

# Регистрация

Давайте все зарегистрируемся на гитхабе

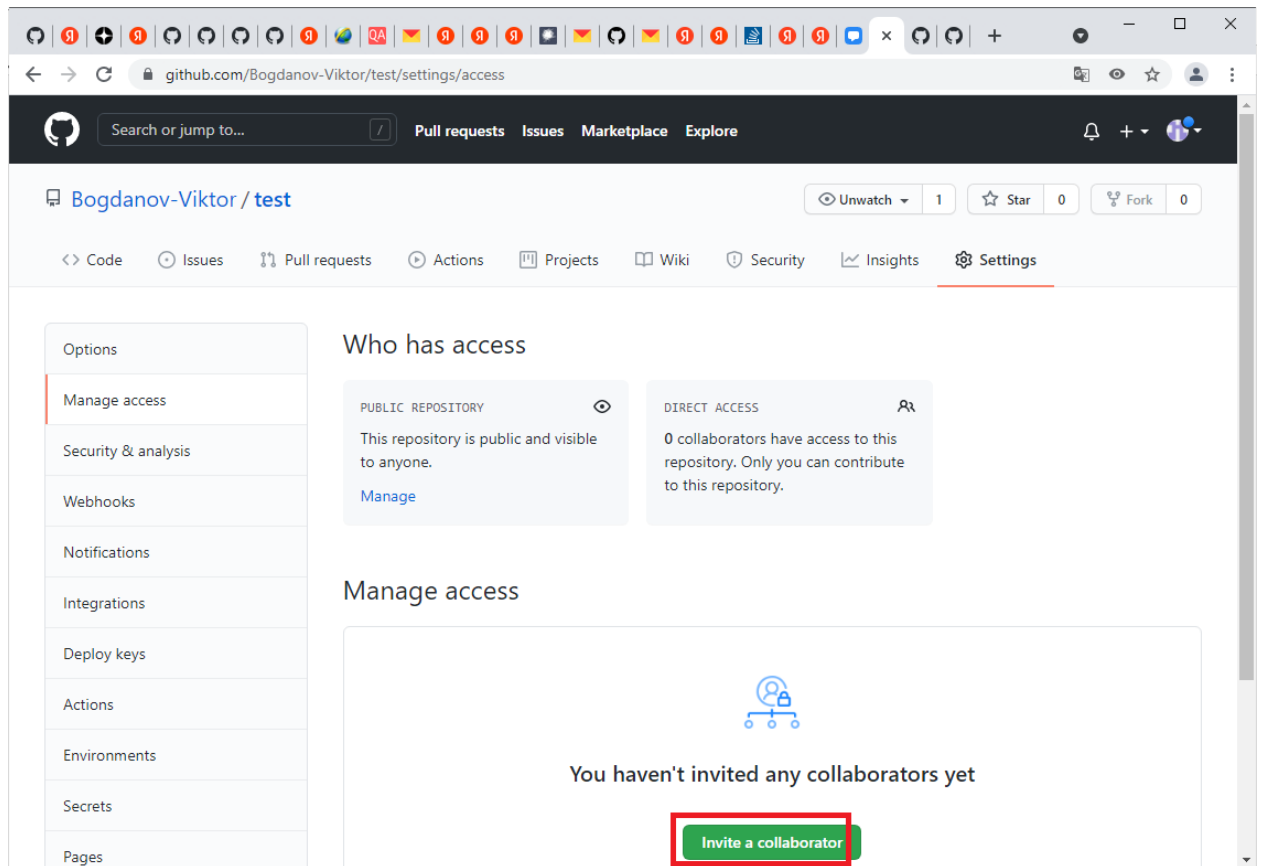
<https://github.com/login>

И подпишемся на меня, чтобы я мог вас найти

<https://github.com/Bogdanov-Viktor>

The screenshot shows a web browser window displaying the GitHub profile of 'Bogdanov-Viktor'. The browser's address bar shows 'github.com/Bogdanov-Viktor'. The GitHub navigation bar at the top includes a search bar, 'Pulls', 'Issues', 'Marketplace', and 'Explore'. The profile header shows the user's name 'Bogdanov-Viktor', a 'Follow' button (highlighted with a red rectangle), and statistics: '1 follower · 1 following · 0 stars'. It also notes 'Joined 2 days ago' and a 'Block or Report' link. The 'Overview' tab is active, showing 'Popular repositories' with 'first-java-spring' (Java) and 'test'. A 'Contribution activity' section for 'August 2021' shows 'Created 14 commits in 1 repository' for 'Bogdanov-Viktor/first-java-spring'. A contribution grid shows activity for August, with a legend indicating 'Less' and 'More' contributions.

