

Создаём pull request

В процессе командной работы следует внимательно следить за изменениями в файлах. Нельзя просто внести правки в своей ветке и сразу залить её в основную. Сначала ваши коллеги должны убедиться, что предложенные вами изменения логичны и эффективны.

Для этого используют механизм **pull request** (англ. «запрос на изменения»; буквально: «запрос на подтягивание»). В обиходе его обычно так и называют — «пул-реквест», или ещё короче — ПР или PR. Алгоритм такой:

1. Вы трудитесь над задачей в своей ветке — например, пишете код новой функциональности.
2. Вы заканчиваете работу, а затем создаёте пул-реквест.
3. Ваши коллеги проверяют, что код выглядит аккуратно и лаконично, а программа работает корректно; также оставляют комментарии. Этот процесс называют **code review** (англ. «рассмотрение кода»), или просто **ревью**.
4. После финального согласования вы заливаете свою ветку в основную.

Из чего состоит pull request и чем он может обернуться

У каждого пул-реквеста есть:

- **Название** — краткое описание предлагаемых изменений. Например: Адаптивный заголовок сайта, Замена альбома на галерею и так далее.
- **Описание** — развёрнутое описание изменений. Это поле заполнять необязательно, но желательно.
- **Исходная ветка** — та, в которой вы работали. Например, `feature/merge-request`.
- **Целевая ветка** — основная ветка проекта, в которую вы хотите внести изменения.

Где заполнять эти параметры, вы увидите далее — в ходе тренировки. Также у каждого пул-реквеста может быть два исхода:

- **merge** (англ. «соединить») — предлагаемые изменения приняты; код вливается в целевую ветку; пул-реквест закрывается.
- **close** (англ. «закрыть») — пул-реквест закрывается без слияния изменений.

Делаем pull request

После того как новая ветка «запущена» в удалённый репозиторий, можно делать пул-реквест. Существует два способа.

Первый способ. При создании новой ветки в удалённом репозитории Git распечатает сообщение. Оно включает ссылку на создание пул-реквеста.

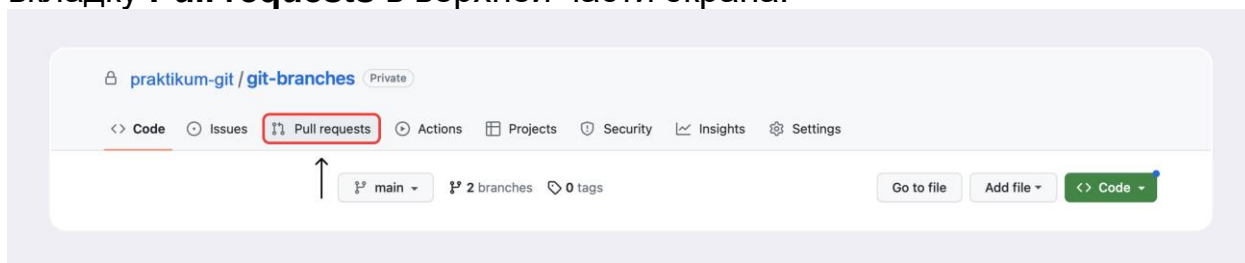
remote: Create a pull request for 'feat/diff' on GitHub by visiting:

remote: https://github.com/%BAШ_AKKAУHT%/git-branches/pull/new/feature/merge-request

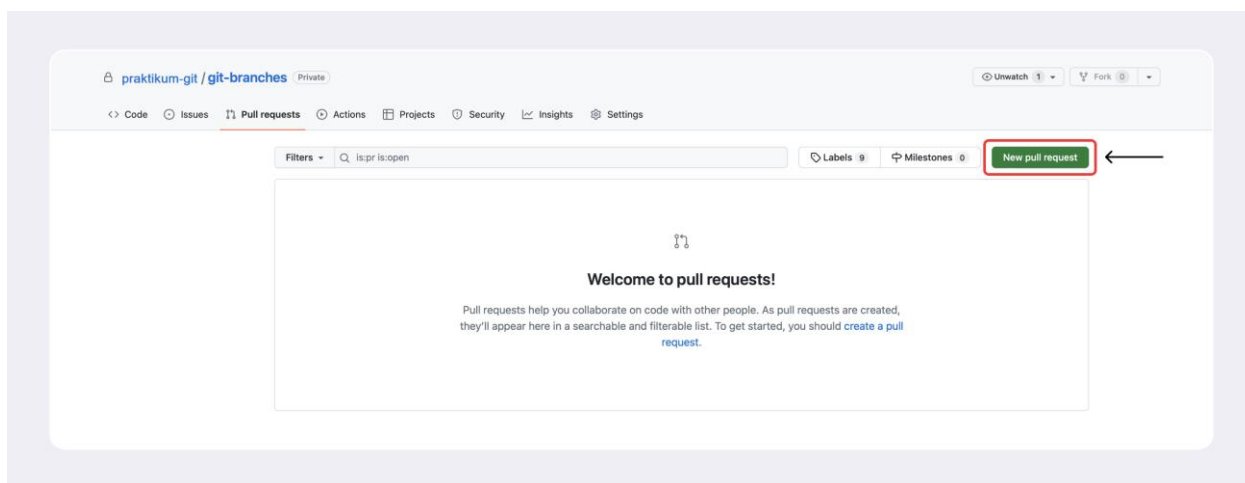
Останется только скопировать её в адресную строку браузера, заполнить необходимые поля и нажать **Create pull request** (англ. «создать запрос на изменения»). Многие терминалы также позволяют кликнуть на эту ссылку — напрямую или через комбинацию `Cmd / Ctrl + клик`.

