**EXEMPLOS DE CONSULTAS SQL**

* Consulta simples

SELECT \* FROM CELULAR

* Consulta de alguns campos

select cod\_CELULAR, MODELO

FROM CELULAR

* WHERE

SELECT \* FROM CELULAR

WHERE modelo = 'S4'

* BETWEEN (ENTRE)

SELECT \* FROM CELULAR

WHERE cod\_celular BETWEEN 4 AND 6

SELECT \* FROM CELULAR

WHERE cod\_celular >= 4 AND cod\_celular >= 6

* IN (LISTA)

SELECT \* FROM CELULAR

WHERE COD\_CLIENTE IN (1,2,7)

* LIKE

SELECT \* FROM CELULAR

WHERE modelo LIKE 'S%'

SELECT \* FROM CELULAR

WHERE modelo LIKE '%S%'

* Selecione todos os campos da tabela funcionario que tenha a letra A no nome

SELECT \* FROM FUNCIONARIO

WHERE nome\_funcionario LIKE '%A%'

* Selecione todos os campos da tabela funcionario que comecem com a letra A no nome

SELECT \* FROM FUNCIONARIO

WHERE nome\_funcionario LIKE 'A%'

* Selecione todos os campos da tabela funcionario que comecem com C, tenha quaisquer 4 caracteres no meio e termine com S

SELECT \* FROM FUNCIONARIO

WHERE nome\_funcionario LIKE 'CS'

* Selecione todos os campos da tabela funcionario que comecem com C, tenha uma quantidade indefinida de caracteres no meio e termine com S

SELECT \* FROM FUNCIONARIO

WHERE nome\_funcionario LIKE 'C%' AND nome\_funcionario LIKE '%S'

* Busca todos os celulares com cod\_marca nulos

SELECT \* FROM CELULAR

WHERE cod\_marca IS NULL

* Cria novo campo com desconto de 10% no preco\_atual da peça

SELECT \* , preco\_atual \* 0.9 AS NOVO\_PRECO

FROM PECA

* Arredondamento

SELECT \* , ROUND(preco\_atual \* 0.9,2) AS NOVO\_PRECO

FROM PECA

* Adiciona o TEXTO NOME antes de cada nome de cada funcionario

SELECT \*, 'NOME: ' + nome\_funcionario AS NOVO\_NOME

FROM FUNCIONARIO

* Coloca NOME DO FUNCIONARIO em maiusculas

SELECT \* , UPPER(NOME\_FUNCIONARIO)

FROM FUNCIONARIO

* Tira espacos antes do nome

SELECT \*, LTRIM(NOME\_FUNCIONARIO)

FROM FUNCIONARIO

* Tira espacos antes e depois do nome

SELECT \*, RTRIM(LTRIM(NOME\_FUNCIONARIO))

FROM FUNCIONARIO

* Puxa 5 primeiros caracteres do nome do funcionario

SELECT \*, SUBSTRING(NOME\_FUNCIONARIO,1,5)

FROM FUNCIONARIO

* Puxa caracteres do 3 ao 13 do funcionario

SELECT \*, SUBSTRING(NOME\_FUNCIONARIO,3,10)

FROM FUNCIONARIO

* Inverte nome do funcionario

SELECT \*, REVERSE(NOME\_FUNCIONARIO) FROM FUNCIONARIO

* Tamanho do nome do funcionario

SELECT \*, LEN(NOME\_FUNCIONARIO) FROM FUNCIONARIO

* Retorna um inteiro de onde está a letra A no nome do funcionario

SELECT \*, CHARINDEX('A',nome\_funcionario)

FROM FUNCIONARIO

* Retorna um inteiro de onde está a ultima letra A no nome do funcionario

SELECT \*, CHARINDEX('A',REVERSE(NOME\_FUNCIONARIO))

FROM FUNCIONARIO

* Ordena tabela de funcionarios alfabeticamente pelo nome do funcionario

SELECT \* FROM FUNCIONARIO

ORDER BY nome\_funcionario

* Ordena tabela de funcionarios alfabeticamente pelo ativo e depois pelo nome do funcionario

SELECT \* FROM FUNCIONARIO

ORDER BY FUNCIONARIO\_ATIVO, NOME\_FUNCIONARIO

* Ordena tabela de funcionarios ordenando pelo codigo do funcionario descendente

SELECT \* FROM FUNCIONARIO

ORDER BY cod\_funcionario

* Puxa a DATA do servidor

SELECT GETDATE()

