

Atividade-08 Banco de Dados

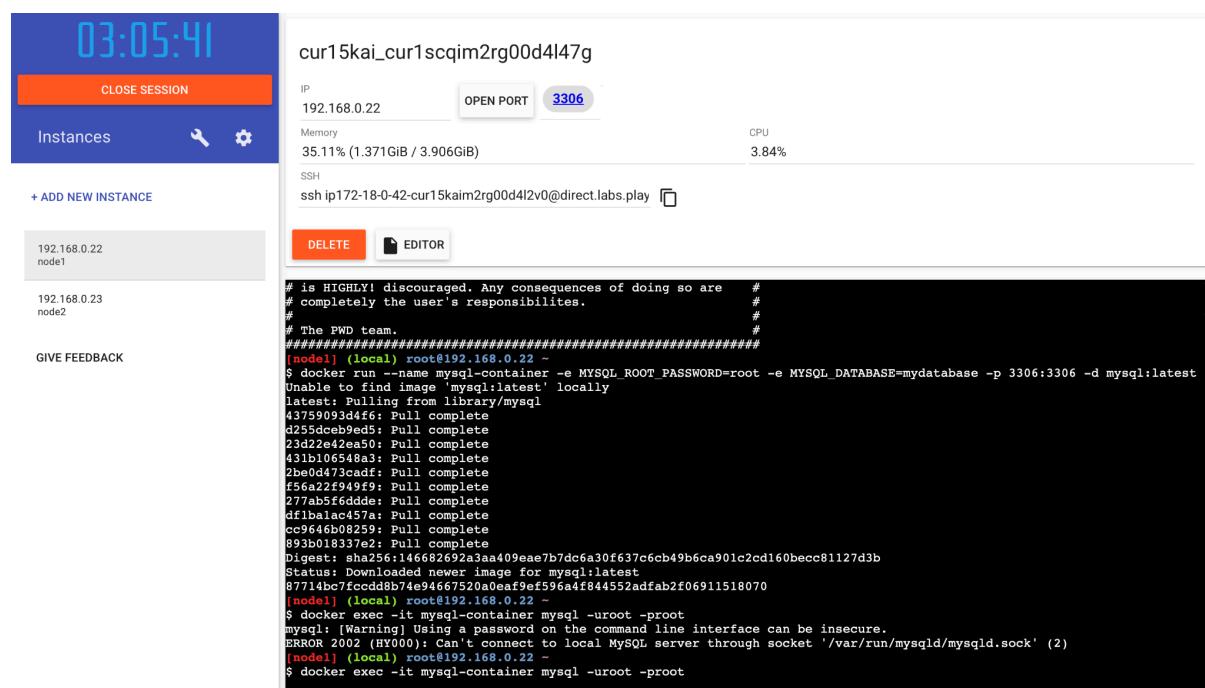
Professor: Ricardo Duarte Taveira

Aluno: Gabriel Souza de Alencar - P4 Informática

Implementar os CRUDs usando o Python (Aula do dia 10/02/2024) das tabelas do banco de dados criado na atividade-07.

Tarefa em dupla postada no GitHub de cada participante com o link do Github referenciado no Google Classroom.

Instância 1 - Criando e acessando o contêiner MySQL no Docker:



Criando as tabelas no banco de dados “mydatabase” (atividade-07) com o seguinte comando:

```
USE mydatabase;
SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
START TRANSACTION;
SET time_zone = "+00:00";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
```

```
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;  
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;  
/*!40101 SET NAMES utf8mb4 */;
```

```
CREATE TABLE `TB_CLIENTES` (  
    `id` int NOT NULL,  
    `nome` varchar(50) NOT NULL,  
    `sobrenome_contato` varchar(50) NOT NULL,  
    `nome_contato` varchar(50) NOT NULL,  
    `telefone` int NOT NULL,  
    `endereco1` varchar(50) NOT NULL,  
    `endereco2` varchar(50) DEFAULT NULL,  
    `cidade` varchar(50) NOT NULL,  
    `estado` varchar(50) NOT NULL,  
    `cep` int NOT NULL,  
    `pais` varchar(50) NOT NULL,  
    `empregado_id` int NOT NULL,  
    `limite_credito` double NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
```

```
CREATE TABLE `TB_DETALHES_DO_PEDIDO` (  
    `id` int NOT NULL,  
    `produto_id` int NOT NULL,  
    `quantidade` int NOT NULL,  
    `preco_cada` double NOT NULL,  
    `linha_pedido` int NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
```

```
CREATE TABLE `TB_EMPREGADOS` (  
    `id` int NOT NULL,  
    `sobrenome` varchar(50) NOT NULL,  
    `nome` varchar(50) NOT NULL,  
    `extensao` varchar(10) NOT NULL,  
    `email` varchar(100) NOT NULL,  
    `escritorio_id` int NOT NULL,  
    `relatorio` int,  
    `cargo` varchar(50) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
```

```
CREATE TABLE `TB_ESCRITORIOS` (  
    `id` int NOT NULL,  
    `cidade` varchar(50) NOT NULL,  
    `telefone` int NOT NULL,  
    `endereco1` varchar(50) NOT NULL,  
    `endereco2` varchar(50) DEFAULT NULL,  
    `estado` varchar(50) NOT NULL,  
    `pais` varchar(50) NOT NULL,
```

```
`cep` int NOT NULL,  
 `territorio` varchar(10) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
```

```
CREATE TABLE `TB_LINHAS_PRODUTO` (  
 `id` int NOT NULL,  
 `descricao_texto` text NOT NULL,  
 `descricao_html` text,  
 `imagem` blob  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
```

```
CREATE TABLE `TB_PAGAMENTOS` (  
 `id` int NOT NULL,  
 `cliente_id` int NOT NULL,  
 `data_pagamento` date NOT NULL,  
 `valor` double NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
```

```
CREATE TABLE `TB_PEDIDOS` (  
 `id` int NOT NULL,  
 `data_pedido` date NOT NULL,  
 `data_requerida` date NOT NULL,  
 `data_envio` date NOT NULL,  
 `status` varchar(15) NOT NULL,  
 `comentarios` text,  
 `cliente_id` int NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
```

```
CREATE TABLE `TB_PRODUTOS` (  
 `id` int NOT NULL,  
 `nome_produto` varchar(70) NOT NULL,  
 `linhas_produto_id` int NOT NULL,  
 `tamanho_produto` varchar(10) NOT NULL,  
 `fornecedor_produto` varchar(50) NOT NULL,  
 `descricao_produto` text NOT NULL,  
 `quantidade_estoque` int NOT NULL,  
 `preco_compra` double NOT NULL,  
 `MSRP` double NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
```

```
ALTER TABLE `TB_CLIENTES`  
 ADD PRIMARY KEY (`id`),  
 ADD KEY `empregado_id` (`empregado_id`);
```

```
ALTER TABLE `TB_DETALHES_DO_PEDIDO`  
 ADD PRIMARY KEY (`id`, `produto_id`),  
 ADD KEY `produto_id` (`produto_id`);
```

```
ALTER TABLE `TB_EMPREGADOS`  
    ADD PRIMARY KEY (`id`),  
    ADD KEY `relatorio`(`relatorio`),  
    ADD KEY `escritorio_id`(`escritorio_id`);  
  
ALTER TABLE `TB_ESCRITORIOS`  
    ADD PRIMARY KEY (`id`);  
  
ALTER TABLE `TB_LINHAS_PRODUTO`  
    ADD PRIMARY KEY (`id`);  
  
ALTER TABLE `TB_PAGAMENTOS`  
    ADD PRIMARY KEY (`id`, `cliente_id`),  
    ADD KEY `cliente_id`(`cliente_id`);  
  
ALTER TABLE `TB_PEDIDOS`  
    ADD PRIMARY KEY (`id`),  
    ADD KEY `cliente_id`(`cliente_id`);  
  
ALTER TABLE `TB_PRODUTOS`  
    ADD PRIMARY KEY (`id`),  
    ADD KEY `linhas_produto_id`(`linhas_produto_id`);  
  
ALTER TABLE `TB_CLIENTES`  
    MODIFY `id` int NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=2;  
  
ALTER TABLE `TB_DETALHES_DO_PEDIDO`  
    MODIFY `id` int NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=2;  
  
ALTER TABLE `TB_EMPREGADOS`  
    MODIFY `id` int NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=2;  
  
ALTER TABLE `TB_ESCRITORIOS`  
    MODIFY `id` int NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=2;  
  
ALTER TABLE `TB_LINHAS_PRODUTO`  
    MODIFY `id` int NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=3;  
  
ALTER TABLE `TB_PAGAMENTOS`  
    MODIFY `id` int NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=2;  
  
ALTER TABLE `TB_PEDIDOS`  
    MODIFY `id` int NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=2;  
  
ALTER TABLE `TB_PRODUTOS`  
    MODIFY `id` int NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=2;  
  
ALTER TABLE `TB_CLIENTES`
```

```
ADD CONSTRAINT `TB_CLIENTES_ibfk_1` FOREIGN KEY (`empregado_id`)
REFERENCES `TB_EMPREGADOS` (`id`);
```

```
ALTER TABLE `TB_DETALHES_DO_PEDIDO`
ADD CONSTRAINT `TB_DETALHES_DO_PEDIDO_ibfk_1` FOREIGN KEY (`id`)
REFERENCES `TB_PEDIDOS` (`id`),
ADD CONSTRAINT `TB_DETALHES_DO_PEDIDO_ibfk_2` FOREIGN KEY (`produto_id`)
REFERENCES `TB_PRODUTOS` (`id`);
```

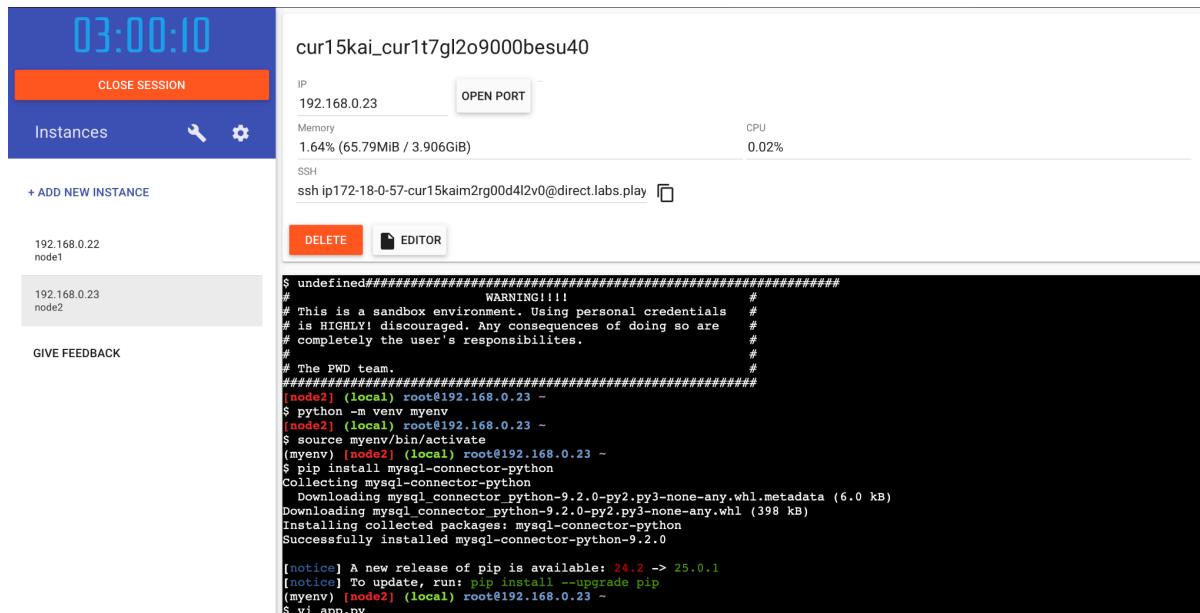
```
ALTER TABLE `TB_EMPREGADOS`
ADD CONSTRAINT `TB_EMPREGADOS_ibfk_1` FOREIGN KEY (`relatorio`)
REFERENCES `TB_EMPREGADOS` (`id`),
ADD CONSTRAINT `TB_EMPREGADOS_ibfk_2` FOREIGN KEY (`escritorio_id`)
REFERENCES `TB_ESCRITORIOS` (`id`);
```

```
ALTER TABLE `TB_PAGAMENTOS`
ADD CONSTRAINT `TB_PAGAMENTOS_ibfk_1` FOREIGN KEY (`cliente_id`)
REFERENCES `TB_CLIENTES` (`id`);
```

```
ALTER TABLE `TB_PEDIDOS`
ADD CONSTRAINT `TB_PEDIDOS_ibfk_1` FOREIGN KEY (`cliente_id`) REFERENCES
`TB_CLIENTES` (`id`);
```

```
ALTER TABLE `TB_PRODUTOS`
ADD CONSTRAINT `TB_PRODUTOS_ibfk_1` FOREIGN KEY (`linhas_produto_id`)
REFERENCES `TB_LINHAS_PRODUTO` (`id`);
COMMIT;
```

