

Алгоритмика

МОДУЛЬ 3. УРОК 4

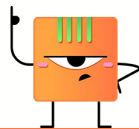
Облако Часть 2



Скачать методичку

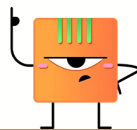
Давайте **проверим знания**, оставшиеся
с последнего урока.

Сыграем в **викторину!**

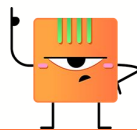


Какие вопросы оказались сложными?

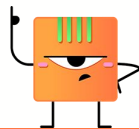
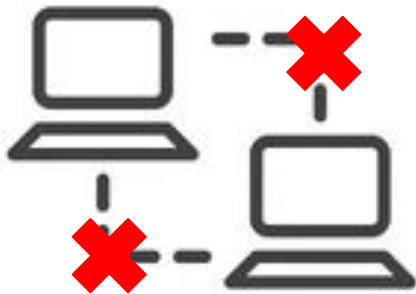
Давайте разберем их вместе!



**На данный момент у нас проект
размещен в 2 местах: на удаленном
сервере и локальном компьютере**



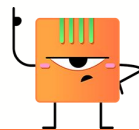
Эти компьютеры друг с другом не
связаны. И изменения в одном месте не
отобразятся в другом.



Для контроля версии и совместной разработки используются git и github



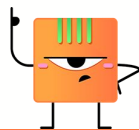
git



GIT

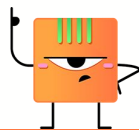


Распределённая система управления версиями. Проект был создан Линусом Торвальдсом для управления разработкой ядра Linux, первая версия выпущена 7 апреля 2005 года.





Крупнейший веб-сервис для хостинга IT-проектов и их совместной разработки. Веб-сервис основан на системе контроля версий Git



Установим GIT

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

Downloads



macOS

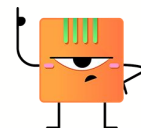


Windows



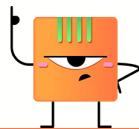
Linux/Unix

Older releases are available and the Git source repository is on GitHub.



После установки в **терминале** должна появиться команда **git**

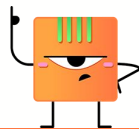
```
PS C:\Users\Сергей\lib\test-1> git
usage: git [-v | --version] [-h | --help] [-C <path>] [-c <name>=<value>]
          [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
          [-p | --paginate | -P | --no-pager] [--no-replace-objects] [--bare]
          [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
          [--config-env=<name>=<envvar>] <command> [<args>]
```



Сообщим **git** кто мы такие:

`git config --global user.name "Ваш username"`

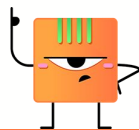
`git config --global user.email "Ваша электронная почта"`



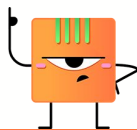
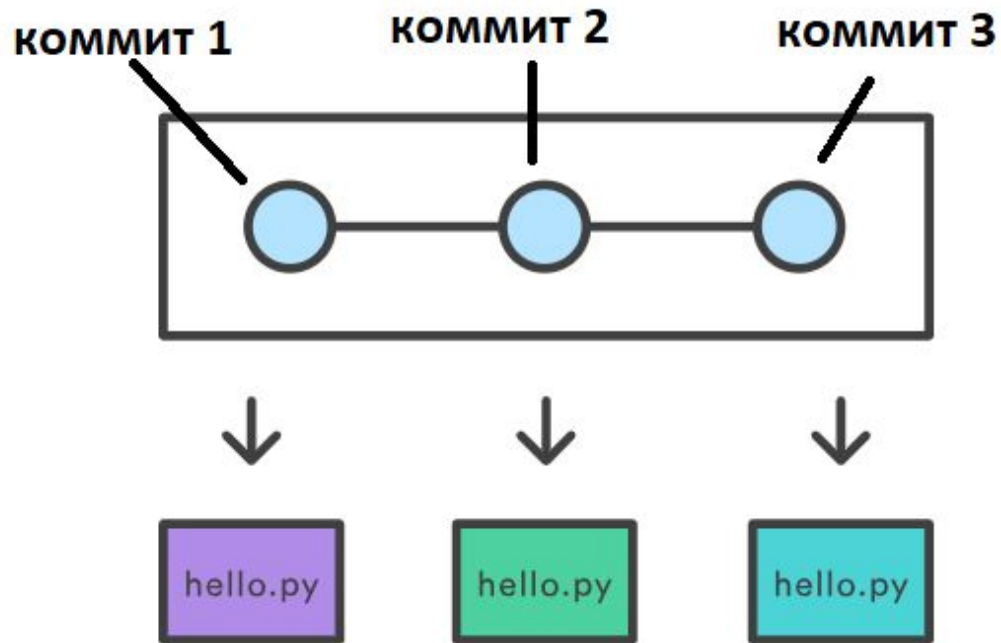
Работа с **git**:

Git позволяет создавать снимоты проекта во времени(**КОММИТЫ**).
Каждый из таких снимков является **самостоятельной неизменяемой версией проекта**.

Такой подход позволяет легко находить, какие изменения привели к **“поломке”** и оперативно **“откатить”** репозиторий на стабильную версию.



