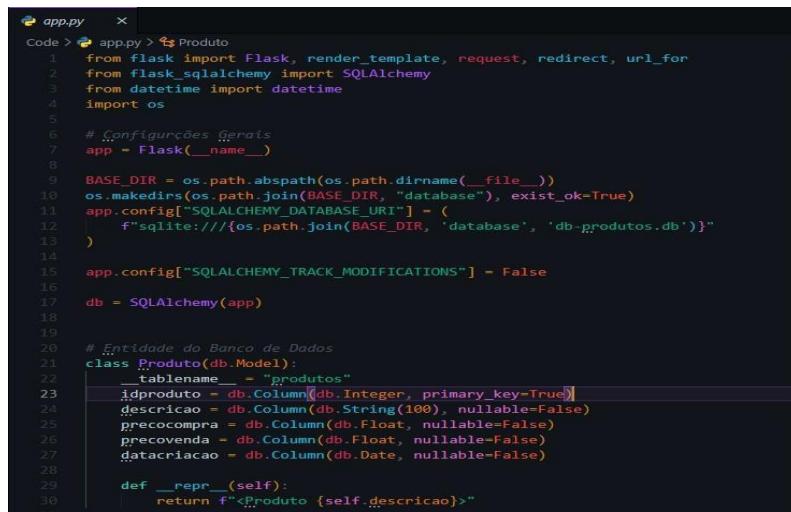
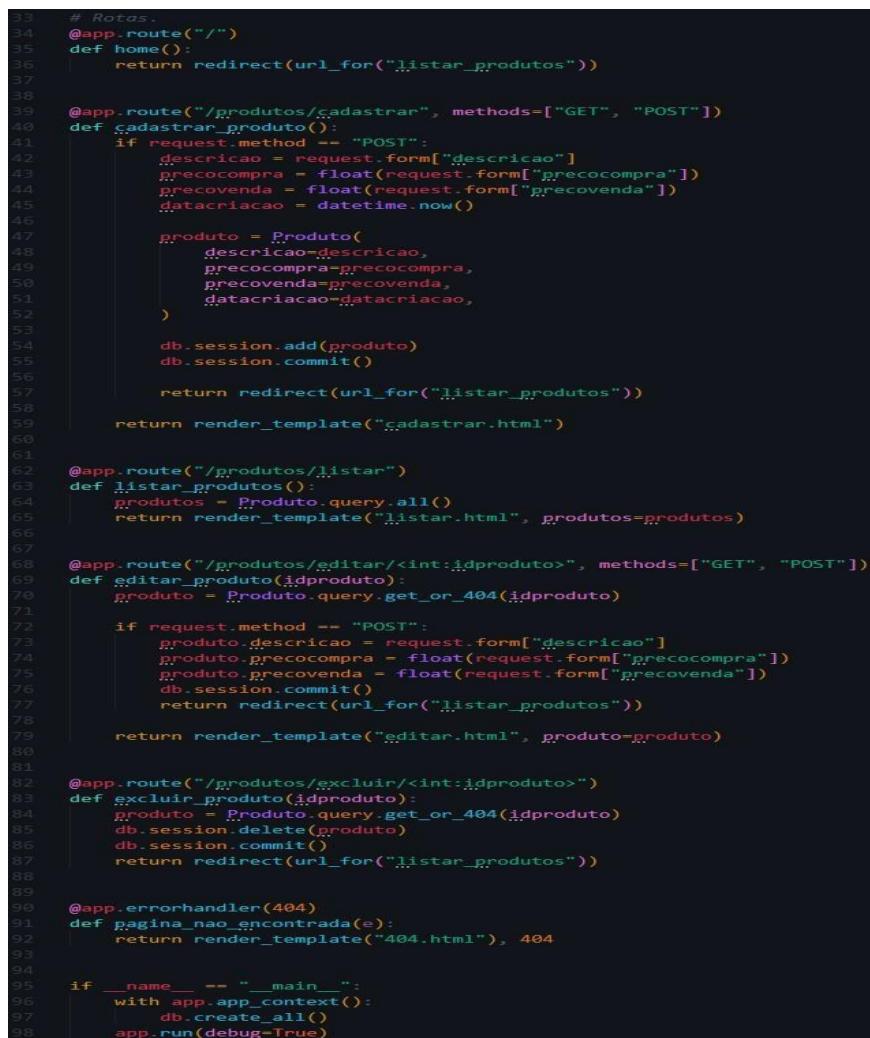


