# **TRABALHO 01: Estoque do Cliente**

Trabalho desenvolvido durante a disciplina de BD1

# **Sumário**

### **1. COMPONENTES**

Integrantes do grupo  
 Eduardo Pitanga Loureiro: edudupilo@gmail.com  
 Theo Mischiatti Gomes: theo.gomes@outlook.com.br  
 Gabriel Dias Brito: cientistadias@gmail.com  
 Julio Cesar: julio.siilvafn2019@gmail.com

### **2.MINI-MUNDO**

O cliente deseja um sistema simples, no qual consiga fazer o controle de estoque, seja de bebidas, carnes, ou outro produtos de um restaurante, e se possível, um controle sobre a validade dos produtos em estoque. O sistema será gerenciado por uma única pessoa, da qual fará a inclusão dos produtos no estoque sempre que chegarem ao restaurante. Da inclusão, deseja-se saber o nome do produto, a quantidade, seja em quilos ou litros, a categoria, seja copa, câmara ou outro, a data de inclusão e a data de validade. Além disso, deseja-se ser possível o cadastro de novos produtos e novas categorias de estoque. Assim como a entrada, o sistema deve permitir a saída de estoque, informando a data de consumo (data de inclusão), quantidade e categoria. Dadas essas funcionalidades do sistema, deseja-se a construção de relatórios com uma visualização detalhada de certas coisas. Deseja-se visualizar um relatório com o total de produtos no estoque atual, separados por categoria. Outro relatório com o total consumido no mês X, separado por categoria. Além disso, também é desejado um relatório informando quais e a quantidade dos produtos que estão para vencer até o dia Y.

### 

### 

### **3. PERGUNTAS A SEREM RESPONDIDAS**

#### **3.1 QUAIS PERGUNTAS PODEM SER RESPONDIDAS COM O SISTEMA PROPOSTO?**

a) O sistema proposto poderá fornecer quais tipos de relatórios e informações?

Relatórios:

* produtos em estoque, separados por categoria
* consumo de produtos por categoria delimitado por um período de tempo
* produtos que estão próximos da data de validade ou que já venceram

b) Crie uma lista com os 5 principais relatórios que poderão ser obtidos por meio do sistema proposto!

O cliente informou que precisa inicialmente dos seguintes relatórios:

* Relatório que mostre o total de produtos em estoque, separados por categoria
* Relatório que mostre o total de produtos consumidos em determinados mês, separados por categoria.
* Relatório que mostre os produtos que estão para vencer e sua quantidade até determinado dia do mês.
* Relatório que mostre todas as movimentações de entrada e saída de produtos, mostrando datas, quantidades, categorias e outras informações relevantes.
* Relatório que mostre quais categorias de produtos estão vencendo com frequência.

### **5.MODELO CONCEITUAL**

A) Utilizar a Notação adequada (Preferencialmente utilizar o BRModelo 3)

B) O mínimo de entidades do modelo conceitual para este trabalho será igual a 3 e o Máximo 5.

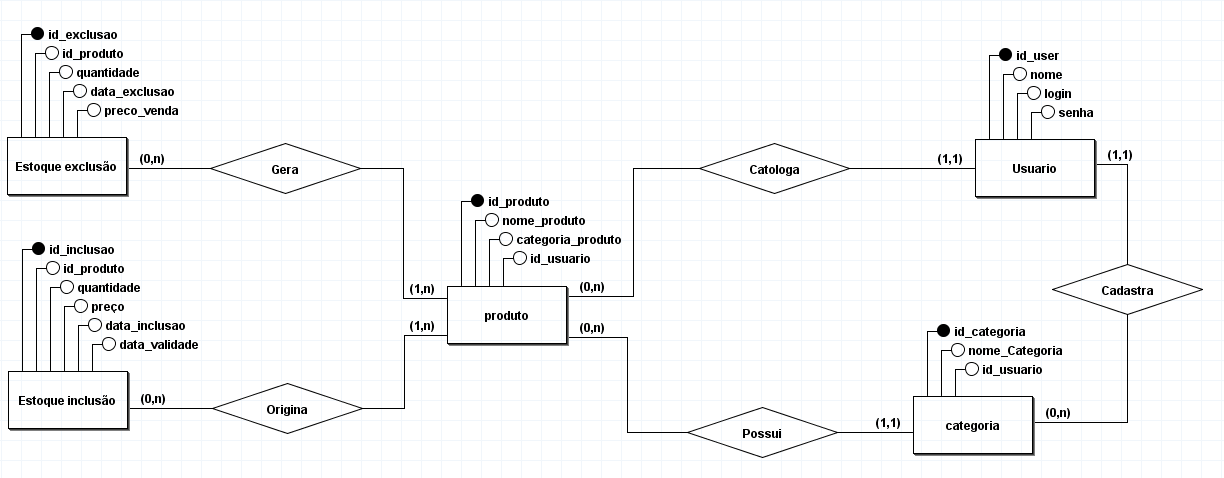
\* informe quais são as 3 principais entidades do sistema em desenvolvimento (se houverem mais de 3 entidades, pense na importância da entidade para o sistema)

C) Principais fluxos de informação/entidades do sistema (mínimo 3).

