pwd - посмотреть где нахожусь

cd …/ -выбрать директорию

ls – посмотреть свойства

git init -добавить в воркинг директорию все содержимое

git add . – добавить все содержимое директории в stagin area

git add index.html -добавить конкретный файл

git status

ls -la посмотреть все содержимое директории

git commit

i :wq

git commit -m “Комментарий”

git push

rm -rf .git -отвязать директорию

vim README.md

cat README.md

clear

.gitignore - файл с игнорируемыми директориями .idea/\* - все содержимое папки

.DS\_Store - пример файла

git rm -r --cached .idea/ - удалить директорию

git branch - ветка

git pull - вытягивание

git checkout bugfix/change\_position переместиться на ветку

git diff – посмотреть изменения

**git checkout -b** bugfix/change\_color -перейти на новую ветку

**git commit** --amend для изменения последнего комментария

**git branch -d** bugfix/change\_color – удалить ветку

**git merge** master – находясь в ветке слить ее с master

git pull --rebase или git pull -r для исключения конфликта при одновременном редактировании

**Quick setup — if you’ve done this kind of thing before**

[Set up in Desktop](x-github-client://openRepo/https:/github.com/sergei-karpovic/stydies)

**or**

Начало формы

HTTPS

Конец формы

Начало формы

SSH

git@github.com:sergei-karpovic/stydies.git

Конец формы



Get started by [creating a new file](https://github.com/sergei-karpovic/stydies/new/main) or [uploading an existing file](https://github.com/sergei-karpovic/stydies/upload). We recommend every repository include a [README](https://github.com/sergei-karpovic/stydies/new/main?readme=1), [LICENSE](https://github.com/sergei-karpovic/stydies/new/main?filename=LICENSE.md), and [.gitignore](https://github.com/sergei-karpovic/stydies/new/main?filename=.gitignore).

**…or create a new repository on the command line**

echo "# stydies" >> README.md

git init

git add README.md

git commit -m "first commit"

git branch -M main

git remote add origin git@github.com:sergei-karpovic/stydies.git

git push -u origin main

**…or push an existing repository from the command line**

git remote add origin git@github.com:sergei-karpovic/stydies.git

git branch -M main

git push -u origin main

**…or import code from another repository**

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

# Генерация нового SSH-ключа и добавление его в ssh-agent

## [В этой статье](https://docs.github.com/authentication/connecting-to-github-with-ssh/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent#in-this-article)

* [О генерации ключей SSH](https://docs.github.com/en/authentication/connecting-to-github-with-ssh/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent#about-ssh-key-generation)
* [Генерация нового ключа SSH](https://docs.github.com/en/authentication/connecting-to-github-with-ssh/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent#generating-a-new-ssh-key)
* [Добавление вашего SSH-ключа в ssh-agent](https://docs.github.com/en/authentication/connecting-to-github-with-ssh/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent#adding-your-ssh-key-to-the-ssh-agent)
* [Создание нового ключа SSH для аппаратного ключа безопасности](https://docs.github.com/en/authentication/connecting-to-github-with-ssh/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent#generating-a-new-ssh-key-for-a-hardware-security-key)
* [дальнейшее чтение](https://docs.github.com/en/authentication/connecting-to-github-with-ssh/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent#further-reading)

После того, как вы проверили существующие ключи SSH, вы можете сгенерировать новый ключ SSH для аутентификации, а затем добавить его в ssh-agent.

MacОкнаLinux

## О генерации ключей SSH

Если у вас еще нет SSH-ключа, вы должны сгенерировать новый SSH-ключ, который будет использоваться для аутентификации. Если вы не уверены, есть ли у вас уже SSH-ключ, вы можете проверить существующие ключи. Для получения дополнительной информации см. « [Проверка существующих ключей SSH](https://docs.github.com/en/github/authenticating-to-github/checking-for-existing-ssh-keys) ».

Если вы хотите использовать аппаратный ключ безопасности для аутентификации на GitHub, вы должны сгенерировать новый ключ SSH для вашего аппаратного ключа безопасности. При аутентификации с помощью пары ключей необходимо подключить аппаратный ключ безопасности к компьютеру. Дополнительные сведения см. В [примечаниях к выпуску OpenSSH 8.2](https://www.openssh.com/txt/release-8.2) .

Если вы не хотите повторно вводить парольную фразу каждый раз, когда используете свой SSH-ключ, вы можете добавить свой ключ к агенту SSH, который управляет вашими SSH-ключами и запоминает вашу парольную фразу.

