

Практическая работа №2.

Работа с GitHub.

Цель работы:

Установка и работа в системе GitHub, установка и регистрация на GitHub.
Команды GitHub'а.

1. Работа с GitHub.

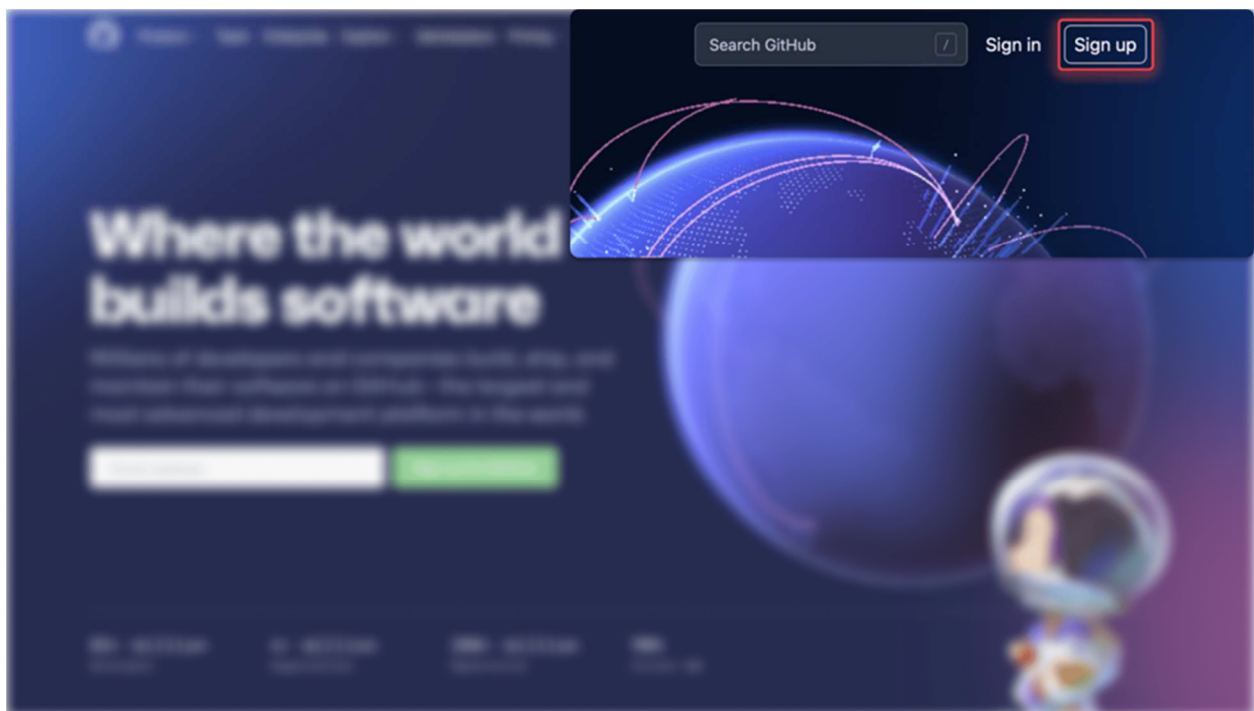
GitHub — крупнейший веб-сервис для хостинга IT-проектов и их совместной разработки. Веб-сервис основан на системе контроля версий Git и разработан компанией GitHub, Inc. Википедия

На его примере разберем, как работать с получением изменений и их отправкой.

Регистрация на GitHub

GitHub — веб-сервис, который основан на системе Git. Это такая социальная сеть для разработчиков, которая помогает удобно вести коллективную разработку IT-проектов. Здесь можно публиковать и редактировать свой код, комментировать чужие наработки, следить за новостями других пользователей.

Для работы с платформой нужно создать аккаунт. Для этого переходим по ссылке: <https://GitHub.com> и нажимаем по кнопке Sign up.

