

Exercícios – Sprint 3

Nesta sprint, iremos realizar uma manutenção em um sistema orientado a objeto. O sistema extrai dados de uma base de dados SQLite. A base de dados possui uma tabela de vendas, com os seguintes campos:

- id
- produto
- preco
- quantidade
- criado_em

O sistema orientado a objetos possui três classes: DAO, Registro e Compras. Esta sprint possui atividades de análise de código, criação/alteração de código e questões dissertativas, que deverão ser respondidas via formulário.

Legenda:

- Atividade marcada desta forma envolve criação de código. Somente os testes unitários estão prontos.
- **Atividade marcada desta forma está implementada e testada.**
- **Atividade marcada desta forma devem ser respondidas no formulário.**

Classe DAO

1) Desenvolver o método **get_registros_por_produto**. Este método recebe como parâmetro uma string com o nome do produto e retorna uma lista de tuplas. Cada tupla possui um registro localizado no banco de dados, na tabela vendas.

2) Desenvolver o método **get_registros_por_data**. Este método recebe como parâmetro uma string com a data procurada e retorna uma lista de tuplas. Cada tupla possui um registro localizado no banco de dados, na tabela vendas.

3) Desenvolver o método **get_registros_quantidade**. Este método recebe como parâmetro duas strings com a quantidade mínima e a quantidade máxima. Este método retorna uma lista de tuplas. Cada tupla possui um registro localizado no banco de dados, na tabela vendas. Se a quantidade mínima e máxima não forem passadas, assumir os valores 0 e 20, respectivamente.

4) Desenvolver o método **get_registros_preco**. Este método recebe como parâmetro duas strings com o preço mínimo e o preço máximo. Este método retorna uma lista de tuplas. Cada tupla possui um registro localizado no banco de dados, na tabela vendas. Se o preço mínimo e máximo não forem passados, assumir os valores 30 e 40, respectivamente.

5) Desenvolver o método **get_registros_todos**. Este método não recebe parâmetro. Este método retorna uma lista de tuplas. Cada tupla possui um registro localizado no banco de dados. Retornar todos os registros encontrados no banco de dados, na tabela vendas.

6) Qual a responsabilidade da classe DAO ?

Classe Registro

7) Desenvolver o método construtor da classe Registro. Este método recebe como parâmetro uma tupla com os dados: id, nome do produto, preço, quantidade e criado em. O parâmetro criado_em deve ser uma string no formato 'aaaa-MM-dd' e ser armazenado como objeto do tipo date.

7.1) Faça alterações no método construtor. Execute os testes via pytest. Veja os testes falharem. Depois ajuste o código para que os testes voltem a funcionar.

8) Pesquise na documentação oficial do Python sobre decoradores @property. Observe as implementações de properties presentes no código. Crie properties para os atributos preco e quantidade.

9) Desenvolver o método **comprado_no_mes_dia**. Este método recebe como parâmetros uma string ou inteiro com o dia e uma string ou um inteiro com o mês. Este método retorna verdadeiro (True) caso a compra tenha sido realizada no dia e mês informado como parâmetro. Caso contrário, deve retornar falso (False)

10) Desenvolver o método **preco_entre**. Este método recebe dois parâmetros: um número mínimo e um número máximo. Se o valor do preço for maior ou igual ao valor mínimo e menor ou igual o valor máximo, este método deve retornar verdadeiro (True).. Caso contrário, deve retornar falso (False)

11) Qual a responsabilidade da classe Registro ?

Classe Compras

12) Desenvolver o método **construtor** da classe Compras. Este método recebe como parâmetro o nome do banco de dados. O construtor deve conectar ao banco de dados, extrair todos os dados para uma lista de objetos Registro e desconectar do banco de dados.

13) Desenvolver o método **localizar_compras_por_mes_dia**. Este método recebe como parâmetros uma string ou inteiro com o dia e uma string ou um inteiro com o mês. Este método retorna uma lista de objetos Registros, com registros em que a compra tenha sido feita no dia e mês passado como parâmetros.

14) Desenvolver o método **localizar_compras_por_produto**. Este método recebe como parâmetro uma string com o nome do produto. Este método retorna uma lista de objetos do tipo Registro, com registros em que o nome do produto da compra seja idêntica ao nome do produto passado como parâmetro.

15) Desenvolver o método **localizar_compras_por_preco**. Este método recebe como parâmetro um número com o preço mínimo e outro número com o preço máximo. Este método retorna uma lista de objetos do tipo Registro, com registros em que o preço da compra esteja entre o valor mínimo e máximo passado como parâmetro.

16) Qual a responsabilidade da classe Compras ?