

## Contents

Общее описание шагов .....	2
Python 3 .....	2
Python 3 → Windows.....	2
Python 3 → Linux.....	2
Python 3 → Проверка установки.....	2
Python IDE .....	3
Python IDE → PyCharm .....	3
Python IDE → VS Code.....	3
Build tools.....	3
Build tools → Linux.....	3
Build tools → Windows.....	4
Git .....	4
Git → Linux.....	4
Git → Windows.....	4
Git → Windows → Настройка Git Bash.....	5
SSH .....	7
SSH → Создание ключей .....	7
SSH → Права доступа .....	8
GitHub .....	8
GitHub → Добавить ключи .....	8
Настройки окружения.....	11
Настройки окружения → Windows .....	11
Настройки окружения → Удобный поиск в истории .....	11
Проект.....	12
Проект → Создать папку для проекта .....	12
Проект → GitHub.....	12
Проект → Добавить ссылку на исходный проект .....	13
Проект → Настройка пользователя Git.....	13
Инициализация проекта IDE.....	13
Инициализация проекта IDE → PyCharm.....	13
Инициализация проекта IDE → VS Code.....	15
Отметка о выполнении.....	15

## Общее описание шагов

Для выполнения тренинга необходимо выполнить ряд шагов:

1. Установить Python
2. Установить IDE
3. Установить средства сборки
4. Установить Git
5. Настроить SSH клиент
6. Зарегистрировать на GitHub и добавить ключи соединения
7. Сделать настройки поиска команд из командной строки
8. Получить исходники проекта

Если у вас возникли трудности в установке и настройке, то напишите сообщение администратору тренинга.

## Python 3

### Python 3 → Windows

Скачайте и установите дистрибутив

<https://www.python.org/downloads/release/python-3100/>.

### Python 3 → Linux

Установите Python, используя ваш менеджер пакетов.

Пример для Ubuntu 20:

```
$ sudo apt update
$ sudo apt install python3 python3-pip python3-venv
$ sudo pip3 install --upgrade pip
$ sudo pip3 install --upgrade wheel
$ sudo pip3 install --upgrade setuptools
```

### Python 3 → Проверка установки

Создайте файл test\_py3.py:

```
try:
    print("Hello")
```

```
except exc as var:  
    ...
```

Выполните его:

```
> c:\Python310\python.exe test_py3.py # windows  
$ /usr/bin/python3 test_py3.py        # linux  
Hello
```

Пусть к интерпретатору Python зависит от вашей ОС и установленной версии Python.  
Если вы увидели сообщение “Hello”, то Python установлен правильно.

## Python IDE

Выберите и установите IDE.

### Python IDE → PyCharm

Скачайте и установите [PyCharm Community Edition](#).

Рекомендуется установить следующие плагины (часто установлены и включены по умолчанию):

- Markdown
- Git
- Code with Me

### Python IDE → VS Code

Скачайте и установите [VS Code](#).

Рекомендуется установить следующие плагины:

- ms-python.python
- visualstudioextteam.vscodeintellicode
- njpwerner.autodocstring

## Build tools

Для использования некоторых пакетов их нужно предварительно скомпилировать.  
Для этого нужно установить компилятор.

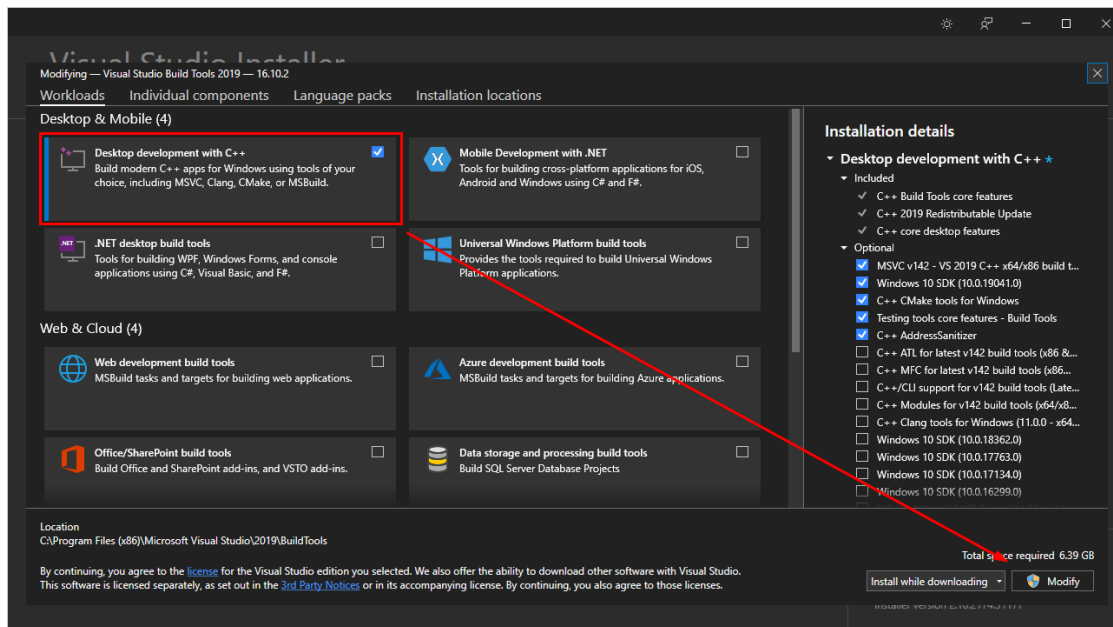
### Build tools → Linux

Пример для Ubuntu 20:

```
$ sudo apt install build-essential
```

## Build tools → Windows

- Скачайте и установите [Microsoft C++ Build Tools Installer](#).
- Запустите Visual Studio Installer из меню Пуск (иногда он запускается после установки автоматически)
- Выберите Desktop Development with C++ и нажмите Install
  - Установка потребует примерно 1 Гб места на диске



