

# Урок 4

## Знакомство с GitHub

2024

 MobyDev



# Что такое GitHub

**GitHub** — это платформа для хостинга и совместной разработки программного обеспечения, основанная на системе контроля версий Git. Она позволяет разработчикам хранить, управлять и отслеживать изменения в коде своих проектов.

GitHub является важным инструментом для современных разработчиков, обеспечивая эффективное управление кодом и облегчая совместную работу над проектами.



# Необходимость в разработке

- **Репозитории (Repositories):** GitHub позволяет создавать репозитории, которые являются хранилищами для кода. Репозитории могут быть публичными и приватными.
- **Контроль версий (Version Control):** С помощью Git можно отслеживать изменения в коде, возвращаться к предыдущим версиям и управлять различными ветками разработки.
- **Совместная работа (Collaboration):** GitHub облегчает совместную работу над проектами, позволяя нескольким разработчикам вносить изменения, предлагать улучшения и обсуждать код через пул-реквесты (pull requests).
- **Пул-реквесты (Pull Requests):** Эта функция позволяет разработчикам предлагать изменения в коде, которые могут быть обсуждены и рассмотрены перед слиянием с основной веткой.

Сайт GitHub <https://github.com/> (для начала необходимо будет пройти регистрацию)

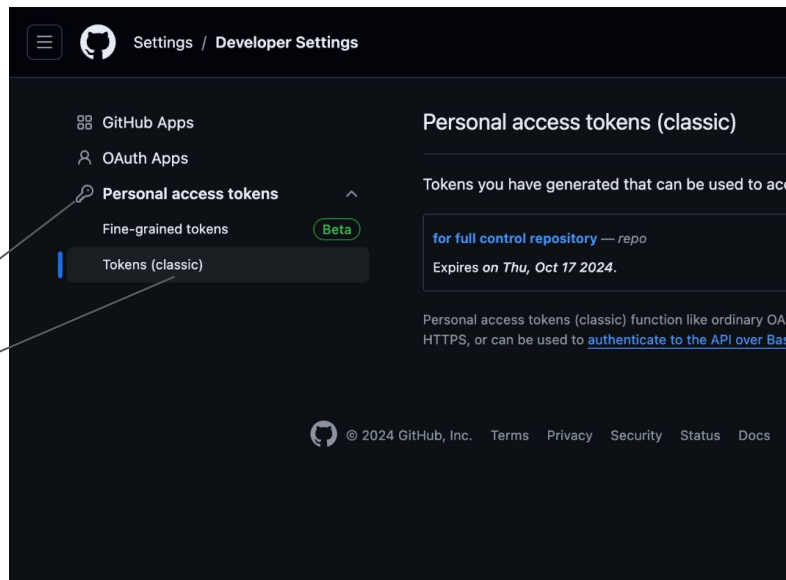


# Access Token

**Access Token (токен доступа)** — это строка, которая предоставляет безопасный и ограниченный доступ к ресурсам и сервисам. В контексте систем контроля версий, таких как GitHub, Access Token используется **для аутентификации и авторизации** пользователей при выполнении операций с репозиториями, таких как **clone, push, pull** и управление репозиториями.

## Получение Access Token:

- Войдите в свою учетную запись GitHub;
- Перейдите к настройкам (Settings): В правом верхнем углу страницы нажмите на значок профиля;
- Перейдите в раздел "Developer settings": В левой боковой панели прокрутите вниз и выберите "Developer settings";
- Выберите "Personal access tokens";
- Выберите Tokens (classic).



# Access Token

## Получение Access Token:

- Нажимаете на кнопку **"Generate new token"**;
- Выбираете **"Generate new token (classic)"**;
- Далее вводите **"Note"** (для чего нужен этот токен) и **"Expiration"** (срок истечения токена);
- Далее чекбокс **"Select scopes"** (на что может влиять данный токен), выбираем только **"repo"**;
- Прокручиваем вниз и нажимаем на **"Generate token"**.

Search type [J] to search

### New personal access token (classic)

Personal access tokens (classic) function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

**Note**

for full control repository

What's this token for?

