

INSTITUTO INFNET
ESCOLA SUPERIOR DE TECNOLOGIA
GRADUAÇÃO EM ENGENHARIA DE SOFTWARE



Fundamentos de Modelagem Relacional e SQL

TP 1

Alunos: Richard de Jesus Cabral Alves.

25 de mai. de 2022

Comercial Nova Era

Contexto: A Comercial Nova Era é uma rede varejista com lojas espalhadas por diferentes estados. O departamento de dados foi encarregado de criar análises cruzando informações de lojas, vendedores, produtos e vendas.

1. A diretoria quer saber quais vendedores estão oficialmente cadastrados em cada loja, com o nome da loja e a cidade onde ela fica. Montar um relatório que liste loja, cidade e os respectivos vendedores.
2. A gerente regional do RJ suspeita que algumas lojas do estado estão vendendo muitos itens de Telefonia. Preparar um levantamento que mostre, para cada venda de produtos da categoria "Telefonia", quem vendeu, em qual loja e em qual cidade essa loja está.
3. O time de *pricing* pediu um material interno que mostre, para cada venda realizada, qual produto foi vendido, o preço unitário atual desse produto e a quantidade vendida. Eles querem cruzar isso para estimar a receita potencial por item.
4. O RH está avaliando desempenho individual. Eles pediram uma lista com todos os vendedores (inclusive quem ainda não fez nenhuma venda), junto com o total de itens vendidos por cada um.

5. A coordenação comercial quer entender o perfil das lojas. Eles pediram um painel que mostre cada loja e todos os produtos que já foram vendidos naquela loja, sem repetir a loja. A ideia é enxergar quais categorias já “encostaram” em cada loja.
6. A gerência de operações está revisando o mix de produtos por categoria. Eles querem ver cada produto da categoria "Móveis" e em qual cidade ele já foi vendido, se é que já foi. É importante também identificar os produtos dessa categoria que ainda não apareceram em nenhuma venda.
7. O pessoal de logística levantou suspeita de que mesas e cadeiras estão saindo em conjunto. Construir uma visão que junte informações das vendas com os dados de produto, mas filtrando apenas itens da categoria "Móveis". Eles querem ver quem vendeu, o que vendeu, em qual data e para qual estado a loja pertence.
8. A diretoria quer priorizar treinamento para vendedores de lojas que vendem eletrônicos caros. Eles consideram “eletrônico caro” qualquer produto com preço acima de 3000. Preparar um extrato listando vendedor, produto vendido, preço do produto e o nome da loja onde esse vendedor atua, apenas para esses casos.
9. O comercial está avaliando se existe sobreposição de esforço entre vendedores que trabalham na mesma loja. Eles querem um

cruzamento que relacione vendedores de uma mesma loja entre si, em pares. A ideia é identificar duplas de vendedores que dividem o mesmo espaço físico.

10. A área de expansão está estudando quais cidades parecem responder melhor à linha de Telefonia (smartphone, fone etc.). Preparar um resumo que associe cidade, produto da categoria "Telefonia" e quantidade vendida nesses itens. Eles querem enxergar em quais cidades essa categoria realmente gira.