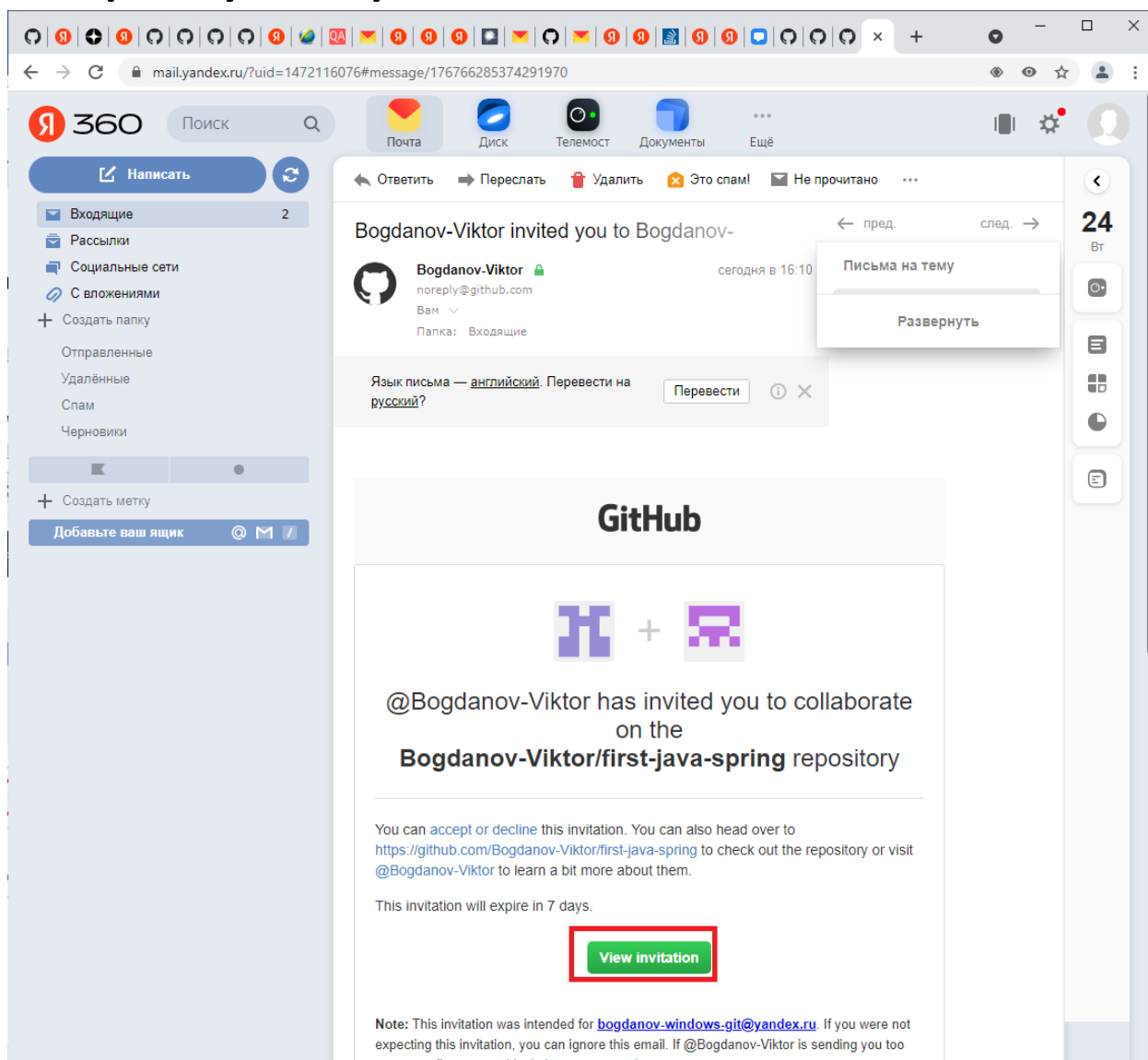
## Я добавляю вас в коллабораторы (т.е. соавторы) репозитория «Test»



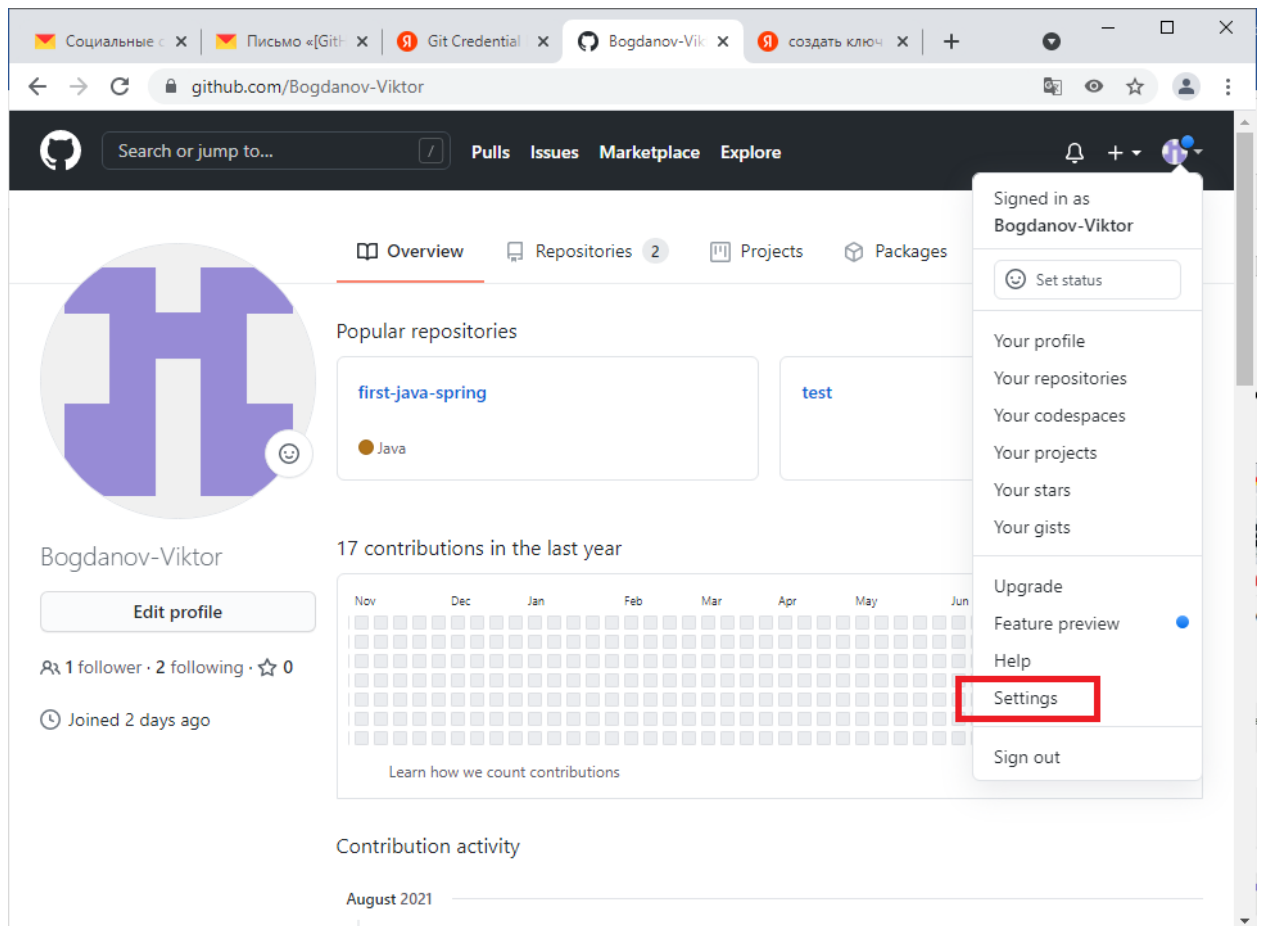
Соавторы программы, могут без моего участия изменять её в репозитории на GitHub