Однако такая ссылка появляется только один раз и для новых веток, поэтому иногда приходится идти более сложным путём.

Второй способ. Чтобы создать пул-реквест для любой существующей ветки на GitHub, перейдите на страницу репозитория, а затем выберите вкладку **Pull requests** в верхней части экрана.

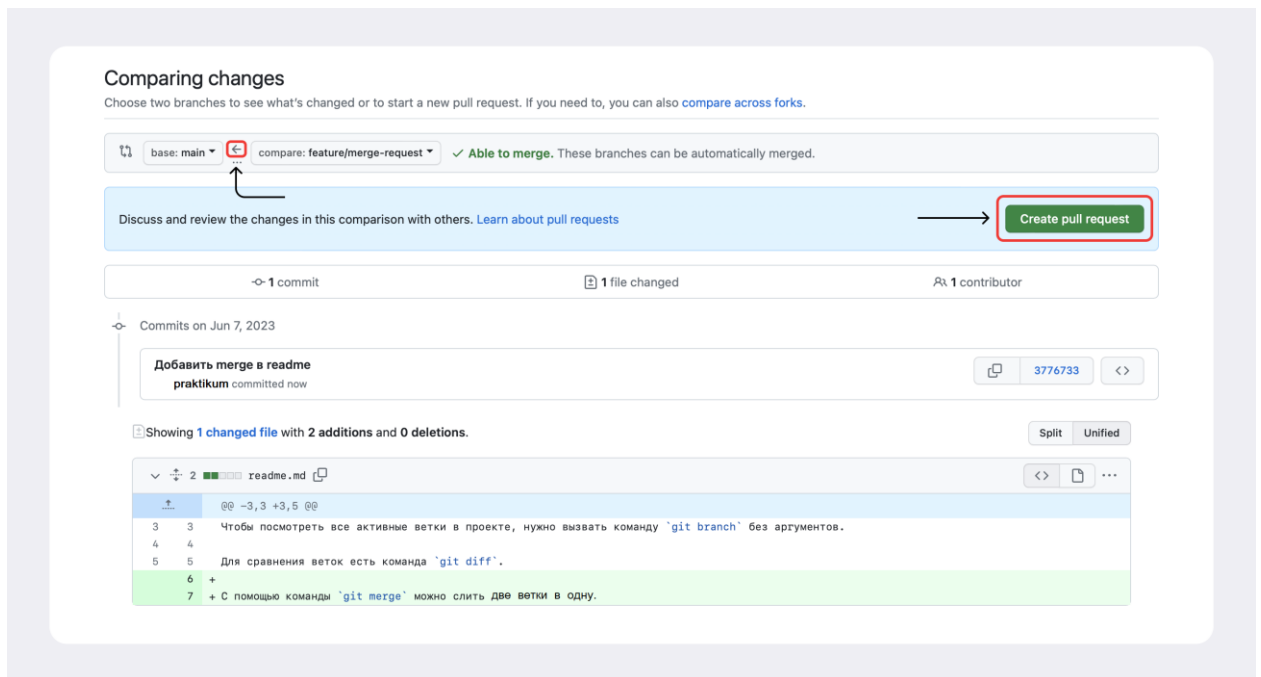


Нажмите на зелёную кнопку **New pull request** (англ. «новый запрос на изменения»).