D) Qualidade e Clareza

Garantir que a semântica dos atributos seja clara no esquema (nomes coerentes com os dados).

Criar o esquema de forma a garantir a redução de informação redundante, possibilidade de valores null, e tuplas falsas (Aplicar os conceitos de normalização abordados).



#### 

#### **5.1 Validação do Modelo Conceitual**

[Grupo01]: [Nomes dos que participaram na avaliação]

#### **5.2 Descrição dos dados**

Estoque inclusão: Tabela que armazena as informações relativas à inclusão de produtos no estoque.

- `id\_inclusão`: Identificador único para cada inclusão.

- `id\_produto`: Identificador do produto incluído.

- `quantidade`: Quantidade de produtos incluídos.

- `preço`: Preço do produto incluído.

- `data\_inclusão`: Data em que o produto foi incluído no estoque.

- `data\_validade`: Data de validade do produto incluído.

Estoque exclusão: Tabela que armazena as informações relativas à exclusão de produtos do estoque.

- `id\_exclusão`: Identificador único para cada exclusão.

- `id\_produto`: Identificador do produto excluído.

- `quantidade`: Quantidade de produtos excluídos.

- `data\_exclusao`: Data em que o produto foi excluído do estoque.

- ‘preço\_venda’: Preço que o produto foi vendido.

Produto: Tabela que armazena as informações relativas aos produtos.

- `id\_produto`: Identificador único do produto.

- `nome\_produto`: Nome do produto.

- `categoria\_produto`: Categoria à qual o produto pertence.

Categoria: Tabela que armazena as informações relativas às categorias de produtos.

- `id\_categoria`: Identificador único da categoria.

- `nome\_Categoria`: Nome da categoria.

Usuario: Tabela que armazena as informações relativas aos usuários.

- `id\_user`: Identificador único do usuário.

- `nome`: Nome do usuário.

- `login`: Nome de login do usuário.

- `senha`: Senha do usuário.

**Marco de Entrega 01: Do item 1 até o item 5.2 (5 PTS)**

### **6 MODELO LÓGICO**

a) inclusão do esquema lógico do banco de dados

b) verificação de correspondência com o modelo conceitual

(não serão aceitos modelos que não estejam em conformidade)

### 

### 

### **7 MODELO FÍSICO**

a) inclusão das instruções de criação das estruturas em SQL/DDL

(criação de tabelas, alterações, etc..)

| DROP TABLE IF EXISTS estoque\_inclusao;  DROP TABLE IF EXISTS estoque\_exclusao;  DROP TABLE IF EXISTS produto;  DROP TABLE IF EXISTS categoria;  DROP TABLE IF EXISTS usuario;  CREATE TABLE estoque\_inclusao (  id\_inclusao int PRIMARY KEY,  id\_produto int,  quantidade int,  preco float,  data\_inclusao date,  data\_validade date );  CREATE TABLE estoque\_exclusao (  id\_exclusao int PRIMARY KEY,  quantidade int,  id\_produto int,  data\_exclusao DATE,  preco\_venda float );  CREATE TABLE produto (  id\_produto int PRIMARY KEY,  nome\_produto varchar,  categoria\_produto int,  id\_usuario int );  CREATE TABLE categoria (  id\_categoria int PRIMARY KEY,  nome\_categoria varchar  id\_ususario int  );  CREATE TABLE usuario (  id\_user int PRIMARY KEY,  nome varchar,  login varchar,  senha varchar );  alter table produto  add constraint fk\_categoria\_produto foreign key (categoria\_produto) references categoria(id\_categoria);  alter table produto  add constraint fk\_id\_usuario foreign key (id\_usuario) references usuario(id\_user);  alter table estoque\_exclusao  add constraint fk\_id\_produto foreign key (id\_produto) references produto(id\_produto);  alter table estoque\_inclusao  add constraint fk\_id\_produto foreign key (id\_produto) references produto(id\_produto); |
| --- |
| alter table categoria  add colum id\_usuario\_usuario int;  alter table categoria  add constraint fk\_id\_usuario foreign key (id\_usuario) references usuario(id\_user); |

