E-BOOK (CONTEÚDO) - BOOTCAMP
CURSO: <BÁSICO EM MACHINE LEARNING>

Tutorial sobre Git e GitHub

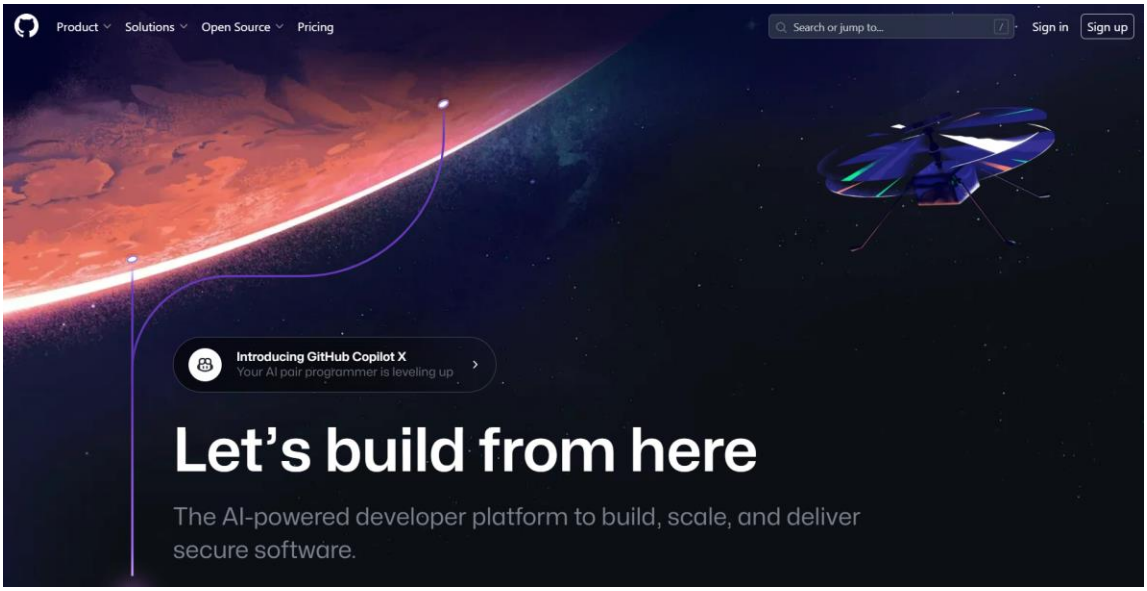
- O que é Git?

  Git é um sistema de controle de versão popular. Foi criado por Linus Torvalds em 2005, e tem sido mantido por Junio Hamano desde então.
  É utilizado para:
  - Rastrear mudanças em códigos;
  - Rastrear quem fez as mudanças;
  - Colaboração em códigos.

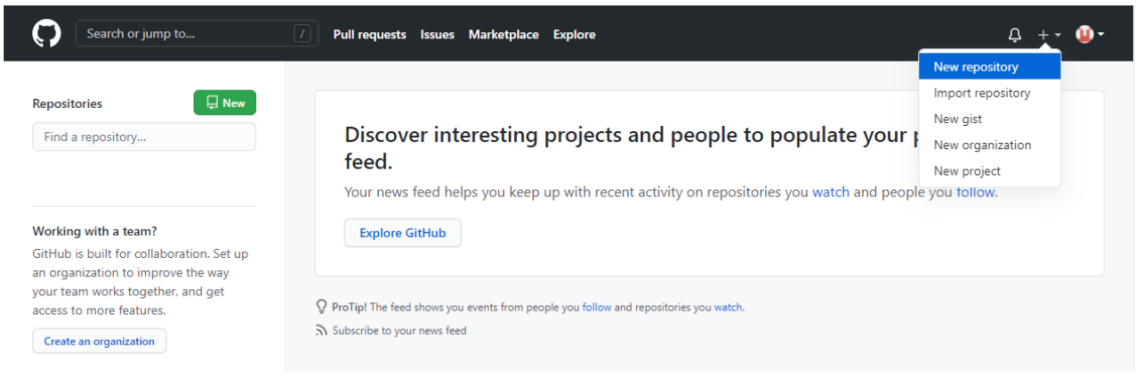
- O que é GitHub?
  - Git não é o mesmo que GitHub;
  - GitHub cria ferramentas que usam o Git;
  - GitHub é a ferramenta de maior hospedagem de códigos fontes no mundo, e pertence à Microsoft desde 2018;
  - Neste tutorial, focaremos na utilização do Git com GitHub.