Use:

```
-- DDL and INSERTS for Dataset 1 (Comercial Nova Era)

CREATE TABLE lojas (
  id SERIAL PRIMARY KEY,
  nome TEXT NOT NULL,
  cidade TEXT NOT NULL,
  estado TEXT NOT NULL
);

CREATE TABLE vendedores (
  id SERIAL PRIMARY KEY,
  nome TEXT NOT NULL,
  loja_id INT NOT NULL REFERENCES lojas(id)
);

CREATE TABLE produtos (
  id SERIAL PRIMARY KEY,
  nome TEXT NOT NULL,
  categoria TEXT NOT NULL,
  preco NUMERIC(10,2) NOT NULL
);

CREATE TABLE vendas (
  id SERIAL PRIMARY KEY,
  vendedor_id INT NOT NULL REFERENCES vendedores(id),
  produto_id INT NOT NULL REFERENCES produtos(id),
  data_venda DATE NOT NULL,
  quantidade INT NOT NULL CHECK (quantidade > 0)
);

INSERT INTO lojas (nome, cidade, estado) VALUES
('Nova Era Centro', 'Rio de Janeiro', 'RJ'),
('Nova Era Norte', 'Niterói', 'RJ'),
('Nova Era Paulista', 'São Paulo', 'SP'),
('Nova Era Pampulha', 'Belo Horizonte', 'MG');

INSERT INTO vendedores (nome, loja_id) VALUES
```

```
('Ana Souza', 1),  
('Bruno Martins', 1),  
('Carlos Lima', 2),  
('Daniela Rocha', 3),  
('Eduardo Farias', 3),  
('Fernanda Alves', 4);
```

```
INSERT INTO produtos (nome, categoria, preco) VALUES  
('Notebook Atlas 14"', 'Informática', 4500.00),  
('Mouse Óptico Pro', 'Informática', 80.00),  
('Cadeira Gamer Storm', 'Móveis', 1299.90),  
('Smartphone Zeta X', 'Telefonia', 2999.99),  
('Fone Bluetooth Wave', 'Telefonia', 199.90),  
('Mesa Escritório Compact', 'Móveis', 650.00);
```

```
INSERT INTO vendas (vendedor_id, produto_id, data_venda, quantidade) VALUES  
(1, 1, '2025-01-05', 2),  
(1, 2, '2025-01-05', 5),  
(2, 4, '2025-01-06', 1),  
(2, 5, '2025-01-06', 3),  
(3, 2, '2025-01-07', 4),  
(3, 6, '2025-01-08', 1),  
(4, 1, '2025-01-09', 1),  
(4, 3, '2025-01-09', 2),  
(5, 4, '2025-01-10', 2),  
(5, 5, '2025-01-11', 4),  
(6, 3, '2025-01-12', 1),  
(6, 6, '2025-01-12', 2);
```

```
INSERT INTO produtos (nome, categoria, preco) VALUES  
('Webcam Crystal HD', 'Informática', 320.00);
```

```
INSERT INTO vendedores (nome, loja_id) VALUES  
('Gustavo Nogueira', 2);
```

CinePlus

Contexto: A CinePlus é uma plataforma de streaming que quer aprofundar suas análises cruzando informações de usuários, planos, filmes, histórico de exibição e pagamentos.

1. A diretoria quer ter clareza de qual serviço está entregando valor real. Eles pediram um levantamento que, para cada usuário, mostre qual plano ele está associado e em qual cidade ele mora.
2. O time de retenção quer descobrir se existe algum risco de churn vindo de contas que não estão mais assistindo nada, mas continuam (ou continuaram recentemente) pagando por plano. Eles solicitaram uma visão relacionando quem fez pagamento e se essa pessoa tem ou não histórico de filmes assistidos.
3. A equipe de marketing quer montar uma campanha segmentada por gênero de filme. Eles pediram um material onde seja possível ver cada filme que foi assistido, seu gênero e quem assistiu. Eles querem bater isso depois com preferências de anúncio personalizado.
4. A análise editorial quer entender quais títulos “seguram” melhor a atenção das cidades grandes. Eles pediram um painel relacionando usuários que moram em São Paulo com os filmes que eles assistiram e a respectiva avaliação dada.

5. O financeiro reclamou que alguns usuários estavam assistindo conteúdo Premium enquanto estavam cadastrados em planos baratos, e quer validar isso. Eles querem um relatório cruzando, para cada sessão assistida, quem é o usuário, que plano essa pessoa tem no cadastro e qual foi o filme visto.
6. O time de catálogo quer identificar filmes que ainda não tiveram nenhuma visualização registrada. Eles pediram uma listagem de todos os filmes disponíveis, destacando os que ainda não aparecem no histórico de quem assistiu.
7. A área de produto quer medir engajamento de quem está no plano mais caro. Eles querem uma visão onde apareçam apenas os usuários do plano Premium, junto com todos os filmes que cada um já assistiu e a nota que deram.
8. A gerência quer saber se o pessoal de "Documentário" é fiel ou se só assiste uma vez e some. Para isso, eles querem cruzar quem assistiu filmes do gênero "Documentário" com a cidade dessas pessoas. A ideia é mapear em quais cidades esse gênero está pegando.
9. A área de cobrança está preocupada com perfis que estão sem plano ativo ou que aparecem sem pagamento registrado recentemente. Eles pediram um levantamento dos usuários que não aparecem na tabela de pagamentos e também dos usuários

cujo plano_id está vazio, com nome e cidade.

