

`git diff` 命令用于显示提交和工作树等之间的更改。此命令比较的是工作目录中当前文件和暂存区域快照之间的差异,也就是修改之后还没有暂存起来的变化内容。

简介

```
git diff [options] [<commit>] [--] [<path>...]  
git diff [options] --cached [<commit>] [--] [<path>...]  
git diff [options] <commit> <commit> [--] [<path>...]  
git diff [options] <blob> <blob>  
git diff [options] [--no-index] [--] <path> <path>
```

描述

在工作树和索引或树之间显示更改, 索引和树之间的更改, 两个树之间的更改, 两个 blob 对象之间的更改或两个文件在磁盘上的更改。

为了防止异常情况发生, 请注意, 上述描述中的所有 `<commit>` 除了使用“..”符号的最后两种形式之外, 都可以是任何 `<tree>`。

示例

以下是一些示例 -

```
git diff <file> # 比较当前文件和暂存区文件差异 git diff
```

```
git diff <id1><id1><id2> # 比较两次提交之间的差异
```

```
git diff <branch1> <branch2> # 在两个分支之间比较
```

```
git diff --staged # 比较暂存区和版本库差异
```

```
git diff --cached # 比较暂存区和版本库差异
```

```
git diff --stat # 仅仅比较统计信息
```

1. 检查工作树的几种方式

```
$ git diff # (1)
```

```
$ git diff --cached # (2)
```

```
$ git diff HEAD # (3)
```

1. 工作树中的更改尚未分段进行下一次提交。
2. 索引和最后一次提交之间的变化; 查看已经 `git add`, 但没有 `git commit` 的改动。

3. 自上次提交以来工作树中的更改；如果运行“`git commit -a`”，查看将会提交什么。
查看尚未暂存的文件更新了哪些部分，不加参数直接输入 -

```
$ git diff
```

此命令比较的是工作目录(Working tree)和暂存区域快照(index)之间的差异
也就是修改之后还没有暂存起来的变化内容。

查看已经暂存起来的文件(staged)和上次提交时的快照之间(HEAD)的差异 -

```
$ git diff --cached
```

```
$ git diff --staged
```

显示的是下一次提交时会提交到 HEAD 的内容(不带 `-a` 情况下)

显示工作版本(Working tree)和 HEAD 的差别

```
$ git diff HEAD
```

直接将两个分支上最新的提交做 diff

```
$ git diff topic master
```

```
$ #或
```

```
$ git diff topic...master
```

输出自 `topic` 和 `master` 分别开发以来，`master` 分支上的变更。

```
$ git diff topic...master
```

查看简单的 diff 结果，可以加上 `--stat` 参数

```
$ git diff --stat
```

查看当前目录和另外一个分支(`test`)的差别

```
$ git diff test
```

显示当前目录和另一个叫 `'test'` 分支的差别

```
$ git diff HEAD -- ./lib
```

显示当前目录下的 `lib` 目录和上次提交之间的差别(更准确的说是在当前分支下)
比较上次提交和上上次提交

```
$ git diff HEAD^ HEAD
```

比较两个历史版本之间的差异

```
$ git diff SHA1 SHA2
```

提示：SHA1，SHA2 是类似 COMMIT ID 的 32 位长度的值。