Instância 2 - Criando o ambiente virtual e instalando o conector python:



The screenshot shows a session titled "cur15kai_cur1t7gl2o9000besu40" with the following details:

- IP: 192.168.0.23
- Memory: 1.64% (65.79MiB / 3.906GiB)
- CPU: 0.02%
- SSH: ssh ip172-18-0-57-cur15kaim2rg00d4l2v0@direct.labs.play
- Actions: CLOSE SESSION, OPEN PORT, DELETE, EDITOR.

The terminal window displays the following Python environment setup and MySQL connector installation logs:

```
$ undefined#####
# WARNING!!!!#
# This is a sandbox environment. Using personal credentials #
# is HIGHLY! discouraged. Any consequences of doing so are #
# completely the user's responsibilites. #
# #
#####
[node2] (local) root@192.168.0.23 ~
$ python -m venv myenv
[node2] (local) root@192.168.0.23 ~
$ source myenv/bin/activate
(myenv) [node2] (local) root@192.168.0.23 ~
$ pip install mysql-connector-python
Collecting mysql-connector-python
  Downloading mysql_connector_python-9.2.0-py2.py3-none-any.whl.metadata (6.0 kB)
  Downloading mysql_connector_python-9.2.0-py2.py3-none-any.whl (398 kB)
Installing collected packages: mysql-connector-python
Successfully installed mysql-connector-python-9.2.0

[notice] A new release of pip is available: 24.2 -> 25.0.1
[notice] To update, run: pip install --upgrade pip
(myenv) [node2] (local) root@192.168.0.23 ~
$ vi app.py
```

Criando e editando o arquivo python (“app.py”):

02:58:02

CLOSE SESSION

IP: 192.168.0.23 OPEN PORT

Instances: node1, node2

+ ADD NEW INSTANCE

Memory: 1.77% (70.62MB / 3.906GB) CPU: 0.08%

SSH: ssh ip172-18-0-57-cur15kaim2rg00d4l2v0@direct.labs.play

DELETE EDITOR

```
1 import mysql.connector
2 from mysql.connector import Error
3
4 def create_connection():
5     """Cria uma conexão com o banco de dados MySQL."""
6     connection = None
7     try:
8         connection = mysql.connector.connect(
9             host= '192.168.0.22',
10            port= 3306,
11            user= 'root',
12            password= 'root',
13            database= 'mydatabase'
14        )
15        print("Conexão com o MySQL bem-sucedida")
16    except Error as e:
17        print(f'Erro ({e}) ocorreu')
18
19    return connection
20
21 #CRUDS
22
23 # CRUD para TB_CLIENTES
24 def create_cliente(connection, id, nome, sobrenome_contato, nome_contato, telefone, enderecol, endereco2, cidade, est
ado, cep, pais, empregado_id, limite_credito):
25     cursor = connection.cursor()
26     query = """
27         INSERT INTO TB_CLIENTES (id, nome, sobrenome_contato, nome_contato, telefone, enderecol, endereco2, cidade, estad
o, cep, pais, empregado_id, limite_credito)
28         VALUES (%s, %s, %s)
29         """
30     cursor.execute(query, (id, nome, sobrenome_contato, nome_contato, telefone, enderecol, endereco2, cidade, estado,
cep, pais, empregado_id, limite_credito))
31     connection.commit()
32     print("Cliente adicionado com sucesso")
33
34 def read_clientes(connection):
35     cursor = connection.cursor()
36     cursor.execute("SELECT * FROM TB_CLIENTES")
37     clientes = cursor.fetchall()
38     for cliente in clientes:
39         print(cliente)
40
41 def update_cliente(connection, cliente_id, nome, sobrenome_contato, nome_contato, telefone, enderecol, endereco2, cid
ade, estado, cep, pais, empregado_id, limite_credito):
42     cursor = connection.cursor()
43     query = """
44         UPDATE TB_CLIENTES
45         SET nome = %s, sobrenome_contato = %s, nome_contato = %s, telefone = %s, enderecol = %s, endereco2 = %s, cidade =
%s, estado = %s, cep = %s, pais = %s, empregado_id = %s, limite_credito = %s
46         WHERE id = %s
47         """
48     cursor.execute(query, (nome, sobrenome_contato, nome_contato, telefone, enderecol, endereco2, cidade, estado, cep
, pais, empregado_id, limite_credito, cliente_id))
```