## Git

- [Git доступен](#) на многих операционных системах, включая Windows, Linux, MacOS X
- Обычно устанавливается из пакетов и/или инсталляторов

## Git → Linux

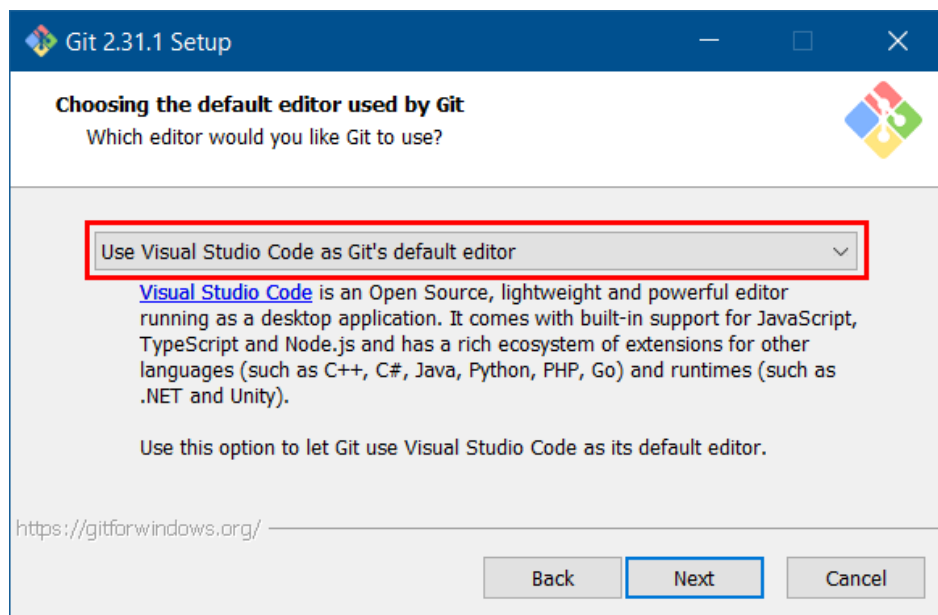
Для Ubuntu необходимо установить пакет git:

```
$ sudo apt install git
```

## Git → Windows

Скачайте и запустите [установщик](#).

По ходу установки выберите используемый вами редактор:

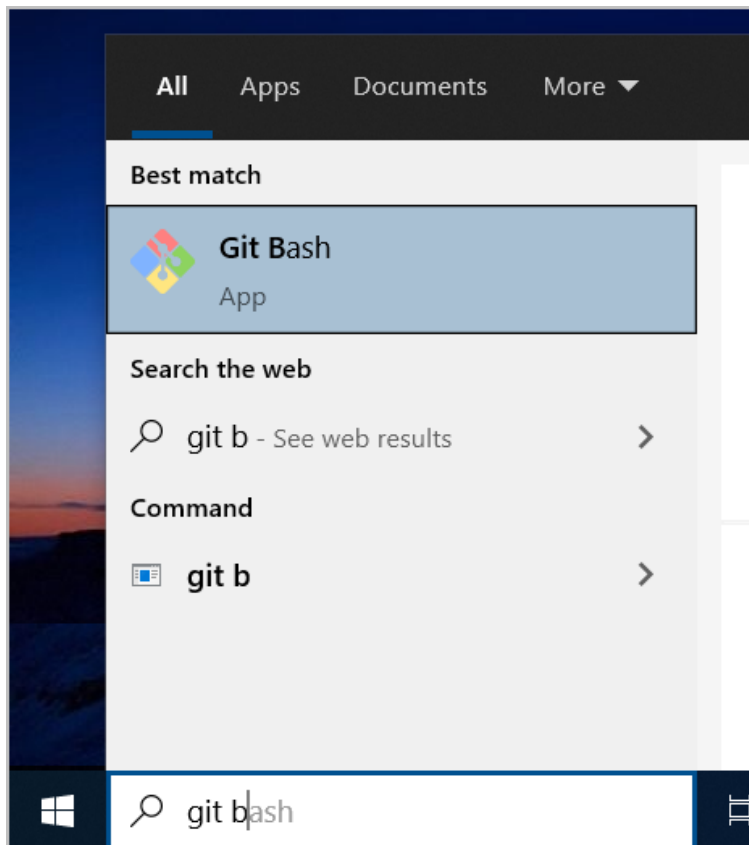


Все остальные опции можно оставить по умолчанию.

