**Шаг 1. Создайте репозиторий**

Обычно **репозиторий** используется для организации единого проекта. Хранилища могут содержать папки и файлы, изображения, видео, электронные таблицы и наборы данных - все, что требуется вашему проекту. Мы рекомендуем включать *README* или файл с информацией о вашем проекте. GitHub упрощает добавление в одно и то же время, когда вы создаете новый репозиторий. *Он также предлагает другие общие параметры, такие как файл лицензии.*

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

**Чтобы создать новый репозиторий**

1. В правом верхнем углу рядом с вашим аватаром или идентификатором нажмите а затем выберите **Новый репозиторий** .
2. Назовите свой репозиторий hello-world.
3. Напишите краткое описание.
4. Выберите « **Инициализировать» этот репозиторий с помощью README** .



Нажмите **Создать репозиторий** .: Тада:

**Шаг 2. Создание ветви**

**Ветвление** - это способ работы с различными версиями репозитория за один раз.

По умолчанию ваш репозиторий имеет одну ветвь с именем, masterкоторая считается окончательной ветвью. Мы используем ветви для экспериментов и редактирования, прежде чем их выполнять master.

Когда вы создаете ветку с masterветки, вы делаете копию или моментальный снимок, masterкак это было в тот момент. Если кто-то другой вносил изменения в masterветку во время работы в вашем филиале, вы можете использовать эти обновления.

Эта диаграмма показывает:

* masterфилиал
* Новая ветвь называется feature(потому что мы делаем «функцию» на этой ветке)
* Путешествие, которое featureпроисходит до того, как оно слилось вmaster



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

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

Филиалы выполняют аналогичные цели в репозиториях GitHub.

Здесь, в GitHub, наши разработчики, писатели и дизайнеры используют филиалы для исправления ошибок и работы функций отдельно от нашей master(производственной) ветки. Когда изменения готовы, они объединяют свою ветвь master.

**Чтобы создать новую ветку**

1. Перейдите в новый репозиторий hello-world.
2. Нажмите на раскрывающийся список в верхней части списка файлов, в котором говорится **: branch: master** .
3. Введите имя readme-editsфилиала в текстовое поле новой ветви.
4. Выберите синюю **кнопку Создать ветку** или нажмите «Ввод» на клавиатуре.



Теперь у вас две ветви, masterи readme-edits. Они выглядят точно так же, но ненадолго! Затем мы добавим наши изменения в новую ветку.

**Шаг 3. Внести изменения и внести изменения**

Браво! Теперь вы находитесь в представлении кода для своей readme-editsветки, которая является копией master. Сделаем некоторые изменения.

На GitHub сохраненные изменения называются *коммитами* . Каждая фиксация имеет связанное *сообщение фиксации* , которое является описанием, объясняющим, почему было сделано определенное изменение. Фиксированные сообщения фиксируют историю ваших изменений, поэтому другие участники могут понять, что вы сделали и почему.

**Сделать и зафиксировать изменения**

1. Щелкните README.mdфайл.
2. Нажмите  значок карандаша в правом верхнем углу просмотра файла для редактирования.
3. В редакторе немного напишите о себе.
4. Напишите сообщение фиксации, которое описывает ваши изменения.
5. Нажмите кнопку « **Зафиксировать изменения»** .



Эти изменения будут внесены только в файл README в вашей readme-editsветке, поэтому теперь эта ветка содержит контент, который отличается от master.

**Шаг 4. Откройте запрос Pull**

Хорошие изменения! Теперь, когда у вас есть изменения в ветке master, вы можете открыть *запрос* на *перенос* .

Pull Requests - это сердце сотрудничества GitHub. Когда вы открываете *запрос* на *вытягивание* , вы предлагаете свои изменения и запрашиваете, чтобы кто-то просмотрел и вложил ваш вклад и объединил их в свою ветку. Выдвижные запросы показывают посмотреть *различия* , или различия, на содержание от обеих ветвей. Изменения, добавления и вычитания показаны зеленым и красным.

Как только вы сделаете фиксацию, вы можете открыть запрос на перенос и начать обсуждение, даже до того, как код будет завершен.

Используя систему [@mention от](https://help.github.com/articles/about-writing-and-formatting-on-github/#text-formatting-toolbar) GitHub в сообщении запроса на тягу, вы можете запросить отзывы от конкретных людей или команд, независимо от того, находятся они в зале или в 10 часовых поясах.

Вы даже можете открыть запросы на pull в своем собственном репозитории и объединить их самостоятельно. Это отличный способ изучить GitHub Flow, прежде чем работать над более крупными проектами.