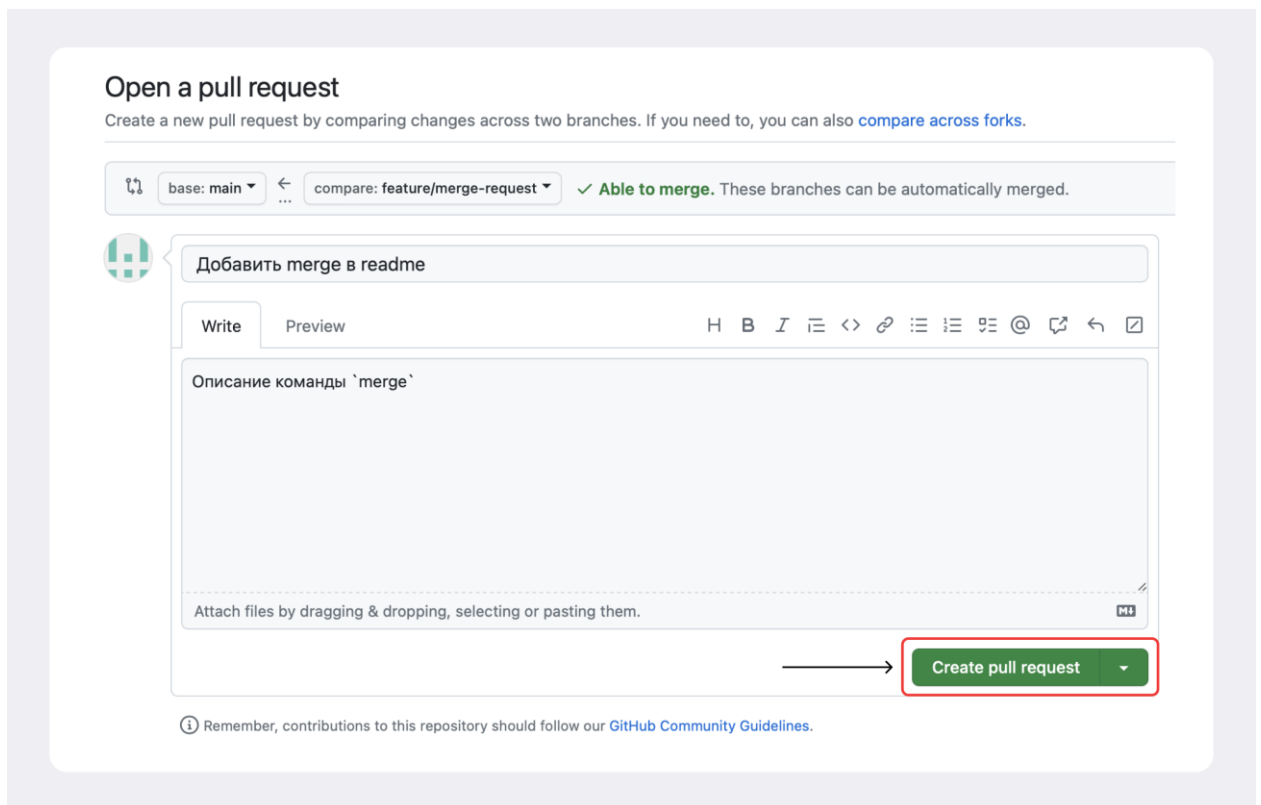
Выберите названия веток: ветка «откуда» (из которой будет происходить пул-реквест) и ветка «куда» (в которую он будет осуществлён). В нашем случае изменения должны попасть из ветки `feature/merge-request` в ветку `main`. На направление изменений указывает стрелка.

В окне ниже отобразится несколько коммитов, а также их изменения. Нажмите на кнопку **Create pull request**.



💡 Вы можете создать пул-реквест из любой ветки, которая отличается от `main`. У многих команд разработки две основные ветки. В ветке `dev` ведётся вся активная работа, а в ветке `main` хранится основная рабочая версия. При этом ветка `dev` время от времени сливается с веткой `main`.

Заполните поля с названием и описанием пул-реквеста. Нажмите **Create pull request**.



Готово: пул-реквест создан! Теперь вы или ваши коллеги можете перейти на вкладку **Files changed** (англ. «изменённые файлы»), чтобы оставить свои комментарии — провести ревью.

По окончании ревью можно посмотреть комментарии и обсудить изменения на вкладке **Pull requests**. Также вы можете в любой момент добавить дополнительные коммиты в ветку — они автоматически попадут в открытый пул-реквест после пуша.

The screenshot shows a GitHub pull request titled "Добавить merge в readme #2". At the top, it indicates that "praktikum-git" wants to merge 1 commit into the `main` branch from the `feature/merge-request` branch. Below this, there are tabs for "Conversation" (0), "Commits" (1), "Checks" (0), and "Files changed" (1). The "Conversation" tab is active, showing a comment from "praktikum-git" with the text "Описание команды merge". Below the comment is a commit diff for "Добавить merge в readme" with a commit hash of 3776733. A message below the diff says: "Add more commits by pushing to the `feature/merge-request` branch on `praktikum-git /git-branches`." At the bottom, there is a section for branch protection rules. It includes a rule "Require approval from specific reviewers before merging" with an "Add rule" button, a note "Continuous integration has not been set up" with links to "GitHub Actions" and "several other apps", and a green checkmark indicating "This branch has no conflicts with the base branch" with the note "Merging can be performed automatically." At the very bottom, there is a green button labeled "Merge pull request" and a link to "open this in GitHub Desktop" or view "command line instructions".

Осталось только нажать на кнопку **Merge pull request** (англ. «принять запрос на изменения») — это действие объединит ветку с вашими изменениями и ветку `main`.

Добавить merge в readme #2

Merged

praktikum-git merged 1 commit into `main` from `feature/merge-request`  now

Conversation 0

Commits 1

Checks 0

Files changed 1



praktikum-git commented 1 minute ago

...

Описание команды `merge`



Добавить merge в readme

3776733



praktikum-git merged commit `4b5ebec` into `main` now

Revert



Pull request successfully merged and closed

Delete branch

You're all set—the `feature/merge-r-` branch can be safely deleted.

Ура: теперь ваши изменения стали частью основной ветки!