ADICIONANDO ARQUIVOS AO GITHUB VIA UPLOAD

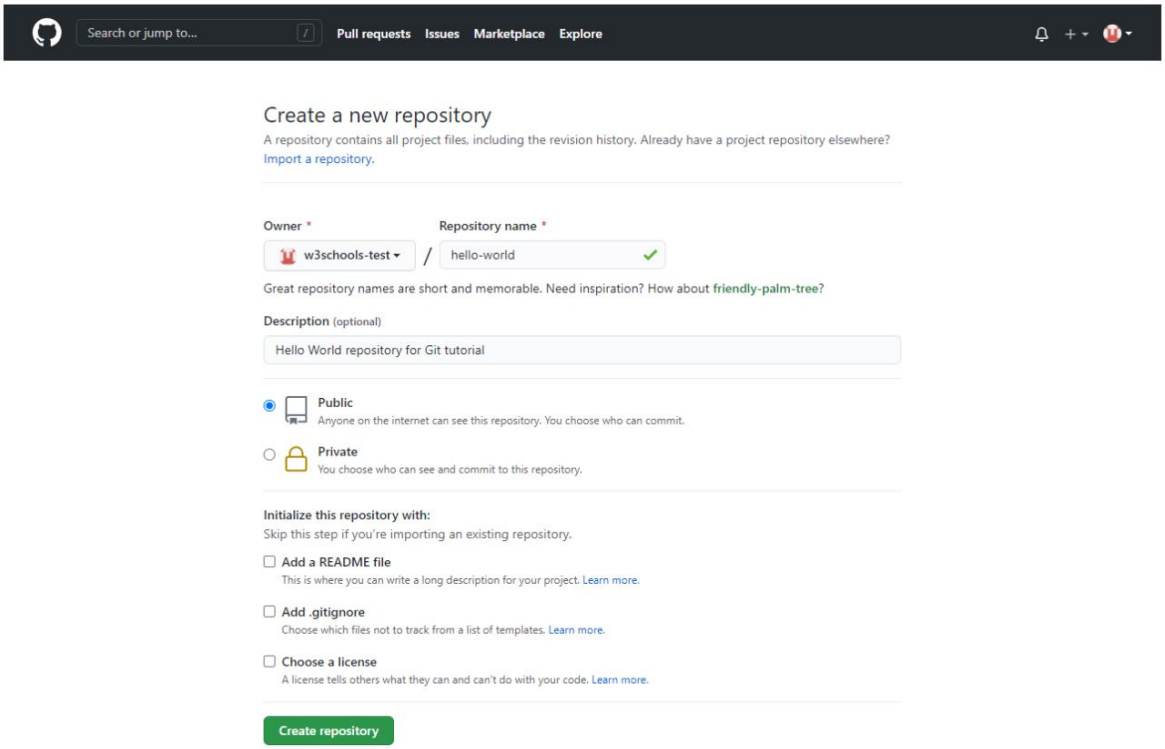
1. Crie uma Conta no GitHub (https://github.com/)



