**Всем привет**

Следуйте этому упражнению Hello World, чтобы начать работу с GitHub

## **Знакомство**

**GitHub** - это платформа для размещения кода для управления версиями и совместной работы. Это позволяет вам и другим работать вместе над проектами из любого места.

Этот учебник научит вас основам GitHub, таким как **репозитории (repositories)**, **ветви (branches), фиксации (commits), и запросы на извлечение (pull requests)**. Вы создадите свой собственный репозиторий Hello World и изучите рабочий процесс запроса на вытягивание GitHub, популярный способ создания и просмотра кода.

**В этом кратком руководстве вы сможете:**

* Создание и использование репозитория
* Запуск нового филиала и управление им
* Внесение изменений в файл и отправка их на GitHub в качестве фиксаций
* Открытие и объединение запроса на извлечение

Для работы с этим учебником вам потребуется [учетная запись GitHub](http://github.com/) и доступ в Интернет. Вам не нужно знать, как кодировать, использовать командную строку или устанавливать Git (программное обеспечение для управления версиями, на котором построен GitHub). Если у вас есть вопрос о каком-либо из выражений, используемых в этом руководстве, перейдите к [глоссарию](https://docs.github.com/en/get-started/quickstart/github-glossary), чтобы узнать больше о нашей терминологии.

## **Создание репозитория**

Репозиторий обычно используется для организации одного проекта. Репозитории могут содержать папки и файлы, изображения, видео, электронные таблицы и наборы данных - все, что нужно вашему проекту. Часто репозитории включают файл README, файл с информацией о вашем проекте. Файлы README написаны на языке Markdown в виде обычного текста. Вы можете использовать эту [шпаргалку](https://www.markdownguide.org/cheat-sheet/), чтобы начать работу с синтаксисом Markdown. GitHub позволяет добавлять файл README одновременно с созданием нового репозитория. GitHub также предлагает другие распространенные параметры, такие как файл лицензии, но теперь вам не нужно выбирать ни один из них.

Ваш репозиторий может быть местом, где вы храните идеи, ресурсы или даже делитесь и обсуждаете вещи с другими.hello-world

# IAPM

## Techmind

### Home work 4

\*\*Theme\*\*VCS

> To Git or not to Git

1. Зарегистрироваться на GitHub

2. Создать проект

3. Создать README файл через интерфейс GitHub

4. Загрузить проект локально через аналог SourceTree (clone)

«На этом шаге может спросить логин и пароль от GitHub»

5. Добавить какие-то файлы (любой формат)

6. Закоммитить в мастер:

7. Запушить в мастер:

a. Make sure you see “1” on push button

8. Убедиться, что изменения появились в GitHub

9. Создать ветку dev (через GitHub)

10. Сделать fetch через аналог SourceTree

11. Найти новую ветку dev в вкладке REMOTES - origin

12. Сделать ей checkout (правый клик)

