**GIT e GITHUB**

**TOKEN DE ACESSO PESSOAL**

Gerar token no GitHub, guardar ele na sua máquina;

Sempre que for realizar um commit o git vai pedir seu usuário e senha (nesse caso ao invés de usar a senha você utiliza sua token pessoal

**Acessar o Git:**

**Settings / Developes settings / Personal access tokens ;**

Clicar no botão **Generate new token;**

Informar o nome do token

Definir tempo de expiração e marcar a opção **repo** e **gerar**;

Basta copiar e salvar ele para utilizar;

Para utilizar o token, por exemplo, se for clonar você deve usar o protocolo HTTPS;



**CHAVE SSH**

Forma de estabelecer uma conexão segura e encriptada entre duas máquinas (Servidor Github e nossa máquina) utilizando uma chave pública e privada. Realizando essa configuração geramos uma chave pública para nossa máquina que será autênticada no GitHub que reconhecerá nossa assinatura para realizarmos os commits.

**Acessar o Github:**

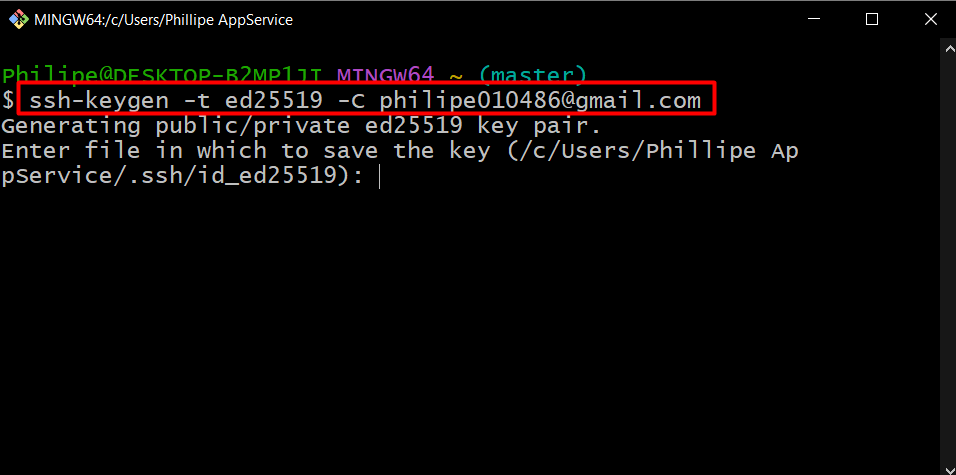
**Settings / SSH and GPG Keys / New SSH key;**

Preencher o título do SSH e devemos passar a Key;

**Como gerar a key:**

Acessar o Git Bash e realizar os seguintes comandos:

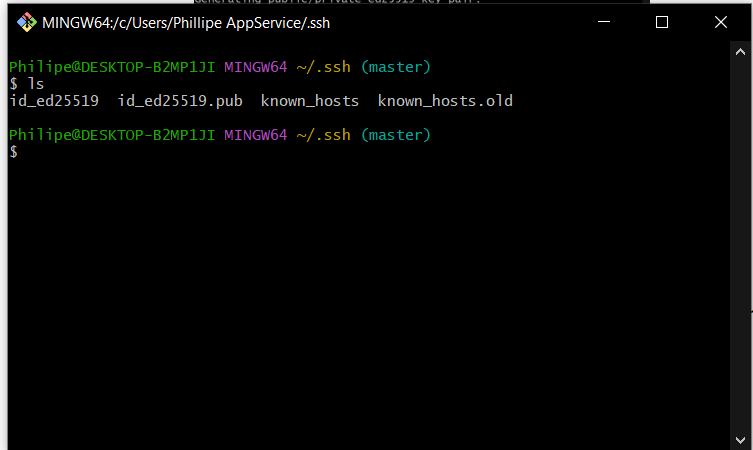
ssh-keygent -t ed25519 -C [philipe010486@gmail.com](mailto:philipe010486@gmail.com)



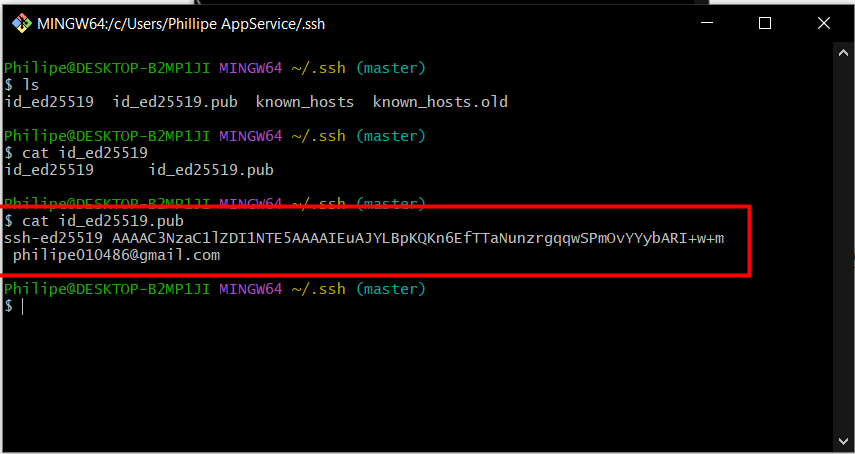
Esse comando ele vai está gerando as chaves pública e privada e informa o local onde vai ser salvo essas chaves.

