-- 1. Quantidade de Funcionários por Departamento

-- Seleciona a coluna depto e faz uma contagem das linhas

select depto, count(\*)

from funcionarios

-- Agrupa os dados com base nos valores unicos da coluna depto

group by depto;

-- 2. Que departamentos tem média salarial maior que R$ 1000?

-- Avg média dos salários dos fungionários de cada departamento

select depto, avg(salario)

from funcionarios

-- Agrupa os dados com bases em valores únicos da coluna depto

group by depto

--having é usada para filtras os grupos de dpto. E mostra os grupos em que a média é maior que 1000.

having avg(salario) > 1000;

-- 3. Média salárial dos funcionários por sexo do setor de Compras

select sexo, avg(salario)

from funcionarios

-- Where filtra os registros da tabela onde dpto = compras

where depto = 'Compras'

-- Agrupa os dados com base nos valores na coluna 'sexo'

group by sexo;

-- 4. O maior salário do departamento que possui pelo menos duas pessoas

-- max busca o valor máximo de salário dentro de cada departamento

select depto, max(salario)

from funcionarios

group by depto

-- Mantém apenas os grupos em que o número de funcionários dentro do grupo é igual ou maior que 2

having count(\*) >= 2;

-- 5. Número de pessoas por sexo, por departamento

-- Consulta sexo, depto e count que conta o número de registros de cada grupo

select sexo, depto, count(\*)

from funcionarios

-- Agrupa os dados nas combinações sexo e depto.

-- Cria grupos distintos para cada combinação de gênero e departamento,

group by sexo, depto;

-- 6. Qual o total da folha de pagamento de cada departamento?

-- Calcula a soma total dos salários dos funcionários em cada departamento

select depto, sum(salario)

from funcionarios

-- Agrupa os resultados com base nos valores únicos encontrados na coluna depto

group by depto;

-- 7. Soma dos salários dos Departamentos com mais de 1 funcionário

-- Seleciona a coluna depto e calcula a soma total dos salários de todos os funcionários em cada departemento

select depto, sum(salario)

from funcionarios

-- Agrupar os resultados com base nos valores únicos encontrados na coluna depto

group by depto

-- Apenas os grupos em que o número de funcionários é maior que 1

having count(\*) > 1;

-- QUESTAO B. Modifique o Esquema --

-- 1. Exporte a coluna de departamento para outra tabela (crie codigo e descrição).

-- Cria uma nova tabela chamada depto com duas colunas depto e nome

create table depto as

-- A coluna cone é preenchida com os valores distintos encontrados na coluna depto da tabela funcionarios

-- Por meio de uma subconsulta que primeiro seleciona os valores distintos da coluna depto na tabela funcionarios

select rownum depto, depto nome

from (

select distinct depto

from funcionarios

);

-- O segundo comando renomeia a coluna depto na tabela funcionarios para depto\_nome

alter table funcionarios rename column depto to depto\_nome;

-- Adiciona uma nova coluna chamada depto à tabela funcionarios

alter table funcionarios add depto number(2);

-- Preenche a coluna depto na tabela funcionarios com os valores corretos com base na

-- correspondência entre depto\_nome e "nome" nas duas tabelas

update funcionarios f

set f.depto = (

select d.depto

from depto d

where f.depto\_nome = d.nome

);

-- Remove a coluna depto\_nome da tabela "funcionarios". A coluna foi renomeada para

-- depto\_nome e agora está sendo removida

alter table funcionarios drop column depto\_nome;

-- Adiciona uma restrição de chave primária à coluna depto da tabela depto

alter table depto add

constraint depto\_pk primary key (depto);

-- adiciona uma restrição de chave estrangeira à coluna depto da tabela funcionarios

-- relaciona os valores na coluna depto da tabela funcionarios com os valores na coluna depto da tabela depto

alter table funcionarios add

constraint depto\_fk foreign key (depto) references depto;

---------------------------------------

-- Exercicios sobre o Estudo de Caso

---------------------------------------

-- EC1. Qual a quantidade de endereços por estado?