Остальные, могут только присылать рекомендации

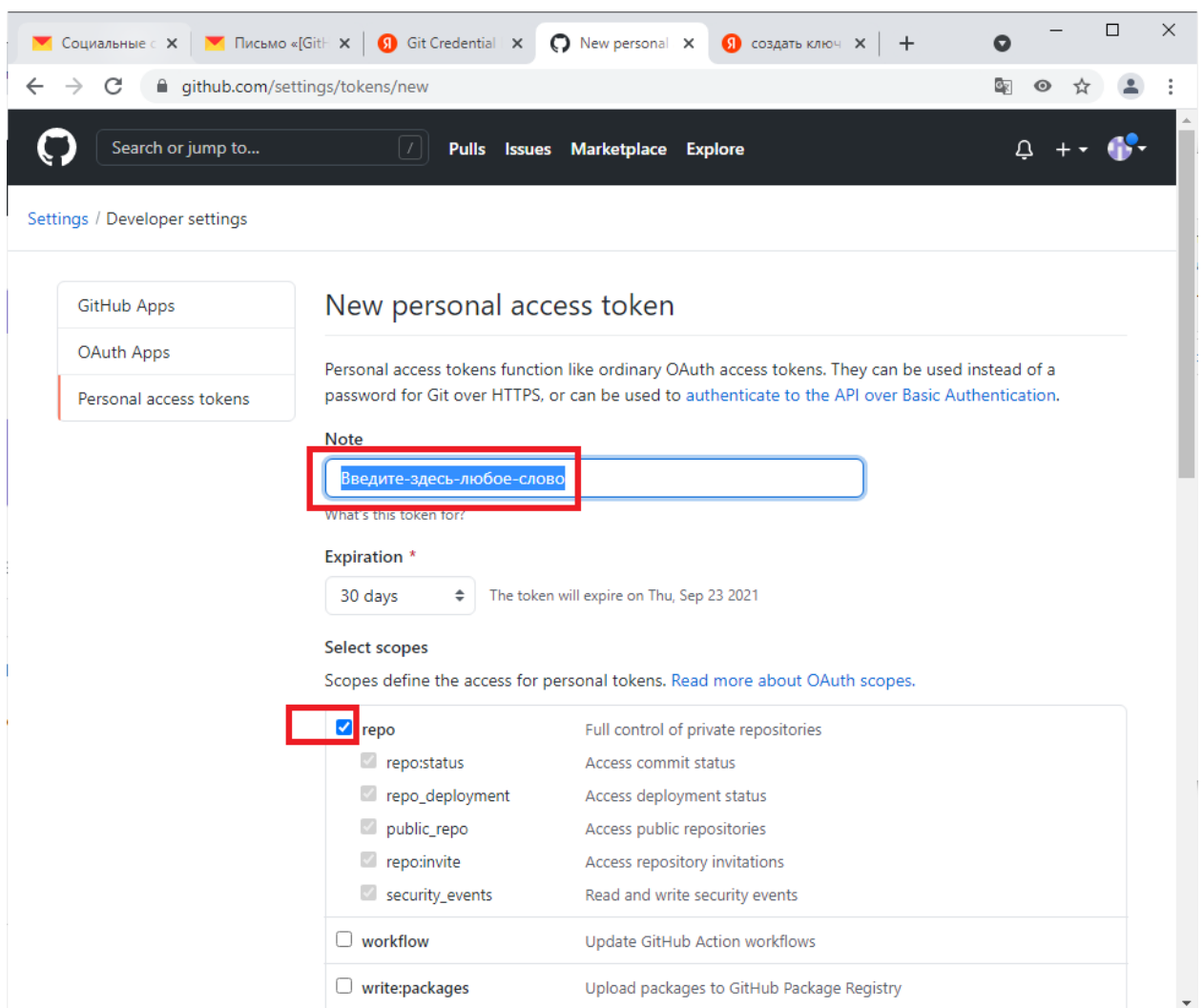
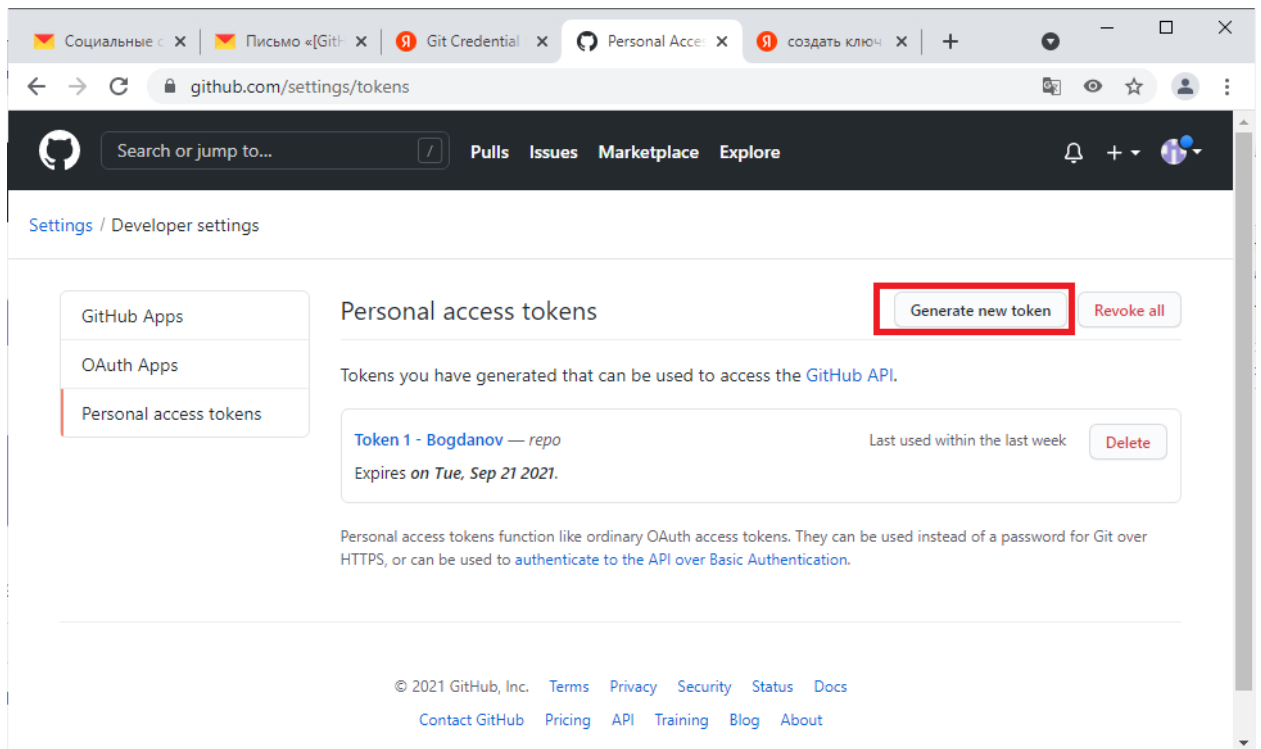
## Вы получаете письмо с приглашением в соавторы, которое нужно принять