* Adiciona 15 dias a data do servidor

SELECT GETDATE() +15

SELECT DATEADD(DD,15,GETDATE())

* Subtrai um ano à data do servidor

SELECT DATEADD(YY,-1,GETDATE())

* Diferença entre datas

SELECT CONVERT(DATETIME,'20150101'), CONVERT(DATETIME,'20150315')

SELECT DATEDIFF(DD,'20150101','20150315')

SELECT DATEDIFF(DD,GETDATE(),'20150101')

SELECT \*, DATEDIFF(DD,DATA\_SERVICO,GETDATE())

FROM SERVICO

* DATANAME

SELECT DATENAME(MM,GETDATE())

SELECT DATENAME(DW,GETDATE())

SELECT \*, DATENAME(MM,DATA\_SERVICO)

FROM SERVICO

* Conversão de datas

SELECT \*, CONVERT(VARCHAR,DATA\_SERVICO,103)

FROM SERVICO

01/03/2014

SELECT \*, CONVERT(VARCHAR,DATA\_SERVICO,108)

FROM SERVICO

08:00:00

SELECT \*, CONVERT(VARCHAR,DATA\_SERVICO,112)

FROM SERVICO

20140112

* Somente parte da data

SELECT \*,DATEPART(HH,DATA\_SERVICO)

FROM SERVICO

* Montando datas

SELECT \*, CONVERT(VARCHAR,DATA\_SERVICO,112) + ' 09:15:15.120'

FROM SERVICO

20140301 09:15:15.120

* Remove duplicadas do campo ativo de funcionarios

SELECT DISTINCT(COD\_FUNCIONARIO) FROM SERVICO

SELECT \* FROM SERVICO

* União de SELECTS

SELECT 'JANEIRO', 1

UNION

SELECT 'FEVEREIRO', 2

SELECT NOME\_FUNCIONARIO FROM FUNCIONARIO

UNION

SELECT NOME\_MARCA FROM MARCA

SELECT DATENAME(MM,GETDATE()-60)

UNION

SELECT DATENAME(MM,GETDATE()-30)

UNION

SELECT DATENAME(MM,GETDATE())

UNION

SELECT DATENAME(MM,GETDATE()+30)

UNION

SELECT DATENAME(MM,GETDATE()+60)

SELECT \*

FROM SERVICO

LEFT JOIN FUNCIONARIO ON

SERVICO.cod\_funcionario = FUNCIONARIO.COD\_FUNCIONARIO

LEFT JOIN CELULAR ON

SERVICO.cod\_celular = CELULAR.COD\_CELULAR

WHERE nome\_funcionario LIKE '%JO%'

SELECT \* FROM CIDADE

LEFT JOIN ESTADO ON CIDADE.COD\_ESTADO = ESTADO.COD\_ESTADO

SELECT \* FROM CLIENTE

LEFT JOIN CIDADE ON CLIENTE.COD\_CIDADE = CIDADE.COD\_CIDADE

LEFT JOIN ESTADO ON CIDADE.COD\_ESTADO = ESTADO.COD\_ESTADO

SELECT CLIENTE.\*, NOME\_CIDADE, NOME\_ESTADO

FROM CLIENTE

LEFT JOIN CIDADE ON CLIENTE.COD\_CIDADE = CIDADE.COD\_CIDADE

LEFT JOIN ESTADO ON CIDADE.COD\_ESTADO = ESTADO.COD\_ESTADO

SELECT \* FROM FUNCIONARIO

* INSERIR NOVO FUNCIONÁRIO POR COMANDO

INSERT INTO FUNCIONARIO

(

nome\_funcionario,

registro,

funcionario\_ativo

)

VALUES

(

'MARIA DA FATEC',

'21215454',

1

)

* Atualizar registro da MARIA, corrigindo o nome

BEGIN TRAN

UPDATE FUNCIONARIO

SET nome\_funcionario = 'MARIA DOS SANTOS',

REGISTRO = 875487

WHERE cod\_funcionario = 16

COMMIT

ROLLBACK

* Atualizando toda a tabela

UPDATE FUNCIONARIO

SET registro = 548787

SELECT \* FROM FUNCIONARIO

* Ativando dois primeiros funcionários

UPDATE FUNCIONARIO

SET FUNCIONARIO\_ATIVO = 0

WHERE cod\_funcionario <=2

* Excluir registro

DELETE FROM FUNCIONARIO

WHERE nome\_funcionario = 'MARIA DOS SANTOS'