-- Seleciona a coluna uf e conta o numero todal de registros

select uf, count(\*)

from estados

-- Faz uma junção entre as tabelas estados e cidades usando a coluna uf

-- Relaciona as unidades federativas nas duas tabelas

JOIN cidades ON estados.uf = cidades.uf

-- Faz uma junção, desta vez entre as tabelas cidades e enderecos usando a coluna cod\_cidade

-- Relaciona informações de endereços com as cidades associadas

JOIN enderecos ON cidades.cod\_cidade = enderecos.cod\_cidade

-- Agrupa os resultados com base nos valores únicos encontrados na coluna uf da tabela estados

GROUP BY estados.uf;

-- EC2. Qual a quantidade de clientes de cada estado?

-- Seleciona a coluna uf da tabela estados e conta o numero de clientes distintos

SELECT estados.uf, COUNT(DISTINCT clientes.cod\_cliente)

FROM estados

-- Faz uma junção entre as tabelas estados e cidades usando a coluna uf

-- Relaciona as unidades federativas nas duas tabelas

JOIN cidades ON estados.uf = cidades.uf

-- Faz uma junção, desta vez entre as tabelas cidades e enderecos usando a coluna cod\_cidade

-- Relaciona informações de endereços com as cidades associadas

JOIN enderecos ON cidades.cod\_cidade = enderecos.cod\_cidade

-- Faz uma junção liga as tabelas enderecos e clientes\_enderecos usando a coluna cod\_endereco

-- Relaciona endereços a clientes

JOIN clientes\_enderecos ON enderecos.cod\_endereco = clientes\_enderecos.cod\_endereco

-- Faz uma junção entre as tabelas clientes\_enderecos e clientes usando a coluna cod\_cliente

-- Relaciona clientes aos seus endereços

JOIN clientes ON clientes\_enderecos.cod\_cliente = clientes.cod\_cliente

-- Agrupa os resultados por unidade federativa (UF), permitindo que a função COUNT conte

-- os clientes distintos para cada UF

GROUP BY estados.uf;

-- EC3. Qual o ranking de estados por quantidade de clientes, ou seja, em ordem decrescente de quantidade de clientes?

-- Seleciona a coluna uf e conta o numero de clientes distintos em cada uf

SELECT estados.uf, COUNT(DISTINCT clientes.cod\_cliente)

FROM estados

-- Faz uma junção entre as tabelas estados e cidades usando a coluna uf

-- Relaciona as unidades federativas nas duas tabelas

JOIN cidades ON estados.uf = cidades.uf

-- Faz uma junção, desta vez entre as tabelas cidades e enderecos usando a coluna cod\_cidade

-- Relaciona informações de endereços com as cidades associadas

JOIN enderecos ON cidades.cod\_cidade = enderecos.cod\_cidade

-- Faz uma junção entre as tabelas endereços e clientes\_enderecos usando a coluna cod\_endereco

-- Relaciona os endereços aos clientes

JOIN clientes\_enderecos ON enderecos.cod\_endereco = clientes\_enderecos.cod\_endereco

-- Agrupa os resultados com base nos valores únicos encontrados na coluna uf da tabela estados

GROUP BY estados.uf

-- Ordena os resultados em ordem decrescente com base na contagem de clientes distintos em cada UF

ORDER BY COUNT(DISTINCT clientes.cod\_cliente) DESC;

-- EC4. Qual o ranking de regiões por quantidade de clientes, ou seja, em ordem decrescente de quantidade de clientes?