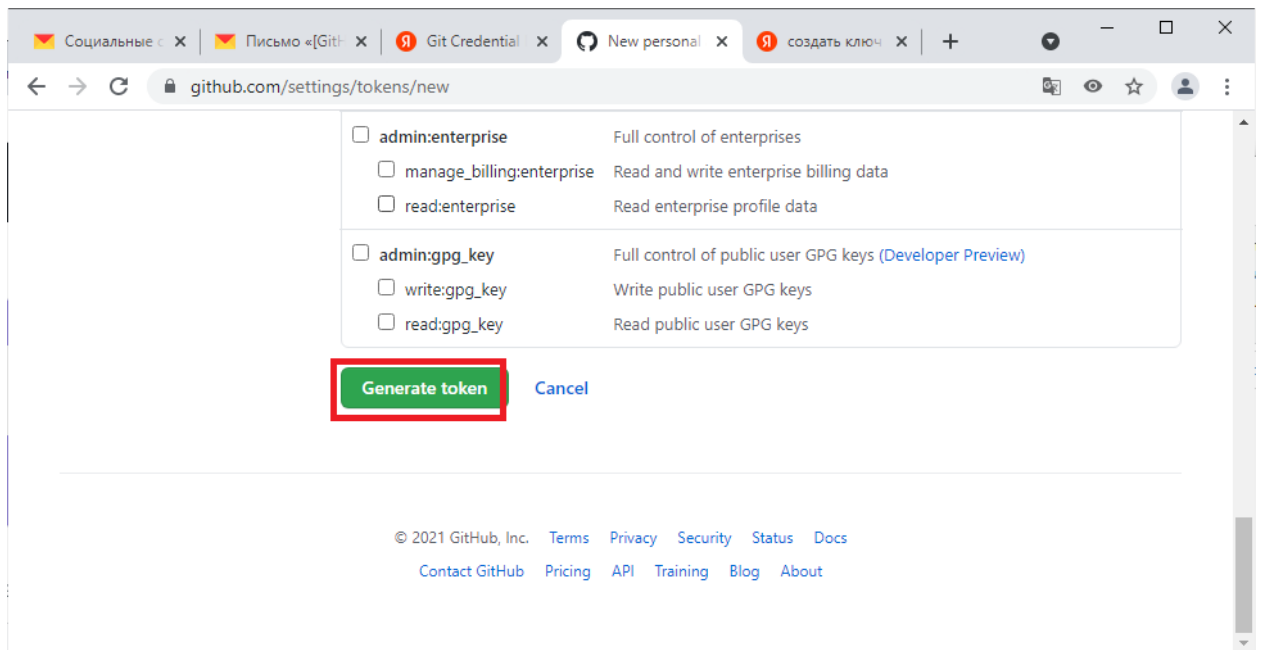


Вы создаете токен для того, чтобы пользоваться GitHub под своей учетной записью в приложениях на своем компьютере (пользоваться, например, в консоли)

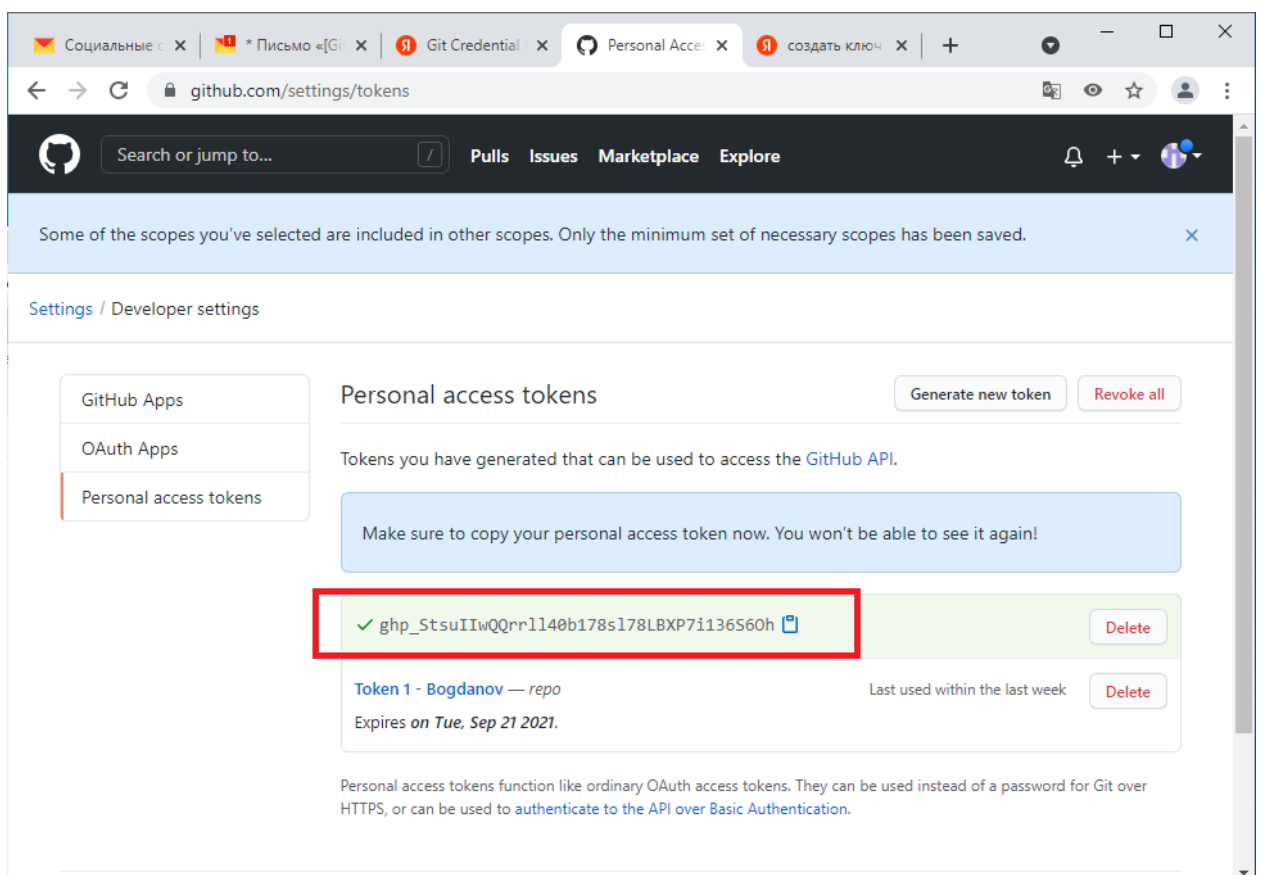


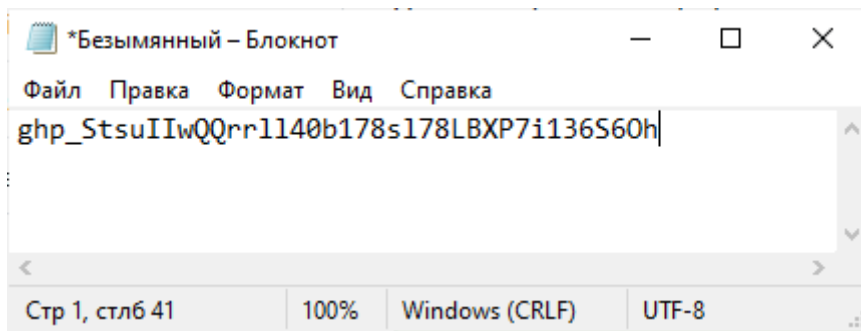




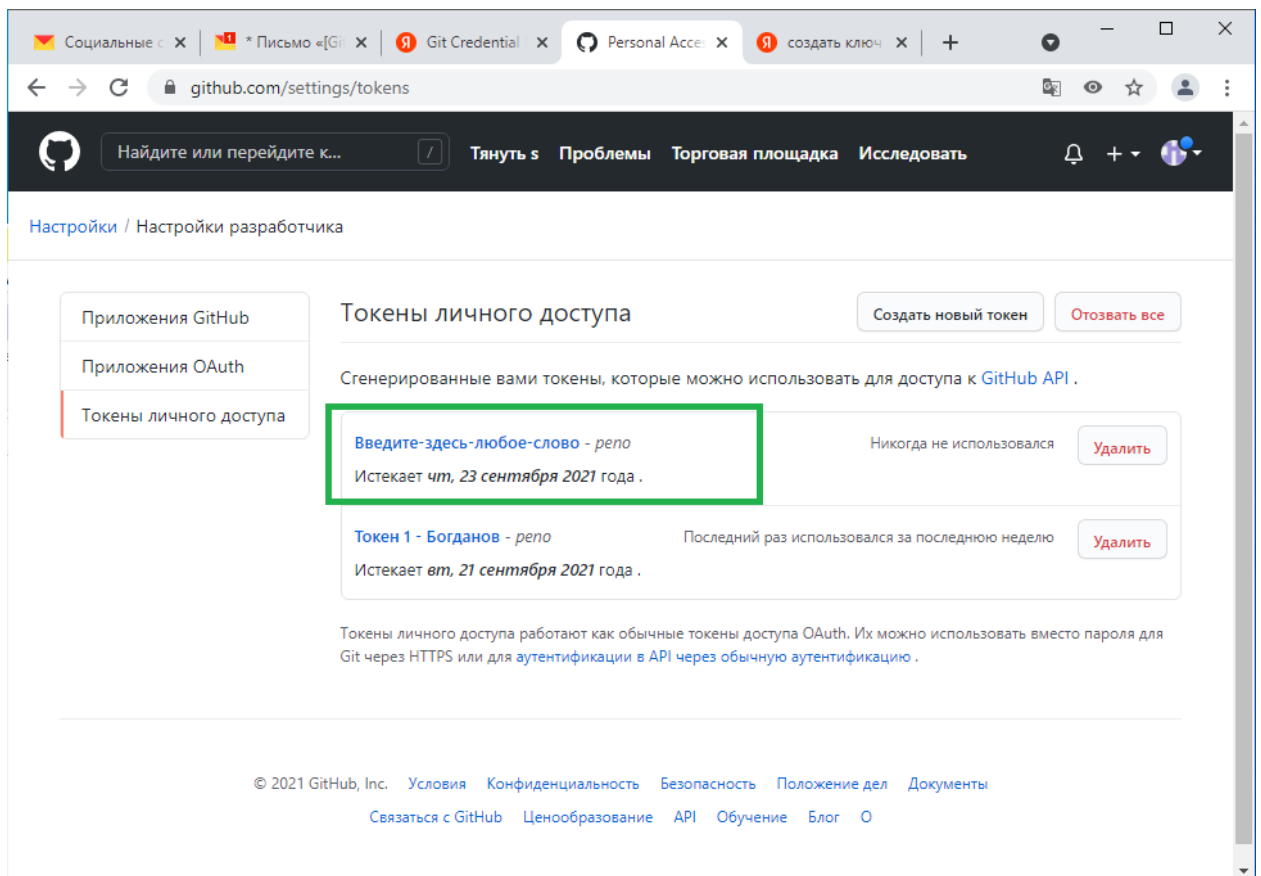


**Скопируйте токен в блокнот (Убедитесь, что вы скопировали токен, вы не увидите его второй раз)**





На странице с токенами вы увидите, что срок действия вашего токена еще не истек.

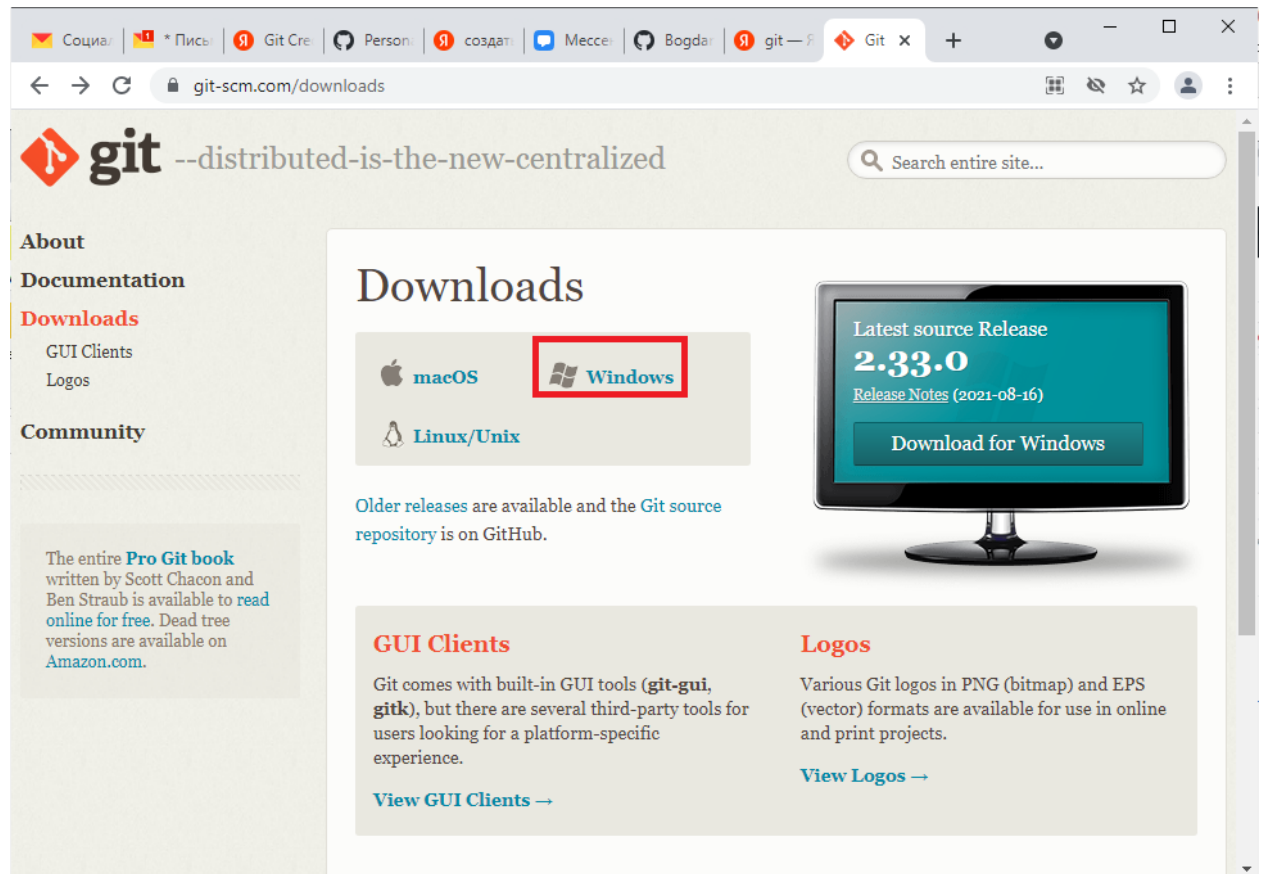




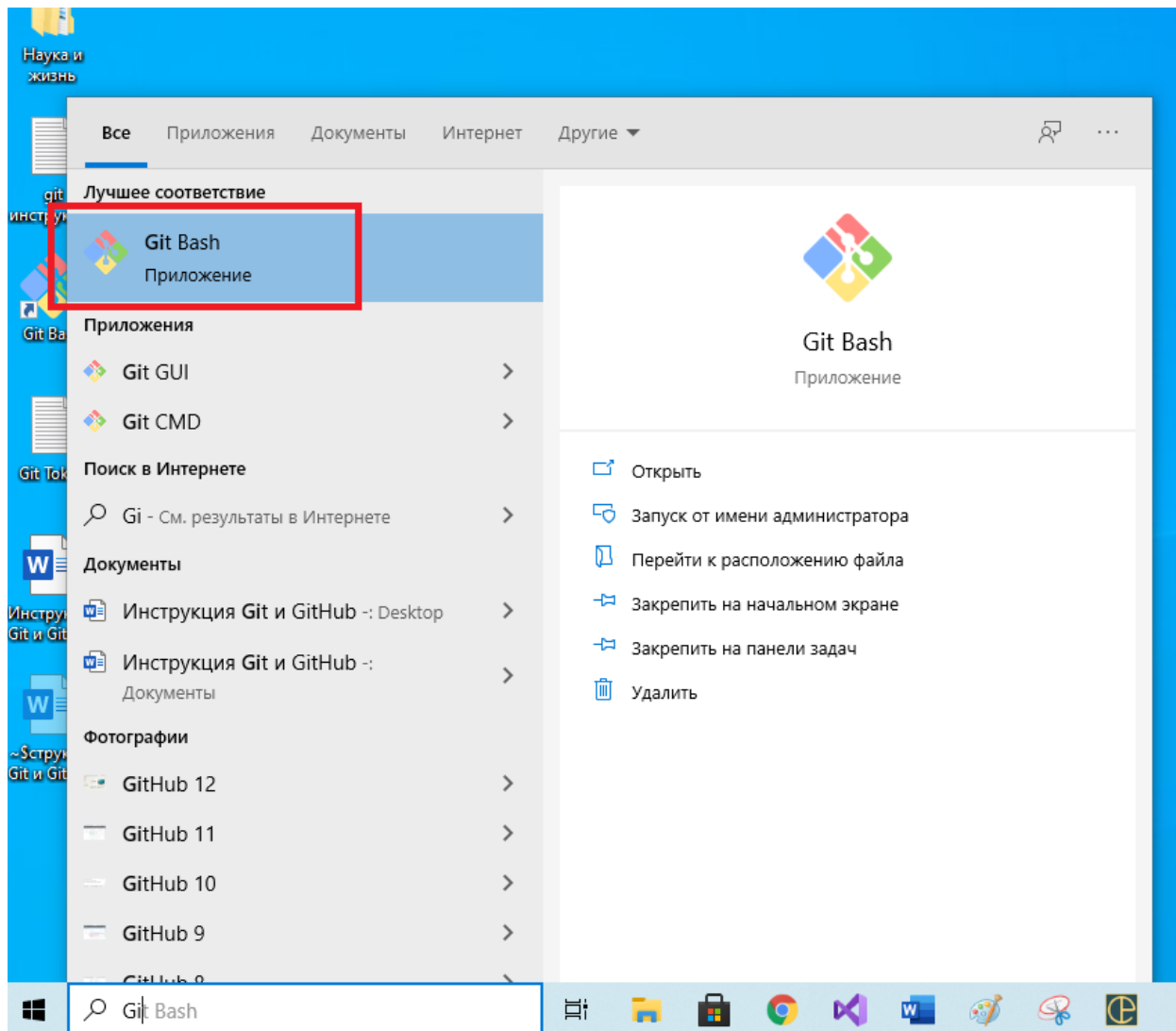
# Вы устанавливаете Git (инструкция для пользователей Windows)

Переходим по ссылке на официальный сайт. Устанавливаем версию для Windows

<https://git-scm.com/downloads>



## Открываем Git Bash

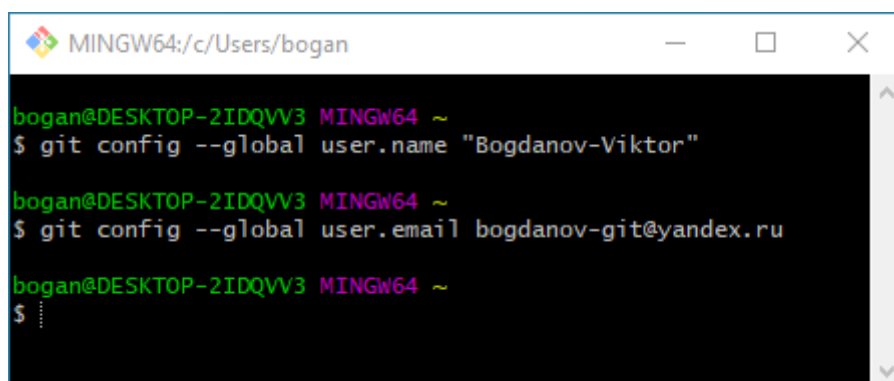


## Вводим свое имя и email в настройки Git

Это нужно, чтобы делать коммиты (объясню позже), т.к. у каждого коммита должно быть указано имя и email автора

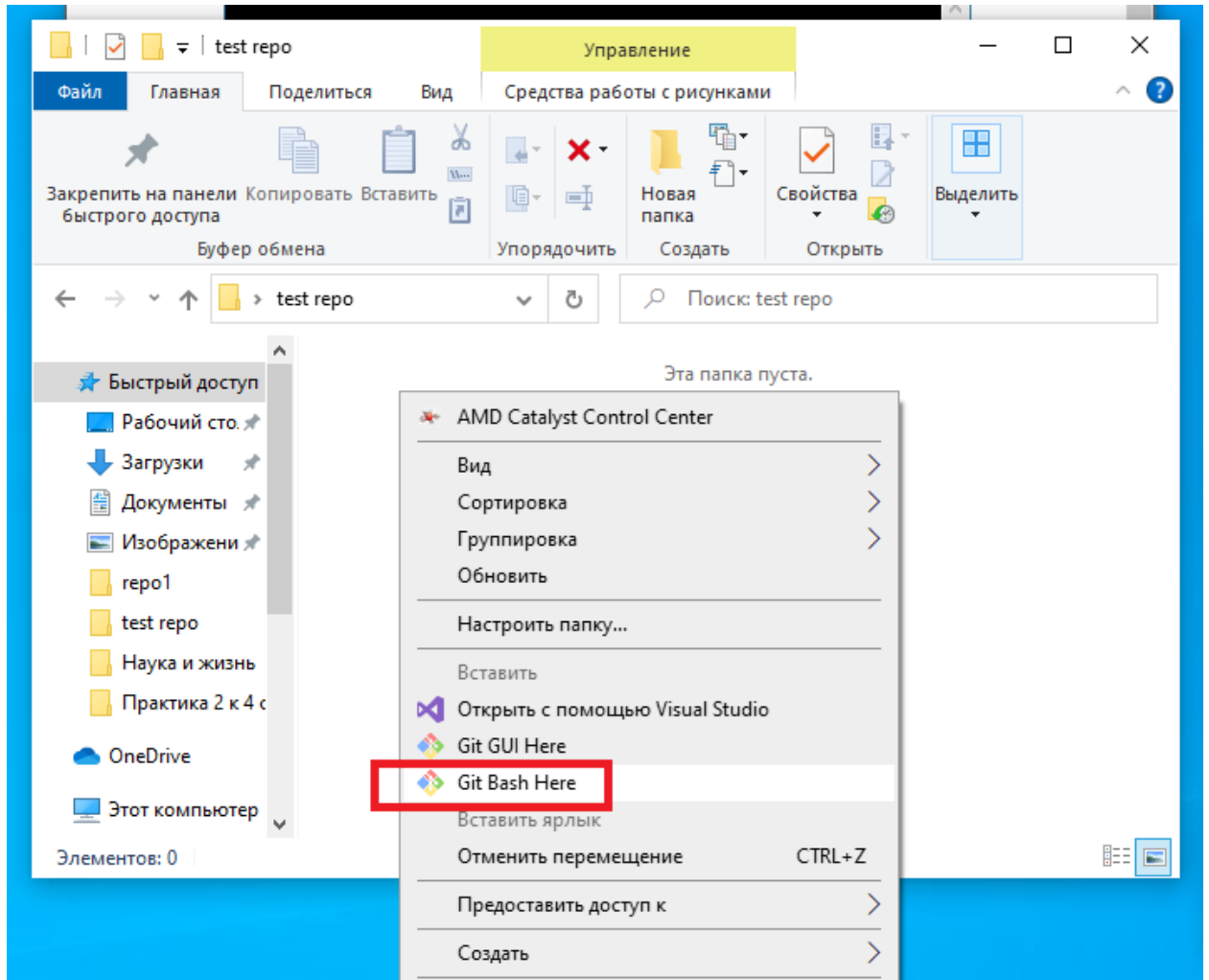
```
git config --global user.name "Bogdanov-Viktor"
```

```
git config --global user.email bogdanov-git@yandex.ru
```