**Expiration**

90 days The token will expire on Thu, Oct 17 2024

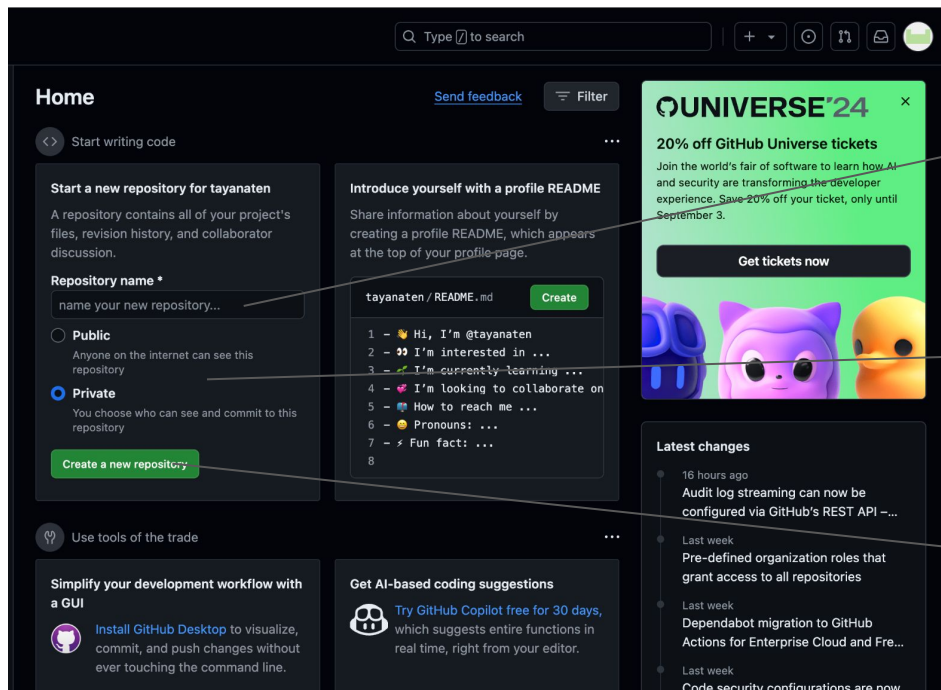
**Select scopes**

Scopes define the access for personal tokens. [Read more about OAuth scopes.](#)

<input checked="" type="checkbox"/> <b>repo</b>	Full control of private repositories
<input checked="" type="checkbox"/> repo:status	Access commit status
<input checked="" type="checkbox"/> repo_deployment	Access deployment status
<input checked="" type="checkbox"/> public_repo	Access public repositories
<input checked="" type="checkbox"/> repo:invite	Access repository invitations
<input checked="" type="checkbox"/> security_events	Read and write security events
<input type="checkbox"/> workflow	Update GitHub Action workflows
<input type="checkbox"/> write:packages	Upload packages to GitHub Package Registry
<input type="checkbox"/> read:packages	Download packages from GitHub Package Registry
<input type="checkbox"/> delete:packages	Delete packages from GitHub Package Registry
<input type="checkbox"/> admin:org	Full control of orgs and teams, read and write org projects



# Создание репозитория в GitHub



Название репозитории

Выбор видимости  
между созданием  
публичного и  
приватного репозитория

Кнопка создания  
репозитории



# Загрузка проектов в GitHub

После создания репозитория в **GitHub**, необходимо будет открыть терминал или командную строку (для Windows) и с помощью команд ниже загрузить проект в созданный вами репозиторий в GitHub.

## ...or create a new repository on the command line

```
echo "# ggg" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/tayanaten/ggg.git
git push -u origin main
```

Здесь должен быть ваш ник (username) в GitHub

Название вашего репозитория



# Загрузка репозиторий с GitHub

Если вы хотите скопировать весь репозиторий с GitHub на ваш локальный компьютер, используйте команду **git clone**. Это создаст копию удаленного репозитория на вашем компьютере.

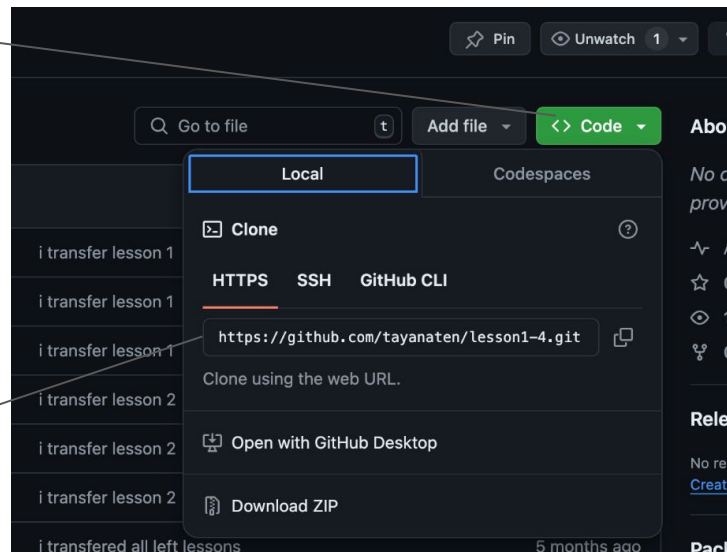
- Перейдите на страницу **репозитория на GitHub**.
- Нажмите на кнопку **"Code"** и скопируйте URL репозитория (HTTPS, SSH или GitHub CLI).
- Затем откройте **терминал** или командную строку и выполните команду:

```
git clone https://github.com/your-username/your-repository.git
```

CodeImage

Замена

**https://github.com/your-username/your-repository.git** на URL вашего репозитория.





# Повторная загрузка изменений

Вы можете внести изменения в ваш локальный репозиторий. Например, вы можете **редактировать, добавлять или удалять** файлы.

- Откройте терминал;
- Чтобы увидеть, какие файлы были изменены, выполните команду: **"git status"**;
- Добавьте измененные файлы в индекс с помощью команды **"git add"** (для загрузки всех файлов)/ **"git add имя\_файла"** (для загрузки определенного файла);
- Создайте коммит, описывающий ваши изменения: **"git commit -m "Описание ваших изменений"";**
- Отправьте изменения на GitHub: **"git push origin main"**.



# Урок 4

## Знакомство с Github

2024

 MobyDev

