



Instituto Federal de Alagoas - IFAL
Campus Maceió
Estrutura de Dados
Prof. Ivo Calado

Lista de exercícios – Semana 02

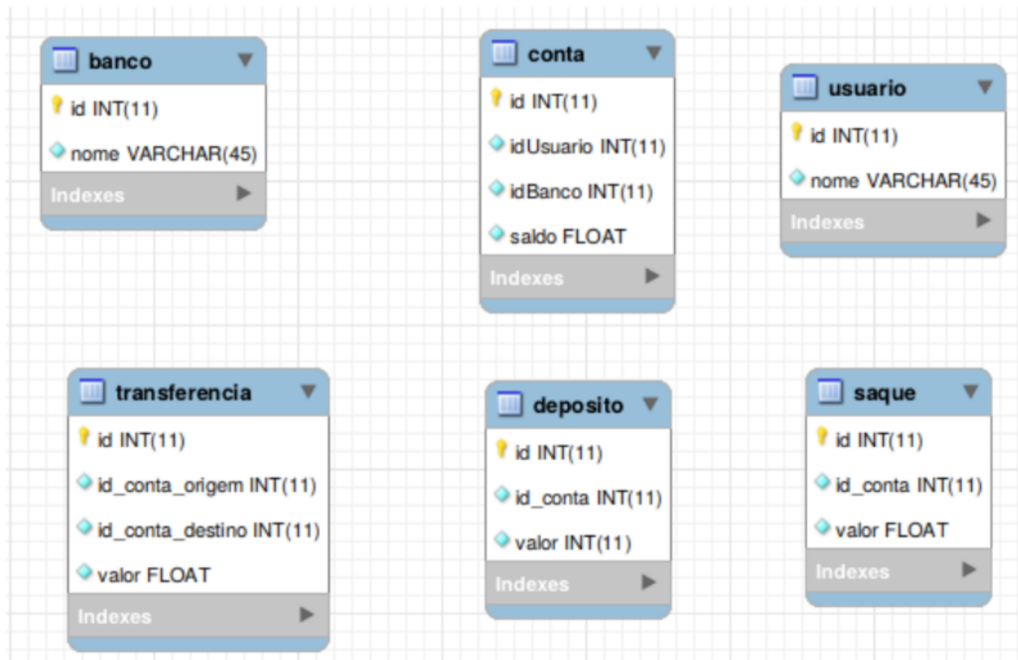
27 de janeiro de 2021

- 1) Você foi contratado por uma rede de supermercados para desenvolver um sistema informatizado para os caixas de sua rede. Para tal, foram passados os seguintes requisitos:
- Deseja-se manter um controle sobre os produtos cadastrados no sistema. Para cada produto, deve-se manter as seguintes informações:
 - Identificador do produto
 - Nome do produto
 - Descrição do produto
 - Valor de venda
 - Deseja-se manter um controle sobre os clientes que já realizaram compras. Para cada cliente, deve ser mantidas as seguintes informações
 - Nome;
 - CPF;
 - Endereço:
 - * Rua;
 - * Cep;
 - * Cidade.
 - Deseja-se manter controle sobre as compras efetivamente realizadas. Para cada compra deve-se manter as seguintes informações:
 - Data da compra (excepcionalmente, considere uma string representando a data);
 - Cliente que realizou a compra;
 - Lista dos itens comprados.

Em termos de menu apresentados, o sistema deve apresentar as seguintes opções:

- 1 - Cadastrar Produto: será solicitado ao usuário os dados sobre o produto que deverá ser armazenado. Considere que o usuário, ao digitar o Identificador do produto se assegurará que será digitado um identificador único, diferente de zero;
- 2 - Cadastrar Compra: deverá ser solicitado a data da compra e os dados do cliente que realizou a compra. Após isso, deverá ser lida a lista dos itens a serem adicionados. Para cada item o seguinte procedimento deverá ser adotado:
 - (a) Ler o código do produto;
 - (b) Procurar o produto na lista de produtos cadastrados no estoque a partir do identificador passado

- Se o produto não for localizado cancelar a inserção desse item em específico.
 - (c) Solicitar ao usuário o número de unidades a serem adquiridas do produto em questão
 - (d) Adicionar o novo item à lista de item comprados
- A leitura de novos itens deve ser interrompida quando o usuário digitar o valor **zero** como valor do identificador.
- 3 - Listar compras: Para cada compra cadastrada, deverá ser impresso a data da compra, o nome e CPF do comprador e o valor total da compra;
 - 4 - Sair: o sistema deve ser finalizado apenas quando o usuário selecionar esta opção.
- 2) A livraria *Livraria2000* deseja automatizar diversas tarefas relacionadas ao funcionamento do estabelecimento. Para tal, foram passados os seguintes requisitos:
- O software a ser desenvolvido tem como propósito gerenciar os processos de gerenciamento de livros, autores e funcionários. Para cada uma dessas entidades deverá ser construído o CRUD (*Create, Read, Update, Delete*).
 - Cada livro é definido pelas seguintes informações:
 - ISBN;
 - Título;
 - Ano;
 - Lista de autores.
 - Além disso, cada exemplar do livro deve ser registrado individualmente, onde para cada cópia as seguintes informações devem ser mantidas:
 - Livro;
 - Data de aquisição;
 - Custo de compra.
 - Cada Autor possui as seguintes informações:
 - Nome;
 - Endereço;
 - CPF;
 - Número de obras publicadas;
 - Descrição pessoal.
 - Em termos de interface o sistema deve apresentar as seguintes opções:
 - (a) Cadastrar autor
 - (b) Listar nomes de autores
 - (c) Recuperar autor por CPF
 - Deve ser exibido todos os dados do autor buscado. Caso o CPF não pertença a nenhum autor deve ser exibida uma mensagem indicando o problema.
 - (d) Deletar autor por CPF
 - (e) Cadastrar livro
 - (f) Listar livros
 - Deve ser listado os dados dos livros bem como as seguintes informações para cada livro:
 - i. Quantitativo de exemplares;
 - ii. Custo total (soma do custo de todos os exemplares do livro).
 - (g) Recuperar livros por ISBN
 - (h) Cadastrar exemplar com base em livro cadastrado