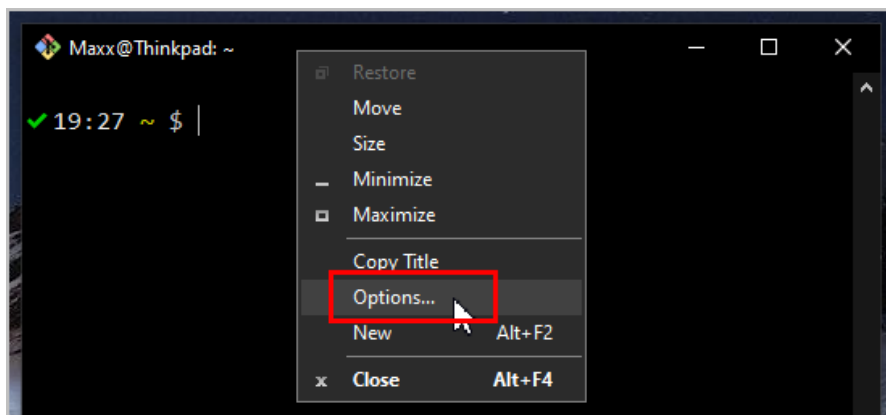
#### Git → Windows → Настройка Git Bash

Для поддержки в Git имён на русском языке рекомендуется сделать следующие настройки.

Запустите Git Bash:



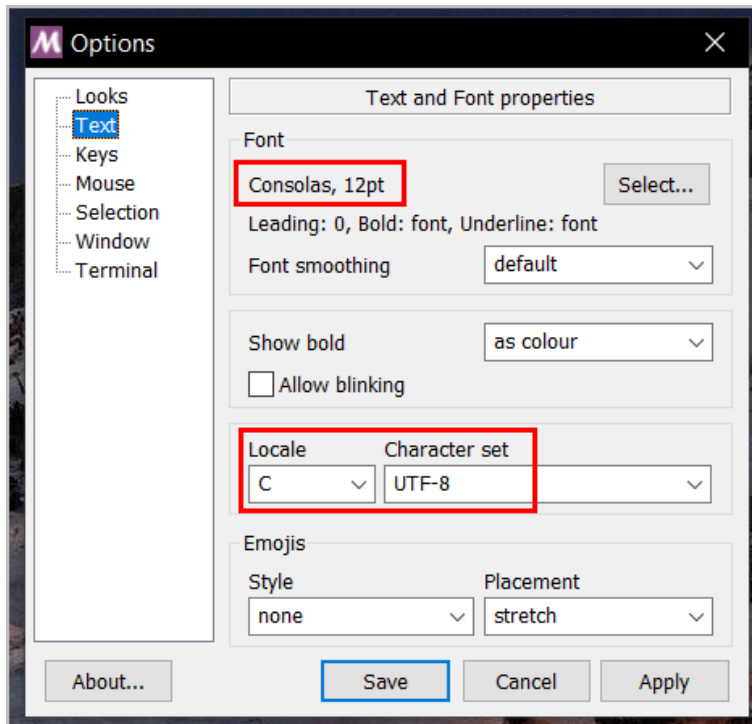
Откройте контекстное меню, нажав правую кнопку мыши на окне. В меню выберите Options....:



Слева в дереве выберите Text.

Измените шрифт на Consolas.

Укажите параметры Locale=C и Character set=UTF-8.



Нажмите Apply и Save.

В терминале введите:

```
$ git config --global core.quotePath false
```

## SSH

Очень часто Git использует протокол SSH для соединения с сервером для обмена информацией. Поэтому требуются SSH-ключи.

### SSH → Создание ключей

Проверьте наличие файла `c:\Users\Username\.ssh\id_rsa`. Если такой файл существует, то проще использовать существующий ключ. Пропустите настройку в этом разделе. Иначе выполните следующие инструкции.

Создайте пару ключей для ssh соединения:

```
$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/c/Users/Maxx/.ssh/id_rsa): <нажимаем enter>
Created directory '/c/Users/Maxx/.ssh'.
Enter passphrase (empty for no passphrase): <нажимаем enter>
Enter same passphrase again: <нажимаем enter>
Your identification has been saved in /c/Users/Maxx/.ssh/id_rsa
Your public key has been saved in /c/Users/Maxx/.ssh/id_rsa.pub
```

👉 Запомните путь к файлу `id_rsa.pub`. Он потребуется чуть позже при настройке учётной записи на сервере.