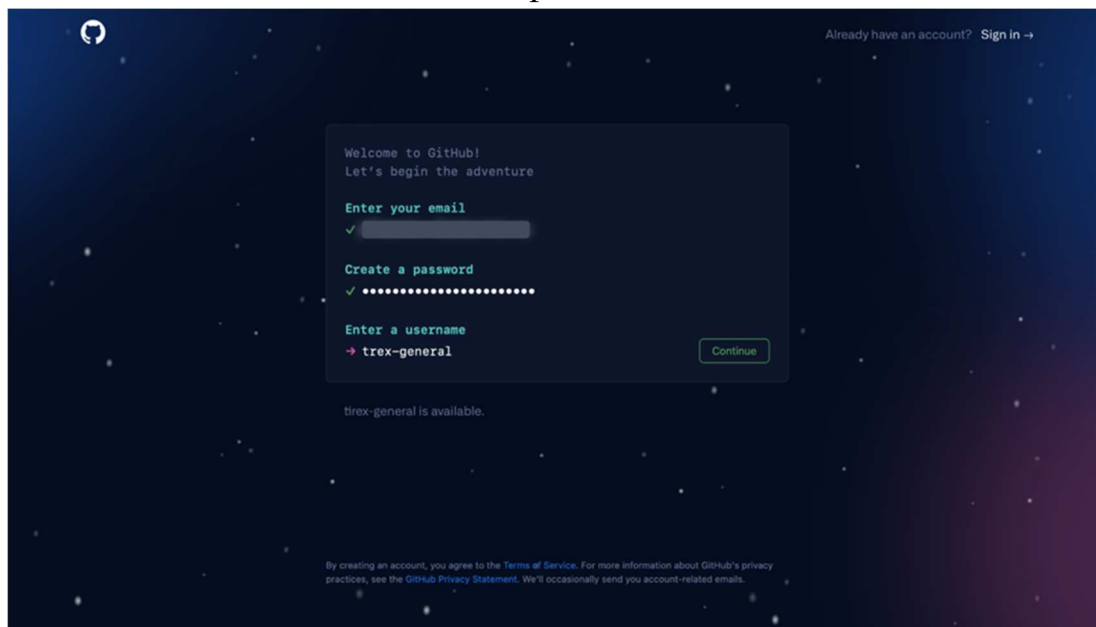


На странице регистрации вводим данные:

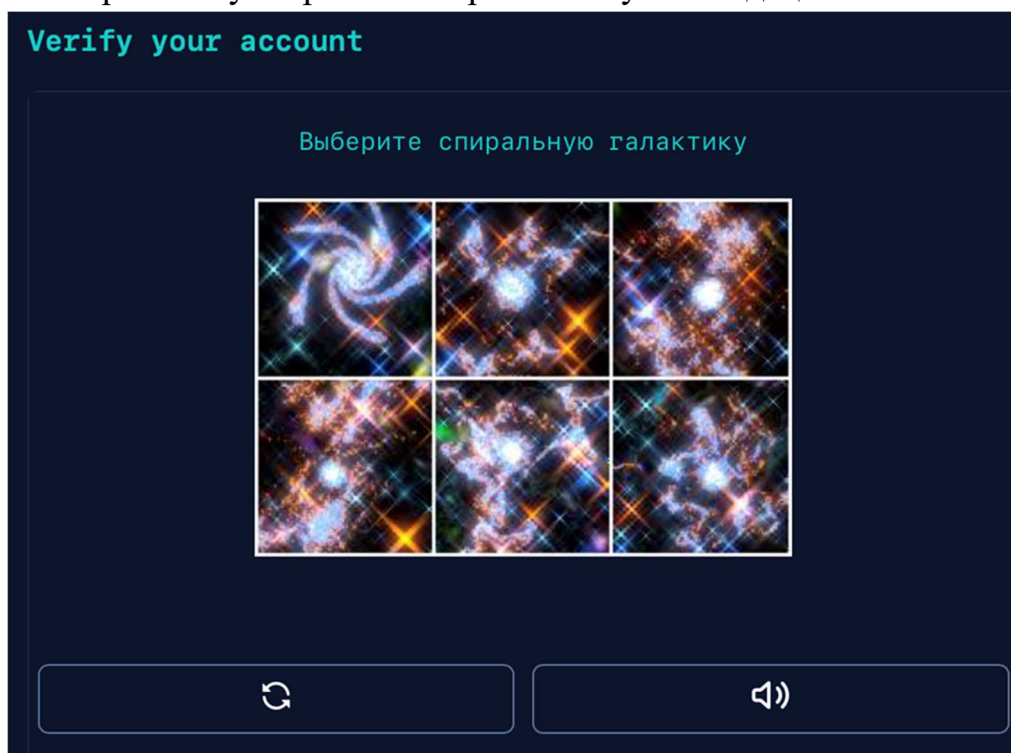
Адрес электронной почты. Если на почту уже был зарегистрирован аккаунт, на странице появится сообщение об ошибке: «Email is invalid or already taken».