3) Tendo como base os tipos de dados apresentados a seguir, construa um programa que apresente o seguinte menu funcionalidades.

1 - Depositar

- Deve solicitar ao usuário o id da conta a ser depositada e o valor, inserir a transação na tabela 'deposito' e atualizar o saldo na tabela conta;

2 - Sacar

- Deve solicitar ao usuário o id da conta a ser depositada e o valor, inserir a transação na tabela 'saque' e atualizar o saldo na tabela conta;

3 - Transferência

- Deve solicitar ao usuário o id da conta de origem, o id da conta de destino e o valor da transferência. Em seguida deve-se inserir a transação na tabela 'transferencia' e atualizar o saldo na tabela conta para ambos os usuários;

4 - Recuperar patrimônio de um usuário

- Deve solicitar o id do usuário, recuperar as contas que esse usuário possui e calcular e somar o saldo de todas as contas que este usuário possui. Em seguida, deve-se exibir os dados do usuário juntamente com o valor do patrimônio.

5 - Recuperar o volume de transações realizadas entre dois bancos.

- Deve solicitar o id de dois bancos distintos, recuperar as transações realizadas entre contas dos mesmos e somar o valor das transações, exibindo em seguida para na tela.

6 - Sair.

4) Você foi contratado para o desenvolvimento de um tipo de dado abstrato denominado *interval_map*. Tal tipo de dado tem como principal característica o fato de que ele associa intervalos de chaves de tipo inteiro com um dado valor K. De certo modo, essa estrutura se assemelha bastante ao dicionário em Python tendo as seguintes diferenças:

- Obrigatoriamente é inicializado com um valor *default* que representa todos os intervalos não atribuídos;

- (b) Pode-se assumir que o valor de entrada de chave é do tipo inteiro;
- (c) O processo de inserção de uma nova entrada consiste da informação de um intervalo [início,final) (fechado no início e aberto no final) de chaves e um valor para o intervalo.

Desta forma, suponha que a sejam realizadas as seguintes inserções:

- Inicialização: $(1, 'A')$
- $(0, 3, 'B')$
- $(4, 6, 'C')$
- $(6, 8, 'D')$

Tal inserção representa o mapeamento:

- $-\infty - > A$
- $-2 - > A$
- $-1 - > A$
- $0 - > B$
- $1 - > B$
- $2 - > B$
- $3 - > A$
- $4 - > C$
- $5 - > C$
- $6 - > D$
- $7 - > D$
- $8 - > A$
- $\infty - > A$

Pode acontecer também de novos intervalos inseridos sobrescreverem total ou parcialmente antigos. Por exemplo, a inserção do intervalo $(-1, 2, 'E')$ alteraria o mapeamento para

- $-\infty - > A$
- $-2 - > A$
- $-1 - > E$
- $0 - > E$
- $1 - > E$
- $2 - > B$
- $3 - > A$
- $4 - > C$
- $5 - > C$
- $6 - > D$
- $7 - > D$
- $8 - > A$
- $\infty - > A$

É importante destacar que o armazenamento dos intervalos deve se dar de modo canônico. Isto é, ocupando o menos espaço possível. Ao afirmar isso, quer dizer dois intervalos consecutivos com a mesma chave e o mesmo valor devem representar, na verdade um único intervalo contínuo.

Sua tarefa é utilizando o código a seguir implementar o tipo de dado abstrato solicitado.

```
class interval_map:
    def __init__(self, valor_inicial):
        print("implementar")

    def inserir_valor(self, inicio, fim, valor_intervalo):
        print("implementar")

    def imprimir_intervalos(self):
        print("[<0, 'A')")
        print("[1, 3, 'B')")
        print("[3, 5, 'C')")
        print("[>5, 'A')")

    def imprimir_valor(self, chave):
        print("imprime valor da chave")
```

Observação: a classe de teste apresentada apresenta apenas os métodos que devem ser adicionados. Eventuais métodos adicionais podem ser adicionados caso necessários. Além disso, pode-se utilizar quaisquer uma das estruturas de dados básicas do Python (listas, dicionários ou conjuntos).