Recording Snapshots (Git)



Основные команды git:

git init - инициализирует гит репозиторий

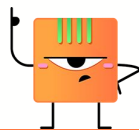
git add - добавляет файлы для коммита

git commit - создает коммит изменений

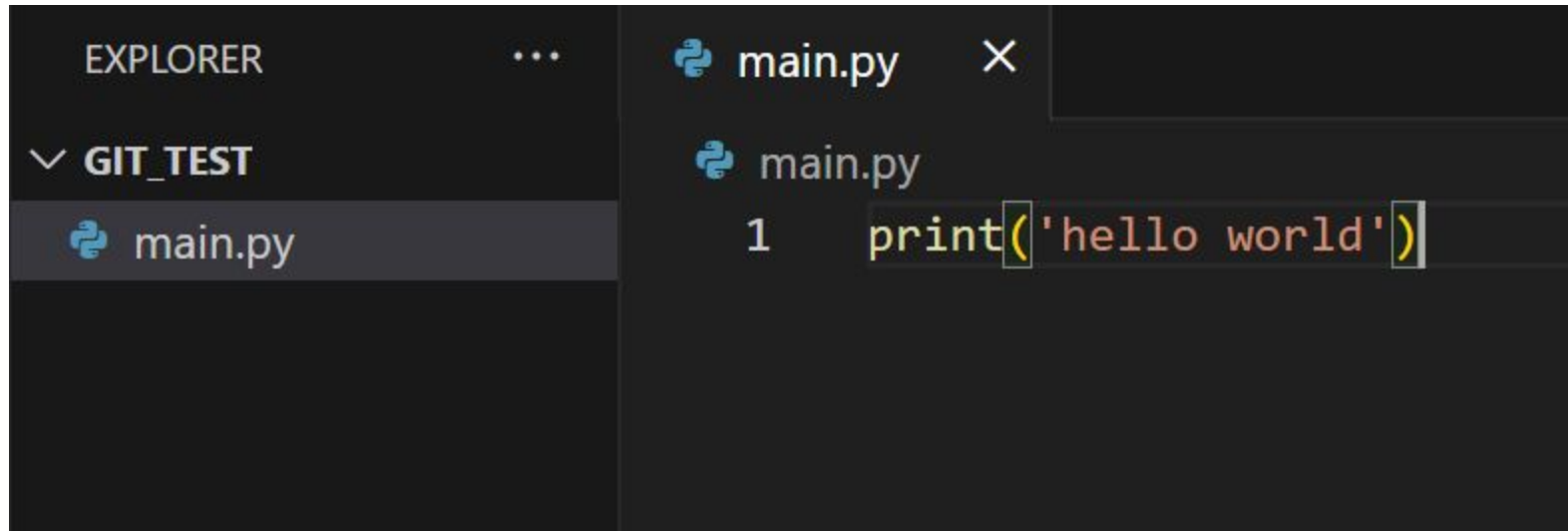
git push - отправляет изменения на удаленный сервер

git pull - получает изменения из удаленного сервера

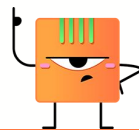
git clone - клонирует существующий репозиторий с удаленного сервера



Создадим **простой проект** с 1 файлом

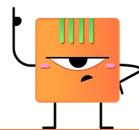
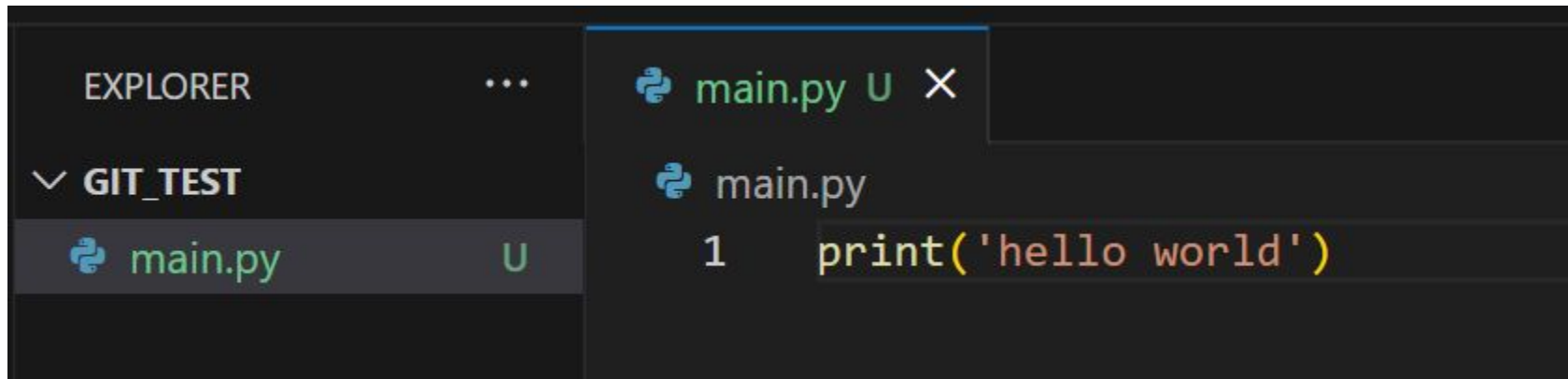


The screenshot shows a code editor interface with a dark theme. On the left, the 'EXPLORER' sidebar displays a project named 'GIT_TEST' with a single file 'main.py' listed below it. The main editor area shows the content of 'main.py', which contains a single line of Python code: `print('hello world')`. The code is written in a monospaced font with syntax highlighting: `print` is blue, `(` and `)` are yellow, and the string `'hello world'` is orange. A cursor is positioned at the end of the line.

