

Exercícios - Semana 10



O objetivo da atividade desta semana consiste em conhecer e alterar o projeto “Integrador”. Abra com o seu editor favorito, execute o arquivo `integracao.py`, observando as saídas, execute os testes unitários.

Desafio 1

Explique, em linhas gerais, como funciona o projeto Integrador.

Resposta: Projeto integrador está composto por 3 arquivos `.py` sendo eles:

`Integracao.py` = Rotina responsável por carregar os arquivos de `credenciais.txt` em uma lista e faz os devidos tratamentos para poder separar os dados entre chave e valor. Após o tratamento o mesmo chama o arquivo de `Contexto.py` passando as credenciais por parâmetro através do comando (`context = Contexto(**credenciais)`)

`Contexto.py` = No contexto a rotina contém um dicionário com 2 tipos de estratégias sendo elas:

'csv': `Estrategia_CSV()`

'sqlite': `Estrategia_SQLite()`

No contexto ele armazena as credenciais e os dados de estratégia, faz algumas verificações para saber se as credenciais

informadas são suficientes para extrair os dados necessários.

`Estrategias.py` = Nesse arquivo contém os algoritmos de seleção dos dados de acordo com a estratégia escolhida, sendo um `select` no banco de dados do `sqlite` ou então uma extração de informações do arquivo `CSV`.

Ao final o arquivo de `integracao.py` printa as informações com os dados selecionados.

Desafio 2

Retirar o comentário da linha 20 do arquivo `integracao.py`. Execute novamente o arquivo `integracao.py`. Ocorreu algum erro? Explique porque.

```
19 arquivos += ['credenciais1.txt', 'credenciais2.txt']
20 arquivos += ['credenciais3.txt']
21 arquivos += ['credenciais4.txt', 'credenciais5.txt']
22 arquivos += ['credenciais6.txt']
```

Resposta: SIM, ocorre o seguinte erro com o arquivo `credenciais3.txt`:

File "c:\NappAcademy\semana10\integrador\classes\Contexto.py", line 27, in `__init__`

raise Exception('Credenciais inválidas. Necessário: ' + esperado)

Exception: Credenciais inválidas. Necessário: ('algoritmo', 'db')

Erro ocorre no arquivo de contexto, no método (`def _dados_credenciais_validos(self):`)

onde o mesmo checka os parametros necessarios que são ('algoritmo', 'db'), porem

no arquivo `credenciais3.txt` possui somente os parâmetros ('algoritmo','arquivo')

Desafio 3

Inserir novamente o comentário da linha 20 e remover o comentário da linha 22 do arquivo `integracao.py`. Execute novamente o arquivo `integracao.py`. Ocorreu algum erro? Explique porque.

Resposta: SIM, ocorre o seguinte erro com o arquivo `credenciais6.txt`:

File "c:\NappAcademy\semana10\integrador\classes\Contexto.py", line 27, in `__init__`

raise Exception('Credenciais inválidas. Necessário: ' + esperado)

Exception: Credenciais inválidas. Necessário: ('algoritmo', 'arquivo')

Erro ocorre no arquivo de contexto, no método (`def _dados_credenciais_validos(self):`)

onde o mesmo checka os parametros necessarios que são ('algoritmo', 'arquivo'), porem

no arquivo `credenciais6.txt` possui os parâmetros ('algoritmo','db', 'user', 'password')

o algoritmo está definido como CSV porem a estrutura da credencial está definida como se fosse do algoritmo sqlite, por isso ocorre o erro.

Desafio 4

Inserir novamente o comentário da linha 22 do arquivo integracao.py.

Editar o arquivo credenciais2.txt. Altere o nome do banco de dados, conforme modelo a seguir:

```
algoritmo==>SQLite
db==>integrador/dados/banco3.db
user==>napp
password==>123mudar
```

Execute novamente o arquivo integracao.py. Ocorreu algum erro ? Explique.

Resposta:SIM, ocorreu erro no comando de seleção de informação:

```
(cursor.execute("SELECT * FROM vendas;"))
```

erro ocorreu pois como foi modificado o arquivo credenciais2.txt apontando para o banco banco3.db e o mesmo não possui a tabela de vendas, erro ocorre ao tentar consultar as informações dessa tabela.

Desafio 5

A função carregar_credenciais tem como objetivo ler um arquivo txt com as credenciais. Entretanto, a função “quebra”, caso o arquivo contenha comentários (# no começo da linha) ou linhas em branco, como no exemplo a seguir.