## Генерация нового ключа SSH

1. Откройте Git Bash .
2. Вставьте текст ниже, подставив свой адрес электронной почты на GitHub.

$ ssh-keygen -t ed25519 -C "your\_email@example.com"

**Примечание.** Если вы используете устаревшую систему, которая не поддерживает алгоритм Ed25519, используйте:

$ ssh-keygen -t rsa -b 4096 -C "your\_email@example.com"

Это создает новый ключ SSH, используя предоставленный адрес электронной почты в качестве метки.

> Generating public/private algorithm key pair.

1. Когда вам будет предложено «Введите файл для сохранения ключа», нажмите Enter. Это принимает расположение файла по умолчанию.

> Enter a file in which to save the key (/c/Users/you/.ssh/id\_algorithm):[Press enter]

1. В командной строке введите безопасную парольную фразу. Для получения дополнительной информации см. [«Работа с ключевыми ключевыми фразами SSH](https://docs.github.com/en/articles/working-with-ssh-key-passphrases) ».
2. > Enter passphrase (empty for no passphrase): [Type a passphrase]

> Enter same passphrase again: [Type passphrase again]

## Добавление вашего SSH-ключа в ssh-agent

Перед добавлением нового SSH-ключа к ssh-agent для управления вашими ключами вы должны были проверить существующие SSH-ключи и сгенерировать новый SSH-ключ.

Если у вас установлен [GitHub Desktop](https://desktop.github.com/) , вы можете использовать его для клонирования репозиториев и не работать с ключами SSH.

1. Убедитесь, что ssh-agent запущен. Вы можете использовать инструкции «Автоматический запуск ssh-agent» в « [Работа с ключевыми ключевыми фразами SSH](https://docs.github.com/en/articles/working-with-ssh-key-passphrases) » или запустить его вручную:
2. # start the ssh-agent in the background
3. $ eval "$(ssh-agent -s)"

> Agent pid 59566

1. Добавьте свой закрытый ключ SSH в ssh-agent. Если вы создали свой ключ с другим именем или если вы добавляете существующий ключ с другим именем, замените id\_ed25519 в команде именем вашего файла закрытого ключа.

$ ssh-add ~/.ssh/id\_ed25519

1. Добавьте SSH-ключ в свою учетную запись на GitHub. Дополнительные сведения см. В разделе « [Добавление нового ключа SSH в учетную запись GitHub](https://docs.github.com/en/github/authenticating-to-github/adding-a-new-ssh-key-to-your-github-account) ».

## Создание нового ключа SSH для аппаратного ключа безопасности

Если вы используете macOS или Linux, вам может потребоваться обновить свой SSH-клиент или установить новый SSH-клиент перед созданием нового SSH-ключа. Дополнительные сведения см. В разделе « [Ошибка: неизвестный тип ключа»](https://docs.github.com/en/github/authenticating-to-github/error-unknown-key-type) .

1. Вставьте аппаратный ключ безопасности в свой компьютер.
2. Откройте Git Bash .
3. Вставьте текст ниже, заменив его адресом электронной почты для своей учетной записи на GitHub.

$ ssh-keygen -t ed25519-sk -C "your\_email@example.com"

**Примечание.** Если команда не выполняется, и вы получаете сообщение об ошибке, invalid formatили feature not supported,вы можете использовать аппаратный ключ безопасности, который не поддерживает алгоритм Ed25519. Вместо этого введите следующую команду.

$ ssh-keygen -t ecdsa-sk -C "your\_email@example.com"

1. Когда вам будет предложено, нажмите кнопку на аппаратном ключе безопасности.
2. Когда вам будет предложено «Введите файл для сохранения ключа», нажмите Enter, чтобы принять расположение файла по умолчанию.

> Enter a file in which to save the key (/c/Users/you/.ssh/id\_ed25519\_sk):[Press enter]

1. Когда вам будет предложено ввести кодовую фразу, нажмите **Enter** .
2. > Enter passphrase (empty for no passphrase): [Type a passphrase]

> Enter same passphrase again: [Type passphrase again]

1. Добавьте SSH-ключ в свою учетную запись на GitHub. Дополнительные сведения см. В разделе « [Добавление нового ключа SSH в учетную запись GitHub](https://docs.github.com/en/github/authenticating-to-github/adding-a-new-ssh-key-to-your-github-account) ».