-- Seleciona a coluna região e conta o número de clientes distintos

SELECT regiao, COUNT(DISTINCT cod\_cliente)

FROM estados

-- Faz uma junção entre as tabelas estados e cidades usando a coluna uf

-- Relaciona as unidades federativas nas duas tabelas

JOIN cidades ON estados.uf = cidades.uf

-- Faz uma junção, desta vez entre as tabelas cidades e enderecos usando a coluna cod\_cidade

-- Relaciona informações de endereços com as cidades associadas

JOIN enderecos ON cidades.cod\_cidade = enderecos.cod\_cidade

-- Faz uma junção entre as tabelas endereços e clientes\_enderecos usando a coluna cod\_endereco

-- Relaciona os endereços aos clientes

JOIN clientes\_enderecos ON enderecos.cod\_endereco = clientes\_enderecos.cod\_endereco

-- Agrupa os resultados com base nos valores únicos encontrados na coluna regiao

GROUP BY regiao

-- Ordena os resultados em ordem decrescente com base na contagem de clientes distintos em cada região

ORDER BY COUNT(DISTINCT cod\_cliente) DESC;

-- EC5. Qual a quantidade de pedidos por região?

-- Seleciona a coluna região e conta o número total de pedidos em cada região

SELECT regiao, COUNT(\*)

FROM estados

-- Faz uma junção entre as tabelas estados e cidades usando a coluna uf

-- Relaciona as unidades federativas nas duas tabelas

JOIN cidades ON estados.uf = cidades.uf

-- Faz uma junção, desta vez entre as tabelas cidades e enderecos usando a coluna cod\_cidade

-- Relaciona informações de endereços com as cidades associadas

JOIN enderecos ON cidades.cod\_cidade = enderecos.cod\_cidade

-- Faz uma junção entre as tabelas endereços e clientes\_enderecos usando a coluna cod\_endereco

-- Relaciona os endereços aos clientes

JOIN clientes\_enderecos ON enderecos.cod\_endereco = clientes\_enderecos.cod\_endereco

-- Faz uma junção entre as tabelas clientes\_endereços e pedidos usando as colunas 'cod\_cliente'

-- e 'cod\_endereco'. Relaciona cada pedido aos seus clientes e endereços correspondentes

JOIN pedidos ON clientes\_enderecos.cod\_cliente = pedidos.cod\_cliente AND clientes\_enderecos.cod\_endereco = pedidos.cod\_endereco

-- Agrupa os resultados com base nos valores únicos encontrados na coluna regiao

GROUP BY regiao;

-- EC6. Qual a quantidade de pedidos por ano e por região, considerando apenas os pedidos feitos nos anos de 2000 até 2004?

-- Extrair o ano da coluna data\_emissao, seleciona a coluna região econta o número total de pedidos

SELECT EXTRACT(YEAR FROM data\_emissao), regiao, COUNT(\*)

FROM estados

-- Faz uma junção entre as tabelas estados e cidades usando a coluna uf

-- Relaciona as unidades federativas nas duas tabelas

JOIN cidades ON estados.uf = cidades.uf

-- Faz uma junção, desta vez entre as tabelas cidades e enderecos usando a coluna cod\_cidade

-- Relaciona informações de endereços com as cidades associadas

JOIN enderecos ON cidades.cod\_cidade = enderecos.cod\_cidade

-- Faz uma junção entre as tabelas endereços e clientes\_enderecos usando a coluna cod\_endereco

-- Relaciona os endereços aos clientes

JOIN clientes\_enderecos ON enderecos.cod\_endereco = clientes\_enderecos.cod\_endereco

-- Faz uma junção entre as tabelas clientes\_endereços e pedidos usando as colunas 'cod\_cliente'

-- e 'cod\_endereco'. Relaciona cada pedido aos seus clientes e endereços correspondentes

JOIN pedidos ON clientes\_enderecos.cod\_cliente = pedidos.cod\_cliente AND clientes\_enderecos.cod\_endereco = pedidos.cod\_endereco

-- Filtra os resultados para incluir apenas os pedidos emitidos entre os anos de 2000 e 2004

WHERE EXTRACT(YEAR FROM data\_emissao) BETWEEN 2000 AND 2004

-- Agrupa os resultados com base nos valores exclusivos da data de emissão no formato de ano

-- e da coluna regiao na tabela estados.

-- Conta quantos pedidos foram emitidos em cada ano e região

GROUP BY EXTRACT(YEAR FROM data\_emissao), regiao

-- Ordena os resultados primeiro por ano em ordem crescente e por região em ordem crescente

ORDER BY EXTRACT(YEAR FROM data\_emissao), regiao;

-- EC7. Qual o valor total gasto por cliente, ordenado em ordem decrescente de valor total?