10. O time de recomendação está testando um algoritmo que conecta pessoas com gostos parecidos. Eles querem um cruzamento de pares de usuários que assistiram ao mesmo filme, para detectar possíveis “grupos de afinidade”. A ideia é gerar pares de usuários que têm pelo menos um filme em comum.

Use:

```
-- DDL and INSERTS for Dataset 2 (CinePlus)

CREATE TABLE planos (
  id SERIAL PRIMARY KEY,
  tipo TEXT NOT NULL,
  valor_mensal NUMERIC(10,2) NOT NULL
);

CREATE TABLE usuarios (
  id SERIAL PRIMARY KEY,
  nome TEXT NOT NULL,
  cidade TEXT NOT NULL,
  plano_id INT REFERENCES planos(id)
);

CREATE TABLE filmes (
  id SERIAL PRIMARY KEY,
  titulo TEXT NOT NULL,
  genero TEXT NOT NULL,
  ano INT NOT NULL
);

CREATE TABLE assistidos (
  usuario_id INT REFERENCES usuarios(id),
  filme_id INT REFERENCES filmes(id),
  data_assistido DATE NOT NULL,
  avaliacao INT CHECK (avaliacao BETWEEN 1 AND 5),
  PRIMARY KEY (usuario_id, filme_id, data_assistido)
);

CREATE TABLE pagamentos (
  id SERIAL PRIMARY KEY,
  usuario_id INT REFERENCES usuarios(id),
  plano_id INT REFERENCES planos(id),
  data_pagamento DATE NOT NULL
);

INSERT INTO planos (tipo, valor_mensal) VALUES
('Básico', 24.90),
('Padrão', 39.90),
```



```
('Premium', 59.90);
```

```
INSERT INTO usuarios (nome, cidade, plano_id) VALUES  
( 'Alice Moraes', 'Rio de Janeiro', 3),  
( 'Bruno Castro', 'São Paulo', 2),  
( 'Carla Nunes', 'Niterói', 1),  
( 'Diego Freitas', 'São Paulo', 3),  
( 'Elisa Prado', 'Belo Horizonte', 2),  
( 'Fernando Alves', 'Curitiba', NULL);
```

```
INSERT INTO filmes (titulo, genero, ano) VALUES  
( 'Noite Sem Fim', 'Ação', 2023),  
( 'Corações de Ferro', 'Drama', 2022),  
( 'Rastros da Neblina', 'Suspense', 2024),  
( 'Amor em São Paulo', 'Romance', 2021),  
( 'Circuito Clandestino', 'Ação', 2024),  
( 'Histórias do Subsolo', 'Documentário', 2020);
```

```
INSERT INTO assistidos (usuario_id, filme_id, data_assistido, avaliacao) VALUES  
(1, 1, '2025-01-03', 4),  
(1, 2, '2025-01-05', 5),  
(1, 3, '2025-01-07', 3),  
(2, 1, '2025-01-04', 5),  
(2, 4, '2025-01-06', 2),  
(3, 6, '2025-01-07', 4),  
(3, 2, '2025-01-10', 3),  
(4, 5, '2025-01-03', 5),  
(4, 1, '2025-01-08', 4),  
(5, 2, '2025-01-02', 4),  
(5, 4, '2025-01-09', 5);
```

```
INSERT INTO pagamentos (usuario_id, plano_id, data_pagamento) VALUES  
(1, 3, '2025-01-01'),  
(2, 2, '2025-01-01'),  
(3, 1, '2025-01-01'),  
(4, 3, '2025-01-01'),  
(5, 2, '2025-01-01');
```