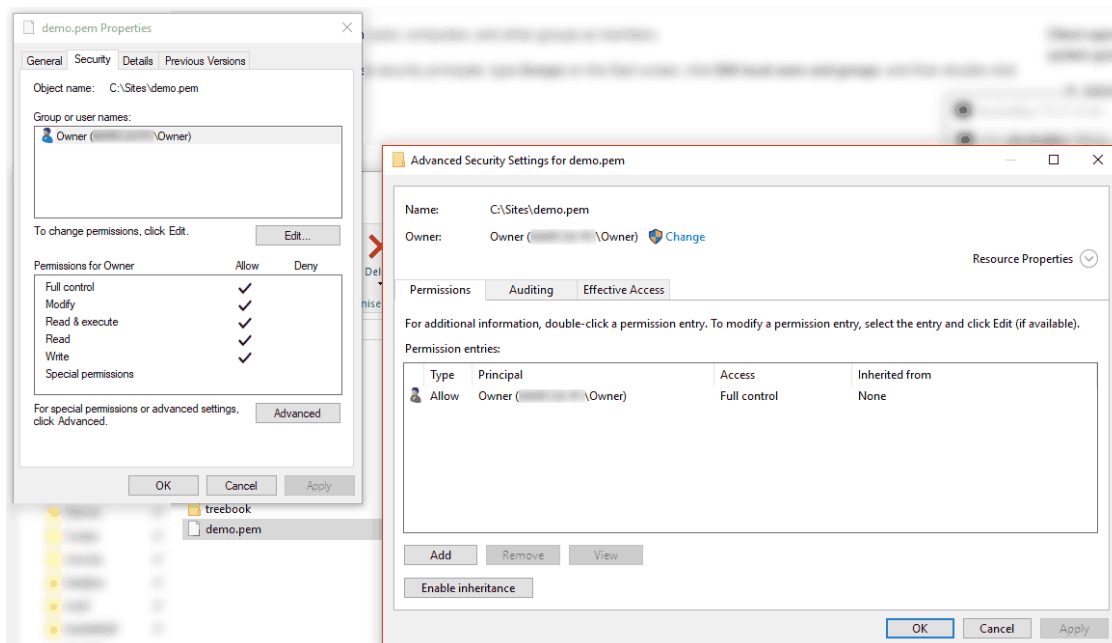
## SSH → Права доступа

На Linux и MacOS X подправить права доступа:

```
$ chmod g-w ~  
$ chmod 700 ~/.ssh  
$ chmod 600 ~/.ssh/*  
$ chmod 644 ~/.ssh/*.pub
```

На Windows нужно сделать **настройку прав**:

- нажать правой кнопкой на файле `id_rsa.pub` и выбрать Свойства
- перейти на вкладку Безопасность
- нажать кнопку Дополнительно
- выставить себя владельцем файла `id_rsa`
- убрать наследование прав
- добавить себе полный доступ на файл



## GitHub

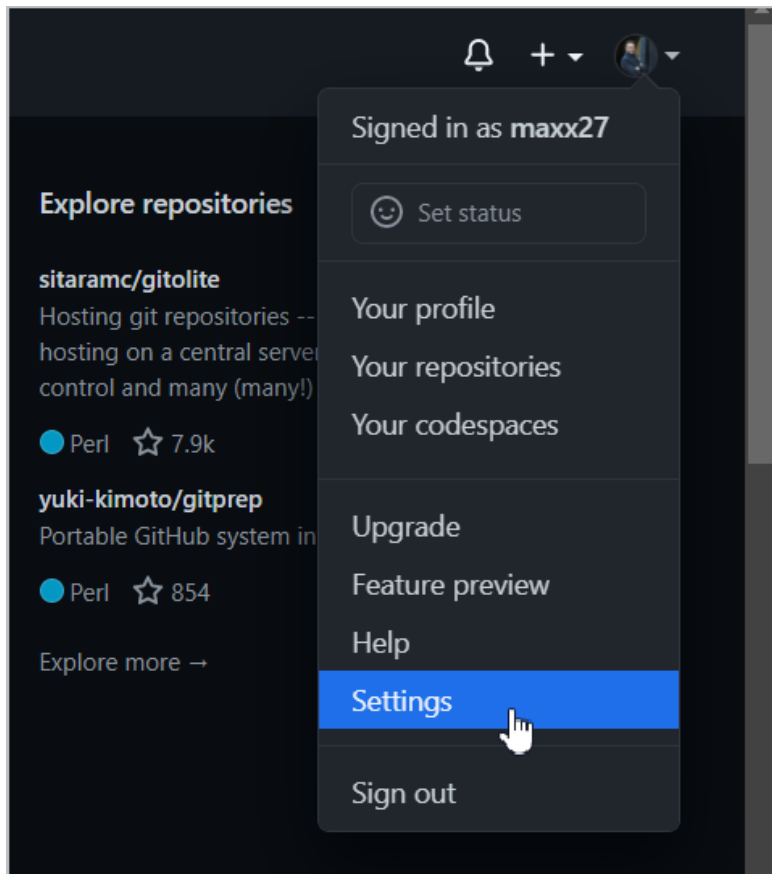
Зарегистрироваться на [github.com](https://github.com).

## GitHub → Добавить ключи

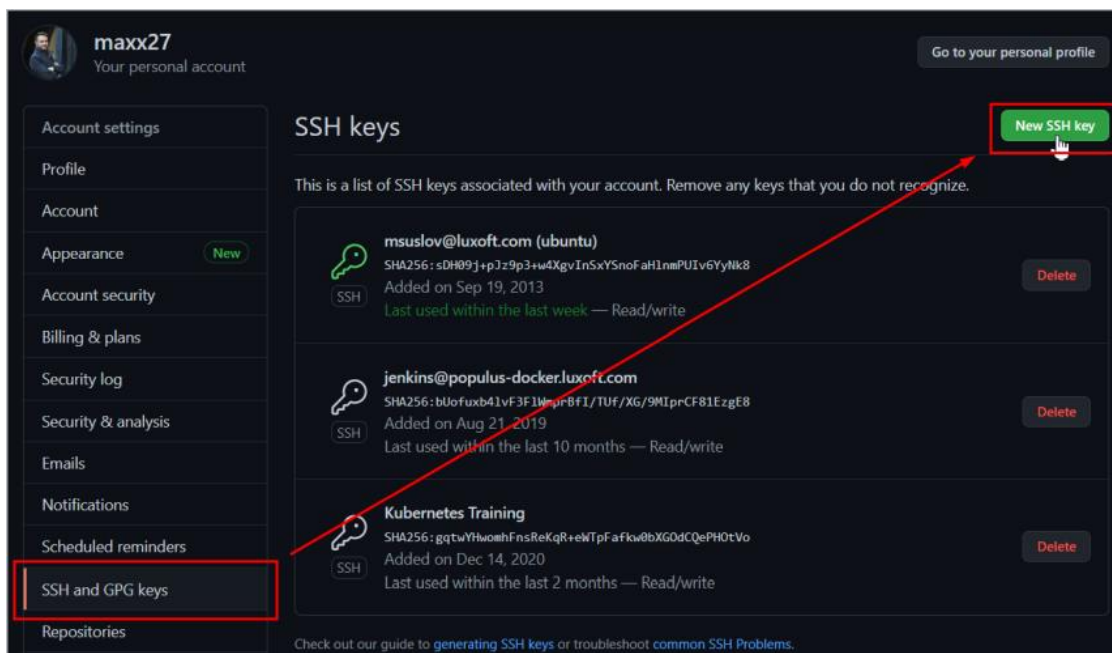
Далее необходимо добавить ssh ключи на [github.com](https://github.com).



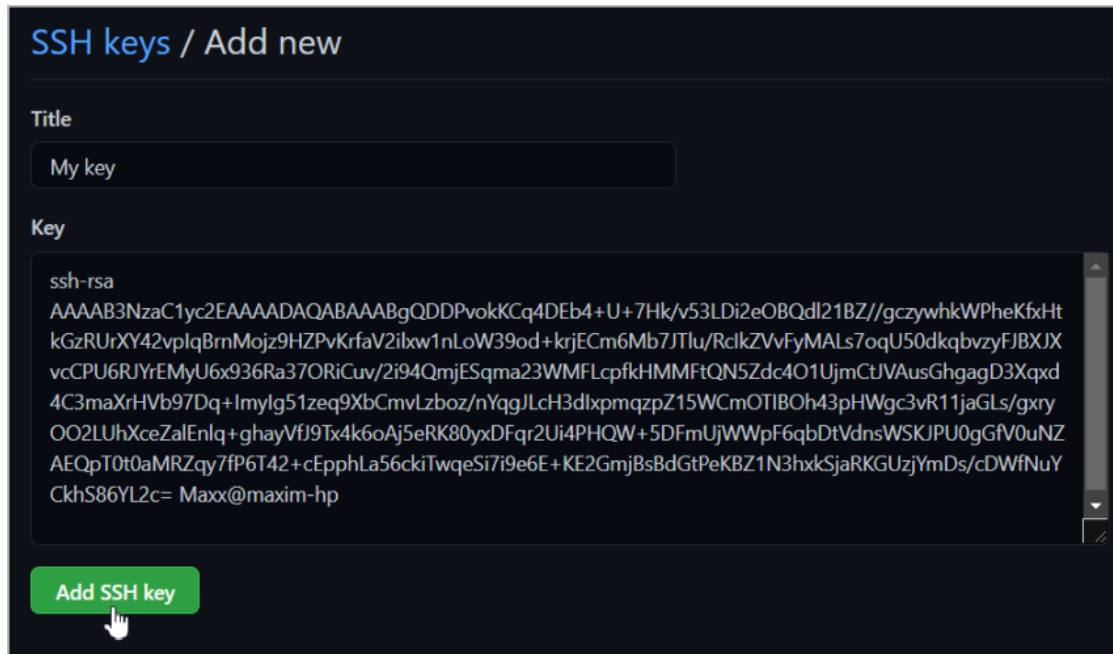
Нажать на аватарку и открыть Settings.



Выбрать SSH and GPG keys. Далее нажать Add SSH key.



Добавить название ключа (например, My key) и вставить содержимое файла `id_rsa.pub` с предыдущих шагов:



SSH keys / Add new

Title

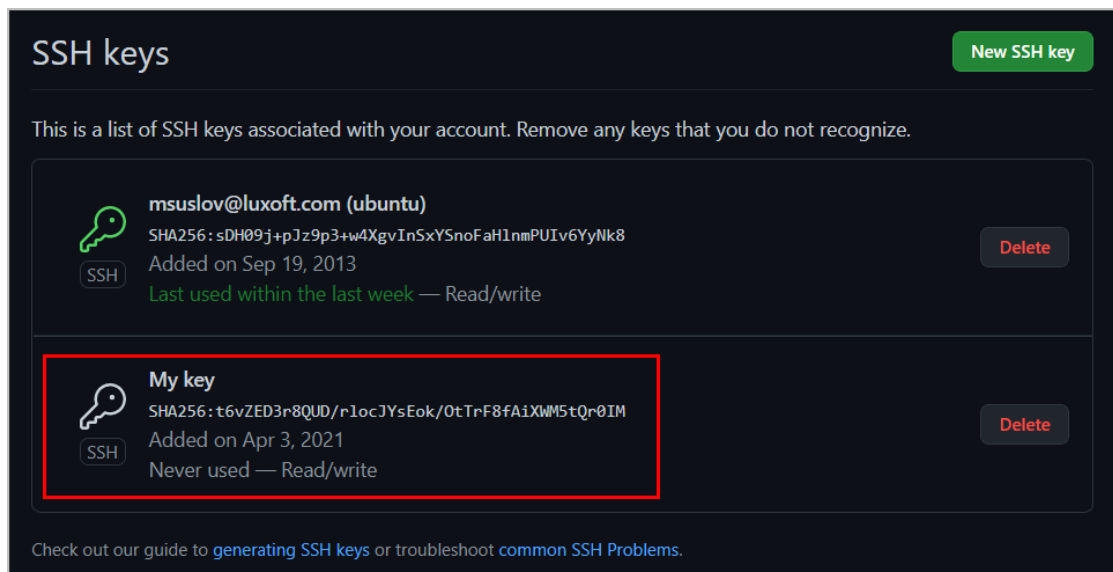
My key

Key

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQgQDDPvokKCq4DEb4+U+7Hk/v53LDi2eOBQdl21BZ//gcywhkWPheKfxHt
kGzRUrXY42vplqBrnMojz9HZPvKrfav2ilxw1nLoW39od+krjECm6Mb7JTLu/RclKZVvFyMALs7oqU50dkqbvyFJBXJX
vcCPU6RJYrEMyU6x936Ra37ORiCuv/2i94QmjESqma23WMFLcpfkHMMFtQN5Zdc4O1UjmCtVAusGhgagD3Xqxd
4C3maXrHVb97Dq+Imylg51zeq9XbCmvLzboz/nYqgJLcH3dlxpmqzpZ15WCmOTIBOh43pHWgc3vR11jaGLs/gxry
OO2LUhXceZalEnlq+ghayVfj9Tx4k6oAj5eRK80yxtDFqr2Ui4PHQW+5DFmUjWWpF6qbDtVdnsWSKJPu0gGFV0uNZ
AEQpT0t0aMRZqy7fP6T42+cEpphLa56ckiTwqeSi7i9e6E+KE2GmjBsBdGtPeKBZ1N3hXkSjaRKGUzjYmDs/cDWfNuY
CkhS86YL2c= Maxx@maxim-hp
```



Add SSH key

Ключ должен появиться в списке ключей:



SSH keys New SSH key

This is a list of SSH keys associated with your account. Remove any keys that you do not recognize.

	<b>msuslov@luxoft.com (ubuntu)</b> SHA256: sDH09j+pJz9p3+w4XgvInSxYSnoFaHlnmPUIv6YyNk8 Added on Sep 19, 2013 Last used within the last week — Read/write <span>Delete</span>
	<b>My key</b> SHA256: t6vZED3r8QUd/r1ocJYsEok/0tTrF8fAiXWM5tQr0IM Added on Apr 3, 2021 Never used — Read/write <span>Delete</span>

Check out our guide to [generating SSH keys](#) or troubleshoot [common SSH Problems](#).

Создать файл `~/.ssh/config` или `c:\Users\Username\.ssh\config` (без расширения!)

```
Host github.com
  User git
  HostName ssh.github.com
  PreferredAuthentications publickey
  IdentityFile ~/.ssh/id_rsa
  Port 443
```

Типовые ошибки:

- изменяют имя пользователя с `git` на своё - нужно оставить `git`
- если используется другой ключ, то указать имя нового ключа вместо `~/.ssh/id_rsa`
- используется папка отличная от `~/.ssh/` - надо делать в указанной папке

Проверить соединение из терминала:

```
$ ssh -T git@github.com
Hi maxx27! You've successfully authenticated, but GitHub does not provide
shell access.
```

В некоторых случаях нужно подтвердить подключение набрав `yes` в ответ:

The authenticity of host 'github.com (IP ADDRESS)' can't be established. RSA key fingerprint is 16:27:ac:....df:a6:48. Are you sure you want to continue connecting (yes/no)?

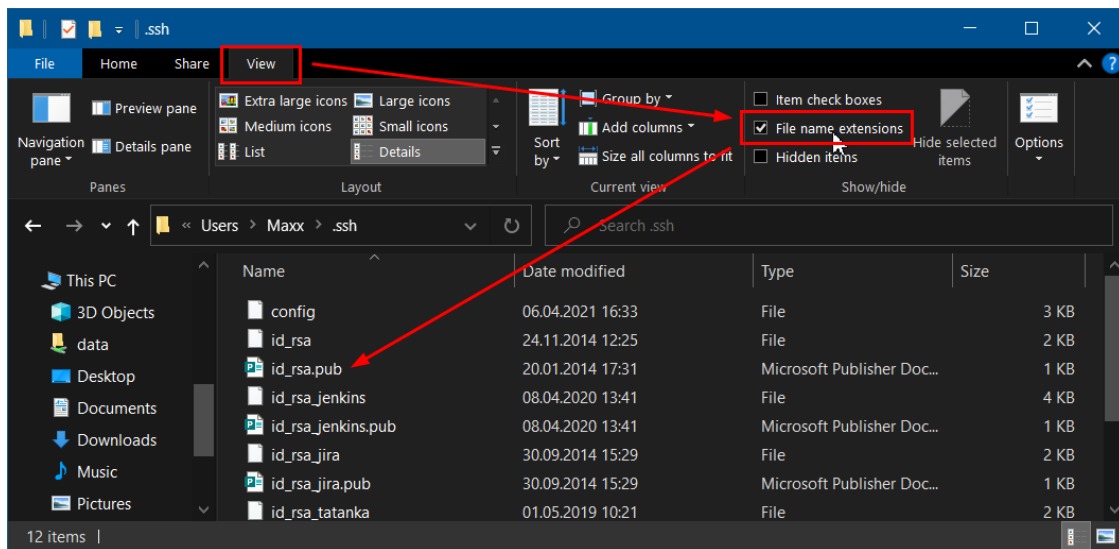
В случае ошибки прочитать <https://help.github.com/en/articles/error-permission-denied-publickey> или спросить знающих людей.

## Настройки окружения

Далее рекомендуется сделать некоторые настройки вашего окружения.

### Настройки окружения → Windows

Рекомендуется включить опцию показа расширения файлов:



### Настройки окружения → Удобный поиск в истории

Создать файл `.inputrc` в домашней папке пользователя

- %userprofile% или c:\Users\Username (Windows)
- \$HOME или /home/username (Linux)

со следующим содержимым:

