Para fazer a conexão com um banco de dados usando Python em um ambiente de desenvolvimento backend e frontend, você pode usar o framework Flask para criar uma aplicação web simples. Vou mostrar um exemplo usando Flask e SQLite, que é um banco de dados leve e embutido. O exemplo inclui um backend básico que conecta ao banco de dados e uma interface frontend para interagir com os dados.

**Backend com Flask e SQLite**

1. **Instalação das Dependências** Primeiro, instale as bibliotecas necessárias:

bash

Copiar código

pip install flask flask\_sqlalchemy

1. **Código do Backend**

python

Copiar código

from flask import Flask, render\_template, request, redirect, url\_for

from flask\_sqlalchemy import SQLAlchemy

app = Flask(\_\_name\_\_)

app.config['SQLALCHEMY\_DATABASE\_URI'] = 'sqlite:///database.db'

db = SQLAlchemy(app)

class User(db.Model):

id = db.Column(db.Integer, primary\_key=True)

name = db.Column(db.String(50), nullable=False)

email = db.Column(db.String(50), nullable=False)

def \_\_repr\_\_(self):

return f'<User {self.name}>'

@app.route('/')

def index():

users = User.query.all()

return render\_template('index.html', users=users)

@app.route('/add', methods=['POST'])

def add\_user():

name = request.form.get('name')

email = request.form.get('email')

new\_user = User(name=name, email=email)

db.session.add(new\_user)

db.session.commit()

return redirect(url\_for('index'))

if \_\_name\_\_ == '\_\_main\_\_':

db.create\_all()

app.run(debug=True)

1. **Frontend (Templates)** Crie uma pasta chamada templates no mesmo diretório do script acima e adicione um arquivo chamado index.html com o seguinte conteúdo:

html

Copiar código

<!DOCTYPE html>

<html>

<head>

<title>Users</title>

</head>

<body>

<h1>Users</h1>

<form action="/add" method="post">

<input type="text" name="name" placeholder="Name" required>

<input type="email" name="email" placeholder="Email" required>

<button type="submit">Add User</button>

</form>

<ul>

{% for user in users %}

<li>{{ user.name }} - {{ user.email }}</li>

{% endfor %}

</ul>

</body>

</html>

**Explicação do Código**

1. **Backend**
   * A aplicação Flask é configurada com uma URI de banco de dados SQLite (sqlite:///database.db).
   * A classe User define um modelo de tabela com colunas para id, name e email.
   * A rota / renderiza uma lista de usuários armazenados no banco de dados.
   * A rota /add permite adicionar novos usuários ao banco de dados via formulário.
2. **Frontend**
   * O template index.html inclui um formulário para adicionar novos usuários e uma lista para exibir os usuários existentes.

**Executando a Aplicação**

1. Salve o código do backend em um arquivo chamado app.py.
2. Crie a pasta templates e adicione o arquivo index.html conforme mostrado.
3. Execute o backend com o comando:

bash

Copiar código

python app.py

1. Abra o navegador e acesse http://127.0.0.1:5000/ para ver a interface web.

Esse exemplo básico demonstra como conectar a um banco de dados, criar um modelo, e interagir com o banco de dados através de uma interface web utilizando Flask para backend e HTML para frontend.