SELECT \* FROM FUNCIONARIO

* CONTROL Z

BEGIN TRAN

DELETE FROM SERVICO\_PECA

SELECT \* FROM SERVICO\_PECA

ROLLBACK -- VOLTAR ATRÁS

COMMIT -- ACEITAR AS ALTERAÇÕES FEITAS

SELECT \* FROM SERVICO\_PECA

* Corrigir registros 7 e 8 das ultimas peças vendidas, retirando 5 reais de cada

UPDATE SERVICO\_PECA

SET preco\_vendido = preco\_vendido - 5

WHERE cod\_servico\_peca IN (7,8)

* Ultimas três peças vendidas, exibindo o nome da peça e data da venda em formato DD/MM/AAAA

SELECT TOP 3 \*, CONVERT(VARCHAR,DATA\_SERVICO,103)

FROM SERVICO\_PECA

LEFT JOIN PECA ON SERVICO\_PECA.COD\_PECA = PECA.COD\_PECA

LEFT JOIN SERVICO ON SERVICO\_PECA.COD\_SERVICO = SERVICO.COD\_SERVICO

ORDER BY DATA\_SERVICO DESC

* Lista de todos os funcionários que fizeram vendas para os clientes do estado de São Paulo, exibindo o nome dos funcionários, removendo os duplicados

SELECT DISTINCT NOME\_FUNCIONARIO FROM SERVICO

LEFT JOIN Funcionario ON SERVICO.COD\_FUNCIONARIO = Funcionario.COD\_FUNCIONARIO

LEFT JOIN CELULAR ON CELULAR.COD\_CELULAR = SERVICO.COD\_CELULAR

LEFT JOIN cliente ON CELULAR.COD\_CLIENTE = cliente.COD\_CLIENTE

LEFT JOIN CIDADE ON cliente.COD\_CIDADE = CIDADE.COD\_CIDADE

LEFT JOIN ESTADO ON CIDADE.COD\_ESTADO = ESTADO.COD\_ESTADO

WHERE NOME\_ESTADO = 'São Paulo'

* Peças vendidas para cada clientes, exibindo o nome da peça, nome do cliente e há quantos dias o serviço foi realizado

SELECT NOME\_PECA, NOME\_CLIENTE, DATA\_SERVICO, DATEDIFF(DD,DATA\_SERVICO,GETDATE()), GETDATE()

FROM SERVICO\_PECA

LEFT JOIN SERVICO ON SERVICO\_PECA.COD\_SERVICO = SERVICO.COD\_SERVICO

LEFT JOIN CELULAR ON SERVICO.COD\_CELULAR = CELULAR.COD\_CELULAR

LEFT JOIN cliente ON CELULAR.COD\_CLIENTE = cliente.COD\_CLIENTE

LEFT JOIN PECA ON SERVICO\_PECA.COD\_PECA = PECA.COD\_PECA

* Criação de VIEW com dados de clientes

CREATE VIEW VW\_CLIENTE

AS

SELECT

CLIENTE.\*,

CIDADE.NOME\_CIDADE,

CIDADE.ATIVO\_CIDADE,

CIDADE.COD\_ESTADO,

ESTADO.NOME\_ESTADO,

ESTADO.ATIVO\_ESTADO

FROM CLIENTE

LEFT JOIN CIDADE ON CLIENTE.COD\_CIDADE = CIDADE.COD\_CIDADE

LEFT JOIN ESTADO ON CIDADE.COD\_ESTADO = ESTADO.COD\_ESTADO

* Consulta dados da VIEW

SELECT \* FROM VW\_CLIENTE

* Quantos serviços registrados

SELECT \* FROM SERVICO

SELECT COUNT(\*) FROM SERVICO

* Quantos funcionários cadastrados

SELECT COUNT(\*) FROM FUNCIONARIO

* Quantos funcionários com nome ZÉ

SELECT COUNT(\*) SOMA\_ZE FROM FUNCIONARIO

WHERE nome\_funcionario LIKE '%ZÉ%'