Пароль. Система рекомендует использовать для пароля последовательность из 15 символов или 8, но с использованием хотя бы одной цифры и строчной буквы.

Имя пользователя. «Юзернейм» должен быть уникальным. При этом он не может начинаться или заканчиваться дефисом.

The image shows the GitHub account creation interface. At the top right, there is a link "Already have an account? Sign in →". The main content area has a dark background with a starry pattern. It contains a form with the following fields: "Enter your email" with a green checkmark, "Create a password" with a green checkmark and a password strength indicator (15 dots), and "Enter a username" with a red arrow and the text "trex-general". A "Continue" button is located to the right of the username field. Below the form, it says "trex-general is available." At the bottom, there is a small disclaimer: "By creating an account, you agree to the Terms of Service. For more information about GitHub's privacy practices, see the GitHub Privacy Statement. We'll occasionally send you account-related emails."

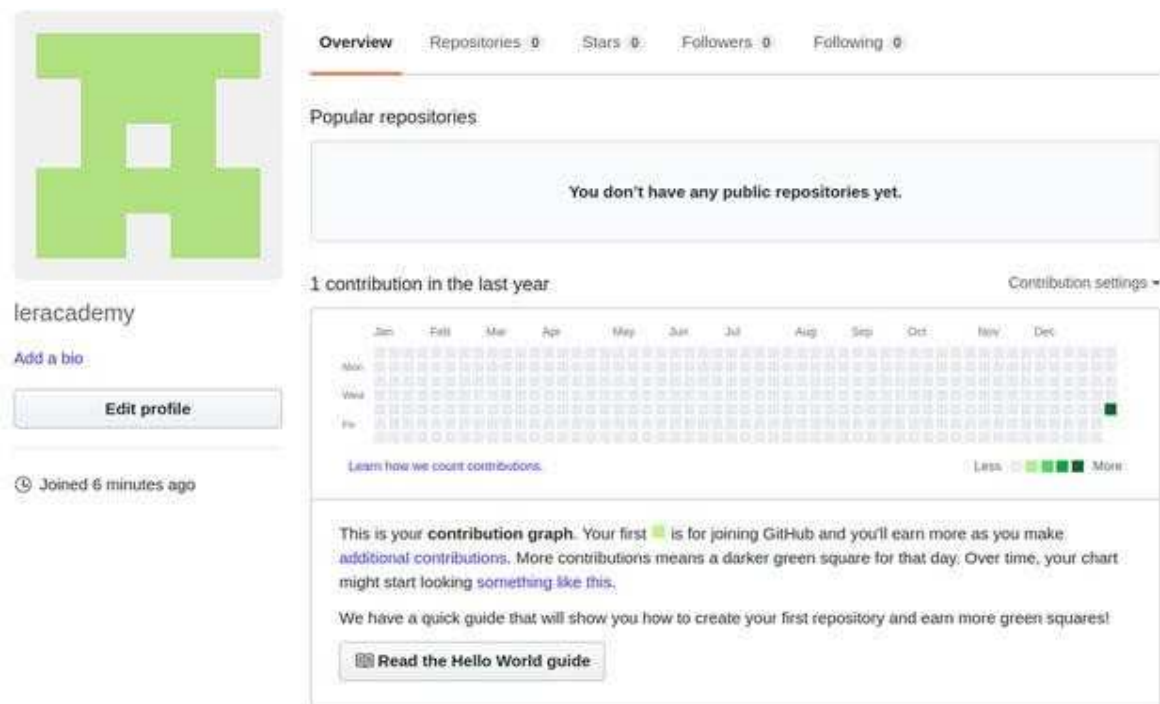
Теперь нужно нажать кнопку Continue, принять или отклонить предложение о подписке на рассылку и пройти экстравагантную валидацию:

The image shows the "Verify your account" screen. The title "Verify your account" is in green. Below it, the instruction "Выберите спиральную галактику" (Choose a spiral galaxy) is in green. There are six images of galaxies arranged in a 2x3 grid. The top-left image is a clear spiral galaxy, while the others are more complex or noisy. At the bottom, there are two buttons: a circular arrow icon for "Refresh" and a speaker icon for "Audio".