```
# arrow up
"\e[A":history-search-backward
# arrow down
"\e[B":history-search-forward
```

Выполните команды в Git Bash:

```
$ git
$ ls
$ ls ~
$ git version
```

Далее введите g и нажатие клавишу вверх ↑ два раза. Должны отображаться команды, начинающиеся только с буквы g.

## Проект

Далее необходимо сделать ряд шагов.

### Проект → Создать папку для проекта

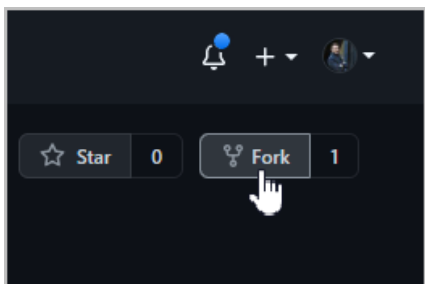
Создать папку для проекта (имя по вашему усмотрению):

```
> mkdir c:\script007 # windows
> cd /d c:\script007
```

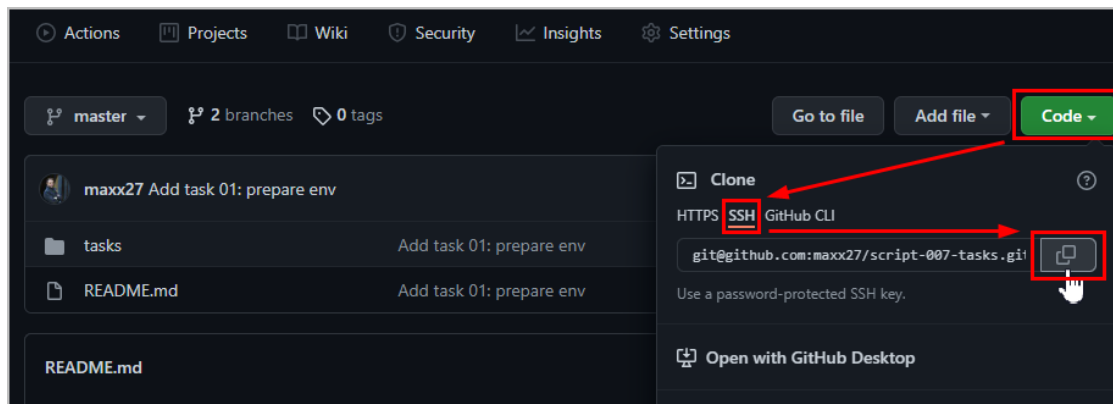
```
$ mkdir ~/script007 # linux
$ cd ~/script007
```

### Проект → GitHub

Откройте <https://github.com/maxx27/script-007-tasks> и форкните проект:



У вас появится собственная версия репозитория. Склонируйте её. Для этого копируем адрес репозитория с использованием протокола SSH:



и используем для клонирования:

```
$ git clone git@github.com:YourGithubUser/script-007-tasks.git .
```

Убедиться, что появились файлы в папке для проекта.

## Проект → Добавить ссылку на исходный проект

Добавить связь на оригинальный репозиторий

```
$ git remote add trainer git@github.com:maxx27/script-007-tasks.git
```

Проверить:

```
$ git remote -v
origin  git@github.com:YourGithubUser/script-007-tasks.git (fetch)
origin  git@github.com:YourGithubUser/script-007-tasks.git (push)
trainer git@github.com:maxx27/script-007-tasks.git (fetch)
trainer git@github.com:maxx27/script-007-tasks.git (push)
```

## Проект → Настройка пользователя Git

Установите параметры пользователя для текущего проекта. Например:

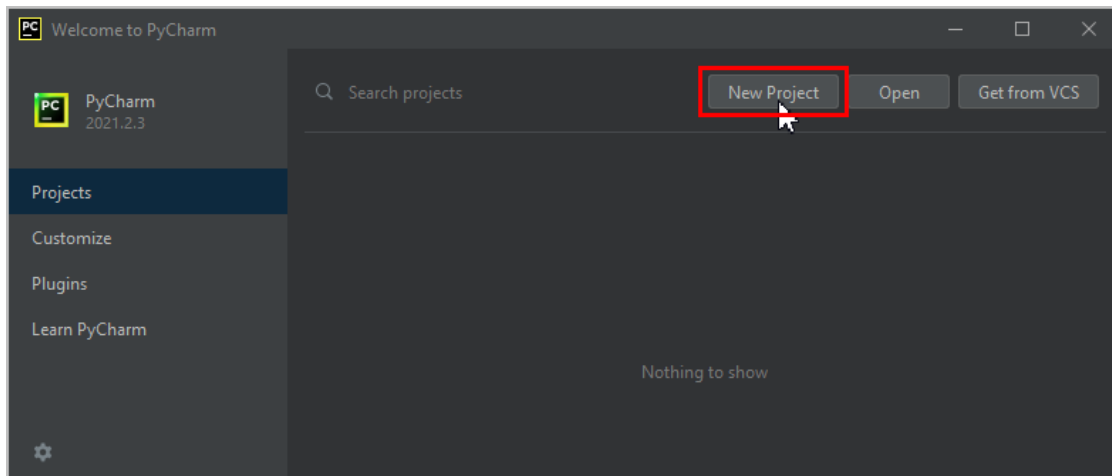
```
$ git config user.name "Maxim Suslov"
$ git config user.email maxim.suslov@dxs.com
```

## Инициализация проекта IDE

### Инициализация проекта IDE → PyCharm

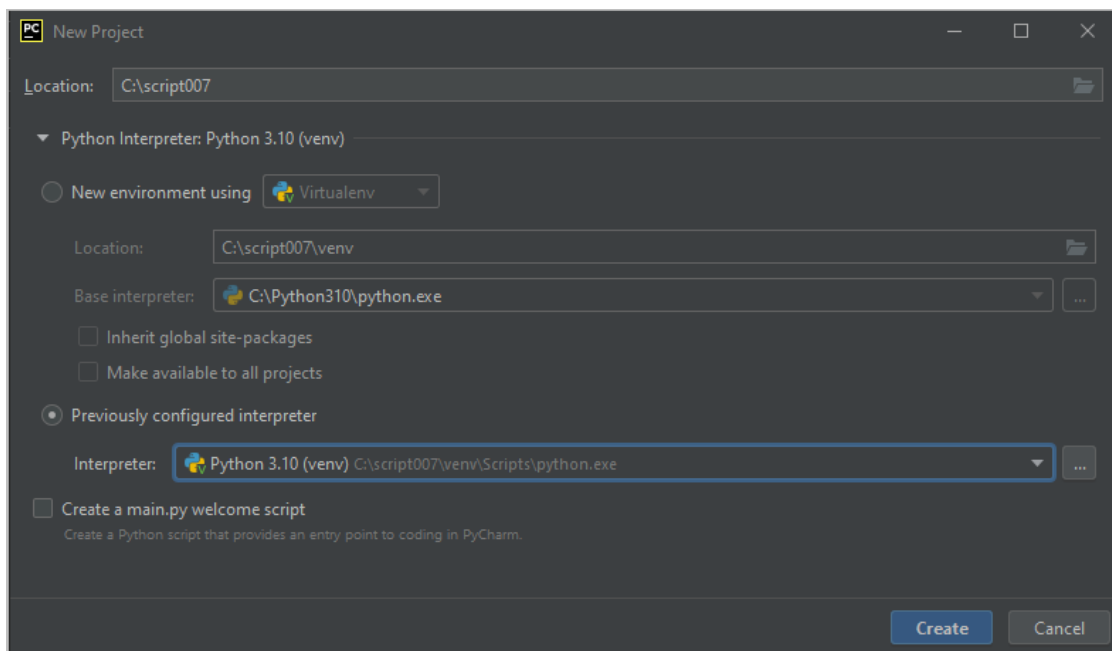
Запустите PyCharm.

Выберите New Project:

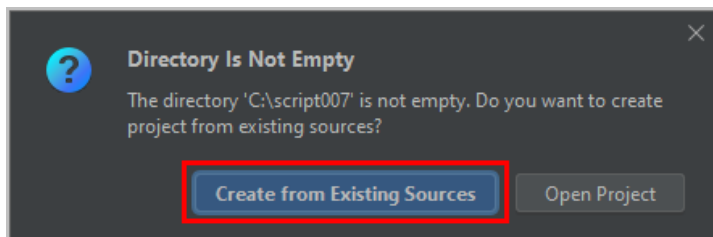


Затем в Location укажите папку проекта.

Укажите Previously configured interpreter (надо выбрать исполняемый файл python внутри папки venv).



Нажмите Create и подтвердите выбор Create from Existing Sources:



## Инициализация проекта IDE → VS Code

TODO: доделать

## Отметка о выполнении

Отметьте выполнение в Google-таблице:

<https://docs.google.com/spreadsheets/d/1HRTDjrojKLyqRSMIE-8wAbRMgXKL0WQdgTu5P1m5dP0/edit#gid=0>