**Справочник**

|  |  |
| --- | --- |
| Инициализация проекта | git init |
| Информация о пользователя | git config user.name имя  git config user.email почта |
| Создание коммита | git add .  git commit –m имя\_коммита  или  git commit –a –m имя\_коммита |
| Проверка состояния репозитория | git status |
| Просмотр коммитов | git log --oneline |
| Переходы по комиммитам/веткам | git checkout имя\_коммита\_или\_ветки |
| Просмотр веток проекта | git branch |
| Создание ветки | git branch имя ветки  или сразу с переходом на ветку  git checkout –b имя ветки |
| Переименование веток | git branch –m новое имя |
| Удаление ветки | git checkout –d имя ветки |
| Слияние веток | git merge имя\_ветки |
| Преобразование веток | git rebase имя\_ветки |
| Связь с удаленным репозиторием | git remote add origin адрес\_репозитория |
| Загрузка данных из удаленного репозитория | git pull |
| Отправка данных в удаленный репозиторий | git push |
| Клонирование репозитория | git clone адрес\_репозитория |
| Создание версии | git tag –a v1.0 –m "имя" |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Работа с git и github**

Сразу загрузим проект на удаленный репозиторий github и будем использовать систему контроля версий в течении всего процесса обучения

Инструкция

1. Авторизуемся в системы github
2. Открываем наш проект laravelproject в VS code
3. Открываем терминал (меню Terminal -> New Terminal)
4. Пишем в консоли команды:

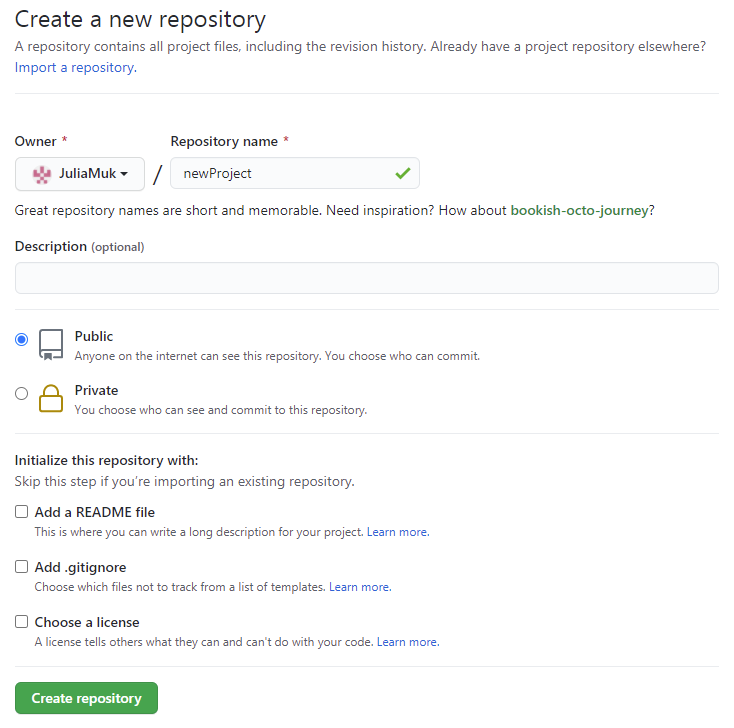
* Создать проект - git init
* Установка имени (имя с githab) git config user.name имя
* Установка email (с githab) config user.email почта

1. Создаем первый коммит, в терминале пишем команды:

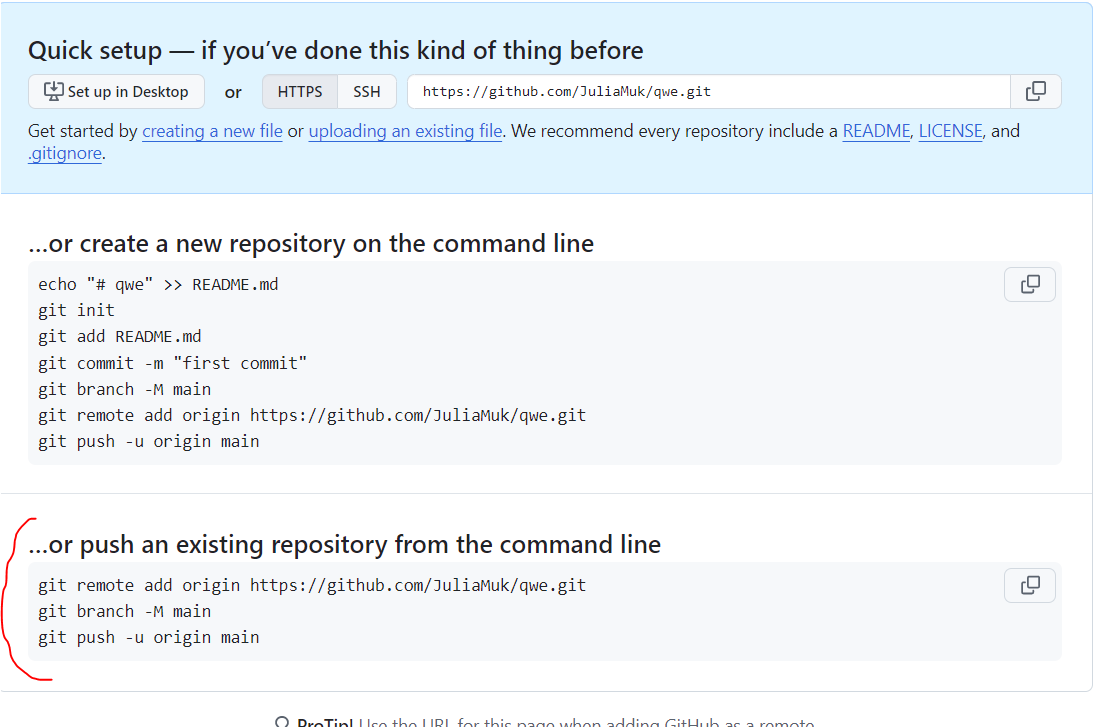
git add .

git commit –m "название изменений"

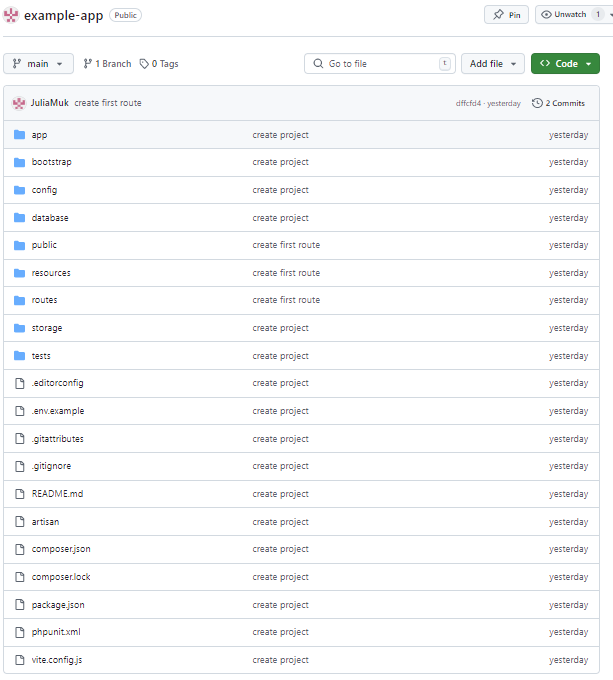
1. Заходим на GitHub.
2. Создаем новый репозиторий на GitHub



1. Нажимаем кнопку Create repository
2. В появившемся окне копируем команды



1. Вставляем их в терминал
2. При запросе авторизуемся через браузер (если система вас попросит)
3. Возвращаемся на страницу репозитория и обнавляем ее, там должен появиться ваш проект



1. В дальнейшем после выполнения каждого задания, сохраняем и отправляем изменения командами

git add .

git commit –m "название изменений"

git push