### **8 INSERT APLICADO NAS TABELAS DO BANCO DE DADOS**

a) Script das instruções relativas a inclusão de dados

| insert into usuario(id\_user ,nome, login, senha) values(100,'eduardo','eduardo\_p','oi') insert into usuario(id\_user ,nome, login, senha) values(200,'theo','theo\_g','parede')  insert into categoria(id\_Categoria, nome\_Categoria,id\_ususario) values(1,'Copa',100); insert into categoria(id\_Categoria, nome\_Categoria, id\_usuario) values(2,'Câmara', 100);  insert into produto(id\_produto, nome\_produto, categoria\_produto, id\_usuario) values(1,'Picanha',2,100); insert into produto(id\_produto, nome\_produto, categoria\_produto, id\_usuario) values(2,'Alcatra',2,100); insert into produto(id\_produto, nome\_produto, categoria\_produto, id\_usuario) values(3,'Contra File',2,100); insert into produto(id\_produto, nome\_produto, categoria\_produto, id\_usuario) values(4,'Filé Mignon',2,100); insert into produto(id\_produto, nome\_produto, categoria\_produto, id\_usuario) values(5,'Cupim',2,100); insert into produto(id\_produto, nome\_produto, categoria\_produto, id\_usuario) values(6,'Coca Cola',1,200); insert into produto(id\_produto, nome\_produto, categoria\_produto, id\_usuario) values(7,'Guarana',1,200); insert into produto(id\_produto, nome\_produto, categoria\_produto, id\_usuario) values(8,'Suco de laranja',1,200); insert into produto(id\_produto, nome\_produto, categoria\_produto, id\_usuario) values(9,'Ancho',2,200); insert into produto(id\_produto, nome\_produto, categoria\_produto, id\_usuario) values(10,'Cerveja',2,100);  insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(1,1,50,600,'07/30/24','08/29/24'); insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(2,1,46,500,'07/21/24','08/20/24'); insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(3,1,60,650,'07/28/24','08/27/24'); insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(4,1,44,430,'08/30/24','09/29/24'); insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(5,1,54,300,'06/18/24','07/18/24'); insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(6,2,10,150,'06/19/24','07/19/24'); insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(7,2,16,300,'06/02/24','07/02/24'); insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(8,2,10,110,'08/21/24','09/20/24'); insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(9,2,11,90,'07/08/24','08/07/24'); insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(10,2,20,320,'07/09/24','08/08/24'); insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(11,3,9,100,'06/24/24','07/24/24'); insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(12,3,1,15,'06/05/24','07/05/24'); insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(13,3,1,30,'08/05/24','09/04/24'); insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(14,3,2,50,'08/26/24','09/25/24'); insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(15,3,7,120,'08/30/24','09/29/24'); insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(16,4,30,650,'06/07/24','07/07/24'); insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(17,4,34,700,'06/06/24','07/06/24'); insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(18,4,32,620,'08/17/24','09/16/24'); insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(19,4,32,625,'08/24/24','09/23/24'); insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(20,4,35,720,'08/13/24','09/12/24'); insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(21,6,60,340,'06/25/24','07/25/24'); insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(22,6,65,350,'06/04/24','07/04/24'); insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(23,6,67,400,'06/26/24','07/26/24'); insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(24,6,80,576,'06/03/24','07/03/24'); insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(25,6,45,300,'07/11/24','08/10/24'); insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(26,5,12,70,'06/25/24','07/25/24'); insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(27,7,15,30,'06/04/24','07/04/24'); insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(28,8,13,42,'06/26/24','07/26/24'); insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(29,9,14,100,'06/03/24','07/03/24'); insert into estoque\_inclusao(id\_inclusao,id\_produto,quantidade,preco,data\_inclusao,data\_validade) Values(30,10,10,50,'07/11/24','08/10/24');  insert into estoque\_exclusao(id\_exclusao,quantidade,id\_produto,data\_exclusao) Values(1,10,1,'07/15/24',200); insert into estoque\_exclusao(id\_exclusao,quantidade,id\_produto,data\_exclusao) Values(2,30,2,'07/16/24',630); insert into estoque\_exclusao(id\_exclusao,quantidade,id\_produto,data\_exclusao) Values(3,12,6,'07/17/24',190); insert into estoque\_exclusao(id\_exclusao,quantidade,id\_produto,data\_exclusao) Values(4,15,6,'07/18/24',230); insert into estoque\_exclusao(id\_exclusao,quantidade,id\_produto,data\_exclusao) Values(5,20,6,'07/19/24',345); insert into estoque\_exclusao(id\_exclusao,quantidade,id\_produto,data\_exclusao) Values(6,15,3,'07/20/24',400); insert into estoque\_exclusao(id\_exclusao,quantidade,id\_produto,data\_exclusao) Values(7,2,4,'07/21/24',120); insert into estoque\_exclusao(id\_exclusao,quantidade,id\_produto,data\_exclusao) Values(8,3,5,'07/22/24',174); insert into estoque\_exclusao(id\_exclusao,quantidade,id\_produto,data\_exclusao) Values(9,10,7,'07/23/24',100); insert into estoque\_exclusao(id\_exclusao,quantidade,id\_produto,data\_exclusao) Values(10,1,8,'07/24/24',30); insert into estoque\_exclusao(id\_exclusao,quantidade,id\_produto,data\_exclusao) Values(11,6,9,'07/25/24',415); insert into estoque\_exclusao(id\_exclusao,quantidade,id\_produto,data\_exclusao) Values(12,6,10,'07/26/24',115); |
| --- |

Requisito mínimo: (Script deve conter: Drop para exclusão de tabelas + create definição de para tabelas e estruturas de dados + insert para dados a serem inseridos)

OBS

1) Criar um novo banco de dados para testar a restauração (em caso de falha na restauração o grupo não pontuará neste quesito)

2) O script deve ser incluído no template em um arquivo no formato .SQL

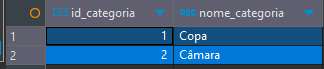
### 

### **9 TABELAS E PRINCIPAIS CONSULTAS**

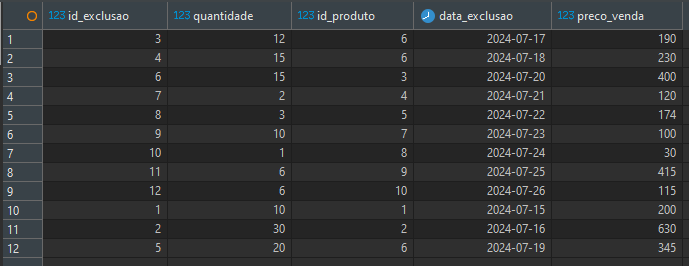
OBS: Usa template da disciplina disponibilizado no Colab.

#### **9.1 CONSULTAS DAS TABELAS COM TODOS OS DADOS INSERIDOS (Todas)**

| select \* from categoria; |
| --- |



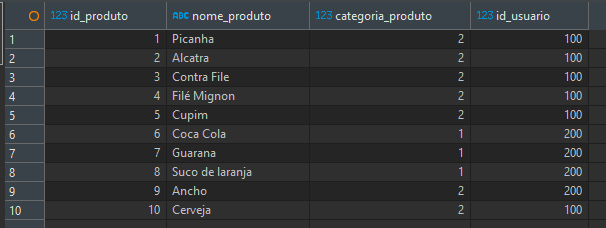
| select \* from estoque\_exclusao; |
| --- |



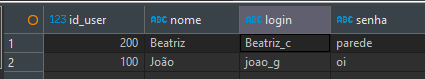
| **select \* from estoque\_inclusao;** |
| --- |

#### 

| **select \* from produto;** |
| --- |

****

| **select \* from usuario;** |
| --- |



#### 

#### **9.2 CONSULTAS DAS TABELAS COM FILTROS WHERE (Mínimo 4)**

| --1) Deseja-se visualizar um relatório com o total de produtos no estoque atual, separados por categoria. select   nome\_produto as Produto,  ((select sum(quantidade) from estoque\_inclusao eis where eis.id\_produto = pp.id\_produto) -   (select sum(quantidade) from estoque\_exclusao exs where exs.id\_produto = pp.id\_produto)) as quantidade from  produto pp where   categoria\_produto = 1 group by   id\_produto,   nome\_produto order by   nome\_produto |
| --- |

| --2) Outro relatório com o total consumido no mês X, separado por categoria. select   nome\_produto as Produto,   sum(quantidade) as Total\_Saida from   estoque\_exclusao  inner join produto on (estoque\_exclusao.id\_produto = produto.id\_produto) where   data\_exclusao between '07/01/2024' and '07/30/2024' and  categoria\_produto = 2 group by   nome\_produto order by   nome\_produto |
| --- |

| --3) relatório informando quais e a quantidade dos produtos que estão para vencer até o dia Y select   nome\_produto as Produto,  (incluido - excluido) as quantidade\_a\_vencer from (  select   pp.id\_produto,  pp.nome\_produto,  (select sum(quantidade) from estoque\_inclusao eis where eis.id\_produto = pp.id\_produto and eis.data\_validade <= '2024-07-30') as incluido,  (select sum(quantidade) from estoque\_exclusao exs where exs.id\_produto = pp.id\_produto) as excluido  from  produto pp  where   pp.categoria\_produto = 2  )  group by   nome\_produto,   incluido,   excluido having   (incluido - excluido) is not null and   (incluido - excluido) > 0 order by   nome\_produto |
| --- |

| --4) gastos totais no mes X select   sum(preco) as Total\_Gasto from  estoque\_inclusao ei  inner join produto pp on (pp.id\_produto = ei.id\_produto) where   pp.categoria\_produto in (1,2) and  data\_inclusao between '07/01/2024' and '07/30/2024' |
| --- |