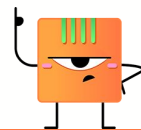
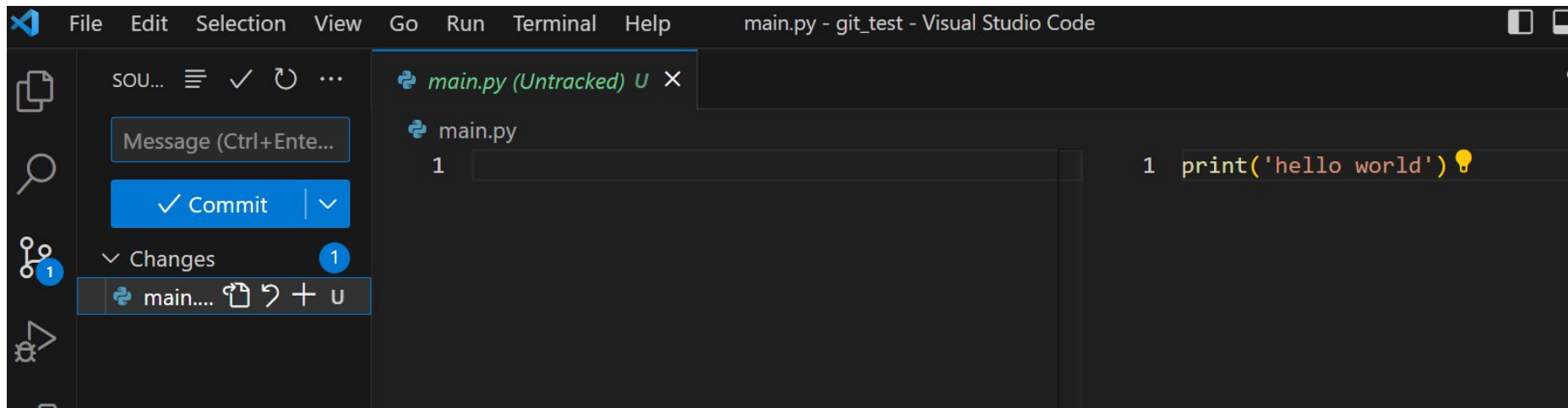


Инициализируем git:

```
PS C:\Users\Сергей\lib\git_test> git init .  
Initialized empty Git repository in C:/Users/Сергей/lib/git_test/.git/
```

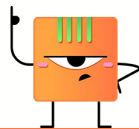
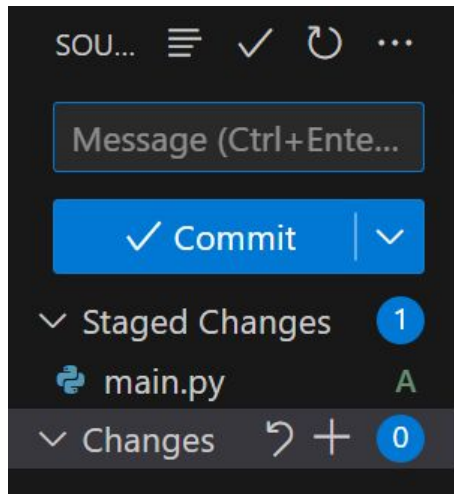


Во вкладке git **VSCode** можно посмотреть **изменения внесенные с** **момента последнего коммита**



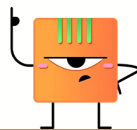
Для того, чтобы сделать **КОММИТ** для
начала нужно добавить файлы для
коммита **git add**

```
PS C:\Users\Сергей\lib\git_test> git add .
```

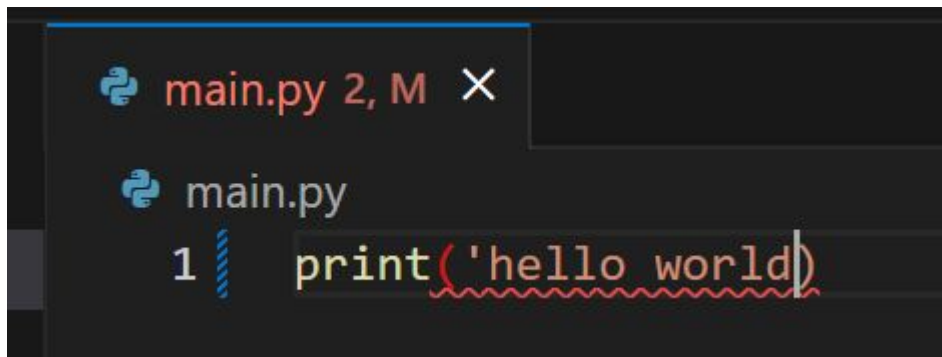


Далее выполняем **git commit -m** с кратким описанием изменений

```
PS C:\Users\Сергей\lib\git_test> git commit -m 'первый комит, создал файл main.py '  
[master (root-commit) 4c75f4a] первый комит, создал файл main.py  
1 file changed, 1 insertion(+)  
create mode 100644 main.py
```

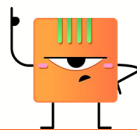


Допустим мы продолжили работу и наш код **сломался. И мы **не знаем** как это исправить**