* Média de valor de servico

SELECT \* FROM SERVICO

SELECT AVG(VALOR\_SERVICO) FROM SERVICO

SELECT AVG(VALOR\_SERVICO) FROM SERVICO

WHERE descricao = 'LIMPEZA'

* Servico com valor mais alto

SELECT MAX(VALOR\_SERVICO) FROM SERVICO

* Servico com valor mais baixo

SELECT MIN(VALOR\_SERVICO) FROM SERVICO

* Valor total de servicos vendidos

SELECT SUM(VALOR\_SERVICO) FROM SERVICO

* Média de valor de serviço para celulares do modelo x10

SELECT AVG(VALOR\_SERVICO)

FROM SERVICO

LEFT JOIN CELULAR ON SERVICO.cod\_celular = CELULAR.COD\_CELULAR

WHERE CELULAR.modelo = 'x10'

SELECT \* FROM VW\_SERVICO

* Soma dos valores de todos os serviços

SELECT SUM(VALOR\_SERVICO) FROM VW\_SERVICO

* Soma dos valores de todos os serviços agrupado por funcionário

SELECT SUM(VALOR\_SERVICO), NOME\_FUNCIONARIO

FROM VW\_SERVICO

GROUP BY NOME\_FUNCIONARIO

* Quantos serviços registrados

SELECT COUNT(\*) FROM SERVICO

* Quantos serviços registrados por cliente

SELECT COUNT(\*) , NOME\_CLIENTE

FROM VW\_SERVICO

GROUP BY NOME\_CLIENTE

* Média de valor de servico

SELECT AVG(VALOR\_SERVICO) FROM SERVICO

* Média de valor de servico por cidade

SELECT AVG(VALOR\_SERVICO), NOME\_CIDADE

FROM VW\_SERVICO

GROUP BY NOME\_CIDADE

* Servico com valor mais alto

SELECT MAX(VALOR\_SERVICO) FROM SERVICO

* Servico com valor mais alto por marca e por cidade

SELECT MAX(VALOR\_SERVICO), NOME\_MARCA, NOME\_CIDADE

FROM VW\_SERVICO

GROUP BY NOME\_MARCA, NOME\_CIDADE

* Servico com valor mais baixo

SELECT MIN(VALOR\_SERVICO) FROM SERVICO

* Servico com valor mais baixo por funcionario

SELECT MIN(VALOR\_SERVICO) , NOME\_FUNCIONARIO

FROM VW\_SERVICO

GROUP BY NOME\_FUNCIONARIO

SELECT \* FROM VW\_SERVICO

* Soma dos valores de todos os serviços

SELECT SUM(VALOR\_SERVICO) FROM VW\_SERVICO

* Soma dos valores de todos os serviços agrupado por funcionário

SELECT SUM(VALOR\_SERVICO), NOME\_FUNCIONARIO

FROM VW\_SERVICO

GROUP BY NOME\_FUNCIONARIO

* Quantos serviços registrados

SELECT COUNT(\*) FROM SERVICO

* Quantos serviços registrados por cliente

SELECT COUNT(\*) , NOME\_CLIENTE

FROM VW\_SERVICO

GROUP BY NOME\_CLIENTE

* Média de valor de servico

SELECT AVG(VALOR\_SERVICO) FROM SERVICO

* Média de valor de servico por cidade

SELECT AVG(VALOR\_SERVICO), NOME\_CIDADE

FROM VW\_SERVICO

GROUP BY NOME\_CIDADE

* Servico com valor mais alto por marca e por cidade

SELECT MAX(VALOR\_SERVICO), NOME\_MARCA, NOME\_CIDADE

FROM VW\_SERVICO

GROUP BY NOME\_MARCA, NOME\_CIDADE

* Servico com valor mais baixo por funcionario

SELECT MIN(VALOR\_SERVICO) , NOME\_FUNCIONARIO

FROM VW\_SERVICO

GROUP BY NOME\_FUNCIONARIO

* Faça um SELECT que traga as colunas: primeiro nome de cada funcionário

SELECT \*,

SUBSTRING(NOME\_FUNCIONARIO,1,CHARINDEX(' ',NOME\_FUNCIONARIO)-1)

FROM FUNCIONARIO

* Quantidade de caracteres do nome completo

SELECT \*, LEN(NOME\_FUNCIONARIO) FROM FUNCIONARIO

* Posição da letra A no nome, se existir

SELECT \*, CHARINDEX('A',NOME\_FUNCIONARIO) FROM FUNCIONARIO

* Nome do funcionário com espaços da esquerda e direita cortados

SELECT \*, RTRIM(LTRIM(NOME\_FUNCIONARIO))