5,1 Top

02:57:26

CLOSE SESSION

IP: 192.168.0.23 OPEN PORT

Instances: node1, node2

+ ADD NEW INSTANCE

Memory: 1.77% (70.79MB / 3.906GB) CPU: 0.01%

SSH: ssh ip172-18-0-57-cur15kaim2rg00d4l2v0@direct.labs.play

DELETE EDITOR

```
27         INSERT INTO TB_CLIENTES (id, nome, sobrenome_contato, nome_contato, telefone, enderecol, endereco2, cidade, estad
o, cep, pais, empregado_id, limite_credito)
28         VALUES (%s, %s, %s)
29         """
30     cursor.execute(query, (id, nome, sobrenome_contato, nome_contato, telefone, enderecol, endereco2, cidade, estado,
cep, pais, empregado_id, limite_credito))
31     connection.commit()
32     print("Cliente adicionado com sucesso")
33
34 def read_clientes(connection):
35     cursor = connection.cursor()
36     cursor.execute("SELECT * FROM TB_CLIENTES")
37     clientes = cursor.fetchall()
38     for cliente in clientes:
39         print(cliente)
40
41 def update_cliente(connection, cliente_id, nome, sobrenome_contato, nome_contato, telefone, enderecol, endereco2, cid
ade, estado, cep, pais, empregado_id, limite_credito):
42     cursor = connection.cursor()
43     query = """
44         UPDATE TB_CLIENTES
45         SET nome = %s, sobrenome_contato = %s, nome_contato = %s, telefone = %s, enderecol = %s, endereco2 = %s, cidade =
%s, estado = %s, cep = %s, pais = %s, empregado_id = %s, limite_credito = %s
46         WHERE id = %s
47         """
48     cursor.execute(query, (nome, sobrenome_contato, nome_contato, telefone, enderecol, endereco2, cidade, estado, cep
, pais, empregado_id, limite_credito, cliente_id))
```

27,1 6%

02:56:17

CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.22 node1

192.168.0.23 node2

GIVE FEEDBACK

cur15kai_cur1t7gl2o9000besu40

IP: 192.168.0.23 OPEN PORT

Memory: 1.77% (70.95MiB / 3.906GiB) CPU: 0.17%

SSH: ssh ip172-18-0-57-cur15kaim2rg00d4l2v0@direct.labs.play

```

51
52 def delete_cliente(connection, cliente_id):
53     cursor = connection.cursor()
54     query = "DELETE FROM TB_CLIENTES WHERE id = %s"
55     cursor.execute(query, (cliente_id,))
56     connection.commit()
57     print("Cliente deletado com sucesso")
58
59 # CRUD para TB DETALHES DO PEDIDO
60 def create_detalhe_do_pedido(connection, id, produto_id, quantidade, preco_cada, linha_pedido):
61     cursor = connection.cursor()
62     query = """
63         INSERT INTO TB_DETALHES_DO_PEDIDO (id, produto_id, quantidade, preco_cada, linha_pedido)
64         VALUES (%s, %s, %s, %s, %s)
65     """
66     cursor.execute(query, (id, produto_id, quantidade, preco_cada, linha_pedido))
67     connection.commit()
68     print("Detalhe do pedido adicionado com sucesso")
69
70 def read_detalhes_do_pedido(connection):
71     cursor = connection.cursor()
72     cursor.execute("SELECT * FROM TB_DETALHES_DO_PEDIDO")
73     detalhes = cursor.fetchall()
74     for detalhe in detalhes:
75         print(detalhe)
76
77 def update_detalhe_do_pedido(connection, id, produto_id, quantidade, preco_cada, linha_pedido):
    
```

76,0-1 13%

02:55:18

CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.22 node1

192.168.0.23 node2

GIVE FEEDBACK

cur15kai_cur1t7gl2o9000besu40

IP: 192.168.0.23 OPEN PORT

Memory: 1.78% (71.04MiB / 3.906GiB) CPU: 0.23%

SSH: ssh ip172-18-0-57-cur15kaim2rg00d4l2v0@direct.labs.play

```

78     cursor = connection.cursor()
79     query = """
80         UPDATE TB_DETALHES_DO_PEDIDO
81         SET produto_id = %s, quantidade = %s, preco_cada = %s, linha_pedido = %s
82         WHERE id = %s
83     """
84     cursor.execute(query, (produto_id, quantidade, preco_cada, linha_pedido, id))
85     connection.commit()
86     print("Detalhe do pedido atualizado com sucesso")
87
88 # CRUD para TB DETALHES_DO_PEDIDO (continuação)
89 def delete_detalhe_do_pedido(connection, id):
90     cursor = connection.cursor()
91     query = "DELETE FROM TB_DETALHES_DO_PEDIDO WHERE id = %s"
92     cursor.execute(query, (id,))
93     connection.commit()
94     print("Detalhe do pedido deletado com sucesso")
95
96 # CRUD para TB EMPREGADOS
97 def create_empregado(connection, id, sobrenome, nome, extensao, email, escritorio_id, relatorio, cargo):
98     cursor = connection.cursor()
99     query = """
100         INSERT INTO TB_EMPREGADOS (id, sobrenome, nome, extensao, email, escritorio_id, relatorio, cargo)
101         VALUES (%s, %s, %s, %s, %s, %s, %s)
102     """
103     cursor.execute(query, (id, sobrenome, nome, extensao, email, escritorio_id, relatorio, cargo))
104
    connection.commit()

```

104,1 20%

02:54:58

CLOSE SESSION

IP: 192.168.0.23 OPEN PORT

Instances

Memory: 1.78% (71.11MiB / 3.906GiB) CPU: 0.14%

SSH: ssh ip172-18-0-57-cur15kaim2rg00d4l2v0@direct.labs.play

+ ADD NEW INSTANCE

192.168.0.22 node1

192.168.0.23 node2

GIVE FEEDBACK

```

105     print("Empregado adicionado com sucesso")
106
107 def read_empregados(connection):
108     cursor = connection.cursor()
109     cursor.execute("SELECT * FROM TB_EMPREGADOS")
110     empregados = cursor.fetchall()
111     for empregado in empregados:
112         print(empregado)
113
114 def update_empregado(connection, id, sobrenome, nome, extensao, email, escritorio_id, relatorio, cargo):
115     cursor = connection.cursor()
116     query = """
117         UPDATE TB_EMPREGADOS
118             SET sobrenome = %s, nome = %s, extensao = %s, email = %s, escritorio_id = %s, relatorio = %s, cargo =
119             %s
120             WHERE id = %s
121
122     cursor.execute(query, (sobrenome, nome, extensao, email, escritorio_id, relatorio, cargo, id))
123     connection.commit()
124     print("Empregado atualizado com sucesso")
125
126 def delete_empregado(connection, id):
127     cursor = connection.cursor()
128     query = "DELETE FROM TB_EMPREGADOS WHERE id = %s"
129     cursor.execute(query, (id,))
130     connection.commit()
131     print("Empregado deletado com sucesso")