#### 

#### **9.3 CONSULTAS QUE USAM OPERADORES LÓGICOS, ARITMÉTICOS E TABELAS OU CAMPOS RENOMEADOS (Mínimo 11)**

a) Criar 5 consultas que envolvam os operadores lógicos AND, OR e Not

| --1) Outro relatório com o total consumido no mês X, separado por categoria. select   nome\_produto as Produto,   sum(quantidade) as Total\_Saida from   estoque\_exclusao  inner join produto on (estoque\_exclusao.id\_produto = produto.id\_produto) where   data\_exclusao between '07/01/2024' and '07/30/2024' and  categoria\_produto = 2 group by   nome\_produto order by   nome\_produto |
| --- |

| --2) gastos totais no mes X select   sum(preco) as Total\_Gasto from  estoque\_inclusao ei  inner join produto pp on (pp.id\_produto = ei.id\_produto) where   pp.categoria\_produto in (1,2) and  data\_inclusao between '07/01/2024' and '07/30/2024' |
| --- |

| --3) lucro total no mes X select   (select   sum(preco\_venda)   from   estoque\_exclusao  where   data\_exclusao between '07/01/2024' and '07/30/2024') -   (select   sum(preco)   from   estoque\_inclusao  where   data\_inclusao between '07/01/2024' and '07/30/2024') as lucro\_total |
| --- |

| --4) Verifica a data de validade dos produtos dado um dia do mês select  nome\_produto,  data\_validade from   produto inner join estoque\_inclusao on (produto.id\_produto = estoque\_inclusao.id\_produto) where  estoque\_inclusao.data\_validade < ‘2024-07-24’; |
| --- |
|  |

| --5) Verifica se algum usuário cadastrou produtos nas duas categorias  SELECT  nome FROM  usuario WHERE  EXISTS (SELECT \* FROM produto WHERE produto.id\_usuario = usuario.id\_user AND produto.categoria\_produto = 1)  AND EXISTS (SELECT \* FROM produto p WHERE p.id\_usuario = usuario.id\_user AND p.categoria\_produto = 2); |
| --- |

b) Criar no mínimo 3 consultas com operadores aritméticos

| --1) Deseja-se visualizar um relatório com o total de produtos no estoque atual, separados por categoria. select   nome\_produto as Produto,  ((select sum(quantidade) from estoque\_inclusao eis where eis.id\_produto = pp.id\_produto) -   (select sum(quantidade) from estoque\_exclusao exs where exs.id\_produto = pp.id\_produto)) as quantidade from  produto pp where   categoria\_produto = 1 group by   id\_produto,   nome\_produto order by   nome\_produto |
| --- |

| --2) lucro total no mes X select   (select   sum(preco\_venda)   from   estoque\_exclusao  where   data\_exclusao between '07/01/2024' and '07/30/2024') -   (select   sum(preco)   from   estoque\_inclusao  where   data\_inclusao between '07/01/2024' and '07/30/2024') as lucro\_total |
| --- |

| --3) média de lucro por produto  select   nome\_produto as Produto,  cast(  (select avg(ee.preco\_venda/ee.quantidade) from estoque\_exclusao ee where ee.id\_produto = pp.id\_produto) -  (select avg(ei.preco/ei.quantidade) from estoque\_inclusao ei where ei.id\_produto = pp.id\_produto)   as decimal(6,2)) as Lucro\_Medio\_kg\_l from   produto pp group by   id\_produto, nome\_produto order by nome\_produto |
| --- |

c) Criar no mínimo 3 consultas com operação de renomear nomes de campos ou tabelas

| ALTER TABLE estoque\_exclusao RENAME COLUMN quantidade TO quantidade\_prod; |
| --- |

| ALTER TABLE estoque\_inclusao RENAME COLUMN quantidade TO quantidade\_prod; |
| --- |