Затем подтвердите адрес электронной почты.

Во всплывающих окнах указывайте настройки на свое усмотрение. Для ознакомления и некоммерческого использования достаточно бесплатного тарифа. Регистрация завершена.

Переход в ваш профиль.



Так выглядит ваш профиль после регистрации.

Теперь у вас есть профиль на GitHub.

Сейчас у нас нет ни одного репозитория, и мы можем либо создать новый репозиторий, либо ответить (fork) от уже существующего чужого репозитория и вести собственную ветку разработки. Затем, при желании, свои изменения можно предложить автору исходного репозитория (Pullrequest).

7.Если Вы находитесь в своем профиле, то в верхней части окна видите один из пунктов меню «Repositories 0». Это означает, что у вас в данный момент нет созданных репозитория. На этой же вкладке есть кнопка

«New», которая позволяет создать новый репозиторий. Нажимаем её.

8.Открылось окно создания репозитория. Даем ему имя (например, GitTest) можно написать описание. Он должен быть публичным. Также установите галочку там, где предлагается поместить в репозиторий текстовый файл README. Нажмите кнопку «Create repository».

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere?
[Import a repository.](#)

Owner * Repository name *

 shl-dmitrij / 

Great repository names are GitTestShlaev is available. Need inspiration? How about [silver-octo-lamp?](#)

Description (optional)

☒  **Public**
Anyone on the internet can see this repository. You choose who can commit.

☐  **Private**
You choose who can see and commit to this repository.

Initialize this repository with:
Skip this step if you're importing an existing repository.

☒ **Add a README file**
This is where you can write a long description for your project. [Learn more.](#)

Add .gitignore
Choose which files not to track from a list of templates. [Learn more.](#)

Choose a license
A license tells others what they can and can't do with your code. [Learn more.](#)

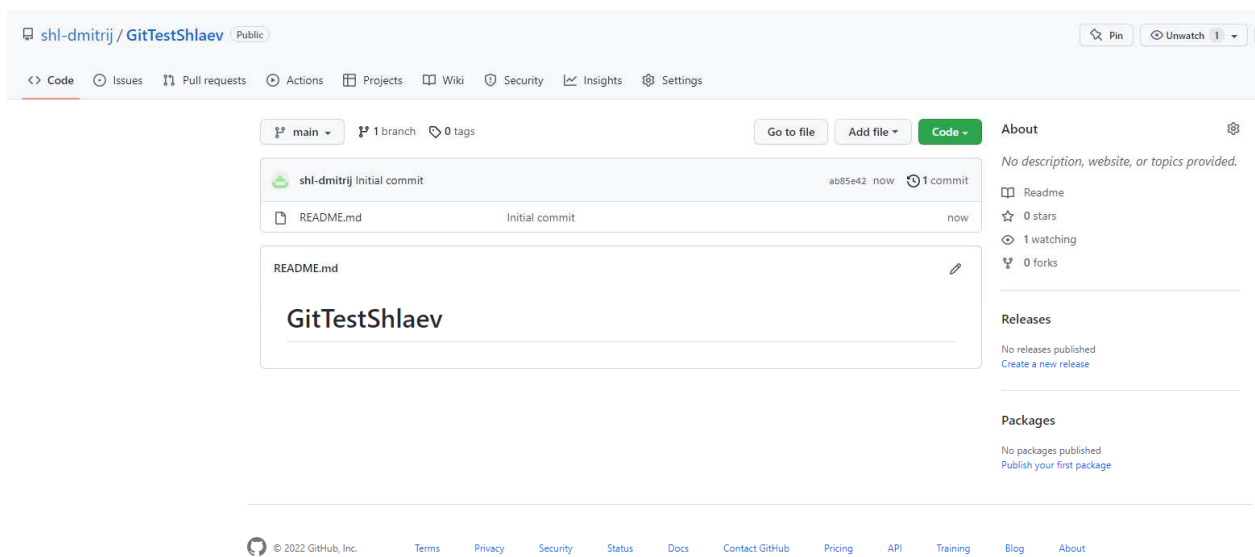
This will set  **main** as the default branch. Change the default name in your [settings](#).