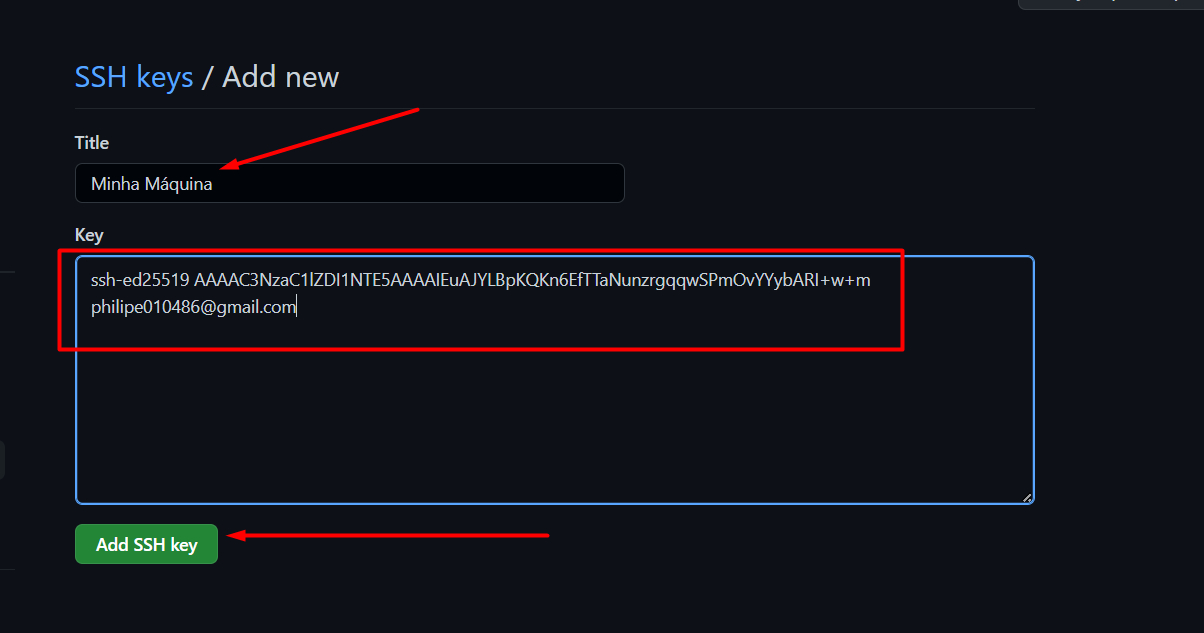
**Basta da um ENTER e configurar uma senha para que seja gerado as chaves. Deve-se acessar o local onde foi armazenado para confirmar.**

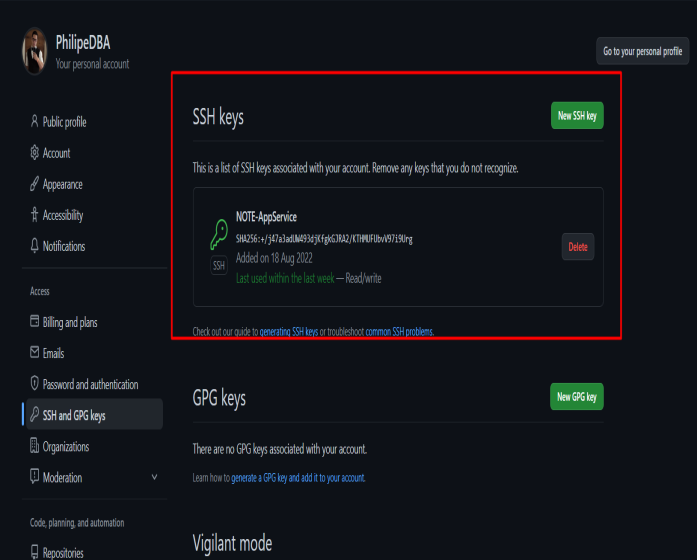
Após acessar a pasta dá um ‘ls’ para listar as chaves



Cat id\_ed25591.pub - realizamos esse comando para buscar o key dentro da chave pública para copiar e colar no GitHub

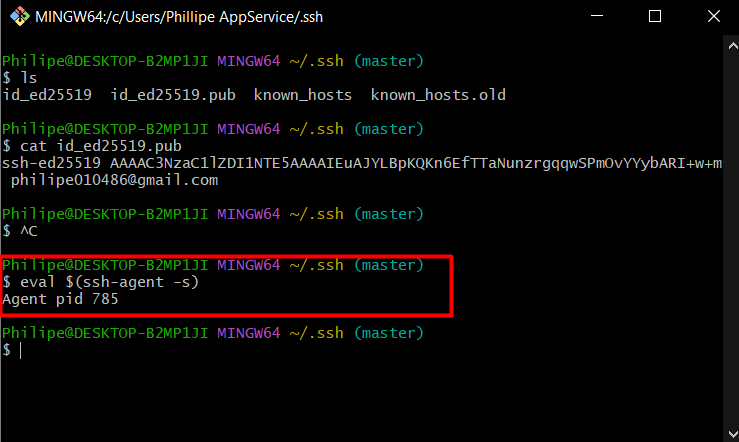






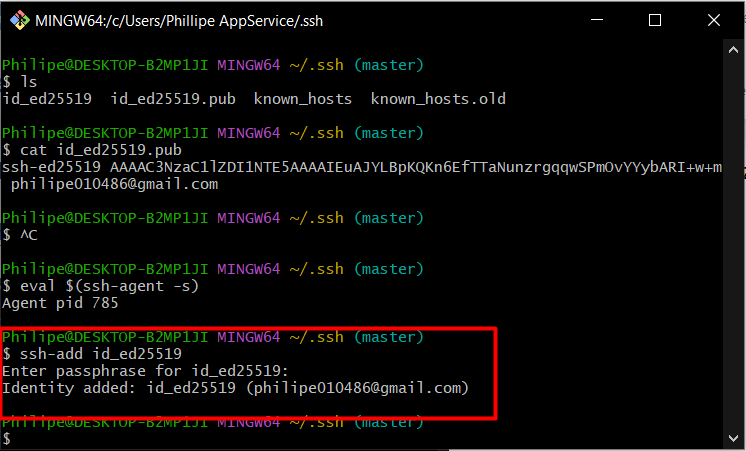
Por fim devemos rodar um comando para inicializar as chaves para podermos utilizá-las:

eval $(ssh-agent -s)

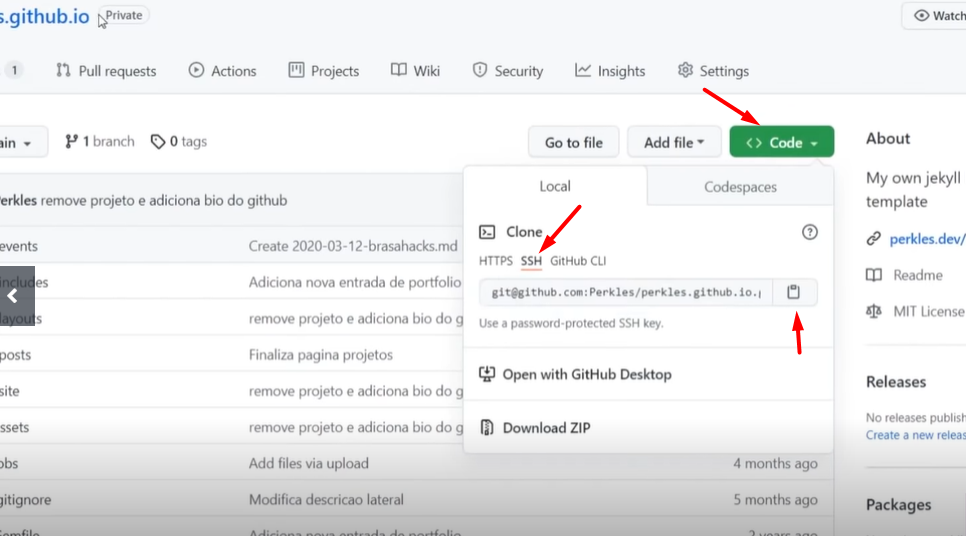


Agora devemos entregar a chave privada para eles

ssh-add id\_ed25519 e informar a senha



Clonar um repositório com SSH na máquina



**COMANDOS GIT**

****git init****: inicializa um repositório local git

****git status****: verifica o estado dos seus arquivos

****git add <nomeDoArquivo>****: envia seu arquivo especificado para o Stage

****git add - -all****: envia todos os arquivos para o Stage

****git add \*****: envia todos os arquivos para o Stage

****git commit -m “tituloDoCommit:****envia o que está no Stage para o HEAD

****git remote add origin urlDoRepositorio:****adiciona e indica a URL do repositório remoto em que os arquivos serão mantidos

****git push origin master:****envia os arquivos para o repositório remoto que você especificou através da URL do comando acima

****git checkout -b <nomeDaBranch>:****cria uma nova branch

****git checkout <nomeDaBranch>:****alterna para a branch especificada

**Ls -a** - mostra pastas ocultas dentro do diretório

Caso seja a primeira que estamos usando os git no pc devemos inserir os seguintes comandos

git config --global user.email ["e-mail"](mailto:\"philipe010486@gmail.com\")

git config --global user.name “nick\_name”