```

130,1 28%

02:54:31

CLOSE SESSION

IP: 192.168.0.23 OPEN PORT

Instances

Memory: 1.78% (71.15MiB / 3.906GiB) CPU: 0.17%

SSH: ssh ip172-18-0-57-cur15kaim2rg00d4l2v0@direct.labs.play

+ ADD NEW INSTANCE

192.168.0.22 node1

192.168.0.23 node2

GIVE FEEDBACK

```

131 # CRUD para TB_ESCRITORIOS
132 def create_escritorio(connection, id, cidade, telefone, enderecol, endereco2, estado, pais, cep, territorio):
133     cursor = connection.cursor()
134     query = """
135         INSERT INTO TB_ESCRITORIOS (id, cidade, telefone, enderecol, endereco2, estado, pais, cep, territorio
136         )
137             VALUES (%s, %s, %s, %s, %s, %s, %s, %s, %s)
138
139     cursor.execute(query, (id, cidade, telefone, enderecol, endereco2, estado, pais, cep, territorio))
140     connection.commit()
141     print("Escritório adicionado com sucesso")
142
143 def read_escritorios(connection):
144     cursor = connection.cursor()
145     cursor.execute("SELECT * FROM TB_ESCRITORIOS")
146     escritorios = cursor.fetchall()
147     for escritorio in escritorios:
148         print(escritorio)
149
150 def update_escritorio(connection, id, cidade, telefone, enderecol, endereco2, estado, pais, cep, territorio):
151     cursor = connection.cursor()
152     query = """
153         UPDATE TB_ESCRITORIOS
154             SET cidade = %s, telefone = %s, enderecol = %s, endereco2 = %s, estado = %s, pais = %s, cep = %s, ter
155             ritorio = %s
156             WHERE id = %s

```

155,1 35%

02:54:10

CLOSE SESSION

IP: 192.168.0.23 OPEN PORT

Instances: node1, node2

+ ADD NEW INSTANCE

Memory: 1.78% (71.16MiB / 3.906GiB) CPU: 0.08%

SSH: ssh ip172-18-0-57-cur15kaim2rg00d4l2v0@direct.labs.play

GIVE FEEDBACK

```

156     """
157     cursor.execute(query, (cidade, telefone, endereco1, endereco2, estado, pais, cep, territorio, id))
158     connection.commit()
159     print("Escritório atualizado com sucesso")
160
161 def delete_escritorio(connection, id):
162     cursor = connection.cursor()
163     query = "DELETE FROM TB_ESCRITORIOS WHERE id = %s"
164     cursor.execute(query, (id,))
165     connection.commit()
166     print("Escritório deletado com sucesso")
167
168 # CRUD para TB LINHAS_PRODUTO
169 def create_linha_produto(connection, id, descricao_texto, descricao_html, imagem):
170     cursor = connection.cursor()
171     query = """
172         INSERT INTO TB_LINHAS_PRODUTO (id, descricao_texto, descricao_html, imagem)
173         VALUES (%s, %s, %s, %s)
174     """
175     cursor.execute(query, (id, descricao_texto, descricao_html, imagem))
176     connection.commit()
177     print("Linha de produto adicionada com sucesso")
178
179 def read_linhas_produto(connection):
180     cursor = connection.cursor()
181     cursor.execute("SELECT * FROM TB_LINHAS_PRODUTO")
182     linhas = cursor.fetchall()

```

182,1 42%

02:53:49

CLOSE SESSION

IP: 192.168.0.23 OPEN PORT

Instances: node1, node2

+ ADD NEW INSTANCE

Memory: 1.78% (71.26MiB / 3.906GiB) CPU: 0.18%

SSH: ssh ip172-18-0-57-cur15kaim2rg00d4l2v0@direct.labs.play

GIVE FEEDBACK

```

183     for linha in linhas:
184         print(linha)
185
186 def update_linha_produto(connection, id, descricao_texto, descricao_html, imagem):
187     cursor = connection.cursor()
188     query = """
189         UPDATE TB_LINHAS_PRODUTO
190         SET descricao_texto = %s, descricao_html = %s, imagem = %s
191         WHERE id = %s
192     """
193     cursor.execute(query, (descricao_texto, descricao_html, imagem, id))
194     connection.commit()
195     print("Linha de produto atualizada com sucesso")
196
197 def delete_linha_produto(connection, id):
198     cursor = connection.cursor()
199     query = "DELETE FROM TB_LINHAS_PRODUTO WHERE id = %s"
200     cursor.execute(query, (id,))
201     connection.commit()
202     print("Linha de produto deletada com sucesso")
203
204 # CRUD para TB_PAGAMENTOS
205 def create_pagamento(connection, id, cliente_id, data_pagamento, valor):
206     cursor = connection.cursor()
207     query = """
208         INSERT INTO TB_PAGAMENTOS (id, cliente_id, data_pagamento, valor)
209         VALUES (%s, %s, %s, %s)

```

209,1 49%

02:53:25

CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.22 node1	
192.168.0.23 node2	