| ALTER TABLE usuario RENAME COLUMN id\_user TO id\_usuario; |
| --- |

#### **9.4 CONSULTAS QUE USAM OPERADORES LIKE E DATAS (Mínimo 12)**

a) Criar outras 5 consultas que envolvam like ou ilike

| SELECT \*  FROM produto WHERE nome\_produto ILIKE 'A%'; |
| --- |

| SELECT estoque\_inclusao.\* FROM estoque\_inclusao JOIN produto ON estoque\_inclusao.id\_produto = produto.id\_produto WHERE produto.nome\_produto ILIKE '%Suco%'; |
| --- |

| SELECT estoque\_exclusao.\* FROM estoque\_exclusao JOIN produto ON estoque\_exclusao.id\_produto = produto.id\_produto WHERE produto.nome\_produto ILIKE '%File%'; |
| --- |

| SELECT \*  FROM produto WHERE nome\_produto LIKE 'F%'; |
| --- |

| SELECT \*  FROM usuario WHERE nome LIKE '%a%'; |
| --- |

b) Criar uma consulta para cada tipo de função apresentada.

| --Retonar os produtos vencidos do dia atual (current\_date) select  produto.nome\_produto,  estoque\_inclusao.data\_validade from  produto inner join estoque\_inclusao on (produto.id\_produto = estoque\_inclusao.id\_produto) where  estoque\_inclusao.data\_validade < (select current\_date); |
| --- |

# **Marco de Entrega 02: Do item 6. até o item 9.1 (5 PTS)**

#### **9.5 INSTRUÇÕES APLICANDO ATUALIZAÇÃO E EXCLUSÃO DE DADOS (Mínimo 6)**

a) Criar mínimo 3 de exclusão

| UPDATE estoque\_exclusao SET preco\_venda = NULL WHERE id\_exclusao = 1; |
| --- |

| UPDATE estoque\_exclusao SET quantidade = NULL WHERE id\_exclusao = 2; |
| --- |

| UPDATE estoque\_exclusao SET quantidade = null WHERE id\_exclusao = 5; |
| --- |

b) Criar mínimo 3 de atualização

| UPDATE usuario SET nome = 'João' WHERE id\_usuario = 100; |
| --- |

| UPDATE usuario SET nome = 'João' WHERE id\_usuario = 100; |
| --- |

| UPDATE usuario SET login = 'Beatriz\_c' WHERE id\_usuario = 200; |
| --- |

#### **9.6 CONSULTAS COM INNER JOIN E ORDER BY (Mínimo 6)**

a) Uma junção que envolva todas as tabelas possuindo no mínimo 2 registros no resultado

| -- Retorna uma visão geral do estoque dos produtos, quantos ainda possuem.  SELECT  produto.nome\_produto,  categoria.nome\_categoria,  usuario.nome AS nome\_usuario,  SUM(estoque\_inclusao.quantidade) - SUM(estoque\_exclusao.quantidade) AS estoque\_atual FROM  produto  INNER JOIN categoria ON (produto.categoria\_produto = categoria.id\_categoria)  INNER JOIN usuario ON (produto.id\_usuario = usuario.id\_user)  LEFT JOIN estoque\_inclusao ON (produto.id\_produto = estoque\_inclusao.id\_produto)  LEFT JOIN estoque\_exclusao ON (produto.id\_produto = estoque\_exclusao.id\_produto) GROUP BY  produto.nome\_produto, categoria.nome\_categoria, usuario.nome ORDER BY  estoque\_atual DESC; |
| --- |

b) Outras junções que o grupo considere como sendo as de principal importância para o trabalho

| --Outro relatório com o total consumido no mês X, separado por categoria. select   nome\_produto as Produto,   sum(quantidade) as Total\_Saida from   estoque\_exclusao  inner join produto on (estoque\_exclusao.id\_produto = produto.id\_produto) where   data\_exclusao between '07/01/2024' and '07/30/2024' and  categoria\_produto = 2 group by   nome\_produto order by   nome\_produto |
| --- |

| **-- quantidade de produtos cadastrados por usuario select   usuario.nome as Usuario,   count(id\_usuario) as Total\_Inclusoes from   produto  inner join usuario on (produto.id\_usuario = usuario.id\_user) group by  usuario.nome order by usuario.nome** |
| --- |

#### 

| **-- historico de preço select   data\_inclusao,  cast((preco/quantidade) as decimal(6,2)) as preco\_kg from   estoque\_inclusao  inner join produto on(produto.id\_produto = estoque\_inclusao.id\_produto) where   produto.nome\_produto = 'Alcatra' order by data\_inclusao** |
| --- |

#### 

| **-- produto mais vendido (quantidade) no mes X   select   nome\_produto as produto,  sum(quantidade) as quant\_total\_vendas from   produto pp  inner join estoque\_exclusao ee on (pp.id\_produto = ee.id\_produto) where   data\_exclusao between '07/01/2024' and '07/30/2024' group by   nome\_produto,   quantidade order by   quantidade  desc limit 1** |
| --- |