FROM FUNCIONARIO

* Crie o login do usuário, considerando o ultimo nome + a primeira letra do primeiro nome.
  + Exemplo: Anne da Silva – Login = silva.a

SELECT \*,

REVERSE(NOME\_FUNCIONARIO),

CHARINDEX(' ',REVERSE(NOME\_FUNCIONARIO)),

SUBSTRING(REVERSE(NOME\_FUNCIONARIO),1,CHARINDEX(' ',REVERSE(NOME\_FUNCIONARIO))),

REVERSE(SUBSTRING(REVERSE(NOME\_FUNCIONARIO),1,CHARINDEX(' ',REVERSE(NOME\_FUNCIONARIO)))),

REVERSE(SUBSTRING(REVERSE(NOME\_FUNCIONARIO),1,CHARINDEX(' ',REVERSE(NOME\_FUNCIONARIO))))+'.',

REVERSE(SUBSTRING(REVERSE(NOME\_FUNCIONARIO),1,CHARINDEX(' ',REVERSE(NOME\_FUNCIONARIO))))+'.'+SUBSTRING(NOME\_FUNCIONARIO,1,1)

FROM FUNCIONARIO

sILVA

* SELECT que traga o preço de cada serviço com um acréscimo de 3%, arredondados para 2 casas decimais, ordenados por CELULAR de forma ascendente

SELECT \*, ROUND(VALOR\_SERVICO \* 1.03,2)

FROM SERVICO

ORDER BY COD\_CELULAR

* Consulta que traga todos os funcionários que atenda a alguma destas condições:
  + Nome comece com a letra A
  + OU Tenha a letra t no nome
  + OU tenha as letras J e A no nome

SELECT \*

FROM FUNCIONARIO

WHERE nome\_funcionario LIKE 'A%'

OR nome\_funcionario LIKE '%T%'

OR nome\_funcionario LIKE '%J%A%'

* Faça uma consulta que traga as colunas:
  + data do serviço em formato dd/mm/aaaa
  + o mês do serviço
  + o ano do serviço
  + a data do serviço no formato yyyymmdd
  + há quantos dias o serviço foi realizado, considerando a data atual do servidor

SELECT \* ,

CONVERT(VARCHAR,DATA\_SERVICO,103) DATA\_SERVICO\_BR,

MONTH(DATA\_SERVICO) MES\_SERVICO,

YEAR(DATA\_SERVICO) ANO\_SERVICO,

CONVERT(VARCHAR,DATA\_SERVICO,112) DATA\_YYYYMMDD,

DATEDIFF(DD,DATA\_SERVICO,GETDATE())

FROM SERVICO

* Faça um select que traga os serviços realizados nos últimos 7 dias.

SELECT \* FROM SERVICO

WHERE DATEDIFF(DD,DATA\_SERVICO,GETDATE()) <= 7

* Faça um SELECT que traga todas as vendas realizadas em qualquer dia, entre às 12:00 e às 18:00

SELECT \*

FROM SERVICO

WHERE DATEPART(HH,DATA\_SERVICO) BETWEEN 12 AND 18

* Faça uma consulta trazendo todos os dados do banco em um único select, exceto os dados de peças, partindo da tabela de serviço

SELECT \*

FROM SERVICO

LEFT JOIN FUNCIONARIO ON SERVICO.COD\_FUNCIONARIO = FUNCIONARIO.COD\_FUNCIONARIO

LEFT JOIN CELULAR ON SERVICO.COD\_CELULAR = CELULAR.COD\_CELULAR

LEFT JOIN MARCA ON CELULAR.cod\_marca = MARCA.cod\_marca

LEFT JOIN CLIENTE ON CELULAR.cod\_cliente = CLIENTE.COD\_CLIENTE

LEFT JOIN CIDADE ON CLIENTE.cod\_cidade = CIDADE.cod\_cidade

LEFT JOIN ESTADO ON ESTADO.cod\_estado = CIDADE.cod\_estado

* Faça uma consulta trazendo os dados de serviços e peças em um único select

SELECT \* FROM SERVICO

LEFT JOIN SERVICO\_PECA ON SERVICO.cod\_servico = SERVICO\_PECA.cod\_servico

LEFT JOIN PECA ON SERVICO\_PECA.cod\_peca = PECA.cod\_peca

* Crie um comando para atualizar em 10% os valores de serviços que contenham a letra A em sua descrição

SELECT \*, valor\_servico \* 1.1

FROM SERVICO

WHERE DESCRICAO LIKE '%A%'

UPDATE SERVICO SET VALOR\_SERVICO = valor\_servico \* 1.1

WHERE DESCRICAO LIKE '%A%'

SELECT \* FROM SERVICO

* Crie um comando para excluir todos os registros de serviços realizados antes de 2010.