## Вы скачиваете с Git Hub на свой компьютер репозиторий с которым будите работать, вносите в него изменения

Заходим в папку, куда будем скачивать, нажимаем правую кнопку мыши, затем Git Bash Here

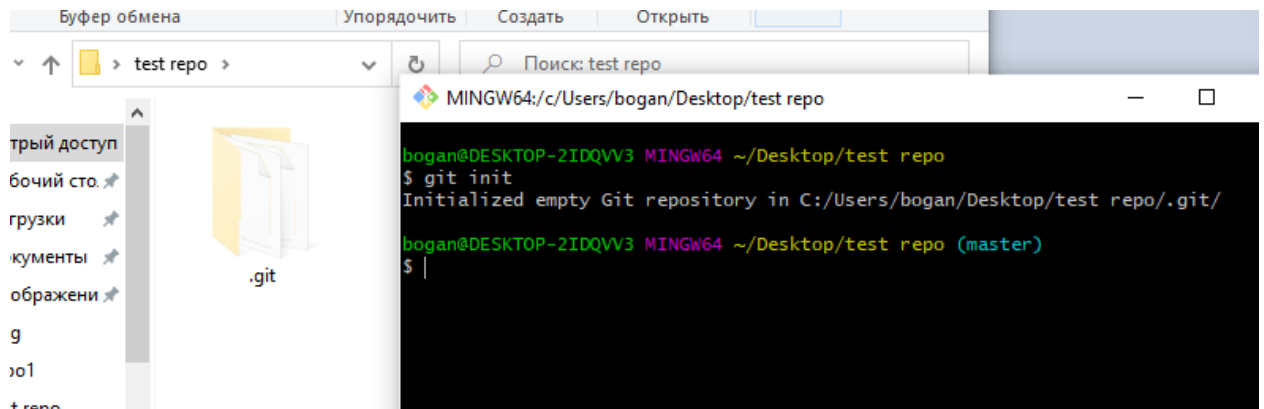


### *git init*

# Этой командой мы:

- # создаем папку .git, где данные о коммитах
- # (т.е. инициализируем пустой репозиторий
- # или переинициализируем существующий)

# В консоли появится слово master – это название ветки, в которой вы находитесь



*git remote add origin* <https://github.com/Bogdanov-Viktor/test.git>

# Этой командой мы:

# Добавляем в ветку origin удаленный репозиторий

*git pull origin master*

# Этой командой мы:

# Скачиваем все файлы из удаленного репозитория, а также коммиты

# (master – ветка в которой вы находитесь,

# origin – которую вы копируете)

# Теперь вы сможете увидеть все файлы проекта в текущей папке

Далее откройте файл «README.md» и добавьте в конец свое имя, а также сделайте скриншот и положите в папку «img»

Затем:

*git add .*

# Этой командой мы:

# Добавляем в будущий коммит все измененные вами файлы

*git commit -m "add bogdanov name to files"*

# Этой командой мы:

# Создаем коммит (после -m указывается название,

# которое мы даем коммиту)

Коммит – это, по-простому, точка восстановления.

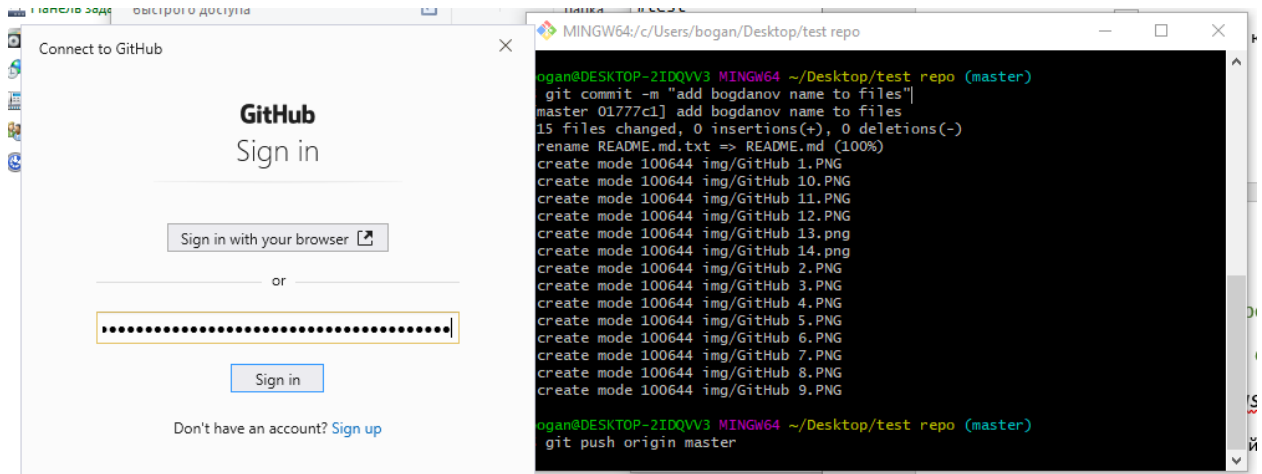
У каждого коммита есть автор, у автора email.

*git push origin master*

# Этой командой мы:

# Отправляем свой коммит в удаленный репозиторий

(В появившееся окошко вставляете скопированный ранее токен)



Чтобы далее можно было использовать *git pull*, *git push* без указания веток: «origin master», задаем настройку для этого репозитория:

*git branch --set-upstream-to=origin/master master*

Рекомендую видео:

<https://www.youtube.com/watch?v=qd74R-rvfsY>

-----всё

Еще пригодятся следующие команды:

*git status*

# проверить, что мы не добавили в коммит

# если мы изменим файл, то он будет помечен, как незакомиченный,

# пока мы не добавим его спомощью git add

*git branch -la*

# смотреть список веток (-а значит в т.ч. из удаленных репозиториях)

*git log --oneline --graph*

# смотреть список коммитов

*git config --list*

# проверка настроек git (например, чтобы удостовериться, что указан

# user.email bogdanov-git@yandex.ru)