-- Seleciona a coluna cod\_cliente e calcula a somo total dos produtos para cada cliente

-- da tabela estados

select cod\_cliente, sum(quantidade \* valor\_unitario)

from estados

-- Faz uma junção entre as tabelas cidades e estados usando a coluna 'uf'

-- Relaciona cada estado às suas cidades correspondentes

JOIN cidades ON estados.uf = cidades.uf

-- Faz uma junção entre as tabelas cidades e endereços usando a coluna 'cod\_cidade'

-- Relaciona cada cidade aos seus endereços correspondentes

JOIN enderecos ON cidades.cod\_cidade = enderecos.cod\_cidade

-- Faz uma junção entre as tabelas endereços e clientes\_endereços usando a coluna 'cod\_endereco'

-- Relaciona cada cidade aos seus endereços

JOIN clientes\_enderecos ON enderecos.cod\_endereco = clientes\_enderecos.cod\_endereco

-- Faz uma junção entre as tabelas clientes\_endereços e pedidos usando as colunas 'cod\_cliente'

-- e 'cod\_endereco'. Relaciona cada pedido aos seus clientes e endereços correspondentes

JOIN pedidos ON clientes\_enderecos.cod\_cliente = pedidos.cod\_cliente AND clientes\_enderecos.cod\_endereco = pedidos.cod\_endereco

-- Faz uma junção entre as tabelas pedidos e pedidos\_produtos usando a coluna 'num\_pedido'

-- Relaciona cada pedido aos produtos associados a ele.

JOIN pedidos\_produtos ON pedidos.num\_pedido = pedidos\_produtos.num\_pedido

-- Ordena os resultados em ordem decrescente com base no valor total gasto por cada cliente.

group by cod\_cliente

order by sum(quantidade \* valor\_unitario) desc;

-- EC8. Qual o valor total gasto por cliente, ordenado em ordem decrescente de valor total, considerando apenas os clientes do Rio Grande do Sul?

-- Seleciona a coluna cod\_cliente e p valor total gasto por cada clientes em seus pedidos

select cod\_cliente, sum(quantidade \* valor\_unitario)

from estados

-- Faz uma junção entre as tabelas cidades e estados usando a coluna 'uf'

-- Relaciona cada estado às suas cidades correspondentes

JOIN cidades ON estados.uf = cidades.uf

-- Faz uma junção entre as tabelas cidades e endereços usando a coluna 'cod\_cidade'

-- Relaciona cada cidade aos seus endereços correspondentes

JOIN enderecos ON cidades.cod\_cidade = enderecos.cod\_cidade

-- Faz uma junção entre as tabelas endereços e clientes\_endereços usando a coluna 'cod\_endereco'

-- Relaciona cada cidade aos seus endereços

JOIN clientes\_enderecos ON enderecos.cod\_endereco = clientes\_enderecos.cod\_endereco

-- Faz uma junção entre as tabelas clientes\_endereços e pedidos usando as colunas 'cod\_cliente'

-- e 'cod\_endereco'. Relaciona cada pedido aos seus clientes e endereços correspondentes

JOIN pedidos ON clientes\_enderecos.cod\_cliente = pedidos.cod\_cliente AND clientes\_enderecos.cod\_endereco = pedidos.cod\_endereco

-- Faz uma junção entre as tabelas pedidos e pedidos\_produtos usando a coluna 'num\_pedido'

-- Relaciona cada pedido aos produtos associados a ele.

JOIN pedidos\_produtos ON pedidos.num\_pedido = pedidos\_produtos.num\_pedido

-- Filtra os resultados apenas para clientes que estão no estado do RS.

where uf = 'RS'

-- Agrupa os resultados com base nos valores exclusivos da coluna cod\_cliente da tabela clientes

group by cod\_cliente

-- Ordena os resultados em ordem decrescente com base no valor total gasto por cada cliente

order by sum(quantidade \* valor\_unitario) desc;

-- EC9. Qual o valor total vendido por autor?

