

Técnico em
DESENVOLVIMENTO DE

SISTEMAS Técnico em



DESENVOLVIMENT

O SISTEMAS DE

PROGRAMAÇÃO FRONT-END

Técnico em
DESENVOLVIMENT
O DE SISTEMAS

Apagando
produtos

Para apagar produtos já

cadastrados, o processo é bem simples, basta adicionar uma rota que irá apagar o produto do banco de dados através de seu id_prod.

Abra o seu arquivo quitanda.py e adicione o seguinte código:



PROGRAMAÇÃO FRONT-END AULA 25

```
#ROTA DE EXCLUSÃO
@app.route("/excluir/<id>")
def excluir(id):
    if verifica_sessao():
        id = int(id)
        conexao = conecta_data
        conexao.execute('DELETE
            id_prod = ?', (id,))
        conexao.commit()
        conexao.close()
        return redirect('/adm'
    else:
        return redirect("/logi
```

O DE SISTEMAS

Editando

Para construir a função de edição de produtos precisamos criar:

- Uma rota que irá fazer a seleção do produto no banco de dados através de seu id_prod e chamar a página editprodutos.html passando o produto selecionado como parâmetro.
- Uma rota que irá receber as novas informações e gravar no banco de dados.
- Uma página que terá um formulário contendo os dados já existentes no banco de dados e que possibilite a edição e a gravação dos novos dados.

Começaremos com a primeira rota:

Técnico em **DESENVOLVIMENTO DE SISTEMAS** Rota do produto a ser editado



```
#CRIAR A ROTA DO EDITAR
```

```
@app.route("/editprodutos/<id_prod>")
```

```
def editar(id_prod):
```

```
    if verifica_sessao():
```

```
        iniciar_db()
```

```
        conexao = conecta_database()
```

```
        produtos = conexao.execute('SELECT * FROM produtos WHERE id_prod = ?',(id_prod,)).fetchall()
```

```
        conexao.close()
```

```
        title = "Edição de produtos"
```

```
        return render_template("editprodutos.html",produtos=produtos,title=title)
```

```
    else:
```

```
        return redirect("/login")
```

Técnico em

DESENVOLVIMEN

TO DE SISTEMAS



PROGRAMAÇÃO FRONT-END AULA 25

39

Rota que irá gravar os novos dados no banco

#CRIAR A ROTA PARA TRATAR A EDIÇÃO

```
@app.route("/editarprodutos", methods=['POST'])
```

```
def editprod():
```

```
    id_prod = request.form['id_prod']
```

```
    nome_prod=request.form['nome_prod']
```

```
    desc_prod=request.form['desc_prod']
```

```
    preco_prod=request.form['preco_prod']
```

```
    img_prod=request.files['img_prod']
```

```
    id_foto=str(uuid.uuid4().hex)
```

```
    filename=id_foto+nome_prod+'.png'
```

```
    img_prod.save("static/img/produtos/"+filename)
```

```
    conexao = conecta_database()
```

```
    conexao.execute('UPDATE produtos SET nome_prod = ?, desc_prod = ?, preco_prod = ?, img_prod = ?
```

```
WHERE id_prod = ?',(nome_prod,desc_prod,preco_prod,filename,id_prod))
```

```
    conexao.commit()
```

```
    conexao.close()
```

```
    return redirect('/adm')
```

Para construir a função de edição de produtos precisamos criar uma rota que irá fazer a seleção do produto no banco de dados através de seu id_prod e chamar a página editprodutos.html passando o produto

selecionado como parâmetro.

Começaremos com a primeira rota:

Técnico em

DESENVOLVIMEN

TO DE SISTEMAS



PROGRAMAÇÃO FRONT-END AULA 25

40

Criando a página de edição de produtos

Crie um novo arquivo chamado editprodutos.html dentro da pasta templates e coloque o código abaixo para herdar o conteúdo da página modelo.html:

```
1 {% extends 'modelo.html' %}
2 {% block conteudo %}
```

Agora vamos

adicionar o restante do código, confira no próximo slide:

Criando a página de edição de produtos – parte 2


```

3  <div class="container">
4      <div class="row justify-content-center text-center">
5          <h1 class="mb-3 display-4">Editar produto</h1>
6          <form action="/editarprodutos" method="post" enctype="multipart/form-data" class="col-sm-10
7              col-md-8 col-lg-6 text-center needs-validation" novalidate>
8              {% for produto in produtos %}
9                  <div class="form-floating mb-3">
10                     <input type="text" name="nome_prod" class="form-control" value="{{ produto['nome_prod'] }}
11                     placeholder="" autofocus />
12                     <label for="nome_prod">Produto</label>
13                 </div>
14                 <div class="form-floating mb-3">
15                     <textarea name="desc_prod" class="form-control" placeholder="">{{ produto['desc_prod'] }}
16                     <label for="desc_prod">Descrição</label>
17                 </div>

```

Técnico em
DESENVOLVIMEN

TO DE SISTEMAS



PROGRAMAÇÃO FRONT-END AULA 25
42

Criando a página de edição de produtos – parte 3

```
16 <div class="input-group mb-3">
17   <span class="input-group-text">R$</span>
18   <div class="form-floating">
19     <input type="text" name="preco_prod" class="form-control" value="{ { produto
20       ['preco_prod'] } }" placeholder="" />
21     <label for="preco_prod">Preço</label>
22   </div>
23 </div>
24 <div class="input-group m2-3">
25   <span class="input-group-text"><i class="bi bi-file-image"></i></span>
26   <div class="form-floating">
27     <input type="file" name="img_prod" id="img_prod" accept="image/jpeg,image/png"
28       class="form-control" required />
29     <div class="invalid-feedback">
30       Você deve enviar a foto do produto
31     </div>
32   </div>
33 </div>
```



Técnico em

DESENVOLVIMENTO DE SISTEMAS

Criando a página de edição de produtos – parte 4



Procure na documentação do Bootstrap pelo script de validação de formulário (vide aula anterior) e copie e cole dentro de um tag script no final da página (antes da linha do final do bloco de conteúdo)



Técnico em

DESENVOLVIMENTO DE SISTEMAS

De volta ao lar (home)

PROGRAMAÇÃO FRONT-END AULA 25

44

De volta à página home.html, falta adicionar a lista de produtos a ser exibida após o administrador da página adicionar produtos.

Abra o arquivo home.html e adicione o seguinte código a partir da linha 3:



Isso fará com que, caso não haja produtos cadastrados, seja exibida uma mensagem ao cliente. Veja a continuação na próxima página:



Técnico em

DESENVOLVIMENTO DE SISTEMAS

Editando a página home – parte 2



Técnico em

DESENVOLVIMEN TO DE SISTEMAS

PROGRAMAÇÃO FRONT-END AULA 25
46



Construindo a página cadastro.html

Leiaute

previsto para a página home com produtos cadastrados.



Técnico em

DESENVOLVIMEN TO DE SISTEMAS

PROGRAMAÇÃO FRONT-END AULA 25
47



Construindo a página cadastro.html

Leiaute

previsto para a página home sem produtos cadastrados.

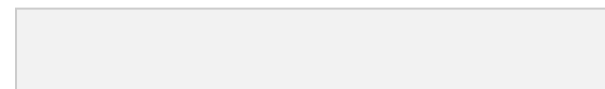


Técnico em

DESENVOLVIMEN

TO DE SISTEMAS

Pesquisando
produtos



PROGRAMAÇÃO FRONT-END AULA 25
48

Para construir a função de pesquisa de produtos precisamos criar uma rota que irá fazer a busca do(s) produto(s) no banco de dados que correspondam ao termo inserido no campo de busca. Essa rota irá armazenar os resultados e chamar novamente a página home.html passando os produtos achados como parâmetro.

Lembre-se de que já temos o campo de busca no cabeçalho da página modelo.html, mas você poderá construir mais locais de buscas caso deseje.

Abra o arquivo quitanda.py e adicione o seguinte código:



Técnico em

DESENVOLVIMENTO DE SISTEMAS

Adicionando a rota de busca:

PROGRAMAÇÃO FRONT-END AULA 25
49



Técnico em

DESENVOLVIMENTO DE SISTEMAS

Finish?

PROGRAMAÇÃO FRONT-END AULA 25

50

Agora já temos todas as funcionalidades previstas no projeto em funcionamento:

- O administrador poderá criar, editar e excluir os produtos.
- O visitante poderá ver e pesquisar entre os produtos existentes.

Fique à vontade para aplicar novas funcionalidades se desejar.

Não se esqueça de atualizar o arquivo readme.md caso implemente novas funções.



Técnico em

DESENVOLVIMENTO DE SISTEMAS

