

Краткая методичка по git

Рустам Гайнутдинов, 2019

Git – это система контроля версий. То есть это ПО, которое позволяет хранить историю изменения нужных файлов, которые используются в программе.

Сначала его надо скачать и установить. После, его нужно настроить:

```
git config --global user.email <твой_адрес>
git config --global user.name <твоё_имя>
```

Если у тебя уже есть программа, для которой ты хочешь инициализировать *git-репозиторий*, то в папке с файлами этой программы нужно прописать команду:

```
git init
```

В директории будет создана папка (скрытая) *.git*, внутри которой в будущем будет храниться история изменений каких-нибудь файлов в репозитории.

Если тебе нужно скачать чью-нибудь программу (это называется клонировать репозиторий), то ты просто заходишь в какую-нибудь директорию, и пишешь там:

```
git clone <url-адрес>
```

Если ты уже клонировал так себе какой-нибудь репозиторий, а его создатель залил на сайт обновления, то не нужно его заново скачивать, достаточно написать:

```
git pull
```

И *git* сам обновит все файлы в директории программы.

Теперь, как создавать *коммиты* (коммит – это как бы конкретная версия, что-то по типу снапшотов в майнкрафте). Пусть у тебя есть несколько файлов. В коммит нужно добавлять те файлы, которые ты изменил или добавил. (Если ты делаешь первый коммит, то все файлы программы). Для этого нужно написать команду:

```
git add <имя_файла>
```

После того, как ты добавил все нужные файлы, нужно собственно сделать коммит:

```
git commit -m <сообщение>
```

Сообщение – это как-бы «название» коммита.

Называть коммиты как-то вроде «1.1», «1.2», «alpha 1.3» считается плохим тоном. Сообщение коммита должно отражать то, что ты изменил в нём. Например «Added python3 support», или «Moved functions into library».

После исполнения этой программы все изменения в файлах, который ты добавил командой *add* будут записаны в коммит с названием, которое ты ему дал. Для того, чтобы сделать потом другой коммит, нужно будет заново добавлять в него файлы.

Если ты хочешь залить свой локальный репозиторий в интернет, то нужно сделать следующее: Заходишь на *github*, регистрируешься там. И создаёшь там репозиторий. Копируешь его ссылку. Потом в своём локальном репозитории пишешь команду:

```
git remote add origin <url_репозитория>
```

И тогда ты привяжешь к локальному репозиторию адрес интернет-репозитория. Чтобы залить свою программу в интернет репозиторий, нужно написать:

```
git push
```

Вроде на этом все основные моменты я перечислил. Чуть более сжатую методичку, но с большим количеством информации и команд, можно найти по адресу:

<https://docs.google.com/presentation/d/1TkifsF41qMppbrNo0DEjNuGKTf2p73wERdvfyudiZE8/>

Это лекции, которые нам читал один из преподавателей по программированию (не Шнейвайс). Там же есть ссылка на более полную документацию по *git*:

<https://git-scm.com/book/ru/v1/Введение-Основы-Git>

Надеюсь, что моя методичка вышла неплохой и что она поможет :)