GIVE FEEDBACK

cur15kai_cur1t7gl2o9000besu40

IP: 192.168.0.23 | OPEN PORT | Memory: 1.77% (70.86MiB / 3.906GiB) | CPU: 0.12%

SSH: ssh ip172-18-0-57-cur15kaim2rg00d4l2v0@direct.labs.play

```

210     """
211     cursor.execute(query, (id, cliente_id, data_pagamento, valor))
212     connection.commit()
213     print("Pagamento adicionado com sucesso")
214
215 def read_pagamentos(connection):
216     cursor = connection.cursor()
217     cursor.execute("SELECT * FROM TB_PAGAMENTOS")
218     pagamentos = cursor.fetchall()
219     for pagamento in pagamentos:
220         print(pagamento)
221
222 def update_pagamento(connection, id, cliente_id, data_pagamento, valor):
223     cursor = connection.cursor()
224     query = """
225         UPDATE TB_PAGAMENTOS
226             SET cliente_id = %s, data_pagamento = %s, valor = %s
227         WHERE id = %s
228     """
229     cursor.execute(query, (cliente_id, data_pagamento, valor, id))
230     connection.commit()
231     print("Pagamento atualizado com sucesso")
232
233 def delete_pagamento(connection, id):
234     cursor = connection.cursor()
235     query = "DELETE FROM TB_PAGAMENTOS WHERE id = %s"
236     cursor.execute(query, (id,))

```

236,1 56%

02:52:46

CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.22 node1	
192.168.0.23 node2	

GIVE FEEDBACK

cur15kai_cur1t7gl2o9000besu40

IP: 192.168.0.23 | OPEN PORT | Memory: 1.77% (70.89MiB / 3.906GiB) | CPU: 0.31%

SSH: ssh ip172-18-0-57-cur15kaim2rg00d4l2v0@direct.labs.play

```

237     connection.commit()
238     print("Pagamento deletado com sucesso")
239
240 # CRUD para TB PEDIDOS
241 def create_pedido(connection, id, data_pedido, data_requerida, data_envio, status, comentarios, cliente_id):
242     cursor = connection.cursor()
243     query = """
244         INSERT INTO TB_PEDIDOS (id, data_pedido, data_requerida, data_envio, status, comentarios, cliente_id)
245         VALUES (%s, %s, %s, %s, %s, %s, %s)
246     """
247     cursor.execute(query, (id, data_pedido, data_requerida, data_envio, status, comentarios, cliente_id))
248     connection.commit()
249     print("Pedido adicionado com sucesso")
250
251 def read_pedidos(connection):
252     cursor = connection.cursor()
253     cursor.execute("SELECT * FROM TB_PEDIDOS")
254     pedidos = cursor.fetchall()
255     for pedido in pedidos:
256         print(pedido)
257
258 def update_pedido(connection, id, data_pedido, data_requerida, data_envio, status, comentarios, cliente_id):
259     cursor = connection.cursor()
260     query = """
261         UPDATE TB_PEDIDOS
262             SET data_pedido = %s, data_requerida = %s, data_envio = %s, status = %s, comentarios = %s, cliente_id = %
263     """

```

243,1 63%

02:52:21

cur15kai_cur1t7gl2o9000besu40

CLOSE SESSION

IP: 192.168.0.23 OPEN PORT

Instances

Memory: 1.77% (70.91MiB / 3.906GiB) CPU: 0.02%

+ ADD NEW INSTANCE

192.168.0.22 node1

192.168.0.23 node2

GIVE FEEDBACK

DELETE **EDITOR**

```

263     WHERE id = %s
264     """
265     cursor.execute(query, (data_pedido, data_requerida, data_envio, status, comentarios, cliente_id, id))
266     connection.commit()
267     print("Pedido atualizado com sucesso")
268
269 def delete_pedido(connection, id):
270     cursor = connection.cursor()
271     query = "DELETE FROM TB_PEDIDOS WHERE id = %s"
272     cursor.execute(query, (id,))
273     connection.commit()
274     print("Pedido deletado com sucesso")
275
276 # CRUD para TB_PRODUTOS
277 def create_produto(connection, id,nome_produto, linhas_produto_id, tamanho_produto, fornecedor_produto, descricao_produto, quantidade_estoque, preco_compra, MSRP):
278     cursor = connection.cursor()
279     query = """
280         INSERT INTO TB_PRODUTOS (id, nome_produto, linhas_produto_id, tamanho_produto, fornecedor_produto, descricao_produto, quantidade_estoque, preco_compra, MSRP)
281         VALUES (%s, %s, %s, %s, %s, %s, %s, %s)
282         """
283     cursor.execute(query, (id, nome_produto, linhas_produto_id, tamanho_produto, fornecedor_produto, descricao_produto, quantidade_estoque, preco_compra, MSRP))
284     connection.commit()
285     print("Produto adicionado com sucesso")
286

```

263,1 70%

02:51:50

cur15kai_cur1t7gl2o9000besu40

CLOSE SESSION

IP: 192.168.0.23 OPEN PORT

Instances

Memory: 1.77% (71MiB / 3.906GiB) CPU: 0.06%

+ ADD NEW INSTANCE

192.168.0.22 node1

192.168.0.23 node2

GIVE FEEDBACK

DELETE **EDITOR**

```

286
287 def read_produtos(connection):
288     cursor = connection.cursor()
289     cursor.execute("SELECT * FROM TB_PRODUTOS")
290     produtos = cursor.fetchall()
291     for produto in produtos:
292         print(produto)
293
294 def update_produto(connection, id, nome_produto, linhas_produto_id, tamanho_produto, fornecedor_produto, descricao_produto, quantidade_estoque, preco_compra, MSRP):
295     cursor = connection.cursor()
296     query = """
297             UPDATE TB_PRODUTOS
298             SET nome_produto = %s, linhas_produto_id = %s, tamanho_produto = %s, fornecedor_produto = %s,
299             quantidade_estoque = %s, preco_compra = %s, MSRP = %s
300             WHERE id = %s
301             """
302     cursor.execute(query, (nome_produto, linhas_produto_id, tamanho_produto, fornecedor_produto, descricao_produto, quantidade_estoque, preco_compra, MSRP, id))
303     connection.commit()
304     print("Produto atualizado com sucesso")
305
306 def delete_produto(connection, id):
307     cursor = connection.cursor()
308     query = "DELETE FROM TB_PRODUTOS WHERE id = %s"
309     cursor.execute(query, (id,))
310     connection.commit()