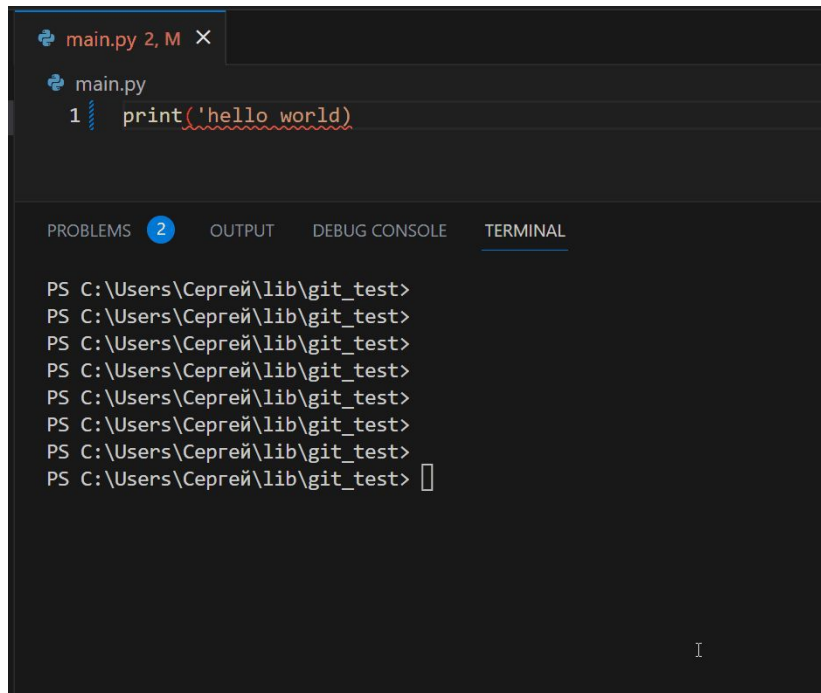
The screenshot shows a code editor window with a dark background. At the top, a tab is labeled 'main.py 2, M' with a Python icon and a close button. Below the tab, the code 'print('hello world')' is visible on line 1. The closing single quote of the string is missing, and the text 'hello world' is underlined with a red wavy line, indicating a syntax error.

```
main.py 2, M X  
main.py  
1 print('hello world)
```

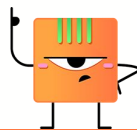


Чтобы **отменить все изменения** не внесенные в коммит выполним:

`$ git reset --hard`

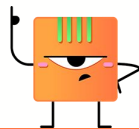
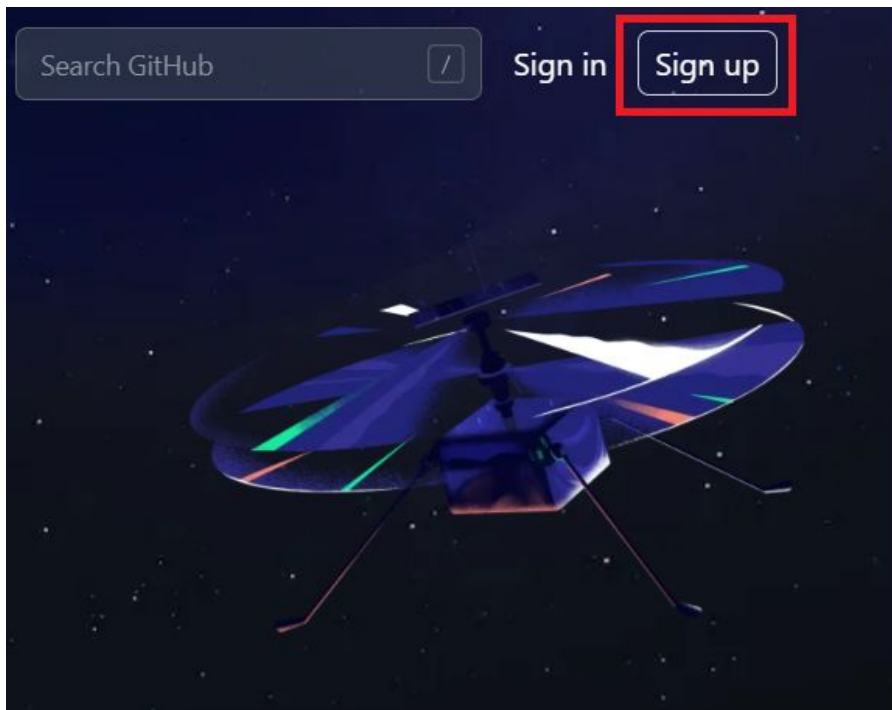


The screenshot shows a code editor with a file named `main.py` containing a single line of Python code: `print('hello world')`. Below the editor, the `TERMINAL` tab is active, displaying a series of PowerShell commands and their outputs. The commands are `PS C:\Users\Ceprей\lib\git_test>`, which are repeated eight times, followed by a cursor. The terminal output is empty, indicating that the commands have not yet been executed.

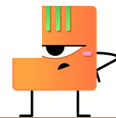
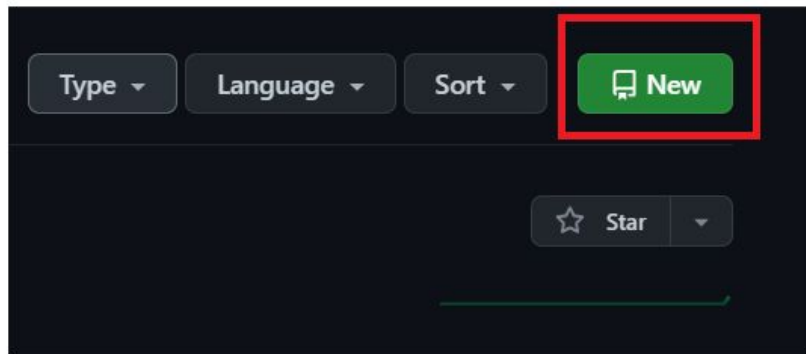
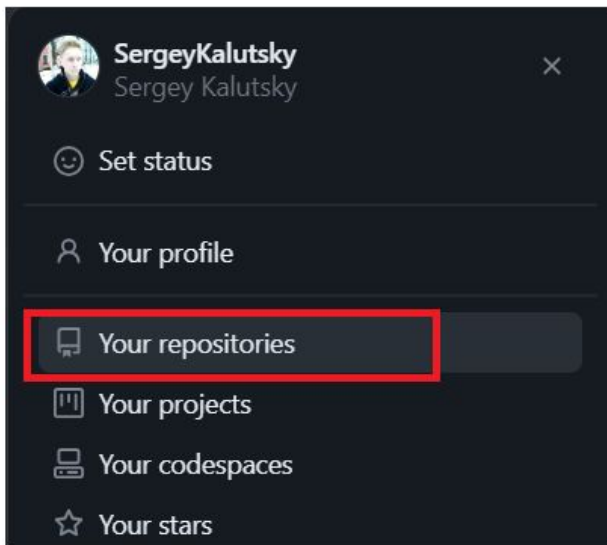


GitHub - создадим аккаунт (если нет)

<https://github.com/>



Создадим удаленный репозиторий под наш **тестовый проект**



Required fields are marked with an asterisk (*).

Owner *



Repository name *

git_test

✔ git_test is available.

Great repository names are short and memorable. Need inspiration? How about **urban-garbanzo** ?

Description (optional)



Public

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



Private

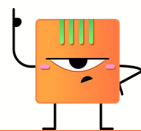
You choose who can see and commit to this repository.

Initialize this repository with:



Add a README file

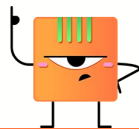
This is where you can write a long description for your project. [Learn more about READMEs.](#)



Вставляем команды постепенно в локальный репо:

...or push an existing repository from the command line

```
git remote add origin https://github.com/SergeyKalutsky/git_test.git  
git branch -M main  
git push -u origin main
```



Выполнение последней команды потребуется авторизации

Connect to GitHub



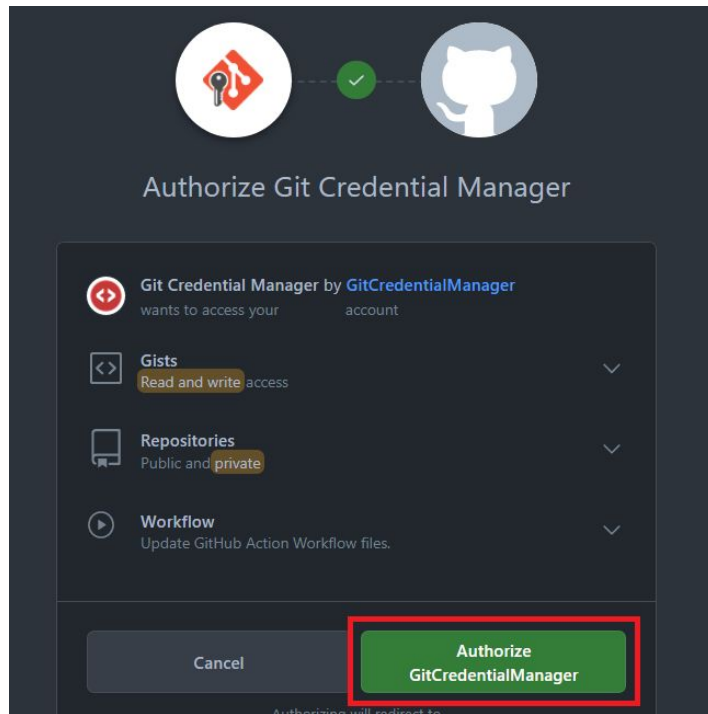
GitHub
Sign in

Browser/Device


Token




Sign in with your browser


Sign in with a code




```
PS C:\Users\Сергей\lib\git_test> git remote add origin https://github.com/SergeyKalutsky/git_test.git
PS C:\Users\Сергей\lib\git_test> git branch -M main
PS C:\Users\Сергей\lib\git_test> git push -u origin main
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 281 bytes | 281.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
```

 **git_test** PublicPin Unwatch 1

 **main**  **1 branch**  **0 tags**Go to file Add file <> Code

 **SergeyKalutsky** первый комит, создал файл main.py

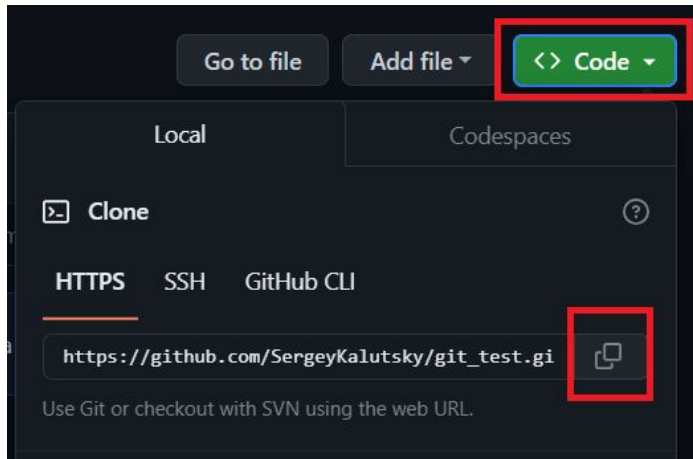
4c75f4a 36 minutes ago 1 commit

 **main.py** первый комит, создал файл main.py 36 minutes ago

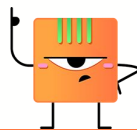
Help people interested in this repository understand your project by adding a README. Add a README