SELECT \* FROM SERVICO

WHERE DATA\_SERVICO < '2010-01-01'

DELETE FROM SERVICO

WHERE DATA\_SERVICO < '2010-01-01'

* Faça uma consulta que traga os as peças vendidas e a diferença entre o preço atual e o preço vendido. A consulta deverá ter as colunas: Nome da peça, preço vendido, preço atual, % de variação do preço

SELECT NOME\_PECA,

PRECO\_ATUAL,

PRECO\_VENDIDO,

preco\_VENDIDO\*100 / PRECO\_ATUAL VARIACAO

FROM SERVICO

LEFT JOIN SERVICO\_PECA ON SERVICO.COD\_SERVICO = SERVICO\_PECA.COD\_SERVICO

LEFT JOIN PECA ON PECA.COD\_PECA = SERVICO\_PECA.COD\_PECA

WHERE NOME\_PECA IS NOT NULL

* Faça uma consulta que traga: todas as peças vendidas para os estados de SP e MG e para as marcas Samsung e Apple no mês de janeiro de 2014 exibindo ainda há quantos dias cada peça foi vendida

SELECT nome\_peca, DATEDIFF(DD,DATA\_SERVICO,GETDATE())

FROM SERVICO

LEFT JOIN FUNCIONARIO ON SERVICO.cod\_funcionario = FUNCIONARIO.cod\_funcionario

LEFT JOIN CELULAR ON SERVICO.cod\_celular = CELULAR.COD\_CELULAR

LEFT JOIN MARCA ON CELULAR.cod\_marca = MARCA.COD\_MARCA

LEFT JOIN CLIENTE ON CLIENTE.COD\_CLIENTE = CELULAR.cod\_cliente

LEFT JOIN CIDADE ON CLIENTE.cod\_cidade = CIDADE.COD\_CIDADE

LEFT JOIN ESTADO ON CIDADE.cod\_estado = ESTADO.COD\_eSTADO

LEFT JOIN SERVICO\_PECA ON SERVICO\_PECA.cod\_servico = SERVICO.COD\_SERVICO

LEFT JOIN PECA ON SERVICO\_PECA.cod\_peca = PECA.COD\_PECA

WHERE nome\_estado IN ('São Paulo','Minas Gerais')

and nome\_marca in ('Samsung','Apple')

and MONTH(data\_servico) = 01 and YEAR(data\_servico) = 2014

* Faça uma consulta que traga: Serviços que ainda estão na garantia exibindo quantos dias ainda tem de garantia, cliente e marca

SELECT DESCRICAO,

NOME\_CLIENTE,

NOME\_MARCA,

DATA\_SERVICO,

GARANTIA,

DATA\_SERVICO + GARANTIA,

GETDATE()

FROM SERVICO

LEFT JOIN CELULAR ON SERVICO.COD\_CELULAR = CELULAR.COD\_CELULAR

LEFT JOIN MARCA ON CELULAR.COD\_MARCA = MARCA.COD\_MARCA

LEFT JOIN CLIENTE ON CELULAR.COD\_CLIENTE = CLIENTE.COD\_CLIENTE

WHERE DATA\_SERVICO + GARANTIA >= GETDATE()

* Crie uma VIEW com todos os dados do banco, exceto as tabelas serviço\_peca e peças.