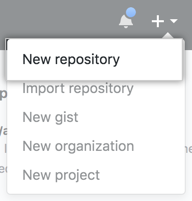
13. Изменить что-то в файлах

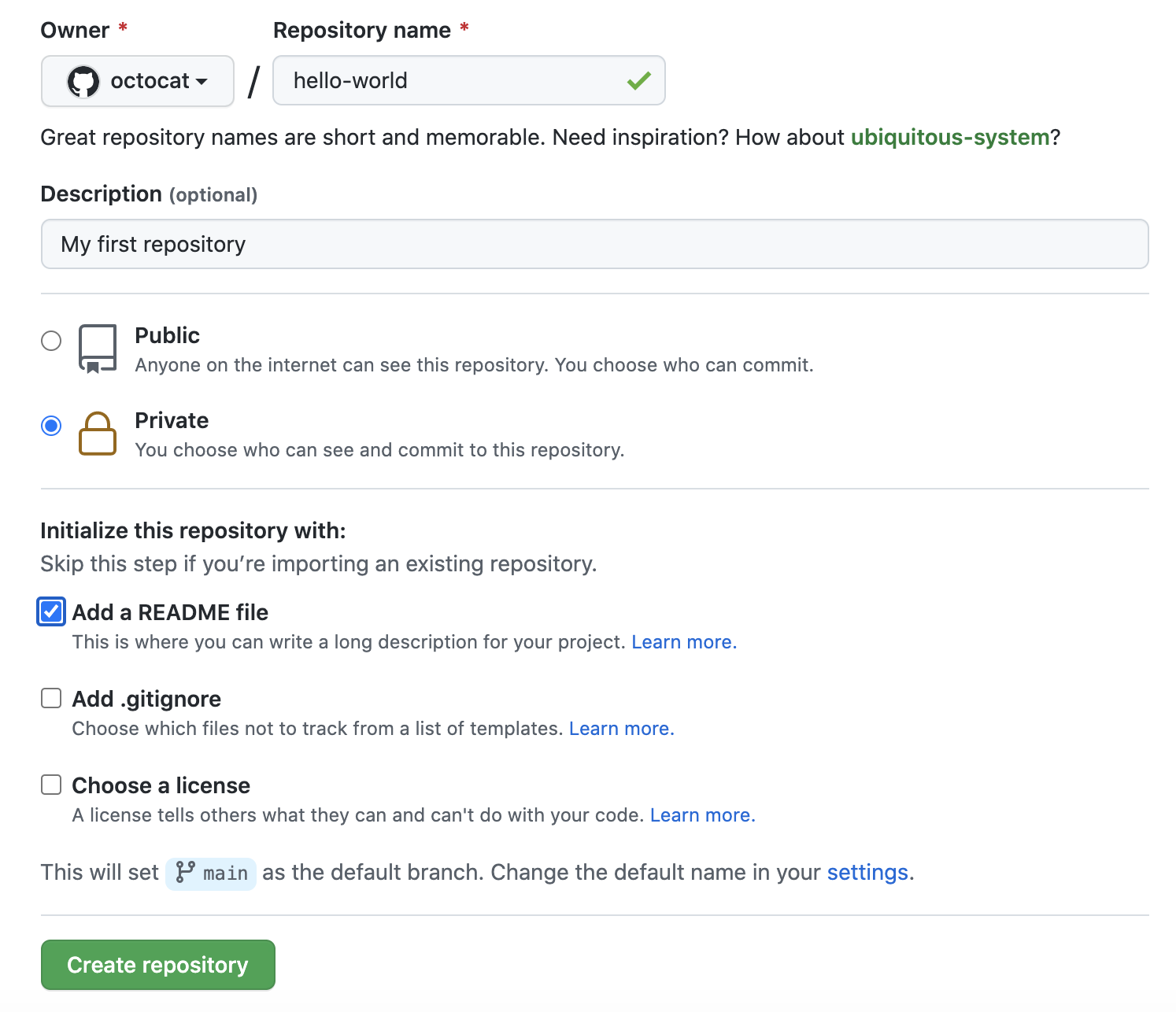
14. Закоммитить в dev

15. Запушить в dev

16. Зайти в GitHub - pull requests - create new one (from dev to master

branch)

1. В правом верхнем углу любой страницы используйте раскрывающееся меню и выберите **Новый репозиторий**. 
2. В поле **Имя репозитория** введите .hello-world
3. В поле **Описание** напишите краткое описание.
4. Выберите **Добавить файл README**.
5. Выберите, будет ли ваш репозиторий **общедоступным** или **частным**.
6. Щелкните **Создать репозиторий**.



