

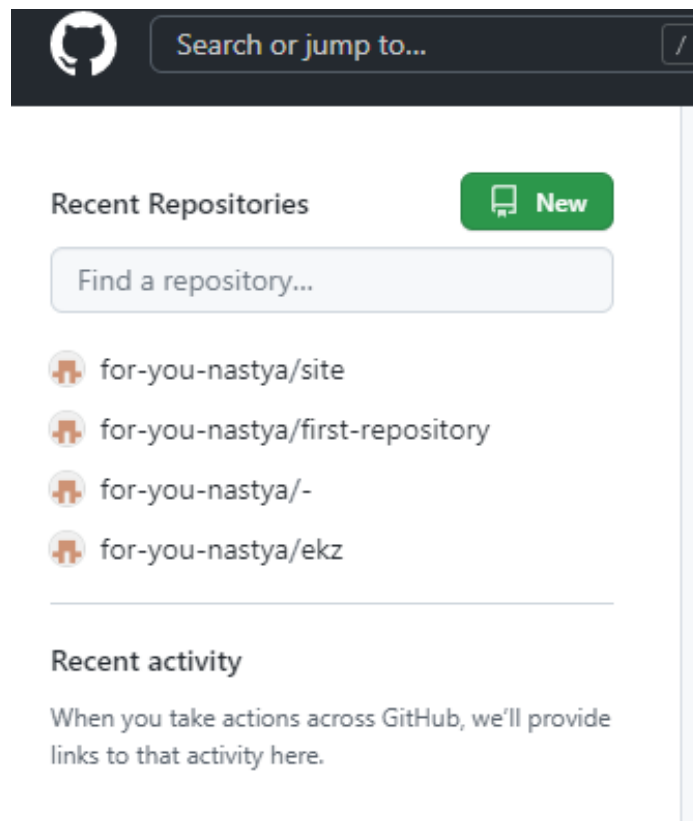
## Лабораторная работа №4.

### Работа с GitHub

**Цель:** научиться работать с Github и загружать локальный репозиторий.



1. До этого мы работали локально, простыми словами у себя на компьютере. Но нам нужно хранить проекты удаленно, чтобы работать с ними из любой точки. Доступ будет у всех, у кого есть ссылка. GitHub необходим проектам с частыми обновлениями, множеством версий, большим количеством файлов, необходимостью синхронизации разработки и удобного развёртывания, что в командной разработке (в основном в крупных и средних IT-компаниях) незаменимая вещь.

Перед подключением локального репозитория Git к удаленному репозиторию GitHub нужно создать этот удаленный репозиторий на Github.



Нажимаем кнопку New и создаем новый репозиторий (заполняем поля).

---

Владелец *	Имя репозитория *
 для тебя-настя ▾	/ studying-Git 

Отличные названия репозитория: обучение-Git доступно. инающиеся. Нужно вдохновение? Как насчет смеха-ракушки?

Описание (необязательно)



First Git project.

---

Поле Описание не обязательно. Но если вы работаете над серьезным проектом, то следует не оставлять поле без внимания. Детально и подробно проект нужно описать в файле README.

Затем указываем, будет ли наш проект публичным (т.е. все смогут его видеть), или приватным.

---

<input checked="" type="radio"/>	 <b>Общественный</b>
	Любой человек в Интернете может увидеть этот репозиторий. Вы сами выбираете, кто может совершать.
<input type="radio"/>	 <b>Частный</b>
	Вы выбираете, кто может видеть и фиксировать этот репозиторий.

---

Кроме создания README, у нас есть еще две опции - добавлять ли файл .gitignore (по умолчанию None - не добавлять, указывает на то, какие файлы необходимо игнорировать при загрузке), и добавлять ли лицензию (по умолчанию тоже None, позволяет или запрещает пользователям использовать ваш код).

Инициализируйте этот репозиторий следующим образом:  
Пропустите этот шаг, если вы импортируете существующий репозиторий.

☐ **Добавление файла README**

Здесь вы можете написать длинное описание для вашего проекта. [Подробнее.](#)

**Добавить .gitignore**

Выберите файлы, которые не следует отслеживать, из списка шаблонов. [Подробнее.](#)

Шаблон .gitignore: Никакой ▾

**Выберите лицензию**

Лицензия говорит другим, что они могут и не могут делать с вашим кодом. [Подробнее.](#)

Лицензия: Никакой ▾

❗ Вы создаете публичный репозиторий в личном кабинете.

**Создать репозиторий**

Нажимаем на кнопку Создать репозиторий и переходим на страницу:

для тебя-настя / учеба-Git Общественный

Булавка Отменить наблюдение 1 Вилка 0 Звезда 0

<> Код Вопросы Запросы на вытягивание Действия Проекты Вики Безопасность Идеи Параметры

**Быстрая настройка — если вы уже делали подобные вещи раньше**

Настройка на рабочем столе или HTTPS SSH git@github.com:for-you-nastya/studying-Git.git

Начните с создания нового файла или отправки существующего файла. Мы рекомендуем каждому репозиторию включать файл README, ЛИЦЕНЗИЮ и .gitignore.