CREATE VIEW [DBO].[VW\_SERVICO]

AS

SELECT

SERVICO.\*,

FUNCIONARIO.NOME\_FUNCIONARIO,

FUNCIONARIO.REGISTRO,

FUNCIONARIO.FUNCIONARIO\_ATIVO,

CELULAR.MODELO,

CELULAR.COD\_MARCA,

CELULAR.NUMERO\_SERIE,

CELULAR.COD\_CLIENTE,

CELULAR.CELULAR\_ATIVO,

MARCA.NOME\_MARCA,

MARCA.MARCA\_ATIVO,

CLIENTE.NOME\_CLIENTE,

CLIENTE.ENDERECO,

CLIENTE.FONE,

CLIENTE.CPF,

CLIENTE.CLIENTE\_ATIVO,

CLIENTE.COD\_CIDADE,

CIDADE.NOME\_CIDADE,

CIDADE.ATIVO\_CIDADE,

CIDADE.COD\_ESTADO,

ESTADO.NOME\_ESTADO,

ESTADO.ATIVO\_ESTADO

FROM SERVICO

LEFT JOIN FUNCIONARIO ON SERVICO.COD\_FUNCIONARIO = FUNCIONARIO.COD\_FUNCIONARIO

LEFT JOIN CELULAR ON SERVICO.COD\_CELULAR = CELULAR.COD\_CELULAR

LEFT JOIN MARCA ON CELULAR.COD\_MARCA = MARCA.COD\_MARCA

LEFT JOIN CLIENTE ON CLIENTE.COD\_CLIENTE = CELULAR.COD\_CLIENTE

LEFT JOIN CIDADE ON CLIENTE.COD\_CIDADE = CIDADE.COD\_CIDADE

LEFT JOIN ESTADO ON CIDADE.COD\_ESTADO = ESTADO.COD\_ESTADO

SELECT \* FROM VW\_SERVICO

* Faça SELECTs para trazer: quantidade de funcionários ativos

SELECT COUNT(\*) FROM FUNCIONARIO

WHERE funcionario\_ativo = 1

* Quantidade de celulares da marca samsung

SELECT COUNT(\*) FROM CELULAR

LEFT JOIN MARCA ON CELULAR.cod\_marca = MARCA.cod\_marca

WHERE nome\_marca = 'SAMSUNG'

* Quantidade de serviços realizados para a cidade de Jundiaí

SELECT COUNT(\*) FROM vw\_servico

WHERE nome\_cidade = 'JUNDIAI'

* Soma do valor de todos os serviços em 2014

SELECT SUM(valor\_servico) FROM VW\_SERVICO

WHERE YEAR(DATA\_SERVICO) = 2014

* Primeiro serviço realizado pelo funcionário 1

SELECT MIN(DATA\_SERVICO) FROM VW\_SERVICO

WHERE cod\_funcionario = 1

* Ultimo serviço realizado para o cliente 2

SELECT MAX(DATA\_SERVICO) FROM vw\_servico

WHERE cod\_cliente = 2

* Média de preço dos serviços de limpeza realizados em 2013 para clientes do estado de São Paulo

SELECT COUNT(\*) FROM VW\_SERVICO

WHERE YEAR(DATA\_SERVICO) = 2013

AND DESCRICAO = 'LIMPEZA'

AND NOME\_ESTADO = 'São Paulo'

* Faça SELECTs para trazer: quantidade de celulares de cada marca

SELECT COUNT(\*), NOME\_MARCA FROM VW\_SERVICO

GROUP BY NOME\_MARCA

* Quantidade de serviços realizados para cada cidade

SELECT COUNT(\*), NOME\_CIDADE

FROM VW\_SERVICO

GROUP BY NOME\_CIDADE

* Soma do valor de todos os serviços por mês e ano

SELECT SUM(VALOR\_SERVICO), MONTH(DATA\_SERVICO) MES, YEAR(DATA\_SERVICO) ANO

FROM VW\_SERVICO

GROUP BY MONTH(DATA\_SERVICO), YEAR(DATA\_SERVICO)

* Primeiro serviço realizado por cada funcionário

SELECT MIN(DATA\_SERVICO), NOME\_FUNCIONARIO

FROM VW\_SERVICO

GROUP BY NOME\_FUNCIONARIO

* Último serviço realizado para cada cliente

SELECT MAX(DATA\_SERVICO), NOME\_CLIENTE

FROM VW\_SERVICO

GROUP BY NOME\_CLIENTE

* Média de preço dos serviços de limpeza realizados em 2013 por estado

SELECT AVG(valor\_servico), NOME\_ESTADO

FROM VW\_SERVICO

WHERE descricao LIKE '%LIMPEZA%' AND YEAR(DATA\_SERVICO) = 2013

GROUP BY NOME\_ESTADO