Теперь мы можем клонировать проект на **pythonanywhere**

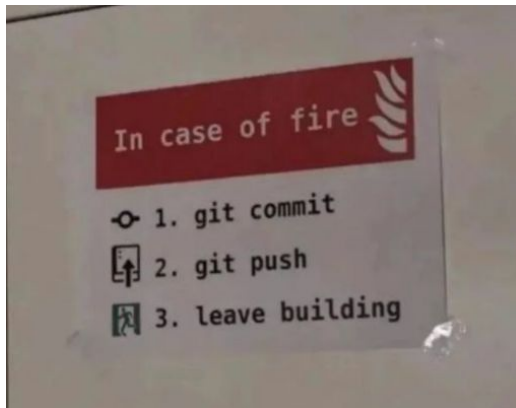
`git clone [URL git репо]`



```
08:03 ~ $ git clone https://github.com/SergeyKalutsky/git_test.git
Cloning into 'git_test'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 3 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 261 bytes | 4.00 KiB/s, done.
08:04 ~ $ ls
git_test
08:04 ~ $ cd git_test/
08:04 ~/git_test (main)$ python git_test
python: can't open file '/home/skalutsky2/git_test/git_test': [Err
08:04 ~/git_test (main)$ python main.py
hello world
08:04 ~/git_test (main)$
```



Для **обновления** проекта на удаленном сервере:

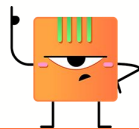


Локально

git add
git commit
git push

Удаленный сервер

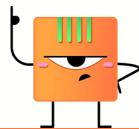
git pull



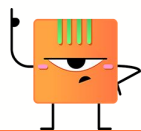
```
main.py ×  
main.py  
1 print('Новые изменения')
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
branch 'main' set up to track 'origin/main'.  
PS C:\Users\Сергей\lib\git_test> git add .  
PS C:\Users\Сергей\lib\git_test> git commit -m 'изменил текст pring'  
[main 752516a] изменил текст pring  
1 file changed, 1 insertion(+), 1 deletion(-)  
PS C:\Users\Сергей\lib\git_test> git push  
Enumerating objects: 5, done.  
Counting objects: 100% (5/5), done.  
Writing objects: 100% (3/3), 307 bytes | 307.00 KiB/s, done.  
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0  
To https://github.com/SergeyKalutsky/git_test.git
```

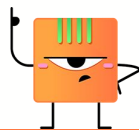


```
08:04 ~/git_test (main)$ git pull
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Total 3 (delta 0), reused 3 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 287 bytes | 5.00 KiB/s, done.
From https://github.com/SergeyKalutsky/git_test
   4c75f4a..752516a  main       -> origin/main
Updating 4c75f4a..752516a
Fast-forward
 main.py | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)
08:24 ~/git_test (main)$ python main.py
Новые изменения
08:24 ~/git_test (main)$
```



Задания:

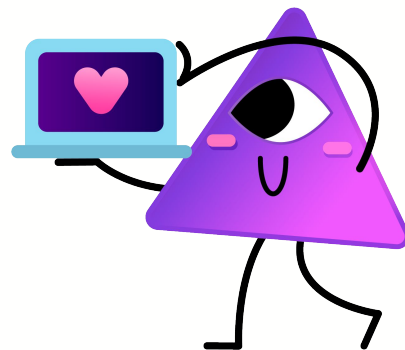
- Удалить проект мафии на удаленном сервере **pythonanywere**
- Инициализировать **git репо** в локальном проекте мафии(`git init .`)
- Сделать коммит (`git add . / git commit -m '...'`)
- Создать **github репо** для мафии
- Добавить **remote** и “запустить” проект мафии на github
- Клонировать проект на удаленном сервере





МОДУЛЬ 3. УРОК 4

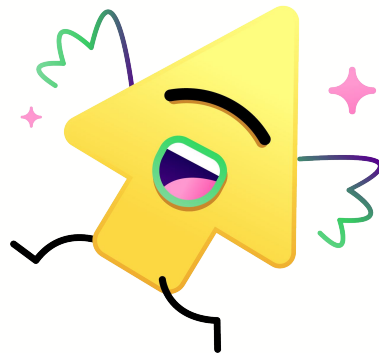
Работа с git и GitHub





МОДУЛЬ 3. УРОК 4

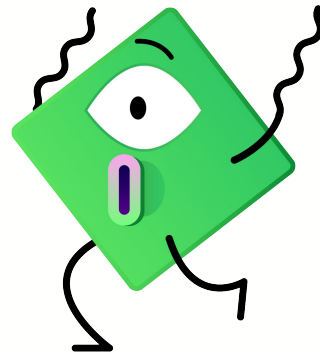
Перерыв





МОДУЛЬ 3. УРОК 4

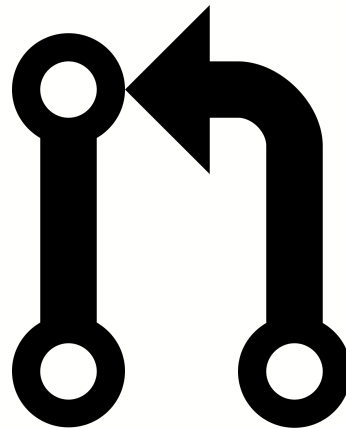
Совместная разработка



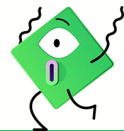
Существует два основных способа совместной разработки через **github**



