

[Главная](#) > [Блог](#) > Git как вернуться (откатиться) к коммиту.

GIT КАК ВЕРНУТЬСЯ (ОТКАТИТЬСЯ) К КОММИТУ.

11 Жовтня 2016



Определим, что именно необходимо:

1. Посмотреть состояние на определенном моменте.
2. Откатиться на какую-либо ветку, индекс, рабочую область.

Выбираем нужный коммит, и в консоли (и сразу откатываемся на нем):

```
git checkout ID_коммита
```

Чтобы посмотреть, состояние проекта на определенном моменте, для этого есть команде git log. Один из способов просмотреть лог (последние 10 коммитов):

```
git log --oneline - 10
```

Теперь чтобы вернуться на последний актуальный коммит:

```
git checkout master
```

Чтобы переключиться на определенный коммит, и создать отдельно ветку – набираем в консоли:

```
git checkout -b имя_новой_ветки ID_коммита
```

2. Теперь рассмотрим случай, когда возвращаться не нужно, а требуется только удалить все свои незакомиченные изменения (которые ещё не индексировались командой git add):

```
git stash save --keep-index
```

при этом появится ещё одно состояние проекта (карман).

Удалить все индексированные и неиндексированные изменения:

```
git stash save
```

Вернуть обратно:

```
git stash apply
```

Если «карман» не нужен, его можно удалить:

```
git stash drop
```

И наконец, если необходимо переключиться на предыдущий коммит, отменить все последующие и продолжить разработку с него, лучший способ – это использовать `git revert`.

Эта команда создает новые коммиты на каждый последующий от текущего состояния.

Синтаксис следующий:

```
git revert  
IDs_коммитов
```

или

```
git revert  
ID_коммита..ID_коммита
```

Замет нужно всё это дело подтвердить коммитом:

```
git commit -m 'что и  
зачем'
```

3. Допустим запустили с помощью `git push --force`

чтобы только ваши изменения были в репозитории (на другие коммиты нам начхать). Крайне нежелательная команда, но что поделаешь...

Протолкнули со свистом. А что же делать с тестовым сайтом? Как на нем теперь вернуться в рабочее состояние?

Используем следующий набор команд:

```
git fetch --all  
git reset --hard origin/master  
git pull origin master
```

А чтобы не фиксировать изменения прав доступа:

```
git config core.filemode  
false
```

И глобально:

```
git config --global core.filemode  
false
```

Теги: 

www.open2web.com.ua

+38 (095) 433-78-87

pavel@open2web.com.ua

skype: open2web

© Павел Викторович, 2006—2023

[rss](#)