

○ 前回コミット時の状態に戻す

```
$ git reset --hard HEAD
```

```
$ git log
```

```
$ git reset --hard ハッシュ値
```

○ 前回コミットに上書き(コメントだけ書き換え)

```
$ git commit --amend
```

○ 過去のコミットの内容を修正

```
$ git rebase -i <commit>
```

```
$ git commit --amend
```

```
$ git rebase --continue
```

○ rebase を途中で中止したい

```
$ git rebase --abort
```

○ 戻りすぎた時(rebase を無かったことにする)

```
$ git reset --hard HEAD^^ # HEAD^と指定するつもりが間違えた!
```

```
$ git reflog
```

```
f5cb888 HEAD@{0}: head^^: updating HEAD
```

```
b0b8073 HEAD@{1}: merge @{-1}: Merge made by the 'recursive' strategy.
```

```
fe3972d HEAD@{2}: checkout: moving from fix/some-bug to master
```

```
$ git reset --hard HEAD@{1} # reset --hard HEAD^^する前に戻れる
```

○ 直前の commit メッセージの書き換え

```
$ git commit -amend -m “新しいメッセージ”
```

○ **現在のワークツリーを一時的に保存する**

```
$ git stash (save)
```

```
$ git stash save "message"  <- message 付き
```

○ **stash に保存されている状態の一覧を見る**

```
$ git stash list
```

○ **stash に保存されている状態に戻し、stash から削除する**

```
$ git stash pop
```

```
$ git stash pop stash@{1}
```

○ **stash から削除する**

```
$ git statsh drop
```

○ **差分のファイル名だけを表示**

```
$ git diff --name-only HEAD^
```

○ **特定のコメントを含むコミットを探す**

```
$ git log -grep "<pattern>"
```

○ **Git の色を変える**

```
$ git config --global color.ui true
```