Стать постоянным коллаборантом

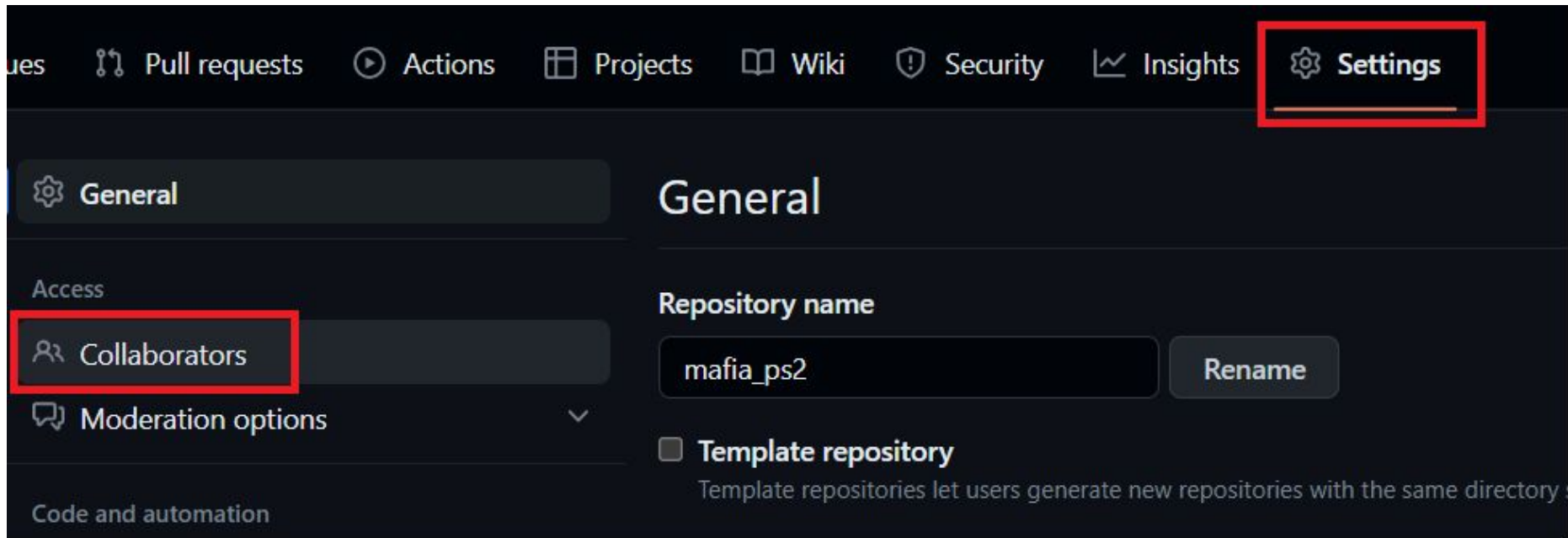


PULL REQUEST



Добавить пользователя в коллабораторы

В настройках выбираем меню **Collaborators**



Добавить пользователя в **коллатораторы**

Находим по имени пользователя, кого хотим добавить.



You haven't invited any collaborators yet

Add people

Q Sergey



Sergey

Invite collaborator



Sergey

SergeySH1980 • Invite collaborator



Sergey

SergeyGvozdetkiy • Invite collaborator



Sergey

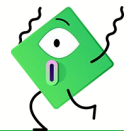
SergeYdeV65Y • Invite collaborator



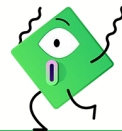
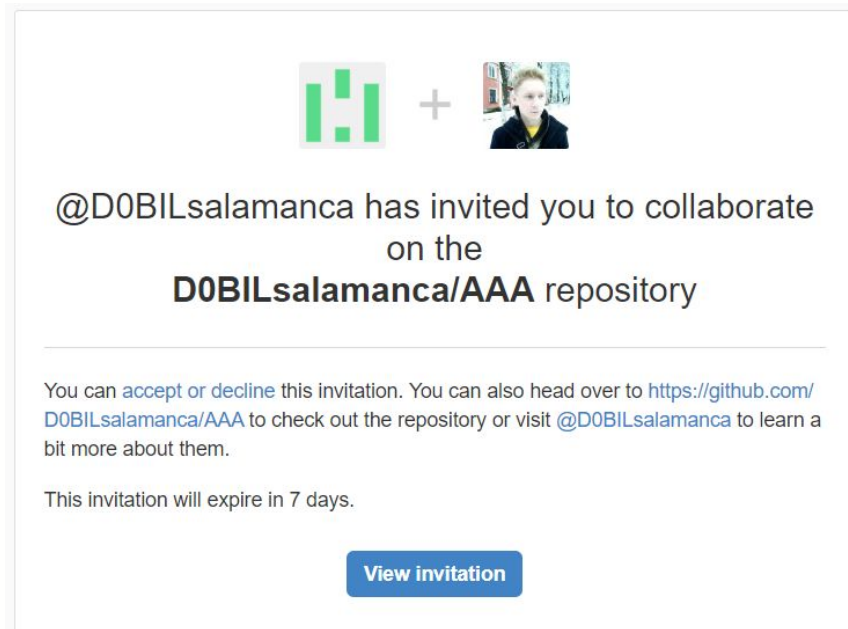
Sergey

Sergey-TLT • Invite collaborator

Select a collaborator above



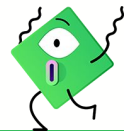
Для того, чтобы стать коллаборантом
надо принять запрос, который придет на
электронную почту.



