**LPBD - Exercícios**

**O modelo abaixo representa um diagrama de banco de dados de vendas**

Diagrama

Descrição gerada automaticamente

1. **Criar e testar as Views para as consultas a seguir:**
   1. v\_total\_notas\_funcionarios

Select

n.notaId, n.dataNota, f.primeiroNome

, SUM(i.quantidade \* p.preco) as Total

FROM

Notas n

Inner Join Funcionarios f ON (f.funcionarioId = n.funcionarioId)

Inner Join Itens i ON (i.notaId = n.notaId)

Inner Join Produtos p ON (p.produtoId = i.produtoId)

GROUP BY n.notaId, n.dataNota, f.PrimeiroNome

ORDER BY Total DESC

* 1. v\_detalhe\_notas\_funcionarios

Select

n.notaId, n.dataNota, f.PrimeiroNome

,p.produtoNome, c.categoriaNome, i.quantidade , p.preco

,i.quantidade \* p.preco As 'Total Produto'

FROM

Notas n

Inner JOIN Funcionarios f On (f.FuncionarioId = n.FuncionarioId)

Inner JOIN Itens i On (i.notaId = n.notaId)

Inner JOIN Produtos p ON (p.produtoId = i.produtoId )

Inner JOIN Categorias c ON (c.categoriaId = p.categoriaId)

* 1. v\_total\_categorias

SELECT

c.categoriaNome,

SUM(i.quantidade \* p.preco) as 'Total de Vendas',

AVG(i.quantidade \* p.preco) as 'Média de Vendas'

FROM

Itens i

INNER JOIN Produtos p ON (p.produtoId = i.produtoId)

INNER JOIN Categorias c ON (c.categoriaId = p.categoriaId )

GROUP BY c.categoriaNome

ORDER BY 3 DESC

SELECT

p.productname,

SUM(d.Quantity \* p.Price) as 'Total de Vendas',

AVG(d.Quantity \* p.Price) as 'Média de Vendas'

FROM

OrderDetails d

INNER JOIN Products p ON (p.ProductID = d.ProductID)

GROUP BY p.productname

ORDER BY 3 DESC

1. Criar e testar uma view para a tabela Produtos chamada *v\_produtos\_categoria4*, que mostre todas as colunas e selecione somente as linhas com “categoriaId = 4”
   1. Insira o seguinte registro através da view *v\_produtos\_categoria4*:
      * insert into v\_produtos\_categoria4 values (1000, 'Bolo de Chocolate', 5, 3, '1 kg pkg.', 22.5);
   2. Faça uma consulta através da view *v\_produtos\_categoria4*:
      * O que aconteceu?
      * O novo produto aparece?
   3. Faça uma consulta na tabela Produtos pelo “produtoId=1000”:
      * O que aconteceu?
      * O novo produto aparece?
2. Alterar a view *v\_produtos\_categoria4* e inserir a opção *with check option*
   1. Insira o seguinte registro através da view *v\_produtos\_categoria4*:
      * insert into v\_produtos\_categoria4 values (2000, 'Bolo de Morango', 5, 3, '1 kg pkg.', 22.5);
      * O que aconteceu?
   2. Agora faça o seguinte insert através da tela *Produtos*:
      * insert into Produtos values (2000, 'Bolo de Morango', 5, 3, '1 kg pkg.', 22.5);
      * O que aconteceu?
3. Execute um comando Delete através da view *v\_produtos\_categoria4* para excluir o “produtoId = 1000”
   1. O que aconteceu? (Consulte a table de produtos)
4. Execute um comando Update através da view *v\_produtos\_categoria4* para alterar os dados que desejar do “produtoId = 1000”
   1. O que aconteceu? (Consulte a table de produtos)
5. Crie a view *v\_produtos\_categoria5*, através da seguinte consulta:

Select

c.CategoriaNome, c.Descricao,p.\*

From produtos p

inner Join categorias c On (p.CategoriaID = c.CategoriaID)

where p.CategoriaID = 5;

* 1. Tente inserir o registro a seguir:

insert into v\_produtos\_categoria5

(CategoriaId, CategoriaNome, Descricao) values (999, 'Outras', 'Outras');

* 1. O que aconteceu?

1. Apague a view v\_produtos\_categoria5
2. Criar uma view *v\_total\_vendas\_ano\_mes* que mostre o total de vendas agrupada pelo ano e mês de venda. A view deve exibir os dados ordenados pela ano e mês de vendas de maneira decrescente.
3. Crie uma view, através de joins, que utilize os dados de todas as tabelas do modelo (notas, itens, produtos, funcionarios, fornecedores, clientes, entregadores e categorias)