Configuração do Flask e banco de dados SQLite.



```
Code > app.py > Produto
 1 from flask import Flask, render_template, request, redirect, url_for
 2 from flask_sqlalchemy import SQLAlchemy
 3 from datetime import datetime
 4 import os
 5
 6 # Configurações Gerais
 7 app = Flask(__name__)
 8
 9 BASE_DIR = os.path.abspath(os.path.dirname(__file__))
10 os.makedirs(os.path.join(BASE_DIR, "database"), exist_ok=True)
11 app.config["SQLALCHEMY_DATABASE_URI"] = (
12     f"sqlite:///{{os.path.join(BASE_DIR, 'database', 'db-produtos.db')}}"
13 )
14
15 app.config["SQLALCHEMY_TRACK_MODIFICATIONS"] = False
16
17 db = SQLAlchemy(app)
18
19
20 # Entidade do Banco de Dados
21 class Produto(db.Model):
22     __tablename__ = "produtos"
23     idproduto = db.Column(db.Integer, primary_key=True)
24     descricao = db.Column(db.String(100), nullable=False)
25     precocompra = db.Column(db.Float, nullable=False)
26     precovenda = db.Column(db.Float, nullable=False)
27     datacriacao = db.Column(db.Date, nullable=False)
28
29     def __repr__(self):
30         return f"<Produto {self.descricao}>"
```

Rotas Flask e suas funções.



```
10 # Rotas.
11 @app.route("/")
12 def home():
13     return redirect(url_for("listar_produtos"))
14
15
16 @app.route("/produtos/cadastrar", methods=["GET", "POST"])
17 def cadastrar_produto():
18     if request.method == "POST":
19         descricao = request.form["descricao"]
20         precocompra = float(request.form["precocompra"])
21         precovenda = float(request.form["precovenda"])
22         datacriacao = datetime.now()
23
24         produto = Produto(
25             descricao=descricao,
26             precocompra=precocompra,
27             precovenda=precovenda,
28             datacriacao=datacriacao,
29         )
30
31         db.session.add(produto)
32         db.session.commit()
33
34         return redirect(url_for("listar_produtos"))
35
36     return render_template("cadastrar.html")
37
38
39 @app.route("/produtos/listar")
40 def listar_produtos():
41     produtos = Produto.query.all()
42     return render_template("listar.html", produtos=produtos)
43
44
45 @app.route("/produtos/editar<int:idproduto>", methods=["GET", "POST"])
46 def editar_produto(idproduto):
47     produto = Produto.query.get_or_404(idproduto)
48
49     if request.method == "POST":
50         produto.descricao = request.form["descricao"]
51         produto.precocompra = float(request.form["precocompra"])
52         produto.precovenda = float(request.form["precovenda"])
53         db.session.commit()
54         return redirect(url_for("listar_produtos"))
55
56     return render_template("editar.html", produto=produto)
57
58
59 @app.route("/produtos/excluir<int:idproduto>")
60 def excluir_produto(idproduto):
61     produto = Produto.query.get_or_404(idproduto)
62     db.session.delete(produto)
63     db.session.commit()
64     return redirect(url_for("listar_produtos"))
65
66
67 @app.errorhandler(404)
68 def pagina_nao_encontrada(e):
69     return render_template("404.html"), 404
70
71
72 if __name__ == "__main__":
73     with app.app_context():
74         db.create_all()
75     app.run(debug=True)
```

Modelo Produto definido para o SQLAlchemy.



```
models.py
Code > models.py > ...
1  from Code.app import db
2
3
4  class Produto(db.Model):
5      __tablename__ = "produtos"
6
7      idproduto = db.Column(db.Integer, primary_key=True)
8      descricao = db.Column(db.String(100), nullable=False)
9      precocompra = db.Column(db.Float, nullable=False)
10     precovenda = db.Column(db.Float, nullable=False)
11     datacriacao = db.Column(db.Date, nullable=False)
12
13     def __repr__(self):
14         return f"<Produto {self.descricao}>"
15
```

Formulário HTML para o cadastro de produtos.



```
index.html
58 <body>
59   <div class="container">
60     <h2>Cadastrar Produto</h2>
61     <form method="post">
62       <input name="descricao" placeholder="Descrição" required>
63       <input name="precocompra" type="number" step="0.01" placeholder="Preço de Compra" required>
64       <input name="precovenda" type="number" step="0.01" placeholder="Preço de Venda" required>
65       <button type="submit">Salvar</button>
66     </form>
67     <a href="/produtos/listar">Listar Produtos</a>
68   </div>
69 </body>
70
71 </html>
```

Banco de dados SQLite demonstrando a tabela de produtos.

The screenshot shows a SQLite database interface with the following details:

- File tabs: app.py and db-produtos.db (active).
- Navigation bar: Code > database > db-produtos.db
- Toolbar buttons: SELECT \*, FROM produtos, Schema, Query Editor, Auto Reload, Find, Other Tools...
- Table structure:

	<code>idproduto INTEGER NOT NULL PRIMARY KEY</code>	<code>descricao VARCHAR(100) NOT NULL</code>	<code>precocompra FLOAT NOT NULL</code>	<code>precovenda FLOAT NOT NULL</code>	<code>datacriacao DATE NOT NULL</code>
1	1	Descrição do Produto	12.0	29.0	2025-11-05
	+				