**README para rodar o Projeto de BD1**

**Github do projeto:** [**kenjiThiago/ProjetoBD1**](https://github.com/kenjiThiago/ProjetoBD1)

**Requisitos**

#### 

**Backend:**

* **Python:** 3.12.7
  + **Nota:** No Windows, pode ser necessário instalar o *Microsoft C++ Build Tools*, pois o Python depende de suporte para compilar alguns componentes em C.

**Frontend:**

* **Node.js:** 22.11.0
* **NPM:** 10.9.0

**Banco de Dados:**

* **PostgreSQL:** 16.6

**Opcional:**

* **Git:** Qualquer versão recente, caso precise clonar o repositório do projeto no GitHub.

**Observação:** Estas são as versões utilizadas durante o desenvolvimento e teste do projeto. Outras versões podem funcionar, mas não foram verificadas.

**Etapas para rodar o projeto**

**Windows**

**Importante:**Para evitar problemas ao utilizar o ambiente virtual do Python, prefira usar o **Prompt de Comando (cmd)** em vez do **PowerShell**. No PowerShell, configurações adicionais são necessárias para executar o ambiente virtual do Python, enquanto no cmd isso não é necessário.

**Linux**

Para rodar no Linux, basta substituir o comando da etapa 3 do backend. Ao invés de usar .\venv\Scripts\activate, utilize source .venv/bin/activate para ativar o ambiente virtual.

#### 

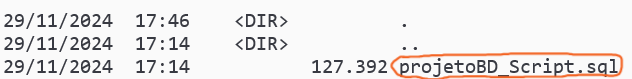
#### 

#### 

**SQL**

O script SQL usado para gerar o banco de dados do projeto está localizado no diretório **“Códigos SQL”**.





**1. Criar o Banco de Dados**

* Acesse o **pgAdmin** (ou outra interface gráfica do PostgreSQL) e crie um novo banco de dados.
* Abra o diretório **"Códigos SQL"** no projeto e execute o script projetoBD\_Script.sql para criar o banco de dados com todas as tabelas e dados necessários.

### 

### 

### 

### 

### 

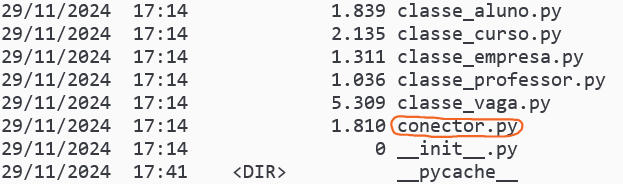
### 

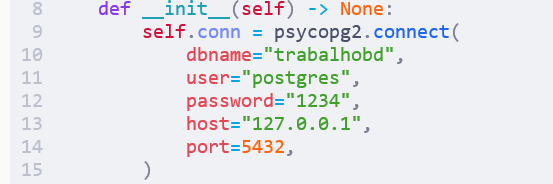
### 

**Importante**

Antes de seguir com as instruções para configurar o backend, edite o arquivo:  
**ProjetoBD1\Backend\Database\conector.py**Este arquivo contém as configurações de conexão com o banco de dados e precisa ser ajustado para refletir o nome, usuário e senha do banco criado.







No arquivo **ProjetoBD1\Backend\Database\conector.py**, atualize os seguintes parâmetros:

* **password**: Substitua pela senha que você utiliza no PostgreSQL.
* **dbname**: Substitua pelo nome do banco de dados que você criou.

### 

### 

### 

### 

### 

**Backend**

**1. Acessar o Diretório do Backend**

* Navegue até o diretório **ProjetoBD1\Backend**.

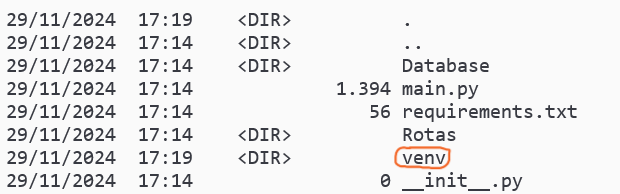


### 

**2. Criando o Ambiente Virtual em python (venv)**

* No diretório **ProjetoBD1\Backend**, rode o comando python -m venv venv**.** Esse comando irá criar um subdiretório no diretório Backend chamado venv.