**Откройте запрос Pull для изменений в README**

*Нажмите на изображение для увеличения*

| **шаг** | **Скриншот** |
| --- | --- |
| Нажмите  **Pull Request** , затем на странице Pull Request щелкните зеленую кнопку**запроса нового запроса** . | [пр-вкладка](https://guides.github.com/activities/hello-world/pr-tab.gif) |
| В поле **Сравнение примеров**выберите ветку, которую вы сделали readme-edits, для сравнения сmaster(оригиналом). | [филиал](https://guides.github.com/activities/hello-world/pick-branch.png) |
| Ознакомьтесь с изменениями на странице сравнения на странице сравнения, убедитесь, что они являются тем, что вы хотите отправить. | [разница](https://guides.github.com/activities/hello-world/diff.png) |
| Когда вы будете удовлетворены тем, что это те изменения, которые вы хотите отправить, нажмите большую зеленуюкнопку **Create Pull Request** . | [создать тянуть](https://guides.github.com/activities/hello-world/create-pr.png) |
| Дайте вашему запросу запрос заголовка и напишите краткое описание ваших изменений. | [пр-формы](https://guides.github.com/activities/hello-world/pr-form.png) |

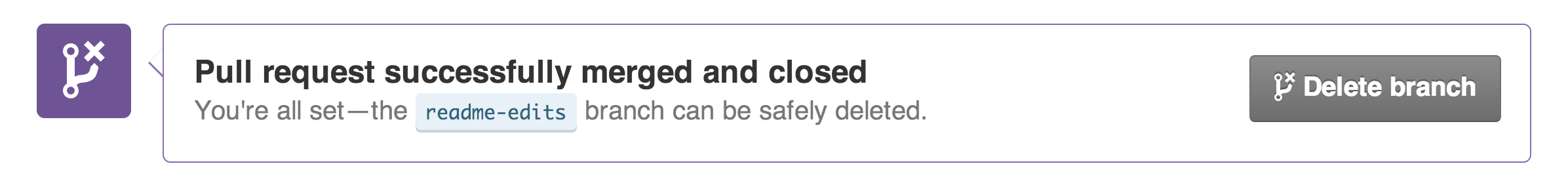
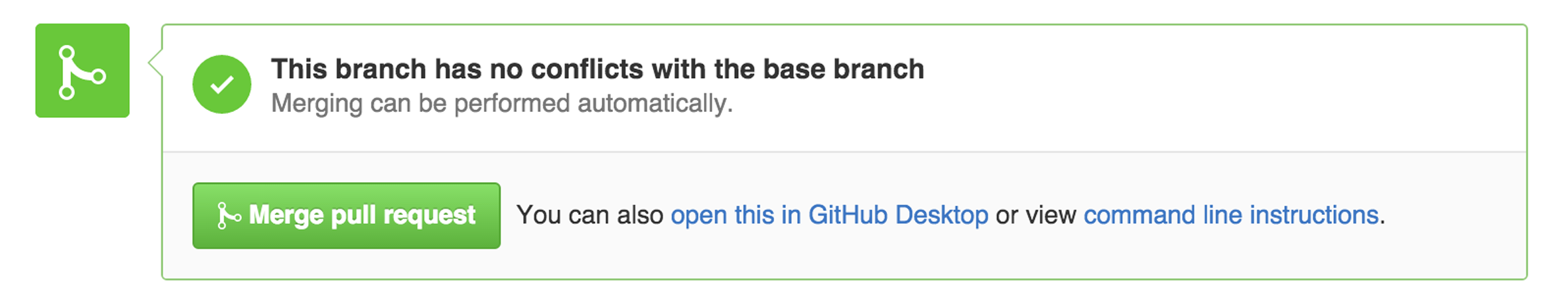
Когда вы закончите свое сообщение, нажмите « **Создать запрос на растяжение»** !

**Совет** . Вы можете использовать [emoji](https://help.github.com/articles/basic-writing-and-formatting-syntax/" \l "using-emoji) и [перетаскивать изображения и gifs](https://help.github.com/articles/file-attachments-on-issues-and-pull-requests/) на комментарии и Pull Requests.

**Шаг 5. Объедините свой запрос на вытягивание**

На этом последнем этапе пришло время объединить ваши изменения - объединение вашей readme-editsветки в masterветку.

1. Нажмите зеленую кнопку **запроса** на слияние, чтобы объединить изменения master.
2. Нажмите « **Подтвердить слияние»** .
3. Идем дальше и удаляем ветку, так как ее изменения были включены, с кнопкой « **Удалить ветвь»** в фиолетовом поле.



**Празднуйте!**

Завершив этот учебник, вы научились создавать проект и делать запрос на получение на GitHub! : Тада: : Octocat: : Зап:

Вот что вы сделали в этом уроке:

* Создан репозиторий с открытым исходным кодом
* Начал и управлял новой веткой
* Изменен файл и внесены изменения в GitHub
* Открытый и объединенный запрос Pull

Взгляните на свой профиль GitHub, и вы увидите свои новые [квадраты вкладов](https://help.github.com/articles/viewing-contributions) !

Чтобы узнать больше о возможностях запроса Pull, рекомендуем прочитать [Руководство по потоку GitHub](http://guides.github.com/overviews/flow/) . Вы также можете посетить [GitHub Explore](http://github.com/explore) и принять участие в проекте с открытым исходным кодом: Octocat:

**Совет** . Ознакомьтесь с нашими другими [руководствами](http://guides.github.com/) , [каналом YouTube](http://youtube.com/githubguides) и по [требованию,](https://services.github.com/on-demand/) чтобы узнать больше о том, как начать работу с GitHub.