## **Создание филиала**

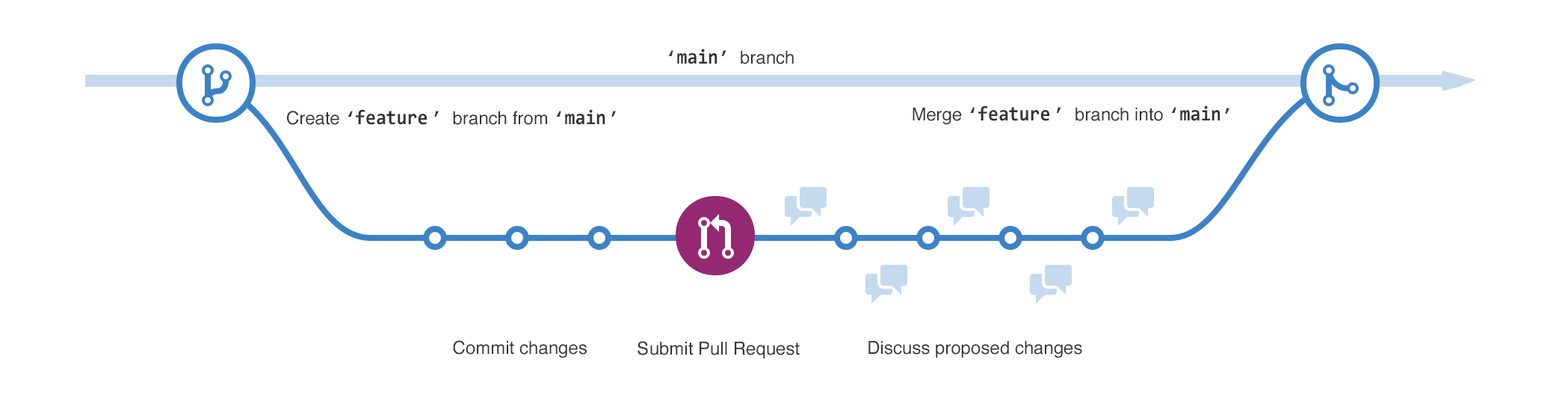
Ветвление позволяет иметь разные версии репозитория одновременно.

По умолчанию репозиторий имеет одну ветвь с именем, которая считается окончательной ветвью. Вы можете создать дополнительные ветви в своем репозитории. Ветви можно использовать для одновременного использования разных версий проекта. Это полезно, если вы хотите добавить новые функции в проект без изменения основного исходного кода. Работа, проделанная над различными ветвями, не будет отображаться на основной ветви, пока вы не объедините ее, о чем мы расскажем далее в этом руководстве. Ветви можно использовать для экспериментов и внесения изменений перед их фиксацией в .mainmainmain

Когда вы создаете ветвь от ветви, вы делаете копию или снимок того, как это было в тот момент времени. Если кто-то другой внес изменения в ветвь во время работы над веткой, вы можете получить эти обновления.mainmainmain

На этой схеме показано:

* Филиалmain
* Новый филиал под названием feature
* Путешествие, которое требуется, прежде чем он будет объединен в featuremain



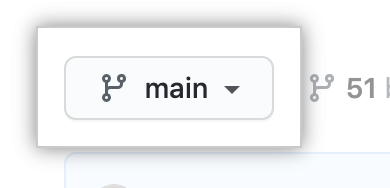
Вы когда-нибудь сохраняли разные версии файла? Что-то вроде:

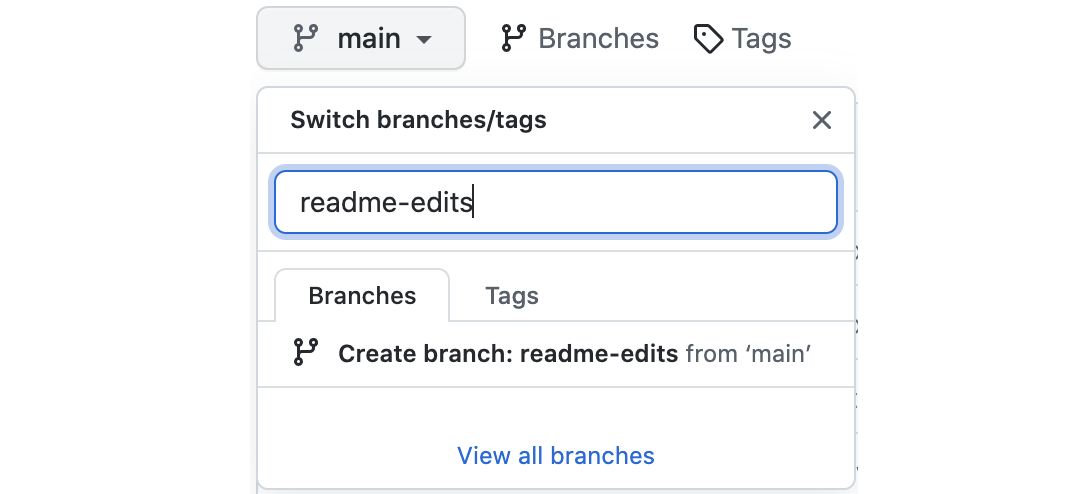
* story.txt
* story-edit.txt
* story-edit-reviewed.txt