#### 

| -- historico de preço select   data\_inclusao,  cast((preco/quantidade) as decimal(6,2)) as preco\_kg from   estoque\_inclusao  inner join produto on(produto.id\_produto = estoque\_inclusao.id\_produto) where   produto.nome\_produto = 'Alcatra' order by data\_inclusao |
| --- |

#### 

#### 

#### 

#### **9.7 CONSULTAS COM GROUP BY E FUNÇÕES DE AGRUPAMENTO (Mín 6)**

a) Criar mínimo 2 envolvendo algum tipo de junção

| select  nome\_produto as Produto,  sum(quantidade) as Total\_Saida from  estoque\_exclusao  inner join produto on (estoque\_exclusao.id\_produto = produto.id\_produto) where  data\_exclusao between '07/01/2024' and '07/30/2024' and  categoria\_produto = 2 group by  nome\_produto order by  nome\_produto; |
| --- |

| select  usuario.nome as Usuario,  count(id\_usuario) as Total\_Inclusoes from  produto  inner join usuario on (produto.id\_usuario = usuario.id\_user) group by  usuario.nome order by usuario.nome |
| --- |

| select  nome\_produto as Produto,  (incluido - excluido) as quantidade from (  select  pp.id\_produto,  pp.nome\_produto,  (select sum(quantidade) from estoque\_inclusao eis where eis.id\_produto = pp.id\_produto) as incluido,  (select sum(quantidade) from estoque\_exclusao exs where exs.id\_produto = pp.id\_produto) as excluido  from  produto pp  ) group by  nome\_produto,  incluido,  excluido having  (incluido - excluido) is not null and  (incluido - excluido) <= 10 order by  nome\_produto |
| --- |

| select  nome\_produto as Produto,  cast(avg(ei.preco/ei.quantidade) as decimal(6,2)) as Preco\_Medio\_kg\_l from  produto  inner join estoque\_inclusao ei on (produto.id\_produto = ei.id\_produto) group by  ei.id\_produto, nome\_produto |
| --- |

| select  nome\_produto as produto,  sum(quantidade) as quant\_total\_vendas from  produto pp  inner join estoque\_exclusao ee on (pp.id\_produto = ee.id\_produto) where  data\_exclusao between '07/01/2024' and '07/30/2024' group by  nome\_produto,  quantidade order by  quantidade asc limit 1 |
| --- |

| select  nome\_produto as Produto,  cast(  (select avg(ee.preco\_venda/ee.quantidade) from estoque\_exclusao ee where ee.id\_produto = pp.id\_produto) -  (select avg(ei.preco/ei.quantidade) from estoque\_inclusao ei where ei.id\_produto = pp.id\_produto)  as decimal(6,2)) as Lucro\_Medio\_kg\_l from  produto pp group by  id\_produto, nome\_produto order by nome\_produto |
| --- |

#### 

#### **9.8 CONSULTAS COM LEFT, RIGHT E FULL JOIN (Mínimo 4)**

a) Criar mínimo 1 de cada tipo

| --Left: Produtos e sua quantidade de inclusão  SELECT  produto.nome\_produto,  SUM(estoque\_inclusao.quantidade) AS total\_inclusao FROM  produto p  LEFT JOIN estoque\_inclusao ON (produto.id\_produto = estoque\_inclusao.id\_produto) GROUP BY  produto.nome\_produto ORDER BY  produto.nome\_produto; |
| --- |

| --Right: Todas as exclusões do estoque, id do produto e o produto  SELECT  estoque\_exclusao.id\_exclusao,  produto.nome\_produto,  estoque\_exclusao.quantidade FROM  estoque\_exclusao RIGHT JOIN produto ON (estoque\_exclusao.id\_produto = produto.id\_produto) ORDER BY  estoque\_exclusao.id\_exclusao; |
| --- |

| --Full join: Mostrar todos os usuários e produtos cadastrados por eles    SELECT  usuario.nome AS nome\_usuario,  produto.nome\_produto FROM  usuario FULL JOIN produto ON (usuario.id\_user = produto.id\_usuario) ORDER BY  usuario.nome, produto.nome\_produto; |
| --- |

#### 

#### 

#### **9.9 CONSULTAS COM SELF JOIN E VIEW (Mínimo 6)**

a) Uma junção que envolva Self Join (caso não ocorra na base justificar e substituir por uma view)

justificativa: Não há dados dos quais possam ser correlacionados de modo a trazer resultados relevantes, visto que o sistema é muito baseado no relacionamento entre as tabelas e não em cada tabela atomicamente.

