

<https://gomesweb.com>

Bem com a linguagem de programação python , bibliotecas ( arquivos prontos reutilizável) e framework (ferramenta que gerencia o site como requisições, serve páginas , processas rotas...,orienta na construção do site , vc não constroi tudo do zero pois ela ja ajuda automaticamente no desenvolvimento do site! interessante ) pode-se desenvolver vários projetos . Essa linguagem é uma das mais utilizada a sintaxe é simplificada , você poder usar um terminal instalar nele ou editor IDE que ja vem integrado a linguagem e um terminal pois precisa as vezes , ja o Flask é um framework voltado no desenvolvimento de site ou api dessa forma é preciso seguir as orientações para o funcionamento correto .

Para quem ta começando na programação web ou outra é uma boa idéia aprender python flask até porque você pode usar o git nesse processo e o github . O git é um programa que gerencia as versões do projeto flask como modificações . É preciso instalar o git no terminal também , depois usa-se os comandos do mesmo para fazer uploads para o github (hospedagem de códigos ) . É lógico que vc

```
=====
Aqui comandos para configurar o git
depois de instalar
git config -- global user.email "teuemail"
git config -- global user.name "teuperfil"
=====
Os comandos 5 e 6 só na primeira vez
1. git init      2. git add .
3. git commit -m "Initial commit"
4. git branch -M main
5. git remote add origin https://
github.com/seu-usuario/seu-repositorio.git
6. git push -u origin main

=====
Comando para enviar para o github os
arquivos
1. git init
2. git add . ou git add nome_do_arquivo
3. git commit -m "primeiro versão
enviando" 4. git push
```

precisa ter uma conta no github para conecta essa conta tendo um repositório remoto( pasta) que vc cria a um repositório local criado com o comando git init dentro do diretório do projeto . Depois de estabelecido essa conexão de repositórios (remota e local - aqui precisa de uma forma de autenticação ou senha/tokens/chaves) ai sim vc pode enviar os arquivos do flask para o github neste caso programaticamente . Vc pode usar o aplicativo termux como ambiente de desenvolvimento no android , ele possui um terminal e editores como o nano etc .Se vc quiser hospedar teu projeto se não não precisa do git ou conta no github ou outra hospedagem não é obrigatório fazer esse procedimento descrito

A forma de colocar online o site dependerá da hospedagem geralmente elas vão precisa que voce saiba o git e siga as orientações deles . E como desenvolver um site simples com o flask do python? Instala o flask primeiro em um editor IDE ou termux, como ?? usa o comando pip install flask . É claro esse pip para funcionar primeiro instale o python com o comando pkg install python ou python ja vem instalado . **Agora** , voce precisa organizar teu projeto , estrutura de pastas e arquivos . O básico criando uma

```
from flask import Flask
app = Flask(__name__)
@app.route('/')
def index():
    return 'ola mundo'

@app.route('/sobre')
def sobre():
    return 'mais texto'

if __name__ == '__main__':
    app.run(debug=True)
```

pasta chamado `teu_diretorio` , dentro coloque o arquivo `app.py` , `requirements.txt`, `procfile`,`README.md`... Esses arquivos vão servir para o servidor conhecer o teu projeto . Dentro do `app.py` você digita as seguintes linhas de códigos , observando a indentação .

Nos arquivos citados acima em `teu_projeto` , como o `requirements.txt`, bem esse arquivo vai lista as dependências do projeto ou bibliotecas . Você pode gerar automaticamente a lista com o seguinte no terminal do editor ou termux , digite o seguinte comando `pip freeze > requirements.txt` bem como sua instalação `pip install -r requirements.txt`. Já no arquivo `procfile` escreva a linha `web:gunicorn app:app` , agora de onde veio isso ? cara , esse primeiro `app` faz referencia a nome do teu arquivo `app.py` sem a extensão , já o outro `app` é uma variavel obtida do `app=Flask(__name__)`. Vc pode instalar o `gunicorn` fazendo `pip install gunicorn`. Lembrando que os sites que hospedam permitem instalar por lar basta vc preencher as opções no `deploy`( processo de publicar teu site online). Já em `README.ME` nesse vc apenas instrui o leitor sobre teu site .

Isso vai acontecer de forma local , o flask usando a metodo `app.run()` vai permitir criar um servidor local possa ser acessado no navegador gerar um link . Então vc consegue visualizar a renderização das rotas páginas pelo navegador, bastando digitar a rota que vc definiu no arquivo `app.py` , tipo `app.route('/sobre')`, `app.route('/')`. Aqui temos duas rotas , para entra digite o link gerado com o nome após a barra tipo `http://127.0.0.1/sobre` , para a rota / não precisa digitar nada o flask entende que vc ta pedindo acesso na raiz. Como dito acima o `deploy` dependendo da hospedagem, vc pode uma vez hospedado teu código no github enviar para outra hospedagem . Outra possibilidade é enviar teu projeto diretamente para uma hospedagem

sem passar pelo github , porém embora não passe por ele , essa hospedagem usa o git . Nesse caso dentro do site que hospeda vc precisa criar um aplicativo , baixar alguns arquivos dele , instalar o git e usar os comandos do git e da hospedagem para o deploy . O git é fundamental saber nesse processo de envio de arquivos seja para o github ou outros website que precisam dele. Algumas hospedagem até facilitam instalar arquivos automaticamente dentro delas para funcionar o código flask , como o arquivo requirements.txt. Pode acontecer dúvidas e erros nisso contudo é normal vc na sua jornada como programador vai tentando identificar e corrigir. Tanto no github como outras provavelmente precisaram ou do teu email associado e também o perfil do repositório remoto que vc pode criar no github na sua conta , pra que??? para conectar o teu repositório local e remoto. Porém existe uma outra maneira de enviar os arquivos para github sem precisar programaticamente , ou seja , usa esses comandos. Como assim ??? dentro do github existe opções de uploads que vc pode usar para enviar sem usar o git. Nos próximos arquivos se aprofundamos como fazer conexão usando tokens ou chaves geradas para repositórios do github , também como usar o html com o python com a função render\_templates , como fazer referência a arquivos css e bancos de dados bem como deploy de algumas hospedagem e desenvolver códigos flask ou python voltados ao ensino da matemática . Sem dúvida a programação com python flask é muito interessante um vasto recursos disponível.