git-prune(1) Manual Page

NAME

git-prune - Prune all unreachable objects from the object database

SYNOPSIS

git prune [-n] [-v] [--progress] [--expire <time>] [--]
[<head>…]

DESCRIPTION

> **NOTE**   In most cases, users should run *git gc*, which calls *git prune*. See the section "NOTES", below.

This runs *git fsck --unreachable* using all the refs available in `refs/`, optionally with additional set of objects specified on the command line, and prunes all unpacked objects unreachable from any of these head objects from the object database. In addition, it prunes the unpacked objects that are also found in packs by running *git prune-packed*. It also removes entries from .git/shallow that are not reachable by any ref.

Note that unreachable, packed objects will remain. If this is not desired, see git-repack(1).

OPTIONS

-n

--dry-run

   Do not remove anything; just report what it would remove.

-v

--verbose

Report all removed objects.

--progress

Show progress.

--expire <time>

Only expire loose objects older than <time>.

--

Do not interpret any more arguments as options.

<head>…

In addition to objects reachable from any of our references, keep objects reachable from listed <head>s.

EXAMPLES

To prune objects not used by your repository or another that borrows from your repository via its `.git/objects/info/alternates`:

```
$ git prune $(cd ../another && git rev-parse --all)
```

NOTES

In most cases, users will not need to call *git prune* directly, but should instead call *git gc*, which handles pruning along with many other housekeeping tasks.

For a description of which objects are considered for pruning, see *git fsck*'s --unreachable option.