Ветви достигают аналогичных целей в репозиториях GitHub.

Здесь, на GitHub, наши разработчики, писатели и дизайнеры используют ветви для хранения исправлений ошибок и работы с функциями отдельно от нашей (производственной) ветви. Когда изменение готово, они объединяют свою ветвь в .mainmain

### Создание филиала

1. Перейдите на вкладку **Код** репозитория.hello-world
2. Щелкните раскрывающийся список в верхней части списка файлов с **надписью main**.
3. Введите имя ветви , в текстовое поле.readme-edits
4. Нажмите **Создать ветвь: readme-edits из main**.



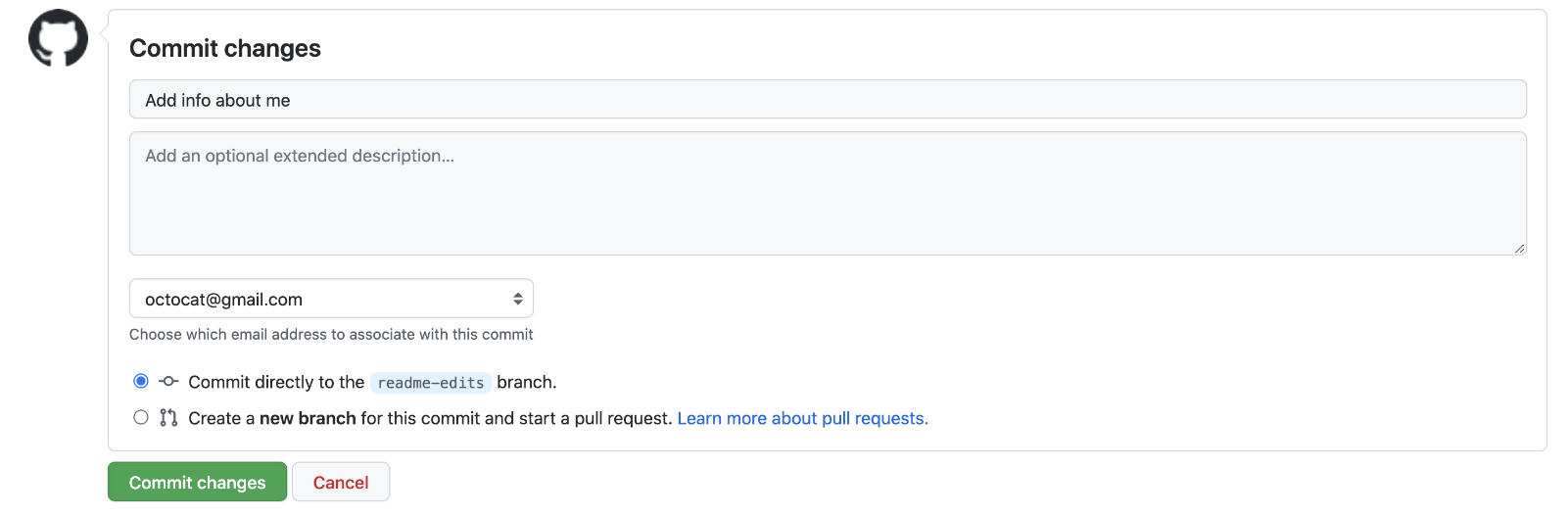
Теперь у вас есть две ветви, и . Сейчас они выглядят точно так же. Далее вы добавите изменения в новую ветвь.mainreadme-edits

## Внесение и фиксация изменений

Когда вы создали новую ветвь на предыдущем шаге, GitHub перевел вас на кодовую страницу для вашей новой ветви, которая является копией .readme-editsmain

Вы можете вносить и сохранять изменения в файлах в репозитории. На GitHub сохраненные изменения называются фиксациями. Каждый коммит имеет связанное сообщение фиксации, которое представляет собой описание, объясняющее, почему было внесено конкретное изменение. Сообщения о фиксации фиксируют историю ваших изменений, чтобы другие участники могли понять, что вы сделали и почему.

1. Под созданной ветвью щелкните файл README.md.readme-edits
2. Нажмите, чтобы отредактировать файл.
3. В редакторе напишите немного о себе. Попробуйте использовать различные элементы Markdown.
4. В поле **Фиксация изменений** напишите сообщение о фиксации, описывающее изменения.
5. Щелкните **Зафиксировать изменения**.



Эти изменения будут внесены только в файл README в вашей ветви, поэтому теперь эта ветвь содержит содержимое, отличное от .readme-editsmain

## **Открытие запроса на извлечение**

Теперь, когда у вас есть изменения в ветви, вы можете открыть запрос на извлечение.main

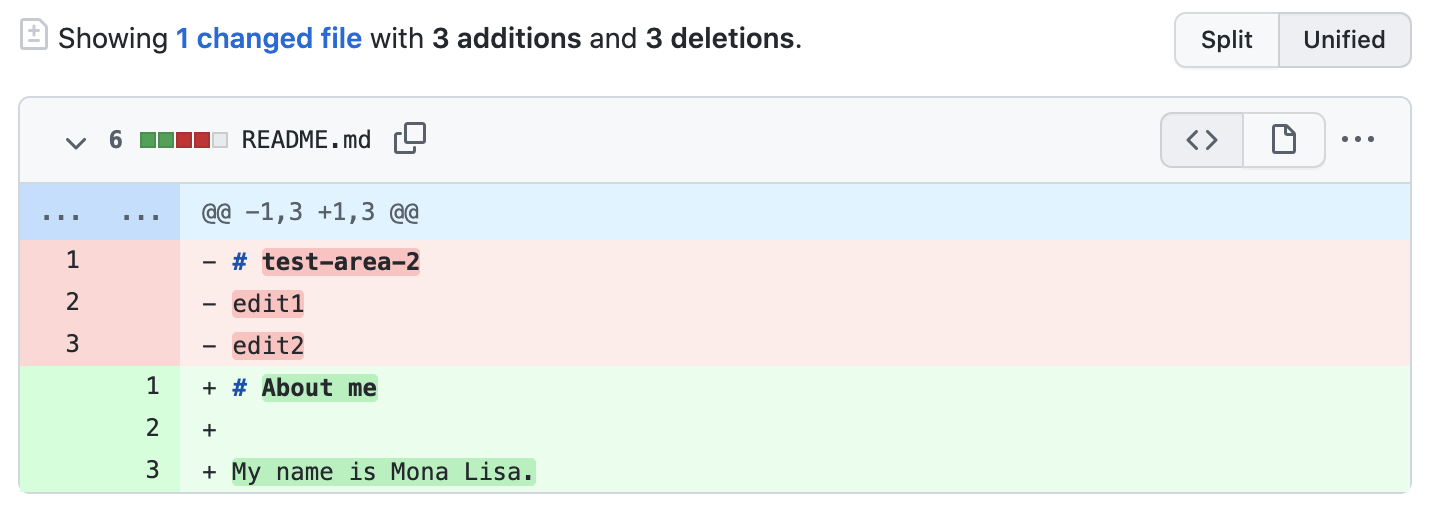
Pull-запросы являются сердцем совместной работы на GitHub. Когда вы открываете запрос на вытягивание, вы предлагаете свои изменения и просите, чтобы кто-то рассмотрел и извлек ваш вклад и объединил их в свою ветвь. Запросы на извлечение показывают различия или различия содержимого из обеих ветвей. Изменения, сложения и вычитания показаны разными цветами.

