Базовый порядок работы с Git (GitHub)

Открываем терминал в Linux (Windows или MacOS) и работаем по следующему примерному алгоритму:

|  |  |  |
| --- | --- | --- |
| **№ п/п** | **Описание действия** | **Команда в терминале** |
| 1 | Проверка установки Git на компьютере. | git version |
| 2 | Установка Git через терминал в Linux Mint . | apt-get install git |
| 3 | Указание имени пользователя в настройках Git.  Флаг —global позволяет сделать это, находясь в любой директории. | git config --global user.name "User Name" |
| 4 | Указание e-mail в настройках Git.  Флаг —global позволяет сделать это, находясь в любой директории. | git config --global user.email username@yandex.ru |
| 5 | Проверка содержимого глобальных настроек Git, хранящихся в файле .gitconfig в домашней директории. | cat ~/.gitconfig  или  git config --list |
| 6 | Создание репозитория Git в папке проекта (предварительно необходимо перейти в папку проекта, используя команду cd в терминале) | cd <папка с проектом>  git -init |
| 7 | Изменение имени ветки «Master» на «Main» | git branch -m «main» |
| 8 | Если под контроль версий неправильно выбрана папка, то снять контроль версий с папки можно удалением скрытой подпапки .git | cd <папка с репозиторием>  rm -rf .git |
| 9 | Проверка текущего состояния репозитория | git status |
| 10 | Добавление конкретного файла или всех файлов папки в репозиторий git. | git add <файл.расширение>  git add -all  или  git add . |
| 11 | Выполнение коммита (флаг -m добавляет сообщение к коммиту) | git commit -m «сообщение об изменениях» |
| 12 | Просмотр истории коммитов | git log |
| 13 | После создания удаленного репозитория на GitHub, для связки его с локальным репозиторием необходимо проверить наличие SSH ключей (публичного и приватного). Они обычно находятся в корневой папке в скрытой директории .ssh/ | cd ~  ls -la .ssh/ |
| 14 | Генерация SSH-ключа. Алгоритм 1.  Если при вводе команды происходит ошибка, то нужно воспользоваться другми алгоритмом | ssh-keygen -t ed25519 -C "электронная почта, к которой привязан ваш аккаунт на GitHub" |
| 15 | Генерация SSH-ключа. Алгоритм 2. | ssh-keygen -t rsa -b 4096 -C "электронная почта, к которой привязан ваш аккаунт на GitHub" |
| 16 | Проверка наличия новых созданных SSH ключей | ls -a ~/.ssh |
| 17 |  |  |
| 18 | Привязывание SSH-ключа с GitHub аккаунта. | cat ~/.ssh/id\_rsa.pub  или  cat ~/.ssh/id\_ed25519.pub  и копируем SSH-ключ в буфер обмена |
| 19 | Переход на GitHub. В аккаунте в «Settings», в пункте «SSH and GPG keys» выбрать «New SSH Keys». В поле «Title» написать «Personal key».  В поле «Key» скопируйте ваш ключ из буфера обмена. |  |
| 20 | После чего нажмите «Add SSH key» |  |
| 21 | Проверка правильности SSH ключа | ssh -T git@github.com |
| 22 | Для привязки удаленного репозитория к локальному необходимо перейти на страницу удаленного репозитория, выбрать тип «SSH» и скопировать URL. |  |
| 23 | Привязка удаленного репозитория к локальному (предварительно необходимо перейти в папку с проектом) | cd <папка с проектом> |
| 24 | Откройте консоль, перейдите в каталог локального репозитория и введите команду «git remote add».  Команде необходимо передать два параметра: имя удалённого репозитория и его URL.  В качестве имени используйте слово «origin», а URL вы скопировали со страницы удалённого репозитория. | git remote add origin git@github.com:%ИМЯ\_АККАУНТА%/first-project.git |
| 25 | Убедиться, что локальный и удаленный репозитории связаны можно с помощью команды | git remote - v |
| 26 | Синхронизация локального репозитория с удаленным. Первый раз это делается с флагом -u и параметрами origin (имя удалённого репозитория) и main или master (название текущей ветки). Флаг -u свяжет локальную ветку с одноимённой удалённой. | git push -u origin main  или  git push -u origin master |
| 27 |  |  |
| 28 |  |  |
| 29 |  |  |
| 30 |  |  |