**1 -** Considerando a tabela MC1\_Order, tabela responsável por armazenar todos os pedidos criados no dispositivo Mobile, informar a quantidade de pedidos criados (dCreated) no mês de Abril

SELECT

COUNT(\*) AS QTD\_PED

FROM MC1\_Order PED

WHERE PED.dCreated BETWEEN '01-04-19' AND '30-04-19'

**2 -** Listar a quantidade de pedidos criados e a soma do valor total dos pedidos (nTotalValue) por cliente (cIDCustomer).   
Ex. 12 - 0010011134 - 1200.0000

SELECT DISTINCT(cIDCustomer),

(SELECT

COUNT(PEDSUB1.nTotalValue)

FROM MC1\_Order PEDSUB1

WHERE PEDSUB1.cIDCustomer = PED.cIDCustomer) AS QTD\_ORD, -- QTD DE PEDIDOS

(SELECT

SUM(PEDSUB1.nTotalValue)

FROM MC1\_Order PEDSUB1

WHERE PEDSUB1.cIDCustomer = PED.cIDCustomer) AS SUM\_CUST -- SOMA DO CLIENTE

FROM MC1\_Order PED

**3 -** Considerando a tabela MC1\_Customer, tabela responsável por armazenar todos os clientes no dispositivo Mobile, listar todos os clientes ATIVOS (mc1Enabled) ordenados por ordem alfabética (cCustomerName)

SELECT \* FROM MC1\_Customer

WHERE mc1Enabled = 1

ORDER BY cCustomerName

# 4 - Um analista está trabalhando em um banco de dados e executou uma transação JOIN. Um comando válido para essa operação está indicado na seguinte opção:

1. SELECT tabela1.\* ON tabela1 INNER JOIN tabela2 WHEN tabela1.tb\_id = tabela2.tb\_id WHERE tabela2.tb\_id IS NULL
2. **SELECT tabela1.\* FROM tabela1 INNER JOIN tabela2 ON tabela1.tb\_id = tabela2.tb\_id WHERE tabela2.tb\_id IS NULL**
3. SELECT tabela1.\* ON tabela1 LEFT JOIN tabela2 OVER tabela1.tb\_id = tabela2.tb\_id WITH tabela2.tb\_id IS NULL
4. SELECT tabela1.\* FROM tabela1 INNER JOIN tabela2 ON tabela1.tb\_id = tabela2.tb\_id WHEN tabela2.tb\_id IS NULL
5. SELECT tabela1.\* FROM tabela1 INNER JOIN tabela2 ON tabela1.tb\_id = tabela2.tb\_id IN tabela2.tb\_id IS NULL

**5 -** Listar todos os clientes da tabela MC1\_Customer (Somente Código: cIDCustomer e Nome: cCustomerName) ativos (mc1Enabled) que possuem pedidos criados na tabela de pedidos (MC1\_Order.cIDCustomer)  
Obs. Podendo utilizar sub select, mas de preferência aplicação de JOIN.

SELECT CUS.cIDCustomer,

CUS.cCustomerName

FROM MC1\_Customer CUS, MC1\_Order ORD

WHERE ORD.cIDCustomer = CUS.cIDCustomer

AND CUS.mc1Enabled = 1

**6 -** Considerando que o cliente 0100015070 (cIDCustomer) da tabela MC1\_Customer não faz mais parte da carteira do vendedor, executar um comando para inativá-lo (mc1Enabled), setando o valor para 0.

UPDATE MC1\_Customer

SET mc1Enabled = 0

WHERE cIDCustomer = '0100015070'

# 7 - O DBA criou uma tabela em um Banco de Dados no SQL utilizando o comando

CREATE TABLE mc1

(

[cidade] [varchar](50),

[temp\_baixa] [int],

[temp\_alta] [int],

[nivel\_precip] [float],

[data] [date]

)

Assinale a alternativa correta para inserção de dados na tabela.

1. INSERT IN mc1(cidade, temp\_baixa, temp\_alta, nivel\_precip, data) VALUES ('Brasilia', 20, 34, 0.2, '2009-01-13')
2. INSERT ON mc1(cidade, temp\_baixa, temp\_alta, nivel\_precip, data) VALUES ('Brasilia', 20, 34, 0.2, '2009-01-13')
3. INSERT FROM mc1(cidade, temp\_baixa, temp\_alta, nivel\_precip, data) VALUES ('Brasilia', 20, 34, 0.2, '2009-01-13')
4. INSERT OVER mc1(cidade, temp\_baixa, temp\_alta, nivel\_precip, data) VALUES ('Brasilia', 20, 34, 0.2, '2009-01-13')
5. **INSERT INTO mc1(cidade, temp\_baixa, temp\_alta, nivel\_precip, data) VALUES ('Brasilia', 20, 34, 0.2, '2009-01-13')**

**8 -** Considerando que não temos registro cadastrado para a cidade de São Paulo na tabela de cidades (MC1\_City), inserir o registro com os seguintes valores: (cIDCity: SP 1234567, cIDCountry: 001, cIDState: SP, cIDCompany: 01, cCityName: SAO PAULO, cCityCode: 1234567, mc1Enabled: 1)

INSERT INTO MC1\_City

(cIDCity , cIDCountry , cIDState , cIDCompany , cCityName , cCityCode, mc1Enabled)

VALUES

('SP 12345/67',001,'SP',01,'SAO PAULO',1234567,1)

**9 -** Deletar somente o registro da cidade de LAVRAS (cIDCity: MG 3138203) da tabela de cidades (MC1\_City).

DELETE FROM MC1\_City WHERE cIDCity = 'MG 3138203'