 You are creating a public repository in your personal account.

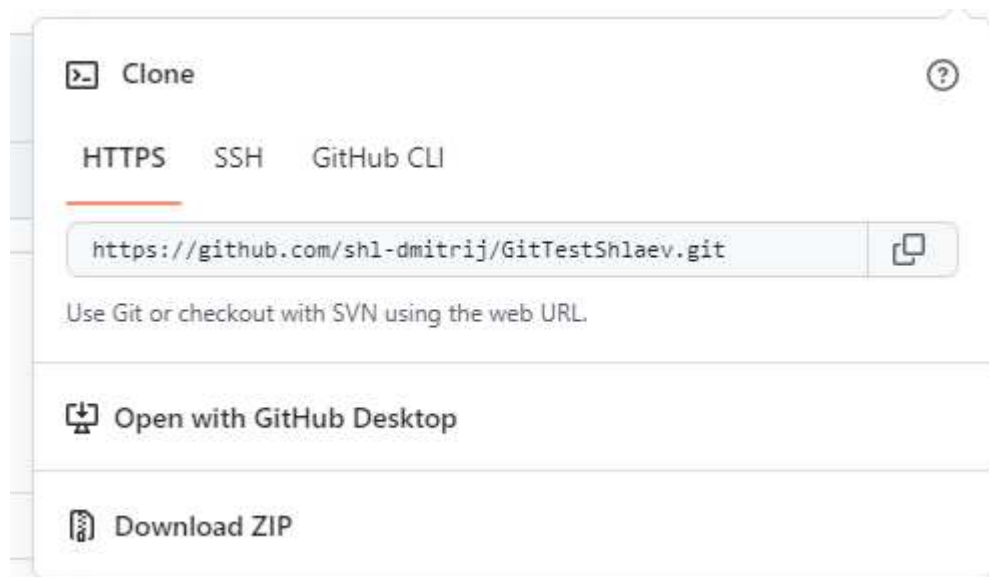
[Create repository](#)

Когда вы создаете новый репозиторий таким образом, на GitHub появляется папка и для нее выполняется команда `git init`, т.е. происходит то же самое, что мы делали выше в ручную.

10. После создания репозитория вы в него попадаете и видите, что в нем в данный момент находится один файл – `README.md`, который был создан автоматически:



11. Далее нам нужно скопировать в буфер обмена адрес нашего репозитория на GitHub. Для этого нажмите кнопку «Code» и в открывшемся окне кнопку справа от адреса ссылки:



12. Откройте консоль GitBash (она должна быть у вас открыта, если нет – перейдите в папку tmp, с которой мы работали). Даем команду

git clone https://GitHub.com/DrovosekovaT/GitTestDrovosekova.git test (у каждого своя)

(вставьте из буфера обмена свой адрес репозитория и после него через пробел название папки, где будет создан клон удаленной папки-репозитория, в данном случае test).

Эта команда создаст в папке test копию репозитория с указанного адреса с GitHub, автоматически GitHub подключит к локальному репозиторию как удалённый и назовёт его origin, т.е. оригинал (перейдите в папку test и проверьте это командой `git remote -v`), автоматически в нашем локальном репозитории создаст

локальную ветку master (проверьте это командой `git gstatus`), которую свяжет с веткой master в удаленном репозитории на GitHub.

