GIT

gitbash
git config --global user.name "Seu Nome"
git config --global user.email "seuemail@exemplo.com"

terminal
git remote add origin https://github.com/seu-usuario/nome-do-repositorio.git

git remote -v # Visualiza o repositório
git init # Inicia um repositório Git
git clone URL # Clona um repositório do GitHub
git status # Mostra arquivos modificados
git add . # Adiciona todos os arquivos para o commit
git commit -m "mensagem" # Cria um ponto de salvamento
git branch -M main # Cria o main no primerio commit
git push origin main # Envia as mudanças pro GitHub
git pull origin main # Baixa as mudanças do GitHub
git remote set-url origin URL # Se conecta a outro repositório
git checkout -b main # Muda a brach local
git checkout -- app/models.py # ignora as alterações para o pull
git push --set-upstream origin main #serve para criar uma tracking connection entre o repositório online e local("--set-upstream" pode ser substituido por "-u")

PASTA .venv

python -m venv .venv
.venv\Scripts\activate

Arquivo requirements

pip freeze > requirements.txt #cria a pasta requirements e a atualiza
pip install -r requirements.txt #instala as bibliotecas da pasta requirements

Bibliotecas

pip install flask
pip install email-validator
pip install Flask-SQLAlchemy Flask-Migrate(importar o flask para bd sql)
pip install Flask-WTF
pip install python-dotenv
pip install flask-login
pip install flask-bcrypt

Banco de dados SQLAlquemy

flask db init(para criar)
flask db migrate(toda as vezes alterar o bd)

flask db upgrade

Limpar banco de dados(executar no python dentro do venv)

```
from app import app, db

with app.app_context():
    db.drop_all()
    db.create_all()

exit()
```

Comandos Docker

```
# Ver containers em execução
docker ps

# Parar container
docker-compose down

# Limpar tudo
docker system prune -a

# Ver logs
docker-compose logs -f
```

Comandos poetry

```
poetry new <nome do pj>                  #cria o ambiente inicial do pj automaticamente
poetry show
poetry add <nome da bib>@1.0.1           #O arroba serve para especificar a versão
poetry remove <nome da bib>
poetry install                           #Instala as parada(tipo o pip install requirements.txt)
poetry self add poetry-plugin-shell      #adiciona o comando shell
poetry shell                             #inicia um terminal dentro do ambiente do projeto
poetry run uvicorn <caminho de pastas>.main:app –reload    #rodar
```