```

286,0-1 76%

02:51:16

CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.22
node1

192.168.0.23
node2

GIVE FEEDBACK

cur15kai_cur1t7gl2o9000besu40

IP: 192.168.0.23 OPEN PORT

Memory: 1.77% (70.75MiB / 3.906GiB) CPU: 0.17%

SSH: ssh ip172-18-0-57-cur15kaim2rg00d4l2v0@direct.labs.play

DELETE **EDITOR**

```

310     print("Produto deletado com sucesso")
311
312 # Função main para demonstrar o uso das funções CRUD
313 def main():
314     connection = create_connection()
315     if connection is None:
316         return
317
318     #CREATE DAS TABELAS
319     create_escritorio(connection, 1, 'São Paulo', 123456789, 'Rua B', 'Apto 2', 'SP', 'Brasil', 12345678, 'SP')
320     create_linha_produto(connection, 1, 'Linha 1', '<p>Descrição HTML</p>', None)
321     create_empregado(connection, 1, 'Silva', 'Carlos', '123', 'carlos@example.com', 1, None, 'Gerente')
322     create_produto(connection, 1, 'Produto 1', 1, 'M', 'Fornecedor 1', 'Descrição do Produto 1', 100, 50.00, 75.00)
323     create_cliente(connection, 1, 'João', 'Silva', 'Maria', 123456789, 'Rua A', 'Apto 1', 'São Paulo', 'SP', 12345678,
324     'Brasil', 1, 5000.00)
324     create_pagamento(connection, 1, 1, '2023-10-01', 100.00)
325     create_pedido(connection, 1, '2023-10-01', '2023-10-05', '2023-10-10', 'Enviado', 'Nenhum', 1)
326     create_detalhe_pedido(connection, 1, 1, 10, 15.50, 1)
327
328
329
330     # Exemplo de uso das funções CRUD para TB_CLIENTES
331     read_clientes(connection)
332     update_cliente(connection, 1, 'João', 'Silva', 'Maria', 987654321, 'Rua A', 'Apto 1', 'São Paulo', 'SP', 87654321
333     , 'Brasil', 1, 6000.00)
333     read_clientes(connection)
334     # read_clientes(connection)

```

334 | 334,1 83%

02:50:51

CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.22
node1

192.168.0.23
node2

GIVE FEEDBACK

cur15kai_cur1t7gl2o9000besu40

IP: 192.168.0.23 OPEN PORT

Memory: 1.77% (70.79MiB / 3.906GiB) CPU: 0.74%

SSH: ssh ip172-18-0-57-cur15kaim2rg00d4l2v0@direct.labs.play

DELETE **EDITOR**

```

337     read_detalhes_pedido(connection)
338     update_detalhe_pedido(connection, 1, 1, 5, 20.00, 1)
339     read_detalhes_pedido(connection)
340     # read_detalhes_pedido(connection)
341
342     # Exemplo de uso das funções CRUD para TB_EMPREGADOS
343     read_empregados(connection)
344     update_empregado(connection, 1, 'Silva', 'Carlos', '456', 'carlos.novo@example.com', 1, None, 'Diretor')
345     read_empregados(connection)
346     # read_empregados(connection)
347
348     # Exemplo de uso das funções CRUD para TB_ESCRITORIOS
349     read_escritorios(connection)
350     update_escritorio(connection, 1, 'São Paulo', 987654321, 'Rua B', 'Apto 3', 'SP', 'Brasil', 87654321, 'SP')
351     read_escritorios(connection)
352     # read_escritorios(connection)
353
354     # Exemplo de uso das funções CRUD para TB_LINHAS_PRODUTO
355     read_linhas_produto(connection)
356     update_linha_produto(connection, 1, 'Linha 1 Atualizada', '<p>Nova Descrição HTML</p>', None)
357     read_linhas_produto(connection)
358     # read_linhas_produto(connection)
359
360     # Exemplo de uso das funções CRUD para TB_PAGAMENTOS
361     read_pagamentos(connection)
362     update_pagamento(connection, 1, 1, '2023-10-02', 150.00)
363     read_pagamentos(connection)

```

363 | 363,1 91%

02:50:03

CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.22 node1

192.168.0.23 node2

GIVE FEEDBACK

cur15kai_cur1t7gl2o9000besu40

IP: 192.168.0.23 OPEN PORT

Memory: 1.77% (70.89MiB / 3.906GiB) CPU: 0.11%

SSH: ssh ip172-18-0-57-cur15kaim2rg00d4l2v0@direct.labs.play

```

366 # Exemplo de uso das funções CRUD para TB_PEDIDOS
367 read_pedidos(connection)
368 update_pedido(connection, 1, '2023-10-01', '2023-10-06', '2023-10-11', 'Entregue', 'Nenhum', 1)
369 read_pedidos(connection)
370 # read_pedidos(connection)
371
372 # Exemplo de uso das funções CRUD para TB_PRODUTOS
373 read_produtos(connection)
374 update_produto(connection, 1, 'Produto 1 Atualizado', 1, 'L', 'Fornecedor 1', 'Descrição do Produto 1 Atualizada'
, 80, 55.00, 80.00)
375 read_produtos(connection)
376 # read_produtos(connection)
377
378 #DELETE DAS TABELAS
379 delete_detalhe_do_pedido(connection, 1)
380 delete_pedido(connection, 1)
381 delete_pagamento(connection, 1)
382 delete_cliente(connection, 1)
383 delete_produto(connection, 1)
384 delete_empregado(connection, 1)
385 delete_linha_produto(connection, 1)
386 delete_escritorio(connection, 1)
387
388
389
390
391 connection.close()

