

# Squash & Merge в Git или сохранение истории коммитов без промежуточных записей

**Squash & Merge** — это способ слияния изменений, при котором несколько коммитов объединяются (squash — "сжимать") в один перед вливанием в основную ветку. Этот метод позволяет сохранить чистую историю коммитов без лишних промежуточных записей.

---

## Как работает Squash & Merge?

Допустим, в ветке `feature/new-feature` есть несколько коммитов:

```
commit a1b2c3d - Fix bug in query handler
commit d4e5f6g - Add logging
commit h7i8j9k - Implement new feature
commit l0m1n2o - Fix typo in previous commit
```

При обычном `merge` они просто добавятся в основную ветку (`main`).

Но если использовать **Squash & Merge**, все эти коммиты объединяются в один:

```
commit p9q8r7s - Implement new feature and fixes
```

Таким образом, история остается чистой.

---

## Когда использовать Squash & Merge?

❑ **Когда нужно объединить "мусорные" коммиты** (например, тестовые исправления, исправления опечаток).

❑ **Когда код писался в несколько итераций**, но результат — одно логическое изменение.

❑ **При работе в feature-ветках**, чтобы в `main` попадали только готовые коммиты.

❑ **Не подходит для веток, где важна детальная история коммитов** (например, в `develop` или `release`).

---

# Как выполнить Squash & Merge?

## 1. Вручную через `git rebase -i`

Переключаемся на нужную ветку:

```
git checkout feature/new-feature
```

Запускаем интерактивный rebase:

```
git rebase -i main
```

Git покажет список коммитов, например:

```
pick a1b2c3d Fix bug in query handler
pick d4e5f6g Add logging
pick h7i8j9k Implement new feature
pick l0m1n2o Fix typo in previous commit
```

Заменяем `pick` на `squash (s)` у всех коммитов, кроме первого:

```
pick a1b2c3d Fix bug in query handler
squash d4e5f6g Add logging
squash h7i8j9k Implement new feature
squash l0m1n2o Fix typo in previous commit
```

После сохранения Git предложит изменить сообщение коммита, оставляем итоговое.

Затем пушим изменения (если уже был push в удаленный репозиторий, потребуется `--force`):

```
git push origin feature/new-feature --force
```

---

## 2. Через GitHub/GitLab UI

При слиянии Pull Request (Merge Request) можно выбрать опцию **Squash & Merge**:

- В **GitHub**: Merge pull request -> Squash & Merge
  - В **GitLab**: Merge Request -> Squash commits
- 

## Вывод

Squash & Merge удобен для поддержания чистоты истории коммитов. Важно применять его осознанно: если детальная история не нужна, squash поможет убрать ненужные коммиты и оставить только финальные изменения.

---

Revision #1

Created 5 February 2025 06:56:46 by Admin

Updated 5 February 2025 06:58:15 by Admin