

`git shortlog` 命令用于汇总 `git` 日志输出。

## 使用语法

```
git log --pretty=short | git shortlog [<options>]
git shortlog [<options>] [<revision range>] [[\--] <path>...]
```

## 描述

适当包含在发布公告中的格式汇总 `git` 日志输出。每个提交将按作者和标题分组。

另外，“`[PATCH]`”将从提交描述中删除。

如果在命令行上没有传递修订版本，并且标准输入不是终端或没有当前的分支，则 `git shortlog` 将输出从标准输入读取的日志的摘要，而不引用当前存储库。

## 场景假设

一个开发小组有 10 个程序员，他们用 `Git` 做版本控制，某一天程序员 A `push` 了当天的几个 `commit` 之后，突然在想“我在这个项目到底一共进行过多少次 `commit`？谁比我 `commit` 更多？多多少？谁是组里面进行最多 `commit` 的？谁是最少的？”

`Git` 非常人性化地支持这样一个命令：

```
$ git shortlog
```

这个命令会返回这个 `git repository` 底下每个用户进行 `commit` 的次数，以及每次 `commit` 的注释。

`-s` 参数省略每次 `commit` 的注释，仅仅返回一个简单的统计。

`-n` 参数按照 `commit` 数量从多到少的顺利对用户进行排序

## 示例

以下是一些示例 -

```
$ git shortlog -s -n
135  Tom Preston-Werner
15   Jack Danger Canty
10   Chris Van Pelt
7    Mark Reid
6    remi
3    Mikael Lind
3    Toby DiPasquale
2    Aristotle Pagaltzis
2    Basil Shkara
```

```
2 John Reilly
2 PJ Hyett
1 Marc Chung
1 Nick Gerakines
1 Nick Quaranto
1 Tom Kirchner
```

比如想要知道一个开源项目(例如 Graphiti )的 commit 统计

```
$ git shortlog -s -n
16 maxsu
5 your_name
2 minsu
1 Maxsu
```

上面的结果表明, **maxsu** 确实是这个开源项目的主要开发者。