```

391,1 98%

02:49:32

CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.22 node1

192.168.0.23 node2

GIVE FEEDBACK

cur15kai_cur1t7gl2o9000besu40

IP: 192.168.0.23 OPEN PORT

Memory: 1.77% (71MiB / 3.906GiB) CPU: 0.07%

SSH: ssh ip172-18-0-57-cur15kaim2rg00d4l2v0@direct.labs.play

```

370 # read_pedidos(connection)
371
372 # Exemplo de uso das funções CRUD para TB_PRODUTOS
373 read_produtos(connection)
374 update_produto(connection, 1, 'Produto 1 Atualizado', 1, 'L', 'Fornecedor 1', 'Descrição do Produto 1 Atualizada'
, 80, 55.00, 80.00)
375 read_produtos(connection)
376 # read_produtos(connection)
377
378 #DELETE DAS TABELAS
379 delete_detalhe_do_pedido(connection, 1)
380 delete_pedido(connection, 1)
381 delete_pagamento(connection, 1)
382 delete_cliente(connection, 1)
383 delete_produto(connection, 1)
384 delete_empregado(connection, 1)
385 delete_linha_produto(connection, 1)
386 delete_escritorio(connection, 1)
387
388
389
390
391 connection.close()
392 if __name__ == "__main__":
393     main()
394

```

395,0-1 Bot

Executando o arquivo python:

02:59:26

CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.22 node1

192.168.0.23 node2

GIVE FEEDBACK

cur15kai_cur1t7gl2o9000besu40

IP: 192.168.0.23 OPEN PORT

Memory: 1.65% (65.88MiB / 3.906GiB) CPU: 0.12%

SSH: ssh ip172-18-0-57-cur15kaim2rg00d4l2v0@direct.labs.play

```
(myenv) [node2] (local) root@192.168.0.23 ~
$ python app.py
Conexão com o MySQL bem-sucedida
Escritório adicionado com sucesso
Linha de produto adicionada com sucesso
Empregado adicionado com sucesso
Produto adicionado com sucesso
Cliente adicionado com sucesso
Pagamento adicionado com sucesso
Pedido adicionado com sucesso
Detalhe do pedido adicionado com sucesso
(1, 'João', 'Silva', 'Maria', 123456789, 'Rua A', 'Apto 1', 'São Paulo', 'SP', 12345678, 'Brasil', 1, 5000.0)
Cliente atualizado com sucesso
(1, 'João', 'Silva', 'Maria', 987654321, 'Rua A', 'Apto 1', 'São Paulo', 'SP', 87654321, 'Brasil', 1, 6000.0)
(1, 1, 10, 15.5, 1)
Detalhe do pedido atualizado com sucesso
(1, 1, 5, 20.0, 1)
(1, 'Silva', 'Carlos', '123', 'carlos@example.com', 1, None, 'Gerente')
Empregado atualizado com sucesso
(1, 'Silva', 'Carlos', '456', 'carlos.novo@example.com', 1, None, 'Diretor')
(1, 'São Paulo', 123456789, 'Rua B', 'Apto 2', 'SP', 'Brasil', 12345678, 'SP')
Escritório atualizado com sucesso
(1, 'São Paulo', 987654321, 'Rua B', 'Apto 3', 'SP', 'Brasil', 87654321, 'SP')
(1, 'Linha 1', '<p>Descrição HTML</p>', None)
Linha de produto atualizada com sucesso
(1, 'Linha 1 Atualizada', '<p>Nova Descrição HTML</p>', None)
(1, 1, datetime.date(2023, 10, 1), 100.0)
Pagamento atualizado com sucesso
```

02:58:51

CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.22 node1

192.168.0.23 node2

GIVE FEEDBACK

cur15kai_cur1t7gl2o9000besu40

IP: 192.168.0.23 OPEN PORT

Memory: 1.65% (65.96MiB / 3.906GiB) CPU: 0.12%

SSH: ssh ip172-18-0-57-cur15kaim2rg00d4l2v0@direct.labs.play

```
[1, 'Silva', 'Carlos', '123', 'carlos@example.com', 1, None, 'Gerente')
Empregado atualizado com sucesso
(1, 'Silva', 'Carlos', '456', 'carlos.novo@example.com', 1, None, 'Diretor')
(1, 'São Paulo', 123456789, 'Rua B', 'Apto 2', 'SP', 'Brasil', 12345678, 'SP')
Escritório atualizado com sucesso
(1, 'São Paulo', 987654321, 'Rua B', 'Apto 3', 'SP', 'Brasil', 87654321, 'SP')
(1, 'Linha 1', '<p>Descrição HTML</p>', None)
Linha de produto atualizada com sucesso
(1, 'Linha 1 Atualizada', '<p>Nova Descrição HTML</p>', None)
(1, 1, datetime.date(2023, 10, 1), 100.0)
Pagamento atualizado com sucesso
(1, 1, datetime.date(2023, 10, 2), 150.0)
(1, datetime.date(2023, 10, 1), datetime.date(2023, 10, 5), datetime.date(2023, 10, 10), 'Enviado', 'Nenhum', 1)
Pedido atualizado com sucesso
(1, datetime.date(2023, 10, 1), datetime.date(2023, 10, 6), datetime.date(2023, 10, 11), 'Entregue', 'Nenhum', 1)
(1, 'Produto 1', 'M', 'Fornecedor 1', 'Descrição do Produto 1', 100, 50.0, 75.0)
Produto atualizado com sucesso
(1, 'Produto 1 Atualizado', 1, 'L', 'Fornecedor 1', 'Descrição do Produto 1 Atualizada', 80, 55.0, 80.0)
Detalhe do pedido deletado com sucesso
Pedido deletado com sucesso
Pagamento deletado com sucesso
Cliente deletado com sucesso
Produto deletado com sucesso
Empregado deletado com sucesso
Linha de produto deletada com sucesso
Escritório deletado com sucesso
(myenv) [node2] (local) root@192.168.0.23 ~
$
```