-- Seleciona a coluna cod\_autor e calcula o valor total das vendas de produtos associados a cada autor

select cod\_autor, sum( quantidade \* valor\_unitario )

from autores

-- Faz uma junção entre as tabelas autores e autores\_produtos usando a coluna cod\_autor

-- Relaciona cada autor aos produtos associados a ele

JOIN autores\_produtos ON autores.cod\_autor = autores\_produtos.cod\_autor

-- Faz uma junção entre as tabelas autores\_produtos e produtos usando a coluna cod\_produto

-- Relaciona cada produto aos seus autores associados

JOIN produtos ON autores\_produtos.cod\_produto = produtos.cod\_produto

-- Faz uma junção entre as tabelas produtos e pedidos\_produtos usando a coluna cod\_produto

-- Relaciona cada produto às vendas associadas a ele

JOIN pedidos\_produtos ON produtos.cod\_produto = pedidos\_produtos.cod\_produto

-- Agrupar os resultados com base nos valores exclusivos da coluna cod\_autor da tabela autores

group by cod\_autor;

-- EC10. Qual o valor médio faturado com as vendas por produto?

-- Seleciona a coluna cod\_produto e a média do valor total de vendas de cada produto

select cod\_produto, avg( quantidade \* valor\_unitario )

from produtos

-- Faz uma junção entre as tabelas produtos e pedidos\_produtos usando a coluna cod\_produto

-- Relaciona cada produto às vendas associadas a ele

JOIN pedidos\_produtos ON produtos.cod\_produto = pedidos\_produtos.cod\_produto

-- Agrupa os resultados com base nos valores exclusivos da coluna cod\_produto da tabela produtos

group by cod\_produto;

-- EC11. Qual o valor total de cada pedido?

-- Seleciona a coluna num\_pedido e o valor total de cada pedido

select num\_pedido, sum( quantidade \* valor\_unitario )

from pedidos

-- Faz uma junção entre as tabelas pedidos e pedidos\_produtos usando a coluna num\_pedido

-- Relaciona cada pedido aos produtos associados a ele

JOIN pedidos\_produtos ON pedidos.num\_pedido = pedidos\_produtos.num\_pedido

-- Agrupa os resultados com base nos valores exclusivos da coluna num\_pedido da tabela pedidos

group by num\_pedido;

-- EC12. Qual o valor médio dos pedidos por estado?

select uf, sum(quantidade \* valor\_unitario) / count(distinct num\_pedido)

from estados

-- Faz uma junção entre as tabelas cidades e estados usando a coluna 'uf'

-- Relaciona cada estado às suas cidades correspondentes

JOIN cidades ON estados.uf = cidades.uf

-- Faz uma junção entre as tabelas cidades e endereços usando a coluna 'cod\_cidade'

-- Relaciona cada cidade aos seus endereços correspondentes

JOIN enderecos ON cidades.cod\_cidade = enderecos.cod\_cidade

-- Faz uma junção entre as tabelas endereços e clientes\_endereços usando a coluna 'cod\_endereco'

-- Relaciona cada cidade aos seus endereços

JOIN clientes\_enderecos ON enderecos.cod\_endereco = clientes\_enderecos.cod\_endereco

-- Faz uma junção entre as tabelas clientes\_endereços e pedidos usando as colunas 'cod\_cliente'

-- e 'cod\_endereco'. Relaciona cada pedido aos seus clientes e endereços correspondentes

JOIN pedidos ON clientes\_enderecos.cod\_cliente = pedidos.cod\_cliente AND clientes\_enderecos.cod\_endereco = pedidos.cod\_endereco

-- Faz uma junção entre as tabelas pedidos e pedidos\_produtos usando a coluna 'num\_pedido'

-- Relaciona cada pedido aos produtos associados a ele

JOIN pedidos\_produtos ON pedidos.num\_pedido = pedidos\_produtos.num\_pedido

-- Agrupa os resultados com base nos valores exclusivos da coluna uf da tabela estados

group by uf;