| CREATE VIEW dados\_inclusao AS SELECT quantidade, preco, data\_inclusao, data\_validade FROM Estoque\_inclusao SELECT \* FROM dados\_inclusao;   CREATE VIEW dados\_exclusao AS SELECT quantidade, preco, data\_exclusao, preco\_venda FROM Estoque\_exclusao SELECT \* FROM dados\_exclusao;  CREATE VIEW dados\_produto AS SELECT nome\_produto, categoria\_produto FROM produto SELECT \* FROM dados\_produto;  CREATE VIEW dados\_usuario AS SELECT nome, login FROM usuario SELECT \* FROM dados\_usuario;  CREATE VIEW dados\_categoria AS SELECT nome\_categoria FROM categoria SELECT \* FROM dados\_categoria;  CREATE VIEW dados\_validade AS SELECT quantidade, preco, data\_inclusao, data\_validade FROM Estoque\_inclusao WHERE data\_validade = '08/27/2024' SELECT \* FROM dados\_validade ; |
| --- |

b) Outras junções com views que o grupo considere como sendo de relevante importância para o trabalho

| CREATE VIEW dados\_inclusão\_preco AS SELECT quantidade, preco, data\_inclusao, data\_validade FROM Estoque\_inclusao WHERE preco >= 100 SELECT \* FROM dados\_inclusão\_preco ; |
| --- |

#### 

#### **9.10 SUBCONSULTAS (Mínimo 4)**

| -- produtos com baixo estoque select   nome\_produto as Produto,  (incluido - excluido) as quantidade from (  select   pp.id\_produto,  pp.nome\_produto,  (select sum(quantidade) from estoque\_inclusao eis where eis.id\_produto = pp.id\_produto) as incluido,  (select sum(quantidade) from estoque\_exclusao exs where exs.id\_produto = pp.id\_produto) as excluido  from  produto pp  )  group by   nome\_produto,   incluido,   excluido having   (incluido - excluido) is not null and   (incluido - excluido) <= 10 order by   nome\_produto |
| --- |

| --3) relatório informando quais e a quantidade dos produtos que estão para vencer até o dia Y select   nome\_produto as Produto,  (incluido - excluido) as quantidade\_a\_vencer from (  select   pp.id\_produto,  pp.nome\_produto,  (select sum(quantidade) from estoque\_inclusao eis where eis.id\_produto = pp.id\_produto and eis.data\_validade <= '2024-07-30') as incluido,  (select sum(quantidade) from estoque\_exclusao exs where exs.id\_produto = pp.id\_produto) as excluido  from  produto pp  where   pp.categoria\_produto = 2  )  group by   nome\_produto,   incluido,   excluido having   (incluido - excluido) is not null and   (incluido - excluido) > 0 order by   nome\_produto |
| --- |

a) Criar mínimo 1 envolvendo GROUP BY

| --1) Deseja-se visualizar um relatório com o total de produtos no estoque atual, separados por categoria. select   nome\_produto as Produto,  ((select sum(quantidade) from estoque\_inclusao eis where eis.id\_produto = pp.id\_produto) -   (select sum(quantidade) from estoque\_exclusao exs where exs.id\_produto = pp.id\_produto)) as quantidade from  produto pp where   categoria\_produto = 1 group by   id\_produto,   nome\_produto order by   nome\_produto |
| --- |

b) Criar mínimo 1 envolvendo algum tipo de junção

| --1) Deseja-se visualizar um relatório com o total de produtos no estoque atual, separados por categoria, e o nome do usuario que adicionou o produto. select  nome\_produto as Produto,  ((select sum(quantidade) from estoque\_inclusao eis where eis.id\_produto = pp.id\_produto) -  (select sum(quantidade) from estoque\_exclusao exs where exs.id\_produto = pp.id\_produto)) as quantidade,  nome as usuario from  produto pp  inner join usuario u on (u.id\_user = pp.id\_usuario) where  nome\_produto in (select nome\_produto from produto where categoria\_produto = 2 group by nome\_produto) group by  id\_produto,  nome\_produto,  nome order by  nome\_produto |
| --- |

# **Marco de Entrega 03: Do item 9.2 até o ítem 9.10 (10 PTS)**

### **10 RELATÓRIOS E GRÁFICOS**

#### **link do colab com os relatórios e graficos:**

[trabalho\_estoque.ipynb](https://colab.research.google.com/drive/1fdIG4SOp8y86fWDeHBF6M6mtqrFEVb48?usp=sharing)

# **Marco de Entrega 04: Itens 10 e 11 (20 PTS)**