### 

### 

### 

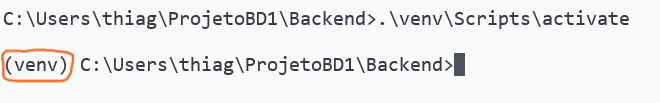
### 

### 

### 

**3. Iniciando o venv**

* Execute o comando **.\venv\Scripts\activate** no terminal.
* Após executar o comando, um indicador **(venv)** aparecerá no início da linha do terminal, confirmando que o ambiente virtual do Python foi ativado.



### 

**4. Instalando as Dependências para o Backend**

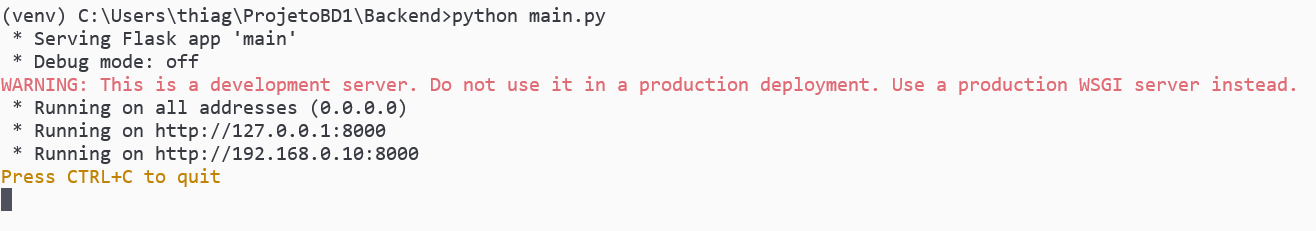
* No diretório **ProjetoBD1\Backend**, execute o comando pip install -r requirements.txt para instalar todas as dependências listadas no arquivo **requirements.txt**, necessárias para rodar o backend.



### 

**5. Rodando o Servidor Backend**

* Execute o comando **python main.py** para rodar o servidor backend.



### 

### 

### 

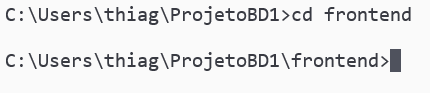
### 

**Frontend**

#### 

**1. Primeiros Passos**

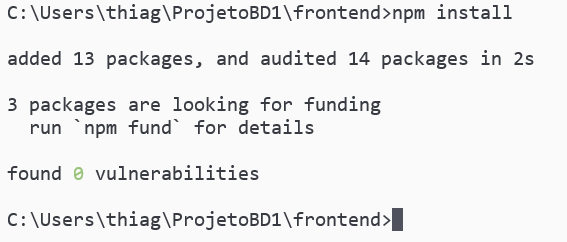
* Abra um novo terminal e navegue até o diretório **frontend**.  
  **Importante:** Não feche o terminal onde o servidor backend está rodando.



### 

**2. Instalando Dependências para Rodar o Frontend**

* No diretório **frontend**, execute o comando npm install para instalar todas as dependências necessárias para rodar o frontend.



### 

### 

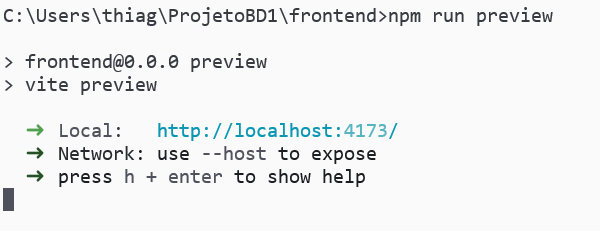
### 

**3. Rodando o Frontend**

* Execute o comando npm run build para realizar o build dos arquivos do frontend.



* Após o build, execute o comando npm run preview para iniciar o servidor do frontend.



* Para entrar no frontend acesse o http://localhost:4173/