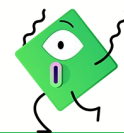
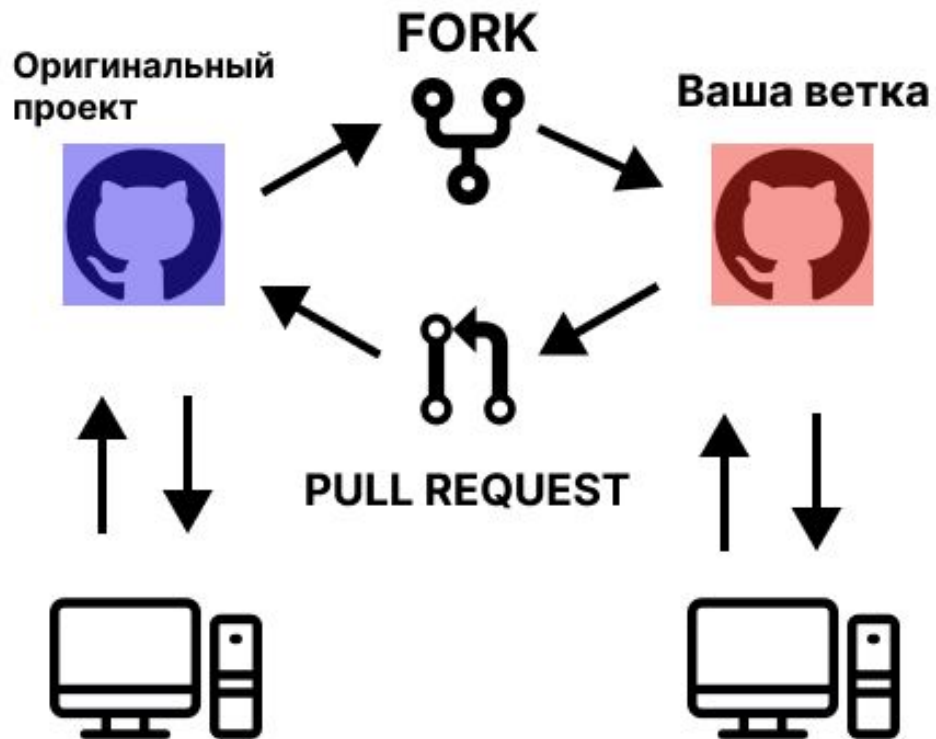
После добавления в коллабораторы вы получаете **такие же права по работе с проектом.**

Задание:

- Добавить 1-2 коллаборантов в свой репо

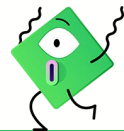


Pull request схема



Pull request описание

- Создаем ветку проекта под своим аккаунтом
- Клонировем проект локально
- Вносим изменения
- Отправляем изменения в **GitHub**
- Делаем **Pull Request**
- Ждем пока автор оригинального проекта добавит(**merge**)
наши изменения в свой проект



Watch 1 Fork 2 Star 0

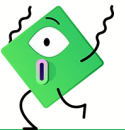
Fork your own copy of kakashi1464/1464

Go to file Add file > <> Code > About


test

 **1464** Public

forked from [kakashi1464/1464](#)



Commits

 main ▾

 Commits on Jul 8, 2023

fixed file name



SergeyKalutsky committed 2 days ago

второй комит

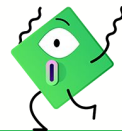


kakashi1464 committed 2 days ago

первый коммит



kakashi1464 committed 2 days ago



Conversation 2

Commits 1

Checks 0



Files changed 1



SergeyKalutsky commented 2 days ago

Переименовал файл



  fixed file name



kakashi1464 commented 2 days ago

SUUUI



kakashi1464 commented 2 days ago

thx

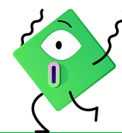


kakashi1464 merged commit **11524cd** into **kakashi1464:main** 2 days ago



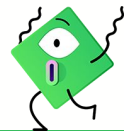
Pull request closed

If you wish, you can delete this fork of kakashi1464/1464 in the [settings](#).



Задание

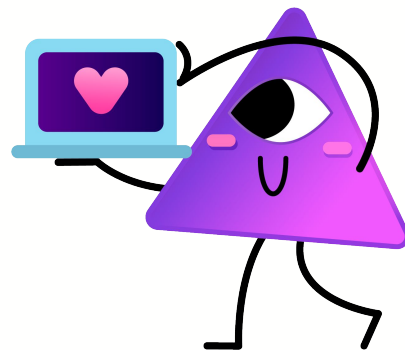
- Сделать **pull request** в аккаунт своего одногруппника





МОДУЛЬ 3. УРОК 4

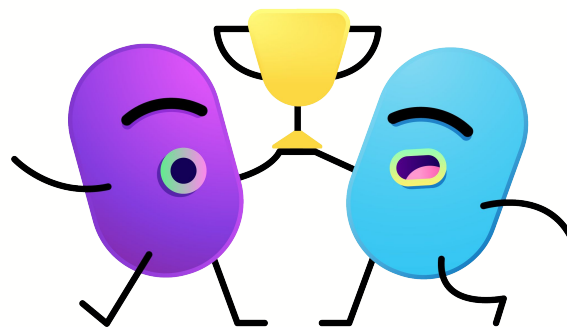
Работа в GitHub





МОДУЛЬ 3. УРОК 4

Завершение урока



Что для вас было самым удивительным на уроке?

Завершение урока



**Что получилось хорошо, что
могло получиться лучше?**