2. Agora que você criou uma conta no GitHub, logue e crie um novo repositório

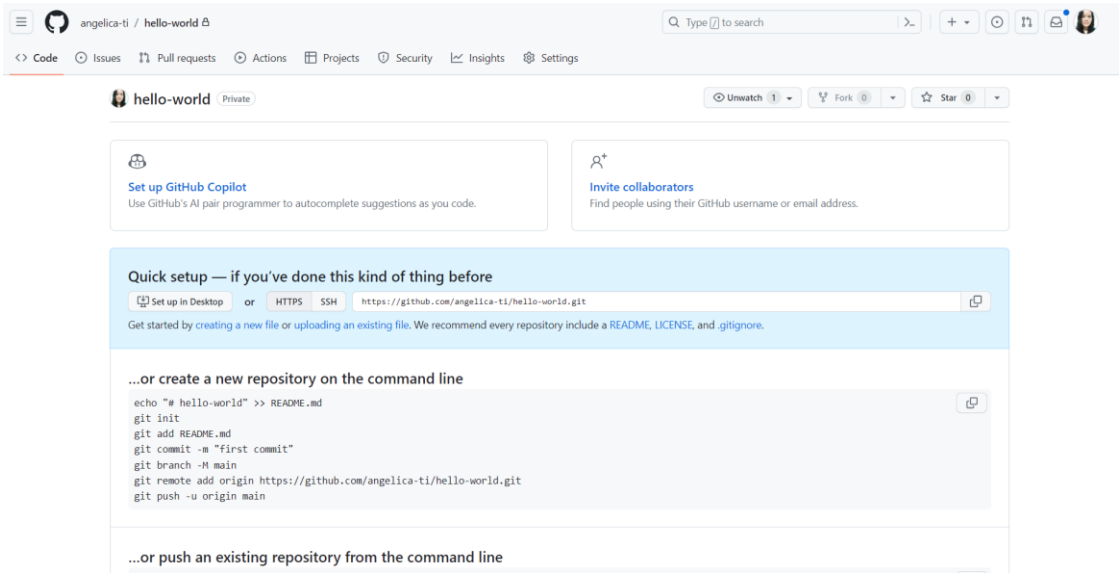


3. Preencha detalhes relevantes:

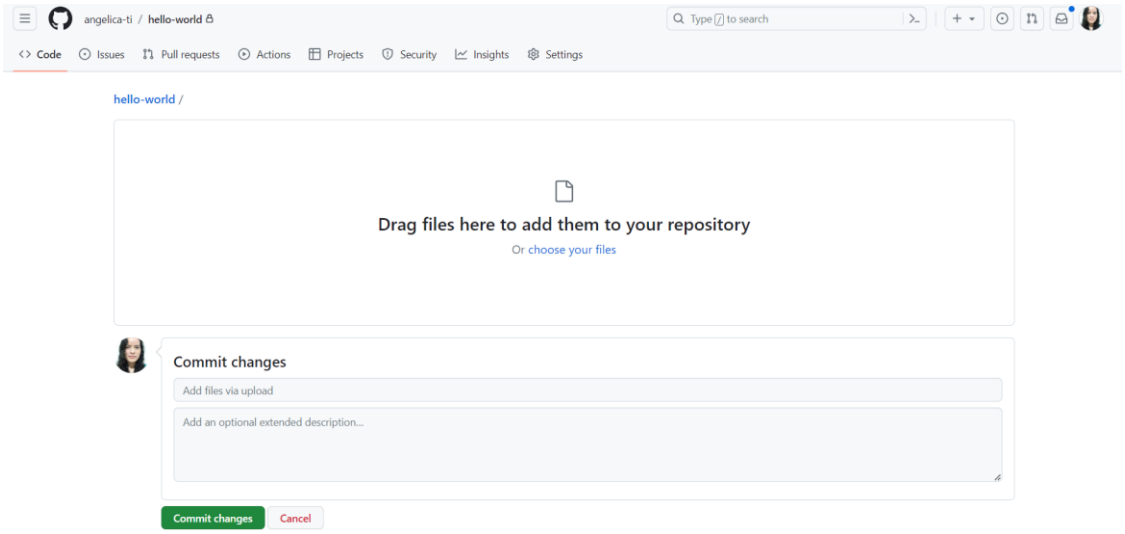


- Dê um nome para o seu repositório remoto, no exemplo acima foi dado o nome *hello-world*.
- Escolha se o seu repositório será público (visível para qualquer pessoa) ou privado (se quiser escolher quem poderá visualizar o repositório). De qualquer maneira, você poderá escolher quem pode contribuir com o repositório.
- Clique em "*Create repository*"

