

`git log` 命令用于显示提交日志信息。

## 使用语法

```
git log [<options>] [<revision range>] [[\--] <path>...]
```

## 描述

`git log` 命令用于显示提交日志信息。

该命令采用适用于 `git rev-list` 命令的选项来控制显示的内容以及如何以及适用于 `git diff-*` 命令的选项，以控制如何更改每个提交引入的内容。

## 示例

以下是一些示例 -

1.显示整个提交历史记录，但跳过合并

```
$ git log --no-merges
commit c5f8a258babf5eec54edc794ff980d8340396592
Author: maxsu <your_email@mail.com>
Date:   Wed Jul 12 22:07:59 2017 +0800

    commit a new file: newfile.txt
... ..
```

2.显示自 `v2.6.12` 版以来所有提交更改 `include/scsi` 或 `drivers/scsi` 子目录中的任何文件的所有提交

```
$ git log master include/scsi drivers/scsi
```

3.显示最近两周的更改文件 `gitk`。“`--`”是必要的，以避免与名为 `gitk` 的分支混淆

```
$ git log --since="2 weeks ago" -- gitk
```

4.显示“`test`”分支中尚未在“`release`”分支中的提交，以及每个提交修改的路径列表

```
$ git log --name-status release..test
```

5.显示更改 `builtin/rev-list.c` 的提交，包括在文件被赋予其现有名称之前发生的提交。

```
$ git log --follow builtin/rev-list.c
```

6.显示在任何本地分支中的所有提交，但不包括任何远程跟踪分支机构的起始点(`origin` 不具有)。

```
git log --branches --not --remotes=origin
```

7.显示本地主服务器中的所有提交，但不显示任何远程存储库主分支。

```
git log master --not --remotes=*/master
```

8.显示历史，包括变化差异，但仅从“主分支”的角度来看，忽略来自合并分支的提交，并显示合并引入的变化的完全差异。只有当遵守在一个整合分支上合并所有主题分支的严格策略时，这才有意义。

```
git log -p -m --first-parent
```

9.显示文件 `main.c` 中的函数 `main()` 随着时间的推移而演变。

```
git log -L '/int main/',/^}/:main.c
```

10.将显示最近三次的提交。

```
git log -3
```

11.根据提交 ID 查询日志

```
$ git log commit_id      #查询 ID(如：6bab70a08afdbf3f7faffaff9f5252a2e4e2d552)之前的记录，包含 commit
```

```
$ git log commit1_id commit2_id #查询 commit1 与 commit2 之间的记录，包括 commit1 和 commit2
```

```
$ git log commit1_id..commit2_id #同上，但是不包括 commit1
```

其中，`commit_id` 可以是提交哈希值的简写模式，也可以使用 `HEAD` 代替。`HEAD` 代表最后一次提交，`HEAD^` 为最后一个提交的父提交，等同于 `HEAD ~ 1`，`HEAD ~ 2` 代表倒数第二次提交

`--pretty` 按指定格式显示日志信息,可选项有：`oneline,short,medium,full,fuller,email,raw` 以及 `format:`,默认为 `medium`，可以通过修改配置文件来指定默认的方式。

```
$ git log (--pretty=)oneline
```

常见的 `format` 选项：

#选项	#说明
%H	提交对象(commit)的完整哈希字符串
%h	提交对象的简短哈希字符串
%T	树对象(tree)的完整哈希字符串
%t	树对象的简短哈希字符串
%P	父对象(parent)的完整哈希字符串
%p	父对象的简短哈希字符串
%an	作者(author)的名字
%ae	作者的电子邮件地址
%ad	作者修订日期(可以用 <code>-date=</code> 选项定制格式)
%ar	作者修订日期，按多久以前的方式显示
%cn	提交者(committer)的名字
%ce	提交者的电子邮件地址
%cd	提交日期

%cr	提交日期, 按多久以前的方式显示
%s	提交说明

注：作者是指最后一次修改文件的人；而提交者是指提交该文件的人。

```
$ git log --pretty=format:"%an %ae %ad %cn %ce %cd %cr %s" --graph
```

**--mergs** - 查看所有合并过的提交历史记录

**--no-merges** - 查看所有未被合并过的提交信息

**--author=someonet** - 查询指定作者的提交记录

```
$ git log --author=maxsu
```

**--since**, **--after** - 仅显示指定时间之后的提交(不包含当前日期)

**--until**, **--before** - 仅显示指定时间之前的提交(包含当前日期)

```
$ git log --before={3,weeks,ago} --after={2018-04-18}
```