no *SCHEMA* is given, list keys in all schemas.

help

Prints help and exits.