



Linux Ubuntu 26.04 Manual Pages on command 'git-ls-remote.1'

\$ man git-ls-remote.1

GIT-LS-REMOTE(1) Git Manual GIT-LS-REMOTE(1)

NAME

git-ls-remote - List references in a remote repository

SYNOPSIS

```
git ls-remote [--branches] [--tags] [--refs] [--upload-pack=<exec>]
               [-q | --quiet] [--exit-code] [--get-url] [--sort=<key>]
               [--symref] [<repository> [<patterns>...]]
```

DESCRIPTION

Displays references available in a remote repository along with the associated commit IDs.

OPTIONS

-b, --branches, -t, --tags

Limit to only local branches and local tags, respectively. These options are not mutually exclusive; when given both, references stored in refs/heads and refs/tags are displayed. Note that --heads and -h are deprecated synonyms for --branches and -b and may be removed in the future. Also note that git ls-remote -h used without anything else on the command line gives help, consistent with other git subcommands.

--refs

Do not show peeled tags or pseudorefs like HEAD in the output.

-q, --quiet

Do not print remote URL to stderr.

--upload-pack=<exec>

Specify the full path of git-upload-pack on the remote host. This allows listing references from repositories accessed via SSH and where the SSH daemon does not use the

PATH configured by the user.

`--exit-code`

Exit with status "2" when no matching refs are found in the remote repository. Usually the command exits with status "0" to indicate it successfully talked with the remote repository, whether it found any matching refs.

`--get-url`

Expand the URL of the given remote repository taking into account any "url.<base>.insteadOf" config setting (See `git-config(1)`) and exit without talking to the remote.

`--symref`

In addition to the object pointed by it, show the underlying ref pointed by it when showing a symbolic ref. Currently, `upload-pack` only shows the symref HEAD, so it will be the only one shown by `ls-remote`.

`--sort=<key>`

Sort based on the key given. Prefix - to sort in descending order of the value. Supports "version:refname" or "v:refname" (tag names are treated as versions). The "version:refname" sort order can also be affected by the "versionsort.suffix" configuration variable. See `git-for-each-ref(1)` for more sort options, but be aware keys like `committerdate` that require access to the objects themselves will not work for refs whose objects have not yet been fetched from the remote, and will give a missing object error.

`-o <option>, --server-option=<option>`

Transmit the given string to the server when communicating using protocol version 2. The given string must not contain a NUL or LF character. When multiple `--server-option=<option>` are given, they are all sent to the other side in the order listed on the command line. When no `--server-option=<option>` is given from the command line, the values of configuration variable `remote.<name>.serverOption` are used instead.

`<repository>`

The "remote" repository to query. This parameter can be either a URL or the name of a remote (see the GIT URLs and REMOTES sections of `git-fetch(1)`).

`<patterns>...`

When unspecified, all references, after filtering done with `--heads` and `--tags`, are shown. When `<patterns>...` are specified, only references matching one or more of the

given patterns are displayed. Each pattern is interpreted as a glob (see glob in gitglossary(7)) which is matched against the "tail" of a ref, starting either from the start of the ref (so a full name like refs/heads/foo matches) or from a slash separator (so bar matches refs/heads/bar but not refs/heads/foobar).

OUTPUT

The output is in the format:

<oid> TAB <ref> LF

When showing an annotated tag, unless --refs is given, two such lines are shown: one with the refname for the tag itself as <ref>, and another with <ref> followed by ^{}. The <oid> on the latter line shows the name of the object the tag points at.

EXAMPLES

? List all references (including symbolics and pseudorefs), peeling tags:

```
$ git ls-remote
```

```
27d43aaaf50ef0ae014b88bba294f93658016a2e    HEAD
950264636c68591989456e3ba0a5442f93152c1a    refs/heads/main
d9ab777d41f92a8c1684c91cfb02053d7dd1046b    refs/heads/next
d4ca2e3147b409459955613c152220f4db848ee1    refs/tags/v2.40.0
73876f4861cd3d187a4682290ab75c9dccadbc56    refs/tags/v2.40.0^{}
```

? List all references matching given patterns:

```
$ git ls-remote http://www.kernel.org/pub/scm/git/git.git master seen rc
5fe978a5381f1fbad26a80e682ddd2a401966740    refs/heads/master
c781a84b5204fb294c9ccc79f8b3baceeb32c061    refs/heads/seen
```

? List only tags matching a given wildcard pattern:

```
$ git ls-remote --tags http://www.kernel.org/pub/scm/git/git.git v\*
485a869c64a68cc5795dd99689797c5900f4716d    refs/tags/v2.39.2
cbf04937d5b9fcf0a76c28f69e6294e9e3ecd7e6    refs/tags/v2.39.2^{}
```

```
d4ca2e3147b409459955613c152220f4db848ee1    refs/tags/v2.40.0
73876f4861cd3d187a4682290ab75c9dccadbc56    refs/tags/v2.40.0^{}
```

SEE ALSO

git-check-ref-format(1).

GIT

Part of the git(1) suite