4. Adicionar arquivos ao repositório remoto



Clique no link "*uploading an existing file*" para subir os seus arquivos do seu computador para o repositório criado.



Clique em "*choose your files*", selecione os arquivos do seu computador que você deseja enviar, adicione uma descrição e clique em "*Commit changes*" para atualizar o repositório remoto com os arquivos enviados.

ADICIONANDO ARQUIVOS AO GITHUB VIA COMANDOS GIT

1. Instalar o Git

   Você pode fazer o *Download* do git pelo *website* https://git-scm.com/

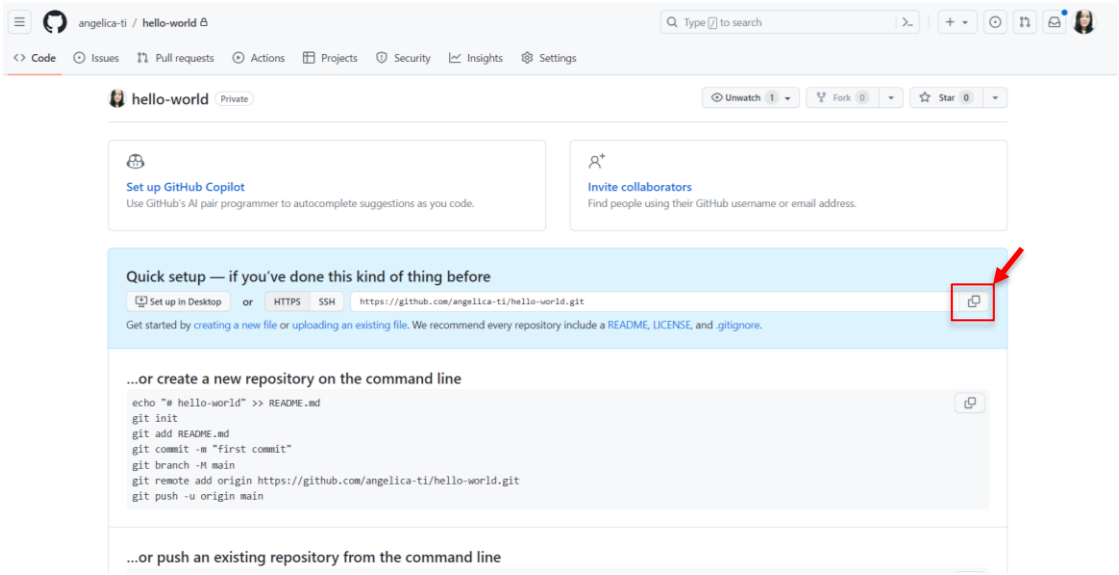
2. Verificar a versão instalada

   Para começar a utilizar o Gir, primeiramente abra o *command shell*. Para Windows, você pode usar o Git Bash, que vem incluído no Git para Windows. Para Mac e Linux você pode usar o terminal integrado. Verifique se o Git foi instalado através do comando:

```
git --version
git version 2.30.2.windows.1
```

3. Para acessar o seu repositório remoto localmente, você precisará de um token de acesso gerado dentro do seu GitHub. Um tutorial detalhado de como gerar esse token está presente na documentação: https://docs.github.com/en/authentication/keeping-your-account-and-data-secure/managing-your-personal-access-tokens