... или создайте новый репозиторий в командной строке

```
echo "# studying-Git" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin git@github.com:for-you-nastya/studying-Git.git
git push -u origin main
```

... или отправьте существующий репозиторий из командной строки

```
git remote add origin git@github.com:for-you-nastya/studying-Git.git
git branch -M main
git push -u origin main
```

Репозиторий создан и теперь нужно загрузить проект на данный удаленный репозиторий.

2. “Зальем” все наши изменения на Github (добавим удаленный репозиторий к локальному), используя команды:

```
git remote add origin https://github.com/for-you-nastya/studying-Git.git
```

```
git branch -M main
```

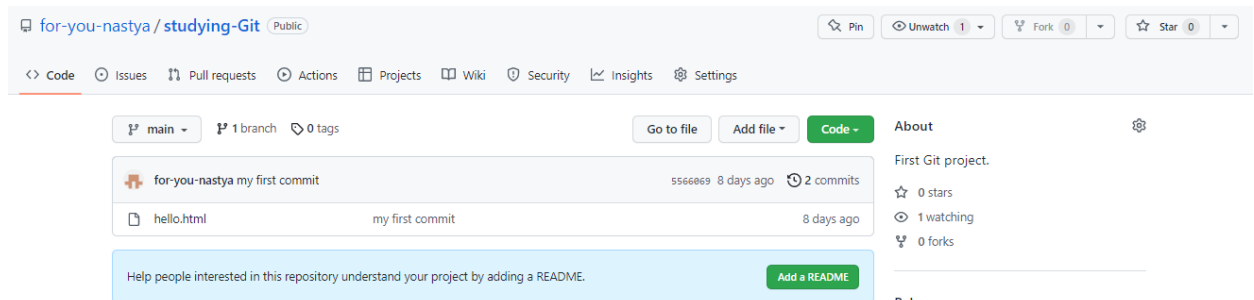
```
git push -u origin main
```

```
пользователь@LAPTOP-27F0P5IS MINGW64 /c/LabGit/hello (master)
$ git remote add origin https://github.com/for-you-nastya/studying-Git.git

пользователь@LAPTOP-27F0P5IS MINGW64 /c/LabGit/hello (master)
$ git branch -M main

пользователь@LAPTOP-27F0P5IS MINGW64 /c/LabGit/hello (main)
$ git push -u origin main
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 4 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (6/6), 643 bytes | 160.00 KiB/s, done.
Total 6 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/for-you-nastya/studying-Git.git
 * [new branch]      main -> main
Branch 'main' set up to track remote branch 'main' from 'origin'.
```

Посмотрим, что изменилось:



Первая команда Git добавляет информацию о связи между удаленным репозиторием с псевдонимом origin с локальным. Команда git branch -M main переименовывает ветку master на локальном репозитории в main. Таким образом, изменения, произошедшие на удаленном репозитории больше не конфликтуют с локальным хранилищем, в котором главная ветка стала также называться main. Третья команда позволяет запустить (отправить) локальный репозиторий на сервер.

3. Измените файл, сделайте второй коммит “My second commit”, проверьте статус и добавьте его к удаленному репозиторию.

# Hello,world!!!

Я учусь работать с Git

Git - система управления версиями с распределенной архитектурой

GitHub - служба размещения в Интернете репозиториях Git

```
пользователь@LAPTOP-27F0P5IS MINGW64 /c/LabGit/hello (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   hello.html

no changes added to commit (use "git add" and/or "git commit -a")

пользователь@LAPTOP-27F0P5IS MINGW64 /c/LabGit/hello (main)
$ git add -u

пользователь@LAPTOP-27F0P5IS MINGW64 /c/LabGit/hello (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   hello.html

пользователь@LAPTOP-27F0P5IS MINGW64 /c/LabGit/hello (main)
$ git commit -m "Second Commit"
[main be72de3] Second Commit
1 file changed, 10 insertions(+)

пользователь@LAPTOP-27F0P5IS MINGW64 /c/LabGit/hello (main)
$ git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 534 bytes | 178.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/for-you-nastya/studying-Git.git
5566069..be72de3  main -> main
```

На GitHub можно посмотреть все наши коммиты:

main

Commits on Oct 15, 2022

Second Commit

for-you-nastya committed 22 minutes ago



be72de3



Commits on Oct 5, 2022

my first commit

for-you-nastya committed 10 days ago



5566069



First Commit

for-you-nastya committed 10 days ago



69664c1

