

Trabalho 7 - Banco de dados

Alunos:

- Caio Cezar
- Levi Cícero
- Lucas Garavaglia

1.

a) Liste o nome de todos os produtos cadastrados:

```
 $\pi$  nomePro(produto)
```

```
SELECT nomePro FROM produto;
```

b) Liste o nome e código de todos os produtos cujo o precoVenda seja maior que 100:

```
 $\pi$  nomePro, codPro ( $\sigma$  precoVenda > 100 (produto))
```

```
SELECT nomePro, codPro FROM produto WHERE precoVenda > 100;
```

c) Liste o código de todos os produtos que tenham registro de compra e venda:

```
 $\pi$  codPro (ItemCompra)  $\cap$   $\pi$  codPro (ItemVenda)
```

```
SELECT codPro FROM ItemCompra INTERSECT SELECT codPro FROM ItemVenda;
```

d) Liste o código de todos os produtos que não tem registro de venda ainda:

```
 $\pi$  codPro (produto) -  $\pi$  codPro(ItemVenda)
```

```
SELECT codPro FROM produto EXCEPT SELECT codPro FROM ItemVenda;
```

e) Liste o código de todos os produtos que não tem nenhum registro de compra e venda:

```
 $\pi \text{ codPro } (\text{produto}) - (\pi \text{ codPro } (\text{ItemCompra}) \cap \pi \text{ codPro } (\text{ItemVenda}))$ 
```

```
SELECT codPro FROM produto Except (  
    SELECT codPro  
    FROM produto  
    Except  
    SELECT codPro  
    FROM ItemVenda  
);
```

f) Liste o nome de todos os clientes cadastrados:

```
 $\pi \text{ nome } (\text{Cliente})$ 
```

```
SELECT nome  
FROM Cliente;
```

2.

a) Liste o nome de todos os produtos que têm venda:

```
 $\pi \text{ nomePro } (  
    \sigma \text{ produto.codPro } = \text{ItemVenda.codPro } (\text{produto} \times \text{ItemVenda})  
)$ 
```

```
SELECT nomePro  
FROM produto,  
    ItemVenda  
WHERE produto.codPro = ItemVenda.codPro;
```

b) Liste o nome dos clientes que tem notaVenda, cuja a venda foi realizada em 2021:

```
π nome (
  σ NotaVenda.codCliente = Cliente.codCliente ^ dataVenda = 2021
  (NotaVenda x Cliente)
)
```

```
SELECT nome FROM NotaVenda, Cliente
WHERE NotaVenda.codCliente = Cliente.codCliente
AND YEAR(NotaVenda.dataVenda) = 2021;
```

c) Liste o nome de todos os produtos que não tem Venda registrada:

```
π nomesPro (
  π codPro,
  nomePro (produto) - π codPro,
  nomePro (
    σ produto.codPro = ItemVenda.codPro (produto x ItemVenda)
  )
);
```

```
SELECT nomePro
FROM produto
Except (
  SELECT produto.nomePro
  FROM produto,
    ItemVenda
  WHERE produto.codPro = ItemVenda.codPro
);
```

d) Liste o nome de todos os produtos que tem Compra, mas não tem venda registrada:

```
π nomesPro (
  π codPro,
  nomePro (
    σ produto.codPro = ItemCompra.codPro (produto x ItemCompra)
  ) - (
    π codPro,
    nomePro (
      σ produto.codPro = ItemVenda.codPro (produto x ItemVenda)
    )
  )
)
```

```
SELECT nomePro FROM produto, ItemCompra
WHERE produto.codPro = ItemCompra.codPro
Except
SELECT nomePro FROM produto, ItemVenda
WHERE produto.codPro = ItemVenda.codPro;
```

e) Liste o nome de todos os clientes que ainda não tem venda registrada em 2021:

```
 $\pi$  codCliente (Cliente) - ( $\pi$  codCliente ( $\sigma$  dataVenda = 2021 (NotaVenda)))
```

```
SELECT nomePro
FROM produto,
ItemCompra
WHERE produto.codPro = ItemCompra.codPro
Except
SELECT nomePro
FROM produto,
ItemVenda
WHERE produto.codPro = ItemVenda.codPro;
```

f) Liste o produto que tem o maior preço de venda cadastrado:

```
(produto x produto) - ( $\sigma$  a1 < a2) ((pa1(produto)))(produto) x
(pa2(produto))(produto)
```

```
select produto.precoVenda, produto.nomePro from produto
Except
select produto.precoVenda, produto.nomePro
from produto join produto v2 on produto.precoVenda < v2.precoVenda;
```

3.

a) Considere que o cliente possa ter profissão informada ou não. Sendo assim liste o nome de todos os Clientes cadastrados com o nome da sua profissão, caso tenham:

```
π nome, nomeProfissao (Cliente _ [x] Profissao)
```

```
SELECT Cliente.nome, Profissao.nomeProfissao
FROM Cliente LEFT JOIN Profissao
ON Profissao.idProfissao = Cliente.idProfissao;
```

b) Considere que o cliente possa ter uma profissão informada ou não. Sendo assim liste o nome dos Clientes cadastrados com o nome de sua profissão, apenas para os clientes com profissão informada:

```
π nomeCliente,
  nomeProfissao (
  σ Cliente.idProfissao = Profissao.idProfissao (Cliente [x] _ Profissao)
)
```

```
SELECT Cliente.nome,
  Profissao.nomeProfissao
FROM Cliente
  RIGHT JOIN Profissao ON Cliente.idProfissao = Profissao.idProfissao
where Cliente.nome is not null;
```

c) Liste o nome dos produtos que tenham registros tanto em ItemVenda, quanto em itemCompra em Março 2021:

```
π nomePro (
  (
    σ codPro = produto.codPro (
      produto x (
        π codPro (
          σ nroVenda = ItemVenda.nroVenda (
            (
              π nroVenda (σ NotaVenda.dataVenda = "2021-03" (NotaVenda))
            )
          ) x ItemVenda
        )
      )
    )
  )
  ) v (
```

```

σ codPro = produto.codPro (
  produto x (
    π codPro (
      σ nroCompra = ItemCompra.nroCompra (
        (
          π nroCompra (σ NotaCompra.dataCompra = "2021-03"
(NotaCompra))
        )
      ) x ItemCompra
    )
  )
)
)
)
)
)
)
)

```

```

SELECT produto.nomePro
FROM NotaVenda
  INNER JOIN ItemVenda on ItemVenda.nroVenda = NotaVenda.nroVenda
  INNER JOIN produto on produto.codPro = ItemVenda.codPro
WHERE YEAR(NotaVenda.dataVenda) = "2021"
  AND MONTH(NotaVenda.dataVenda) = "03"
UNION
SELECT produto.nomePro
FROM NotaCompra
  INNER JOIN ItemCompra on ItemCompra.nroCompra = NotaCompra.nroCompra
  INNER JOIN produto on produto.codPro = ItemCompra.codPro
WHERE YEAR(NotaCompra.dataCompra) = "2021"
  AND MONTH(NotaCompra.dataCompra) = "03";

```

Exercício (NotaVenda x Cliente)

```

SELECT *
FROM NotaVenda
JOIN Cliente;

```