4. Uma vez que o token foi gerado, você o utilizará para realizar a autenticação local. Vá ao seu repositório remoto no GitHub e copie o link do repositório, clicando no ícone mostrado abaixo:



5. Abra o *command shell* e crie uma pasta localmente no seu computador para que você possa realizar o clone do repositório remoto e utilize o comando cd para navegar para dentro da pasta criada:

```
$ mkdir bootcamp-avanti-ml
```

```
$ cd bootcamp-avanti-ml
```

6. Clone o repositório remoto para o seu ambiente local:

colar nome do repositório

```
$ git clone https://github.com/angelica-ti/hello-world.git
```

7. Copie os arquivos de extensão *.py* ou *.ipynb* para dentro da pasta local /bootcamp-avanti-ml/hello-world

8. Você adicionou os arquivos localmente, para que os arquivos sejam enviados para o repositório remoto, adicione todas as mudanças com o comando *git add .* (você pode optar por adicionar uma a uma das mudanças com o comando *git add* <nome do arquivo>) e, registre essas mudanças com uma mensagem utilizando o *git commit* <msg> e suba as alterações para o seu ambiente remoto com o *git push.*

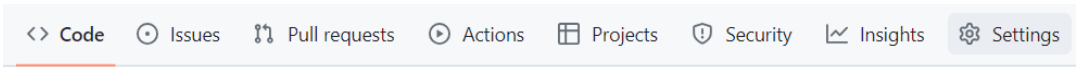
```
$ git add .
```

```
$ git commit “atividade 1 - ML”
```

