Trabalho M2

**Aluno(a)s:** Ana Clara Kniss e Mariana Ferreira

**Explains**

**D)**

**EXPLAIN ANALYZE SELECT Produto.\***

**FROM Produto**

**LEFT JOIN Venda ON Produto.descricao = Venda.produto**

**AND Venda.data >= DATE\_SUB(CURDATE(), INTERVAL 1 MONTH)**

**WHERE Venda.data IS NULL;**

-> Filter: (venda.`data` is null) (cost=11.9 rows=34.8) (actual time=0.27..0.319 rows=5 loops=1)\n

-> Nested loop left join (cost=11.9 rows=34.8) (actual time=0.0908..0.313 rows=24 loops=1)\n

-> Table scan on Produto (cost=1.35 rows=11) (actual time=0.0433..0.0552 rows=11 loops=1)\n

-> Filter: (venda.`data` >= <cache>((curdate() - interval 1 month))) (cost=0.604 rows=3.17) (actual time=0.0132..0.0224 rows=1.73 loops=11)\n

-> Index lookup on Venda using idx\_venda\_produto (produto=produto.descricao) (cost=0.604 rows=3.83) (actual time=0.0114..0.0204 rows=2.09 loops=11)\n'

**E)**

**EXPLAIN ANALYZE SELECT Vendedor.\***

**FROM Vendedor**

**LEFT JOIN Venda ON Vendedor.matricula = Venda.vendedor**

**AND Venda.data >= DATE\_SUB(CURDATE(), INTERVAL 20 DAY)**

**WHERE Venda.data IS NULL;**

**'-> Filter: (venda.`data` is null) (cost=11.9 rows=34.8) (actual time=0.0748..0.289 rows=5 loops=1)\n -> Nested loop left join (cost=11.9 rows=34.8) (actual time=0.0731..0.284 rows=24 loops=1)\n -> Table scan on Vendedor (cost=1.35 rows=11) (actual time=0.0457..0.0556 rows=11 loops=1)\n -> Filter: (venda.`data` >= <cache>((curdate() - interval 20 day))) (cost=0.604 rows=3.17) (actual time=0.014..0.0199 rows=1.73 loops=11)\n -> Index lookup on Venda using idx\_venda\_vendedor (vendedor=vendedor.matricula) (cost=0.604 rows=3.83) (actual time=0.0129..0.0185 rows=2.09 loops=11)\n'**

**H)**

**EXPLAIN ANALYZE SELECT Unidade.cidade, SUM(Venda.quantidade \* Produto.preco) AS valor\_total**

**FROM Venda**

**JOIN Produto ON Venda.produto = Produto.descricao**

**JOIN Unidade ON Venda.unidade = Unidade.cidade**

**WHERE Venda.data BETWEEN DATE\_SUB(CURDATE(), INTERVAL 1 YEAR) AND CURDATE()**

**GROUP BY Unidade.cidade**

**HAVING valor\_total > 100000;**

**'-> Filter: (valor\_total > 100000) (actual time=0.222..0.222 rows=0 loops=1)\n -> Table scan on <temporary> (actual time=0.218..0.218 rows=4 loops=1)\n -> Aggregate using temporary table (actual time=0.217..0.217 rows=4 loops=1)\n -> Nested loop inner join (cost=12.3 rows=14) (actual time=0.0696..0.168 rows=14 loops=1)\n -> Nested loop inner join (cost=7.45 rows=14) (actual time=0.059..0.129 rows=14 loops=1)\n -> Filter: ((venda.`data` between <cache>((curdate() - interval 1 year)) and <cache>(curdate())) and (venda.unidade is not null) and (venda.produto is not null)) (cost=2.55 rows=14) (actual time=0.032..0.0797 rows=14 loops=1)\n -> Table scan on Venda (cost=2.55 rows=23) (actual time=0.0266..0.0659 rows=23 loops=1)\n -> Single-row covering index lookup on Unidade using PRIMARY (cidade=venda.unidade) (cost=0.257 rows=1) (actual time=0.00314..0.00316 rows=1 loops=14)\n -> Single-row index lookup on Produto using PRIMARY (descricao=venda.produto) (cost=0.257 rows=1) (actual time=0.00241..0.00244 rows=1 loops=14)\n'**

**I)**

**EXPLAIN ANALYZE SELECT Produto.descricao, SUM(Venda.quantidade) AS quantidade\_total**

**FROM Produto**

**LEFT JOIN Venda ON Produto.descricao = Venda.produto**

**AND Venda.data BETWEEN DATE\_SUB(CURDATE(), INTERVAL 3 MONTH) AND CURDATE()**

**GROUP BY Produto.descricao**

**HAVING quantidade\_total < 20 OR quantidade\_total IS NULL;**

**'-> Filter: ((quantidade\_total < 20) or (quantidade\_total is null)) (cost=16.1 rows=11) (actual time=0.148..0.254 rows=10 loops=1)\n -> Group aggregate: sum(venda.quantidade) (cost=16.1 rows=11) (actual time=0.122..0.245 rows=11 loops=1)\n -> Nested loop left join (cost=11.9 rows=42.2) (actual time=0.0709..0.223 rows=20 loops=1)\n -> Covering index scan on Produto using idx\_produto\_descricao (cost=1.35 rows=11) (actual time=0.0364..0.0435 rows=11 loops=1)\n -> Filter: (venda.`data` between <cache>((curdate() - interval 3 month)) and <cache>(curdate())) (cost=0.61 rows=3.83) (actual time=0.0103..0.0155 rows=1.27 loops=11)\n -> Index lookup on Venda using idx\_venda\_produto (produto=produto.descricao) (cost=0.61 rows=3.83) (actual time=0.00772..0.0142 rows=2.09 loops=11)\n'**

**J)**

**EXPLAIN ANALYZE SELECT Vendedor.nome, soma FROM (**

**SELECT Venda.vendedor as vendedor, SUM(Venda.quantidade \* Produto.preco) as soma**

**FROM loja.venda**

**JOIN Produto ON Venda.produto = Produto.descricao**

**WHERE Venda.data >= DATE\_SUB(CURDATE(), INTERVAL 1 YEAR)**

**GROUP BY Venda.vendedor**

**) as medias**

**JOIN Vendedor ON medias.vendedor = Vendedor.matricula**

**HAVING soma > (**

**SELECT AVG(soma2) FROM (**

**SELECT SUM(Venda.quantidade \* Produto.preco) as soma2**

**FROM loja.venda**

**JOIN Produto ON Venda.produto = Produto.descricao**

**WHERE Venda.data >= DATE\_SUB(CURDATE(), INTERVAL 1 YEAR)**

**GROUP BY Venda.vendedor**

**) a**

**);**

**'-> Filter: (medias.soma > (select #3)) (cost=6.85 rows=0) (actual time=0.511..0.526 rows=2 loops=1)\n -> Nested loop inner join (cost=6.85 rows=0) (actual time=0.207..0.23 rows=6 loops=1)\n -> Table scan on Vendedor (cost=1.35 rows=11) (actual time=0.025..0.034 rows=11 loops=1)\n -> Index lookup on medias using <auto\_key0> (vendedor=vendedor.matricula) (cost=0.259..0.518 rows=2) (actual time=0.017..0.0172 rows=0.545 loops=11)\n -> Materialize (cost=0..0 rows=0) (actual time=0.175..0.175 rows=6 loops=1)\n -> Table scan on <temporary> (actual time=0.152..0.153 rows=6 loops=1)\n -> Aggregate using temporary table (actual time=0.151..0.151 rows=6 loops=1)\n -> Nested loop inner join (cost=9.2 rows=19) (actual time=0.0304..0.125 rows=19 loops=1)\n -> Filter: ((venda.`data` >= <cache>((curdate() - interval 1 year))) and (venda.produto is not null)) (cost=2.55 rows=19) (actual time=0.012..0.0596 rows=19 loops=1)\n -> Table scan on venda (cost=2.55 rows=23) (actual time=0.0076..0.0475 rows=23 loops=1)\n -> Single-row index lookup on Produto using PRIMARY (descricao=venda.produto) (cost=0.255 rows=1) (actual time=0.00308..0.00313 rows=1 loops=19)\n -> Select #3 (subquery in condition; run only once)\n -> Aggregate: avg(a.soma2) (cost=2.5..2.5 rows=1) (actual time=0.269..0.269 rows=1 loops=1)\n -> Table scan on a (cost=2.5..2.5 rows=0) (actual time=0.264..0.265 rows=6 loops=1)\n -> Materialize (cost=0..0 rows=0) (actual time=0.264..0.264 rows=6 loops=1)\n -> Table scan on <temporary> (actual time=0.255..0.256 rows=6 loops=1)\n -> Aggregate using temporary table (actual time=0.254..0.254 rows=6 loops=1)\n -> Nested loop inner join (cost=9.2 rows=19) (actual time=0.0158..0.225 rows=19 loops=1)\n -> Filter: ((venda.`data` >= <cache>((curdate() - interval 1 year))) and (venda.produto is not null)) (cost=2.55 rows=19) (actual time=0.0099..0.164 rows=19 loops=1)\n -> Table scan on venda (cost=2.55 rows=23) (actual time=0.0073..0.0539 rows=23 loops=1)\n -> Single-row index lookup on Produto using PRIMARY (descricao=venda.produto) (cost=0.255 rows=1) (actual time=0.00277..0.00284 rows=1 loops=19)\n'**