Alterar a função carregar_credenciais, para que esta quebra não ocorra mais.

```
# Algoritmo utilizado
algoritmo==>SQLite

# Localização do banco
db==>integrador/dados/banco2.db
user==>napp
password==>123mudar
```

Resposta: Adicionei o seguinte tratamento abaixo do comando (for line in file:)

```
if((line != '\n') and (line[0] != '#')):
```

A primeira condição identifica quebra de linha, e a segunda verificar as linhas que são comentarios.

Caso nao seja uma quebra de linha e não seja um comentario a rotina lê a linha e guarda as informações, caso contrario ignora a linha.

Desafio 6

O método **execute** da classe **Estrategia_SQLite** extrai todos os registros da tabela vendas, no banco de dados informado. Retorna em uma lista de tuplas. Entretanto, as informações necessárias para compor o histórico de vendas são: total e vendido_em. Altere o método para que retorne uma lista de tuplas, contendo apenas dados dos campos *total* e *vendido_em*. Copie e cole seu método, com seus ajustes.

Resposta: Apenas alterei o comando SQL:

Antigo: cursor.execute("SELECT * FROM vendas;")

Alterado: cursor.execute("SELECT total, vendido_em FROM vendas;")

Exemplo de retorno obtido utilizando o arquivo de credenciais2.txt

```
[(348.01, '2021-03-20'), (132.91, '2021-03-20'), (375.01, '2021-03-20'), (589.59, '2021-03-20'),
(106.55, '2021-03-20'),(327.83, '2021-03-20'), (1080.58, '2021-03-20'), (1846.18, '2021-03-20'),
(605.95, '2021-03-20'), (430.34, '2021-03-20'),(220.57, '2021-03-20'), (330.41, '2021-03-20'),
(741.34, '2021-03-20'), (333.87, '2021-03-20'), (192.14, '2021-03-20'), (1031.99, '2021-03-20'),
(1931.7, '2021-03-20'), (411.2, '2021-03-20'), (497.1, '2021-03-20'), (1505.09, '2021-03-20'),
(1276.01, '2021-03-21'), (189.06, '2021-03-21'), (308.86, '2021-03-21'), (798.62, '2021-03-21'),
```

(133.59, '2021-03-21'), (1132.85, '2021-03-21'), (841.88, '2021-03-21'), (86.84, '2021-03-21'), (594.97, '2021-03-21'), (985.05, '2021-03-21'), (158.02, '2021-03-21'), (903.62, '2021-03-21'), (464.39, '2021-03-21'), (656.24, '2021-03-21'), (449.79, '2021-03-21'), (383.36, '2021-03-21'), (174.7, '2021-03-21'), (99.18, '2021-03-21'), (133.2, '2021-03-21'), (52.74, '2021-03-21'), (1286.02, '2021-03-22'), (107.65, '2021-03-22'), (126.12, '2021-03-22'), (19.28, '2021-03-22'), (67.74, '2021-03-22'), (1476.72, '2021-03-22'), (632.07, '2021-03-22'), (180.8, '2021-03-22'), (45.81, '2021-03-22'), (7.59, '2021-03-22'), (34.51, '2021-03-22'), (286.56, '2021-03-22'), (1101.26, '2021-03-22'), (1319.83, '2021-03-22'), (779.38, '2021-03-22'), (966.33, '2021-03-22'), (81.84, '2021-03-22'), (1222.16, '2021-03-22'), (1506.53, '2021-03-22'), (616.3, '2021-03-22'), (180.14, '2021-03-23'), (412.33, '2021-03-23'), (240.7, '2021-03-23'), (205.72, '2021-03-23'), (184.39, '2021-03-23'), (261.59, '2021-03-23'), (1483.63, '2021-03-23'), (997.3, '2021-03-23'), (458.7, '2021-03-23'), (894.04, '2021-03-23'), (218.17, '2021-03-23'), (583.19, '2021-03-23'), (469.06, '2021-03-23'), (1070.56, '2021-03-23'), (319.91, '2021-03-23'), (348.11, '2021-03-23'), (37.87, '2021-03-23'), (242.06, '2021-03-23'), (455.82, '2021-03-23'), (125.62, '2021-03-23)]]

Desafio 7

O método **execute** da classe **Estrategia_CSV** extrai todos os registros da planilha informada. Retorna em uma lista de dicionários. Entretanto, as informações necessárias para compor o histórico de vendas são: *total* e *vendido_em*. Altere o método para que retorne uma lista de dicionários, contendo apenas dados dos campos *total* e *vendido_em*. Copie e cole seu método, com seus ajustes.

Resposta: Adicionei o seguinte tratamento abaixo do comando (for line in reader:)

```
total = line['total']
vendido_em = line['vendido_em']
lista_registros.append({total, vendido_em})
```

Adicionei o valor *total* e *vendido_em* em variáveis separadas, depois inclui na lista em forma de dicionário, retornando apenas os dois campos necessários.