13. Теперь рассмотрим, как синхронизировать локальный репозиторий (клон) с оригиналом, который находится на GitHub. В клонированной папке test создайте локально файл Hello.txt, напишите в нем текст «Hello!!!», закройте его, сохранив изменения. Теперь у нас репозиторий содержит 2 файла, а на GitHub только один. Для того, чтобы зафиксировать изменения, выполните команду `git add Hello.txt` и создайте коммит:

```
Танюшка@NoteBook MINGW64 /d/tmp/test
git add Hello.txt

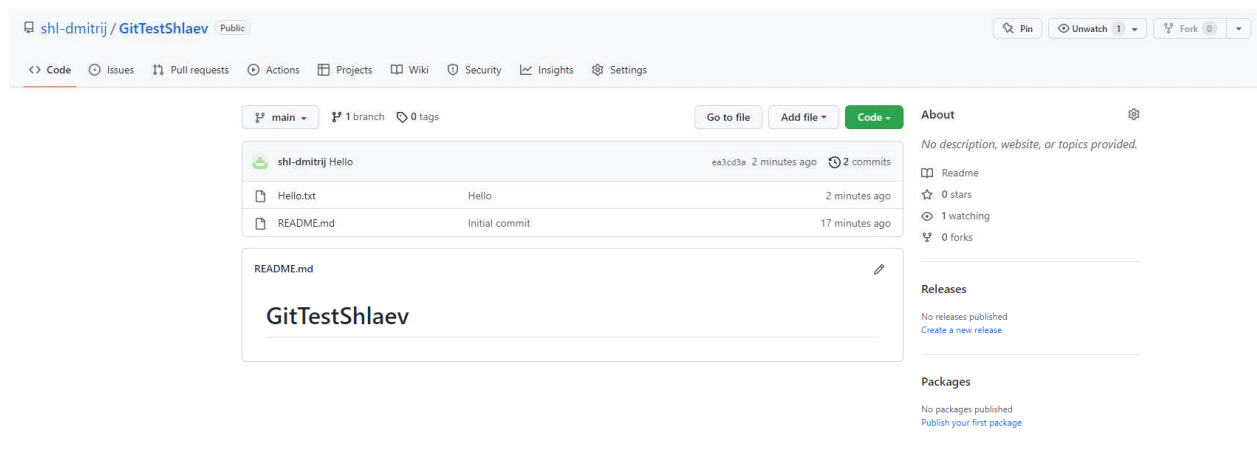
Танюшка@NoteBook MINGW64 /d/tmp/test
git commit -m "Hello"
[master 3075eda] Hello
1 file changed, 1 insertion(+)
create mode 100644 Hello.txt
```

14. Выполните команду `git push`. Эта команда берет имеющиеся локальные изменения и отправляет их в удалённый репозиторий. В ответ на запрос введите своё имя на GitHub и пароль.

```
Танюшка@NoteBook MINGW64 /d/tmp/test
git push
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 306 bytes | 306.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/DrovosekovaT/GitTestDrovosekova.git
2519672..3075eda master -> master

Танюшка@NoteBook MINGW64 /d/tmp/test
```

15. Откройте репозиторий на сайте GitHub, убедитесь, что файл добавился:



Если вы откроете добавившийся файл Hello.txt, то увидите свой текст. Также вы видите название коммита во втором столбце.

Итоги:

`gitpull` – не просто «достает изменения» из удаленного репозитория, но и пытается применить изменения из удалённого репозитория к нашему (вливать удалённую ветку в локальную). Ветки будут изучаться далее.

`gitpush` – отправляет наши наборы изменений на удалённый репозиторий.

Таким образом, для чего применяется GitHub в реальной работе?

1. GitHub очень удобно использовать для быстрого бесплатного создания репозитория в интернете.

Для чего может понадобиться репозиторий в интернете?

1. Дать кому-то на него ссылку, чтобы этот человек мог удобно посмотреть на код, хранящийся там, удобно просмотрел историю изменений.

2. GitHub может служить посредником для коллективной работы.

3. Если вы работаете за несколькими компьютерами, можно клонировать на них репозиторий с GitHub и синхронизировать через него все версии.

Задание для самостоятельной работы

1. Создайте у себя на компьютере еще один репозиторий ("второй").

2. Свяжите его с локальным. так же созданным вторым, получив результат, произведите его коммит.

3. Создайте файл в блокноте скопируйте из Git Bush результат и разместите файл на платформе для проверки преподавателем.