

Alunas: Ana Paula Merencia,

Joana Pacheco Rolim e

Milena Lucas dos Santos.

EXERCÍCIO 1 -> Álgebra Relacional e equivalente em SQL.

a) Nome de todos os produtos cadastrados

AR: π nomePro (produto)

SQL: "SELECT nomePro FROM produto;"

b) Nome e Código de todos os produtos cujo precoVenda > 100

AR: π codPro, nomePro (σ precoVenda > 100) (produto)

SQL: "SELECT codPro,nomePro FROM produto WHERE precoVenda > 100;"

c) Código de todos os produtos com Registro de Compra e Venda

AR: π codPro (produto) \cup π codPro (σ produto.codPro = itemcompra.codPro (produto X itemcompra)) \wedge (produto.codPro = itemvenda.codPro (produto X itemvenda))

SQL: "SELECT codPro FROM produto WHERE codPro IN (SELECT codPro FROM produto, itemcompra, itemvenda WHERE produto.codPro = itemcompra.produto_codPro AND produto.codPro = itemvenda.produto_codPro GROUP BY codPro);"

d) Código de todos os produtos sem registro de Venda ainda

AR: π codPro (produto) - π codPro (σ produto.codPro = itemvenda.codPro (produto X itemvenda))

SQL: "SELECT codPro FROM produto WHERE codPro NOT IN (SELECT codPro FROM produto, itemvenda WHERE produto.codPro = itemvenda.produto_codPro GROUP BY codPro);"

e) Código de todos os produtos sem registro de compra e venda

AR: π codPro (produto) - π codPro (σ produto.codPro = itemcompra.codPro (produto X itemcompra)) \wedge (produto.codPro = itemvenda.codPro (produto X itemvenda))

SQL: "SELECT codPro FROM produto WHERE codPro NOT IN (SELECT codPro FROM produto, itemcompra, itemvenda WHERE produto.codPro = itemcompra.produto_codPro AND produto.codPro = itemvenda.produto_codPro GROUP BY codPro);"

f) Nome de todos os clientes cadastrados

AR: π nome (cliente)

SQL: "SELECT nome FROM cliente;"

EXERCÍCIO 2 -> Álgebra Relacional e equivalente em SQL.

a) Nome de todos os produtos que tem venda

AR: π nomePro (σ produto.codPro = itemvenda.codPro (produto X itemvenda))

SQL: "SELECT nomePro FROM produto, itemvenda WHERE produto.codPro = itemvenda.produto_codPro GROUP BY nomePro;"

b) Nome de todos os clientes que tem NotaVenda realizada em 2021

AR: π nome (σ cliente.codCliente = notavenda.codCliente ^ (notavenda.dataVenda > '01/01/2021' ^ notavenda.dataVenda < '31/12/2021') (cliente X notavenda))

SQL: "SELECT nome FROM cliente, notavenda WHERE cliente.codCliente = notavenda.cliente_codCliente AND str_to_date(notavenda.dataVenda, '%Y-%m-%d') between str_to_date('2021-01-01', '%Y-%m-%d') and str_to_date('2021-12-31', '%Y-%m-%d') GROUP BY nome;"

c) Nome de todos os produtos que não tem venda registrada

AR: π nomePro (produto) - π nomePro (σ produto.codPro = itemvenda.codPro (produto X itemvenda))

SQL: "SELECT nomePro FROM produto WHERE nomePro NOT IN (SELECT nomePro FROM produto,itemvenda WHERE produto.codPro = itemvenda.produto_codPro GROUP BY nomePro);"

d) Nome de todos os produtos que tem compra, mas não tem venda registrada

AR: π nomePro (σ produto.codPro = itemcompra.codPro (produto X itemcompra)) - π nomePro (σ produto.codPro = itemvenda.codPro (produto X itemvenda))

SQL: "SELECT nomePro FROM produto,itemcompra WHERE produto.codPro = itemcompra.produto_codPro AND nomePro NOT IN (SELECT nomePro FROM produto,itemvenda WHERE produto.codPro = itemvenda.produto_codPro GROUP BY nomePro);"

e) Nome dos clientes que ainda não tem venda registrada em 2021

AR: π nome (σ cliente.codCliente = notavenda.codCliente ^ notavenda.dataVenda < '2021-01-01' (cliente X notavenda))

SQL: "SELECT nome FROM cliente WHERE nome NOT IN (SELECT nome from cliente,notavenda WHERE cliente.codCliente = notavenda.Cliente_codCliente and str_to_date(notavenda.dataVenda, '%Y-%m-%d') between str_to_date('2021-01-01', '%Y-%m-%d') and str_to_date('2021-12-31', '%Y-%m-%d') GROUP BY nome);"

f) Produto que tem o maior preço de venda cadastrado

AR: π nomePro (produto) - π produto.nome (σ produto.precoVenda < d.precoVenda (produto X (pd (produto))))

SQL: "SELECT nomePro,precoVenda FROM produto WHERE precoVenda = (SELECT max(precoVenda) FROM produto;"

EXERCÍCIO 3 -> Álgebra Relacional e equivalente em SQL.

a) Nome de todos os clientes com profissão informada ou não

AR: π nome, nomeProfissao ((σ profissão.idprofissao = idprofissao) (cliente IXI profissão))

SQL: "SELECT nome, nomeProfissao FROM cliente, profissao WHERE profissao_idprofissão = idprofissão GROUP BY nomeProfissao;"

b) Nome de todos os clientes que possuem profissão informada

AR: π nome, nomeProfissao (((σ profissão.idprofissao = idprofissao) – (π nomeProfissao (σ profissão.nomeProfissao = ' ') (cliente IXI profissão))))

SQL: "SELECT nome, nomeProfissao FROM cliente, profissao WHERE cliente.profissao_idprofissão = profissao.idprofissão AND profissao.nomeProfissao != '' GROUP BY nomeProfissao;"

c) Nome de todos os produtos que tenham registros tanto em ItemVenda como em ItemCompra em março de 2021.

AR: π nomePro (σ produto.codPro = itemvenda.codPro ^ produto.codPro = itemcompra.codPro ^ (nroVenda.itemvenda = nroVenda.notavenda (itemvenda X notavenda)) ^ (notavenda.dataVenda > '01/03/2021' ^ notavenda.dataVenda < '31/03/2021') ^ (notacompra.data > '01/03/2021' ^ notacompra.data < '31/03/2021') (produto IXI notavenda))

SQL: "SELECT nomePro FROM produto, itemvenda, itemcompra, notacompra, notavenda WHERE produto.codPro = itemvenda.produto_codPro AND produto.codPro = itemcompra.produto_codPro AND itemvenda.nroVenda = notavenda.itemvenda_nroVenda AND itemcompra.nroCompra = notacompra.itemcompra_nroCompra AND ((MONTH(data) = '03' AND YEAR(data) = '2021') OR (MONTH(dataVenda) = '03' AND YEAR(dataVenda) = '2021')) GROUP BY nomePro

4- EXTRA – PRODUTO CARTESIANO

notavenda X cliente

SQL: "SELECT * FROM notavenda, cliente;"