Команды Git’a

git init – инициализация git’a в папке

rm -rf .git – финализировать git в папке

git status – проверить состояние репозитория

git add – отслеживать состояние файла

--all – выбрать все файлы

. – добавляет всю текущую папку

git commit – сделать коммит (отпечаток)

-m ‘message’ – добавить сообщение

--amend – изменить сообщение последнего коммита

git push – отправить изменения на гитхаб

-u origin – связывание локальной ветки с удаленной

git log – просмотр истории изминений и хеш коммитов

--oneline – получить сокращенный лог

--amend – добавить забытый файл в последний коммит

git restore --staged <file> - вернуть файл из staged area в untracked, убрать из коммита

git reset --hard <commit hash> - откатить коммит

git restore <file> - октатить файл к последней версии

git diff – проверить разницу в изменениях файла исходной с moddified

--staged – для просмотра в staged area

git clone – клонирование репозитория

.gitignore создание файла в в .git для игнорирования файлов

Conventional Commits предлагает такой формат коммита: <type>: <сообщение>. Первая часть type — это тип изменений. Таких типов достаточно много. Вот два примера:

* feat (сокращение от англ. *feature*) — для новой функциональности;
* fix (от англ. «исправить», «устранить») — для исправленных ошибок.

Более подробный список можно увидеть [на сайте с описанием этого стиля](https://www.conventionalcommits.org/ru/v1.0.0-beta.4/#%D1%81%D0%BF%D0%B5%D1%86%D0%B8%D1%84%D0%B8%D0%BA%D0%B0%D1%86%D0%B8%D1%8F).

Например, сообщение может быть таким.