

Atividade Prática referente a aula 5

Aluno:	Bruna Martins, Camille Petersen, Caroline Ortiz, Lorenzo Ribeiro e Luiza Fagundes.	Data:	25/08/2023
--------	--	-------	------------

Responda a atividade a seguir **com as suas palavras**, expressando seu entendimento do assunto estudado.

OBS.: Entregar via T2K correção até o final da aula.

ETAPA: 1 de 5

- Escolher alguma realidade a ser representada em um banco de dados (idéias inicial de entidades).
- Deve ser claramente algo original (não copiado de algum outro lugar).
- Entrega desta etapa: Informar ao professor, que determinará se o tema é válido e tomará nota do que cada um resolveu fazer para ajudar a evitar duplicações.

ETAPA: 2 de 5

- Modelar 4 ou mais tabelas de acordo com a realidade escolhida na etapa 1.
- Entrega desta etapa: Um script SQL com os comandos DDL (create table, primary keys e foreign keys).

ETAPA: 3 de 5

Escrever 1 views que explorem subqueries aplicando corretamente os conceitos:

- Subquery como uma nova coluna da consulta.
- Subquery como filtro de uma consulta
- Entrega desta etapa: Um script SQL com os comandos DDL. Pode ser um trecho adicional no mesmo script da **etapa 2**.

ETAPA: 4 de 5

- Escrever pelo menos 1 triggers que faça sentido com a realidade representada na **etapa 2**.
- Aqui também vale a originalidade.
- Entrega desta etapa: Um script SQL com os comandos de consulta. Pode ser um trecho adicional no mesmo script da Etapa 2.

ETAPA: 5

- Criar perfil no [github](https://github.com) e criar repositório com todos os códigos de todas as etapas do trabalho.

INSTRUÇÕES GERAIS

Os scripts SQL devem ser arquivos com extensão SQL.

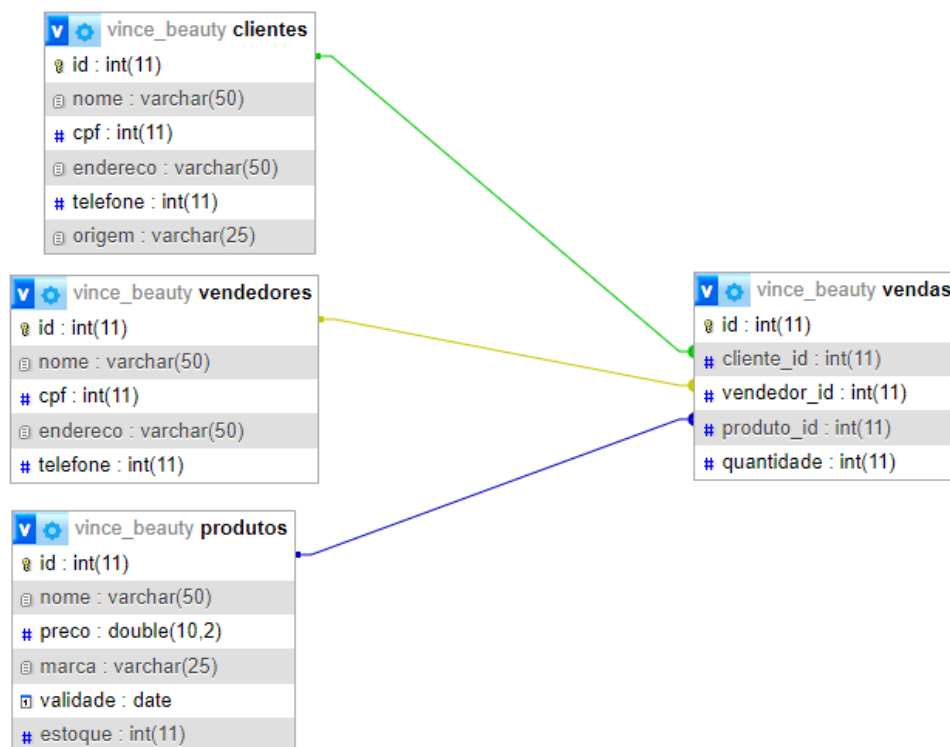
Nenhum outro formato será aceito (txt, doc, pdf, corpo do e-mail, etc).

Qualquer dúvida ou demanda siga à disposição.