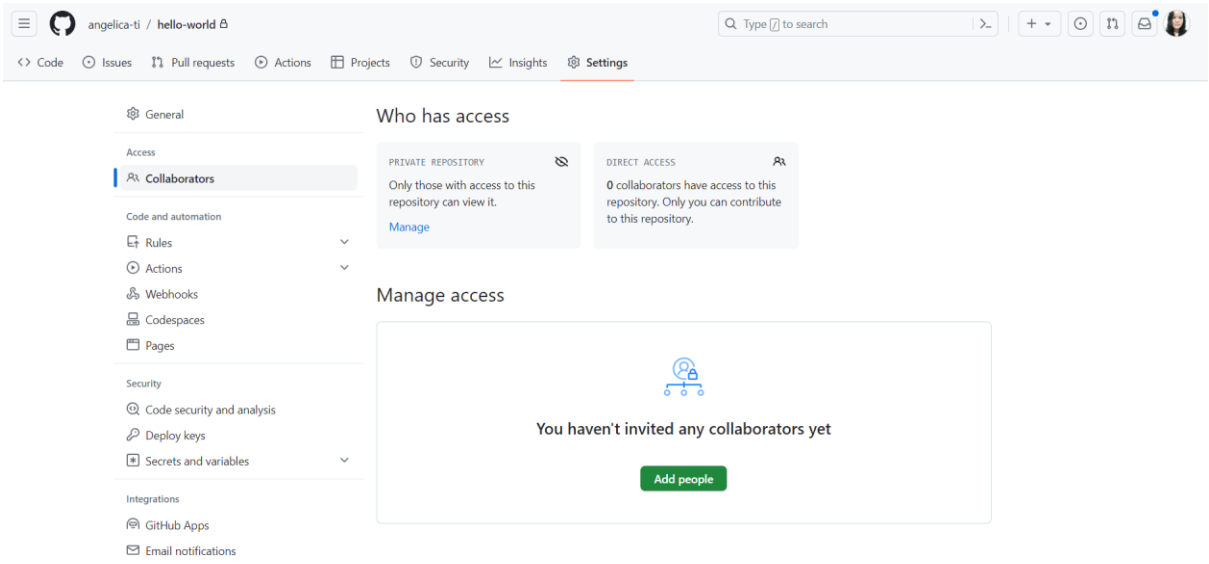
```
$ git push
```

10. Verifique o repositório remoto para conferir se seus arquivos foram enviados corretamente.

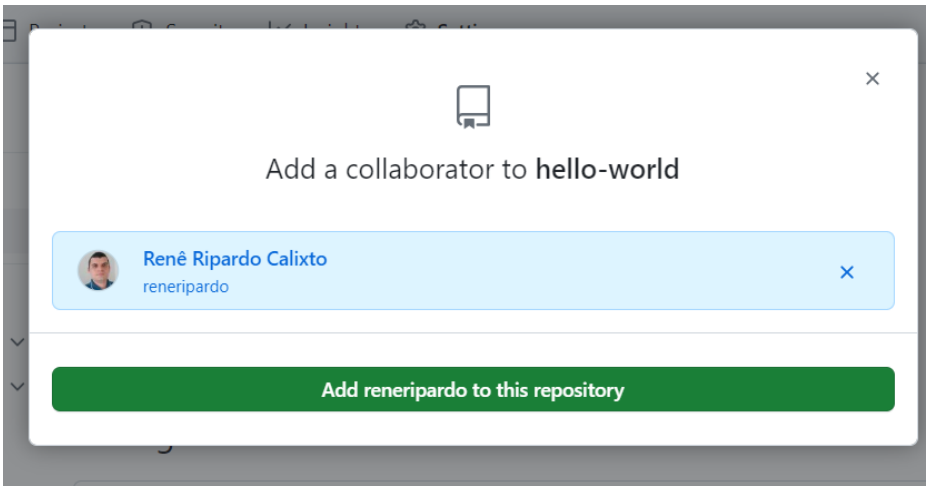
COMPARTILHANDO O REPOSITÓRIO COM COLABORADORES



No menu superior, clique em *Settings* > *Collaborators* > *Add people*



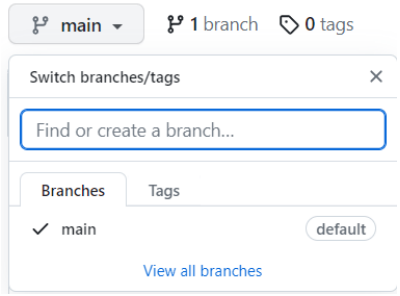
Adicione o e-mail do colaborador e clique em "*Add reneripardo to this repository*".



OUTROS COMANDOS GIT IMPORTANTES

1. **Atualizar Repositório Local – git pull**
   Dentro do seu repositório local via terminal, digite esse comando para obter as atualizações mais recentes de seu repositório remoto.

2. **Criar Branch – git branch**
   Branchs são ramificações do código principal que permitem fazer alterações sem nenhum problema. Sempre que um novo repositório remoto é criado, uma branch principal com o nome main é criada:



   Utilize o comando **git branch <nome da branch>** para criar uma nova branch para o seu repositório.

3. **Verificar status de alterações locais – git status**
   Ao digitar esse comando no terminal dentro do seu repositório local, você verá a listagem de modificações realizadas localmente. Aparecerão em vermelho as alterações não adicionadas e em verde as já adicionadas.

4. **Navegar entre branchs – git checkout**
   O comando **git checkout <nome da branch>** permite que você mude de uma branch para outra facilmente.

5. **Juntar alterações de uma branch em outra – git merge**
   Suponha que uma branch develop foi criada e possui alterações mais recentes que a branch main, para atualizar a branch main com as alterações da develop, você primeiro deve garantir que está na branch main (use o git checkout) e realizar o comando **git merge develop** no terminal.

6. **Listar o histórico de commits – git log**

Esse comando permite verificar o histórico de *commits* realizados juntamente com o seu código *hash*, com essas informações é possível remover algum *commit* realizado indevidamente ou retornar para algum *commit* anterior por exemplo.

Para ver mais detalhes e conhecer outros comandos git, consulte a documentação https://git-scm.com/docs/git.