Как только вы сделаете фиксацию, вы можете открыть запрос на извлечение и начать обсуждение еще до завершения кода.

Используя функцию GitHub в сообщении запроса на вытягивание, вы можете запросить обратную связь от определенных людей или команд, независимо от того, находятся ли они внизу зала или в 10 часовых поясах.@mention

Вы даже можете открыть pull-запросы в своем собственном репозитории и объединить их самостоятельно. Это отличный способ изучить поток GitHub перед работой над более крупными проектами.

1. Перейдите на вкладку **Запросы на извлечение** репозитория.hello-world
2. Нажмите **Новый запрос на извлечение**
3. В поле **Сравнения примеров** выберите созданную ветвь , для сравнения с (оригинал).readme-editsmain
4. Просмотрите свои изменения в диффах на странице Сравнение, убедитесь, что это то, что вы хотите отправить.



1. Щелкните **Создать запрос на вытягивание**.
2. Присвойте запросу на вытягивание заголовок и напишите краткое описание изменений. Вы можете включать смайлики и перетаскивать изображения и GIF-файлы.
3. При необходимости справа от заголовка и описания щелкните рядом с **пунктом Рецензенты**. **Получатели**, **метки**, **проекты** или **вехи**, чтобы добавить любой из этих параметров в запрос на получение. Вам пока не нужно ничего добавлять, но эти параметры предлагают различные способы совместной работы с помощью запросов на вытягивание. Дополнительные сведения см. в разделе «[О запросах на вытягивание](https://docs.github.com/en/pull-requests/collaborating-with-pull-requests/proposing-changes-to-your-work-with-pull-requests/about-pull-requests)».
4. Щелкните **Создать запрос на вытягивание**.

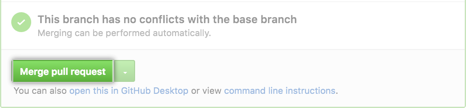
Теперь ваши коллеги могут просматривать ваши изменения и вносить предложения.

## **Объединение запроса на извлечение**

На этом заключительном шаге вы объедините свою ветвь в ветвь. После объединения запроса на извлечение изменения в ветви будут включены в .readme-editsmainreadme-editsmain

Иногда запрос на извлечение может внести изменения в код, которые конфликтуют с существующим кодом на . Если возникнут какие-либо конфликты, GitHub предупредит вас о конфликтующем коде и предотвратит слияние до тех пор, пока конфликты не будут разрешены. Вы можете сделать фиксацию, которая разрешает конфликты, или использовать комментарии в запросе на извлечение для обсуждения конфликтов с членами вашей команды.main

В этом пошаговом руководстве не должно возникнуть конфликтов, поэтому вы готовы объединить ветвь в основную ветвь.

1. Щелкните **Объединить запрос на извлечение**, чтобы объединить изменения в . main
2. Нажмите **кнопку Подтвердить слияние**. Вы получите сообщение о том, что запрос был успешно объединен и запрос закрыт.
3. Щелкните **Удалить ветвь**. Теперь, когда запрос на извлечение объединен и изменения включены, можно безопасно удалить ветвь. Если вы хотите внести дополнительные изменения в свой проект, вы всегда можете создать новую ветвь и повторить этот процесс.mainreadme-edits

## **Дальнейшие шаги**

Пройдя этот учебник, вы научились создавать проект и делать запрос на извлечение на GitHub.

Вот что вы достигли в этом учебнике:

* Создан репозиторий с открытым исходным кодом
* Запуск и управление новым филиалом
* Изменен файл и зафиксированы эти изменения на GitHub
* Открытие и объединение запроса на извлечение

Взгляните на свой профиль GitHub, и вы увидите, что ваша работа отражена на графике вашего вклада.

Дополнительные сведения о возможностях ветвей и запросов на извлечение см. в разделе «[Поток GitHub](https://docs.github.com/en/get-started/quickstart/github-flow)». Дополнительные сведения о начале работы с GitHub см. в других руководствах краткого [руководства по началу работы](https://docs.github.com/en/get-started/quickstart).