Att. Márcio Lemos

ETAPA 1: Banco De Dados Da VINCE BEAUTY DISTRIBUIDORA

ETAPA 2: Tabelas -> Produtos, Clientes, Vendedores, Vendas.



```
--  
-- Banco de dados: `vince_beauty`  
--
```

```
-- Estrutura para tabela `clientes`
--

CREATE TABLE `clientes` (
  `id` int(11) NOT NULL,
  `nome` varchar(50) DEFAULT NULL,
  `cpf` int(11) DEFAULT NULL,
  `endereco` varchar(50) DEFAULT NULL,
  `telefone` int(11) DEFAULT NULL,
  `origem` varchar(25) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

-----

-- Estrutura para tabela `produtos`
--

CREATE TABLE `produtos` (
  `id` int(11) NOT NULL,
  `nome` varchar(50) DEFAULT NULL,
  `preco` double(10,2) DEFAULT NULL,
  `marca` varchar(25) DEFAULT NULL,
  `validade` date DEFAULT NULL,
  `estoque` int(11) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

-----

-- Estrutura para tabela `vendas`
--

CREATE TABLE `vendas` (
  `id` int(11) NOT NULL,
  `cliente_id` int(11) DEFAULT NULL,
  `vendedor_id` int(11) DEFAULT NULL,
  `produto_id` int(11) DEFAULT NULL,
  `quantidade` int(11) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

-----

-- Estrutura para tabela `vendedores`
--

CREATE TABLE `vendedores` (
  `id` int(11) NOT NULL,
  `nome` varchar(50) DEFAULT NULL,
  `cpf` int(11) DEFAULT NULL,
```

```
`endereco` varchar(50) DEFAULT NULL,  
`telefone` int(11) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;  
  
--  
-- Índices para tabelas despejadas  
--  
-- Índices de tabela `clientes`  
--  
ALTER TABLE `clientes`  
  ADD PRIMARY KEY (`id`);  
  
--  
-- Índices de tabela `produtos`  
--  
ALTER TABLE `produtos`  
  ADD PRIMARY KEY (`id`);  
  
--  
-- Índices de tabela `vendas`  
--  
ALTER TABLE `vendas`  
  ADD PRIMARY KEY (`id`),  
  ADD KEY `fk_cliente_id` (`cliente_id`),  
  ADD KEY `fk_vendedor_id` (`vendedor_id`),  
  ADD KEY `fk_produto_id` (`produto_id`);  
  
--  
-- Índices de tabela `vendedores`  
--  
ALTER TABLE `vendedores`  
  ADD PRIMARY KEY (`id`);  
  
-- Restrições para tabelas despejadas  
--  
-- Restrições para tabelas `vendas`  
--  
ALTER TABLE `vendas`  
  ADD CONSTRAINT `fk_cliente_id` FOREIGN KEY (`cliente_id`) REFERENCES `clientes` (`id`),  
  ADD CONSTRAINT `fk_produto_id` FOREIGN KEY (`produto_id`) REFERENCES `produtos` (`id`),  
  ADD CONSTRAINT `fk_vendedor_id` FOREIGN KEY (`vendedor_id`) REFERENCES `vendedores` (`id`);  
COMMIT;
```

ETAPA 3:

`SELECT nome from vendedores where id = 2;`

☐ Perfil [[Editar em linha](#)] [[Editar](#)] [[Demonstrar SQL](#)] [[Criar código PHP](#)] [[Atualizar](#)]

☐ Mostrar tudo | Número de linhas: 25 ▼ Filtrar linhas:

Opções extras

	nome
<input type="checkbox"/> Editar Copiar Remover	Jair Sousa

`SELECT nome as nome_cliente, endereco as endereco_cliente from clientes where cpf = 123456789;`

☐ Perfil [[Editar em linha](#)] [[Editar](#)] [[Demonstrar SQL](#)] [[Criar código PHP](#)] [[Atualizar](#)]

☐ Mostrar tudo | Número de linhas: 25 ▼ Filtrar linhas:

Opções extras

	nome_cliente	endereco_cliente
<input type="checkbox"/> Editar Copiar Remover	Maira Silva	Rua 5, casa 7

ETAPA 4:

```
--
-- Acionadores `vendas`
--
DELIMITER $$
CREATE TRIGGER `atualiza_estoque_venda` AFTER INSERT ON `vendas` FOR EACH ROW BEGIN
    DECLARE produto_id int;
    DECLARE quantidade_vendida int;

    SET produto_id = new.produto_id;
    SET quantidade_vendida = new.quantidade;

    UPDATE produtos
    SET estoque = estoque - quantidade_vendida
    WHERE id = produto_id;
```

```
END  
$$  
DELIMITER ;
```

ETAPA 5: