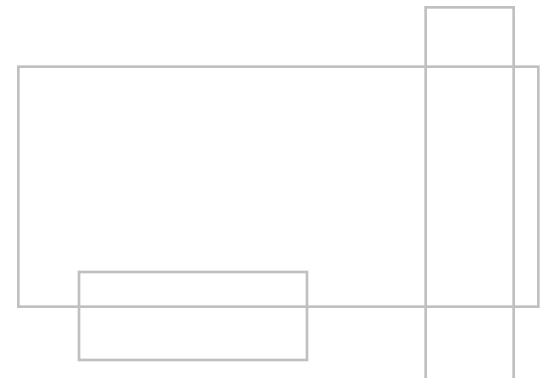




Banco de Dados

Introdução ao SQL

João Marcelo Borovina Josko



- **O MODELO RELACIONAL**
- **SQL E SEUS COMPONENTES**
- **SQL-DML**
 - **SELECT BÁSICO e RESTRIÇÃO DE LINHAS**
 - **CLASSIFICAÇÃO E AGREGAÇÃO SIMPLES**
 - **AGRUPAMENTO DE DADOS**
 - **FILTRANDO AGRUPAMENTOS e NULOS**
 - **JUNÇÃO INTERNA e JUNÇÃO EXTERNA**
 - **SUBCONSULTAS**
 - **OPERADORES DE CONJUNTO**
 - **INCLUSÃO ou MANUTENÇÃO DE INSTÂNCIAS DE DADOS**
- **Objetos Programados**

O MODELO RELACIONAL

MODELO RELACIONAL

COMPOSIÇÃO DE UM BANCO DE DADOS RELACIONAL

- BANCO DE DADOS FORMADOS POR “TABELAS” (*Naive notion*)

Coluna (atributo)

nome do campo (nome do atributo)

Emp

CódigoEmp	Nome	CodigoDepto	CategFuncional
E5	Souza	D1	C5
E3	Santos	D2	C5
E2	Silva	D1	C2
E1	Soares	D1	—

valor do campo (valor do atributo)

linha (tupla)

Fonte: HEUSER, Carlos A Projeto de banco de Dados

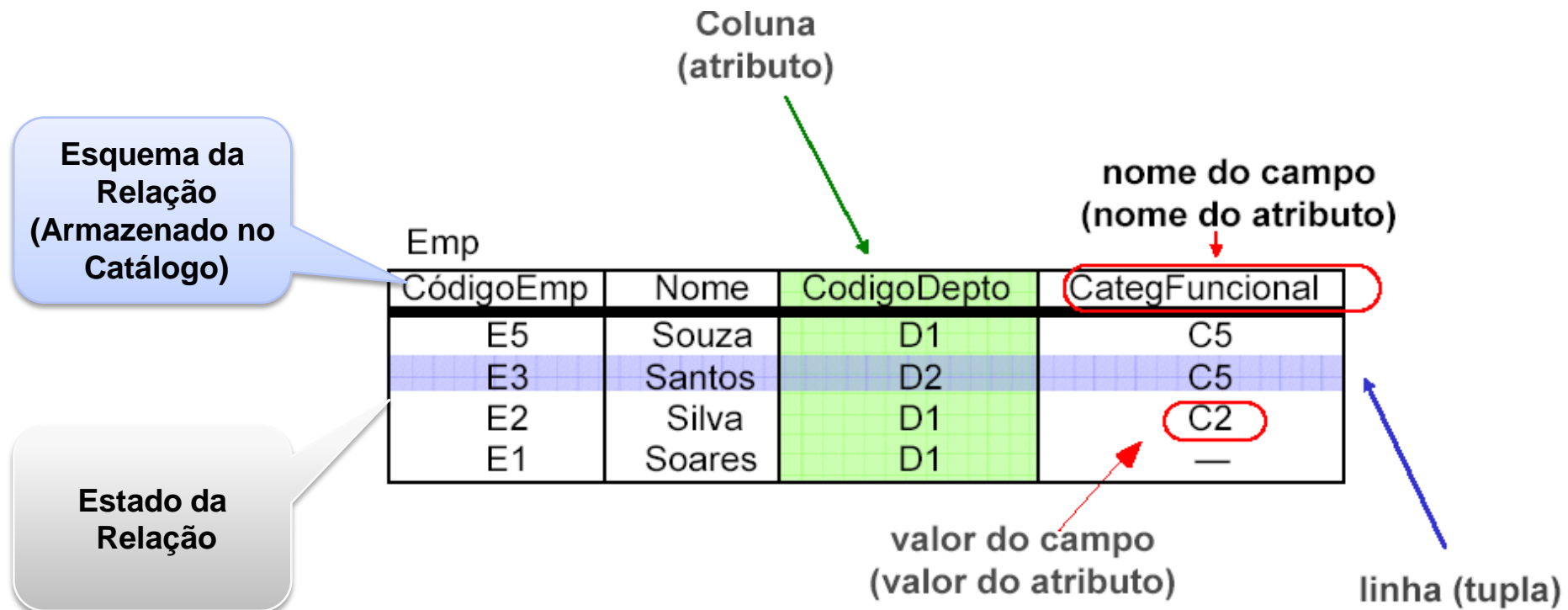
MODELO RELACIONAL

COMPOSIÇÃO DE UM BANCO DE DADOS RELACIONAL

- BANCO DE DADOS CONTÉM UMA COLEÇÃO DE RELAÇÕES N-ÁRIAS
 - A RELAÇÃO É UM CONCEITO MATEMÁTICO QUE DENOTA “CONDIÇÃO” DE LIGAÇÃO ENTRE ELEMENTOS DE DIFERENTES CONJUNTOS
- ESQUEMA DA RELAÇÃO
 - DEFINE UM TIPO DE ASSERÇÃO ONDE CADA TUPLA REPRESENTA UM FATO DA MESMA
 - Conjunto de k atributos $A = \{a_1, a_2, \dots, a_k\}$, onde cada a_s , $s = [1, k]$, é regulado por um domínio D_s
- ESTADO DA RELAÇÃO
 - REFLETE OS VALORES DE DADOS PRESENTES EM UMA RELAÇÃO EM UM DETERMINADO INSTANTE DO TEMPO
 - Conjunto de n tuplas, $r = \{t_1, t_2, \dots, t_n\}$, onde cada t_p , $p = [1, n]$, é uma lista de q valores $t_p = \{v_1, v_2, \dots, v_q\}$

MODELO RELACIONAL

COMPOSIÇÃO DE UM BANCO DE DADOS RELACIONAL



MODELO RELACIONAL

COMPOSIÇÃO DE UM BANCO DE DADOS RELACIONAL

- **SUAS LINHAS NÃO APRESENTAM ORDENAÇÃO**
 - MATEMATICAMENTE, RELAÇÕES NÃO APRESENTAM ORDEM
- **QUALQUER ATRIBUTO PODE SER UTILIZADA EM PESQUISAS**
- **CADA VALOR EM UMA TUPLA É ATÔMICO (NÃO DIVISÍVEL) E MONOVALORADO**
- **CHAVES**
 - MECANISMO QUE IDENTIFICA TUPLA E ESTABELECE RELACIONAMENTOS ENTRE RELAÇÕES
 - DOIS TIPOS PRINCIPAIS:
 - CHAVE PRIMÁRIA
 - CHAVE ESTRANGEIRA

MODELO RELACIONAL

COMPOSIÇÃO DE UM BANCO DE DADOS RELACIONAL

- **CHAVES PRIMÁRIA**

- **COLUNA OU CONJUNTO DE ATRIBUTOS CUJOS VALORES IDENTIFICAM UNIVOCAMENTE CADA TUPLA**
- **TODA RELAÇÃO DEVE TER UMA CHAVE PRIMÁRIA**

Dependente

CódigoEmp	NoDepen	Nome	Tipo	DataNasc
E1	01	João	Filho	12/12/91
E1	02	Maria	Esposa	01/01/50
E2	01	Ana	Esposa	05/11/55
E6	01	Paula	Esposa	04/07/60
E6	02	José	Filho	03/02/85

HEUSER, Carlos A Projeto de banco de Dados

MODELO RELACIONAL

COMPOSIÇÃO DE UM BANCO DE DADOS RELACIONAL

- **CHAVES ESTRANGEIRA**

- **COLUNA OU CONJUNTO DE ATRIBUTOS CUJOS VALORES APARECEM NECESSARIAMENTE COMO CHAVE PRIMÁRIA NA RELAÇÃO REFERENCIADA**
- **MECANISMO QUE PERMITE A IMPLEMENTAÇÃO DE RELACIONAMENTOS ENTRE AS RELAÇÕES**

DEPTO

CodigoDepto	NomeDepto
D1	Compras
D2	Engenharia
D3	Vendas

CodigoDepto em EMP é uma chave estrangeira em relação a tabela DEPTO

EMP

CodigoEmp	Nome	CodigoDepto	CategFuncional	CIC
E1	Souza	D1	-	132.121.331-20
E2	Santos	D2	C5	891.221.111-11
E3	Silva	D2	C5	341.511.775-45
E5	Soares	D1	C2	631.692.754-88

HEUSER, Carlos A Projeto de banco de Dados

MODELO RELACIONAL

COMPOSIÇÃO DE UM BANCO DE DADOS RELACIONAL

- **VALIDAÇÕES GARANTIDAS POR CHAVES ESTRANGEIRAS**
 - **VALIDAÇÃO DE INCLUSÃO**
 - GARANTIR QUE O VALOR DA CHAVE ESTRANGEIRA DEVE APARECER NA RESPECTIVA COLUNA DA CHAVE PRIMÁRIA REFERENCIADA
 - **VALIDAÇÃO DE ALTERAÇÃO**
 - GARANTIR QUE O NOVO VALOR DA CHAVE ESTRANGEIRA DEVE APARECER NA RESPECTIVA COLUNA DA CHAVE PRIMÁRIA REFERENCIADA
 - **VALIDAÇÃO DE ELIMINAÇÃO**
 - GARANTIR QUE O NOVO NA CHAVE ESTRANGEIRA NÃO DEVE APARECER O VALOR DA CHAVE PRIMÁRIA A SER ELIMINADA

MODELO RELACIONAL

COMPOSIÇÃO DE UM BANCO DE DADOS RELACIONAL

- **RESTRIÇÕES QUE DEFINEM O ESTADO VÁLIDO DO BD**
 1. **INERENTES A ABORDAGEM RELACIONAL**
 2. **EXPRESSAS NO ESQUEMA DE DADOS**
 - RESTRIÇÕES DE DOMÍNIO (INCLUSIVE PERMISSÃO DE NULOS)
 - RESTRIÇÕES DE UNICIDADE
 - RESTRIÇÕES DE INTEGRIDADE REFERENCIAL
- **RESTRIÇÕES DE TRANSIÇÃO DE ESTADO**
 1. **EXPRESSAS EM PROGRAMAS DE APLICAÇÃO ou OBJETOS PROGRAMADOS**
 - RESTRIÇÕES DE ORDEM SEMÂNTICA

LINGUAGENS de CONSULTA do BANCO DE DADOS

- MEIO DE REQUISITAR ALGO AO BANCO DE DADOS
- CATEGORIZADAS EM:
 - PROCEDURAIS
 - INSTRUIR O SISTEMA POR UMA SÉRIE DE OPERAÇÕES PARA PRODUZIR ALGO
 - *Álgebra e Cálculo Relacional*
 - NÃO PROCEDURAIS
 - DESCRIVER QUAIS DADOS SÃO DESEJADOS E NÃO COMO OBTÊ-LOS
 - *SQL e QBE (Query by Example)*

SQL e SEUS COMPONENTES

SQL E SEUS COMPONENTES

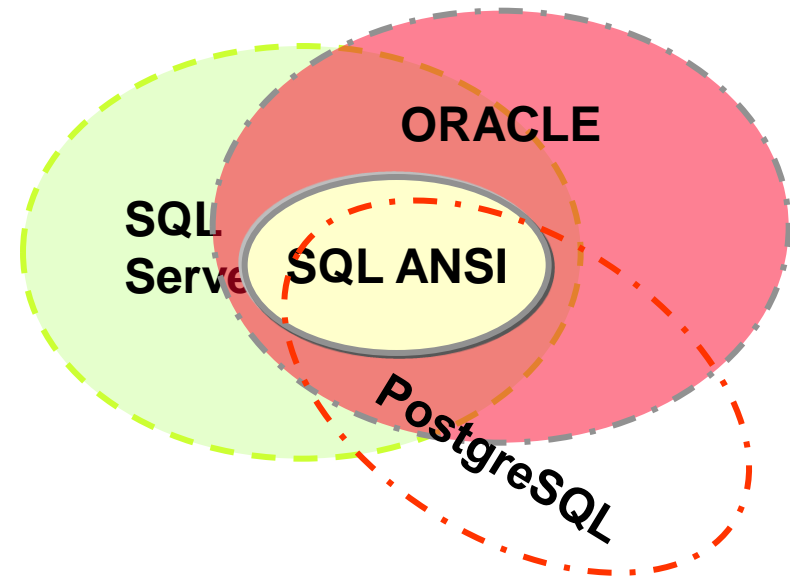
O QUE É SQL - STRUCTURED QUERY LANGUAGE?

- **LINGUAGEM COMERCIAL PARA DEFINIÇÃO E MANIPULAÇÃO DE BANCO DE DADOS RELACIONAIS**
- **LINGUAGEM NÃO PROCEDURAL**
 - SURTIU NA IBM NO PROJETO SYSTEM R
- **COMANDOS NÃO SÃO *CASE SENSITIVE***
 - PODEM CONTER MAIÚSCULOS E MINÚSCULOS
- **PODEM TER MAIS DO QUE UMA LINHA**
 - NO ORACLE UM SENTENÇA É ENCERRADA COM “.”
- **PALAVRAS CHAVES NÃO PODEM SER ABREVIADAS**

SQL E SEUS COMPONENTES

CARACTERÍSTICAS GERAIS

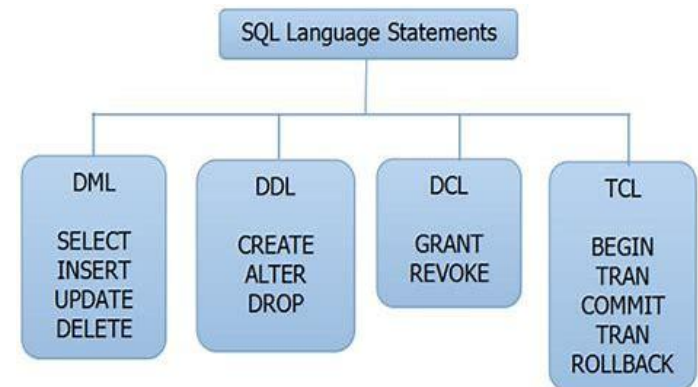
- É UM PADRÃO CONTROLADO PELA ISO/ANSI
- VERSÕES:
 - SQL/89, SQL/92, SQL/99, SQL/04, SQL/08, SQL/11
- MAS.....
 - CADA PRODUTO de SGBD APRESENTA EXTENSÕES AO PADRÃO ISO/ANSI
 - SENTENÇAS SQL QUE UTILIZAM TAIS EXTENSÕES NÃO SÃO PORTÁVEIS



SQL E SEUS COMPONENTES

USOS ou FUNÇÕES DO SQL

- **MANUTENÇÃO DE OBJETOS DE BANCO DE DADOS**
(DDL - DATA DEFINITION LANGUAGE)
- **MANIPULAÇÃO DOS DADOS EM BANCOS DE DADOS**
(DML - DATA MANIPULATION LANGUAGE)
- **MANUTENÇÃO DE AUTORIZAÇÃO DE ACESSO**
(DCL - DATA CONTROL LANGUAGE)
- **CONTROLE DE TRANSAÇÕES E CONCORRÊNCIA**
(TCL – TRANSACTION CONTROL LANGUAGE)

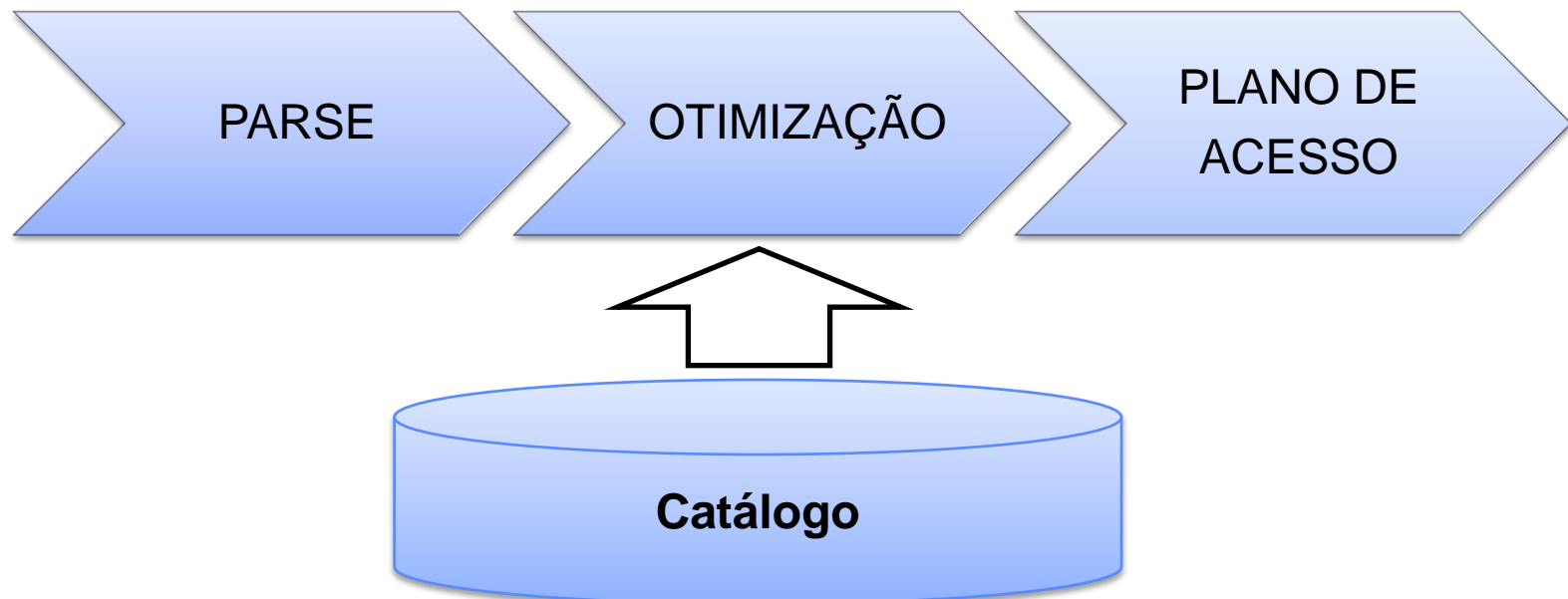


Processamento de Consultas – Visão do Espaço!

PROCESSAMENTO DE CONSULTAS

VISÃO GERAL

- Atividades envolvidas na extração de dados via SQL
 - Tradução da linguagem DECLARATIVA (O que) para PROCEDIMENTAL (Como)
- Envolve vários passos de TRADUÇÃO:



SQL-DML (DATA MANIPULATION LANGUAGE)

SQL - DML (DATA MANIPULATION LANGUAGE)

ELEMENTOS DA LINGUAGEM

- ***SELECT:***

PERMITE SELECIONAR DADOS DE UMA OU MAIS RELAÇÕES

- ***UPDATE:***

PERMITE ATUALIZAR UMA OU MAIS TUPLAS DE UMA RELAÇÃO

- ***DELETE:***

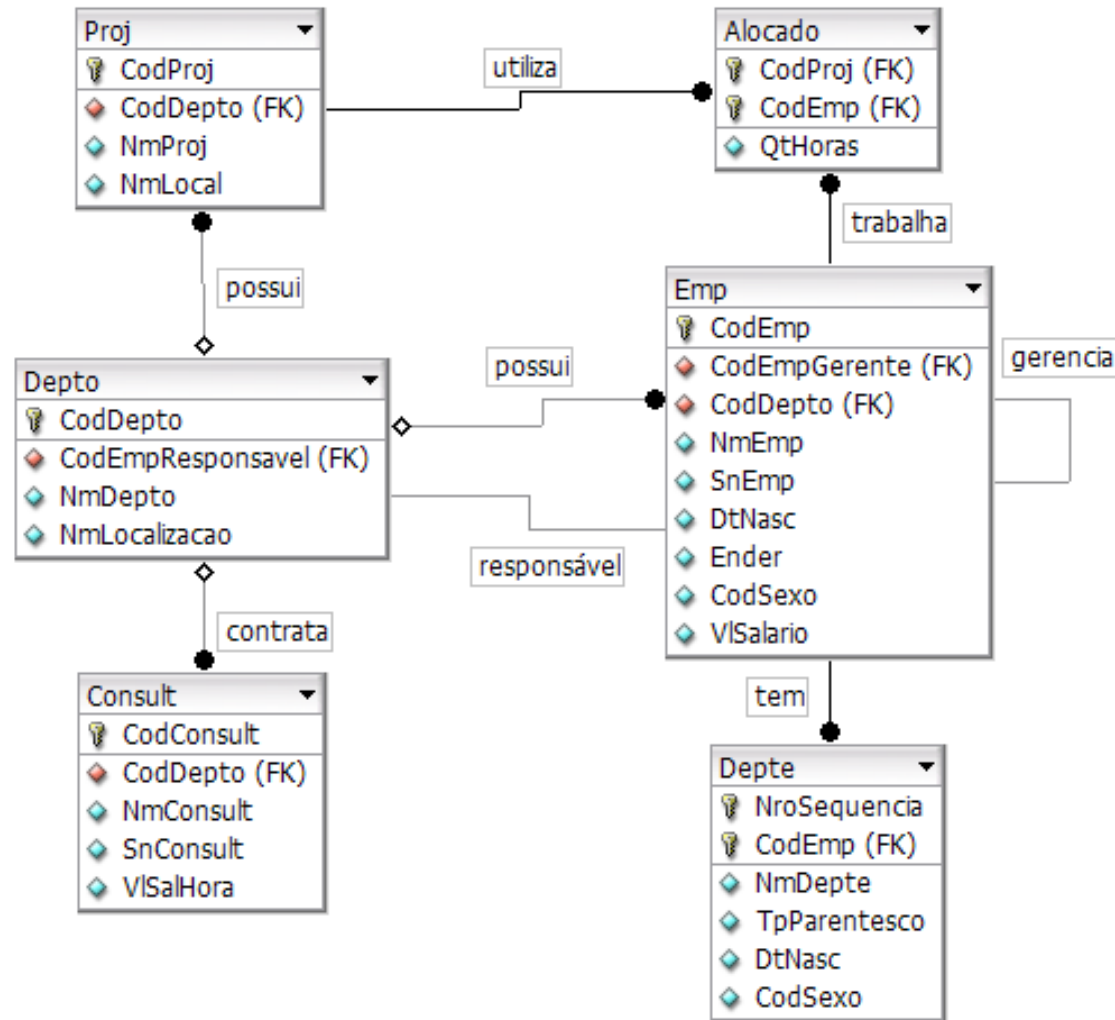
PERMITE ELIMINAR UMA OU MAIS TUPLAS DE UMA RELAÇÃO

- ***INSERT:***

PERMITE ACRESCENTAR UMA OU MAIS TUPLAS EM UMA RELAÇÃO

SQL - DML (DATA MANIPULATION LANGUAGE)

BANCO DE DADOS DE REFERÊNCIA



SQL - DML (DATA MANIPULATION LANGUAGE)

BANCO DE DADOS DE REFERÊNCIA

Emp

CodEmp	NmEmp	SnEmp	DtNasc	Ender	CodSexo	VISalario	CodDepto	CodEmpGerente
1	Jose	Santos	10/01/1961	R. Prof. Sebatião, 15	M	5000	1	13
2	Mariana	Campos	22/09/1973	R. Cerro Corá, 190	F	4500	1	13
13	Gilberto	Vurg	22/01/1960	R. Prof. Faustino, 111	M	8000	1	55
16	Fabiana	Shultz	30/03/1977	Av. Ibirapuera, 91	F	4500	3	29
17	Daniela	Smith	10/12/1980	Av. Ibirapuera, 1200	F	2600	3	29
29	Tereza	Kohn	22/04/1963	R. Prof. Mario Sá, 198	F	7500	3	55
39	Ana Paula	Borges	19/01/1976	R. Prof. Isídio Cunha, 1	F	4000	2	55
55	Paulo	Shazan	19/04/1952	R. Prof. Castilho, 2003	M	21000	2	<i>null</i>

Alocado

CodProj	CodEmp	QtHoras
2	13	80
2	1	135
4	13	60
4	2	160
4	1	50
1	55	40
1	30	165
1	13	25
3	55	30
4	29	15
5	29	70
5	16	160
5	17	125
3	17	40

Depto

CodDepto	NmDepto	NmLocalização	CodEmpResponsavel
1	Pesquisa	São Paulo	13
2	Administração	Campinas	55
3	Financeiro	Campinas	29

SQL - DML (DATA MANIPULATION LANGUAGE)

BANCO DE DADOS DE REFERÊNCIA

Depte

CodEmp	NroSequencia	NmDepte	TpParentesco	DtNasc	CodSexo
1	1	Ramiro Santos	filho	22/01/1988	M
1	2	Samuel Santos	filho	01/09/1990	M
13	3	Carla Vurg	Esposa	09/09/1961	F
29	4	Gabriela Kohn	Filha	11/02/1982	F
29	5	Ricardo Kohn	Filho	31/01/1980	M

Proj

CodProj	NmLocal	NmProj	CodDepto
1	SP	Reengenharia	2
2	SP	Produto X	1
3	MG	Reengenharia	2
4	SP	Produto Y	1
5	SP	Automatização	3

Consult

CodConsult	NmConsult	SnConsult	VISalHora	CodDepto
1	Carlos	Figueroa	32	2
2	Sebastião	Sá	NULL	2
3	Cristina	Fernandes	28	2
4	Humberto	Osvaldo	31	3
5	Gustavo	Bianco	NULL	3
6	Paula	Gilt	41	3
7	Fernanda	Dantas	NULL	3
8	Francisca	da Silva	33	2

SQL - DML (DATA MANIPULATION LANGUAGE)

CAPACIDADES DO SELECT

Projeção

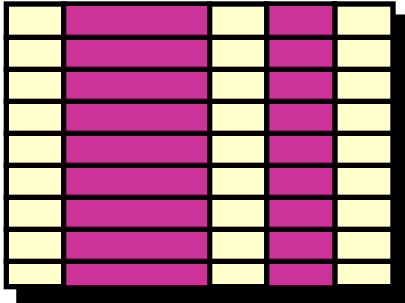


Tabela 1

Seleção

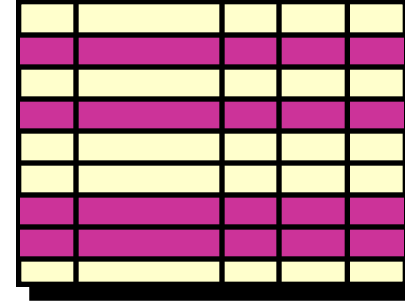


Tabela 1

Junção

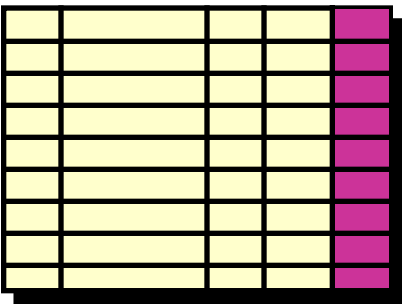


Tabela 1

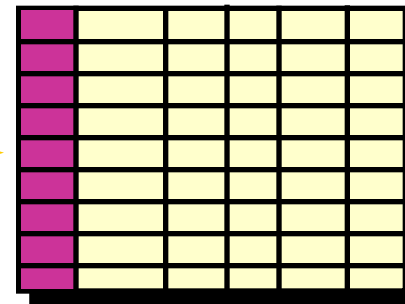


Tabela 2

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - SELECIONANDO TODAS AS COLUNAS

- O ASTERISCO (*) INDICA A RECUPERAÇÃO DE TODAS AS COLUNAS DAS TABELAS ENVOLVIDAS NO *FROM*
- AUSÊNCIA DA CLÁUSULA *WHERE* INDICA LEITURA DE TODO CONTEÚDO DAS TABELAS ENVOLVIDAS NO *FROM*

```
SELECT * FROM DEPTO
```

CODDEPTO	CODEMPRESPONSAVEL	NMDEPTO	NMLOCALIZACAO
1	13	pesquisa	sao paulo
2	55	administracao	campinas
3	29	financeiro	campinas

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - APLICANDO A PROJEÇÃO

- **INFORMA-SE OS NOMES DAS COLUNAS PARA RESTRINGIR QUAIS ESTARÃO DISPONÍVEIS NA RELAÇÃO RESULTANTE**

```
Select nmemp, snemp, dtnasc from Emp
```

NMEMP	SNEMP	DTNASC
PÁulo	Shazan	19/4/1952
Ana	Paula	19/1/1976
Tereza	Kohn	22/4/1963
Gilberto	Vurg	22/1/1960
Daniela	Smith	10/12/1980
Fabiana	Schultz	30/3/1977
Mariana	Campos	22/9/1973
Jose	Santos	10/1/1961

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - EXPRESSÕES ARITMÉTICAS


- **CRIE EXPRESSÕES SOBRE NÚMEROS E DATA**

Operador	Descrição
+	Soma
-	Subtração
*	Multiplicação
/	Divisão

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - EXPRESSÕES ARITMÉTICAS

```
select  NmEmp, SnEmp, vlsalario / 160 from Emp
```



NMEMP	SNEMP	VLHORA
Paulo	Shazan	131,25
Ana	Paula	25
Tereza	Kohn	46,875
Gilberto	Vurg	50
Daniela	Smith	16,25
Fabiana	Schultz	28,125
Mariana	Campos	28,125
Jose	Santos	31,25

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - SINÔNIMOS PARA COLUNAS E TABELAS

- **INDICAÇÃO DE UMA NOVA IDENTIFICAÇÃO OU NOME PARA UMA COLUNA E/OU TABELA**
- **RECURSO IMPORTANTE QUANDO:**
 - **NECESSÁRIO EFETUAR UMA JUNÇÃO ENTRE A MESMA TABELA (AUTO-RELACIONAMENTO);**
 - **DENOMINAÇÃO CURTA DA TABELA EM EXPRESSÕES SQL**
 - **IDENTIFICAR DE QUAL TABELA PERTENCE UM COLUNA CUJO NOME É ENCONTRADO EM MÚLTIPLAS TABELAS**
 - **RENOMEAR O NOME DA COLUNA**

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - SINÔNIMOS PARA COLUNAS E TABELAS

```
SELECT NMEMP || ' ' || SNEMP Empregado FROM EMP;
```

Empregado	
Ana Paula	
Tereza Kohn	
Gilberto Vurg	
Daniela Smith	
Fabiana Schultz	
Jose Santos	
Mariana Campos	
PAulo Shazan	

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - USO DO DISTINCT

- O RETORNO DO SELECT APRESENTA TODAS AS LINHAS, INCLUSIVE AS DUPLICADAS

```
SELECT CODSEXO FROM DEPTE;
```

Codsexo
m
m
f
f
m

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - USO DO DISTINCT

- OPERADOR *DISTINCT*

- RETORNA OS VALORES DIFERENTES DA COLUNA OU COLUNAS INFORMADAS

```
SELECT DISTINCT CODSEXO  
FROM DEPTE;
```

CODSEXO
f
m

```
SELECT DISTINCT TPPARENTESCO,  
CODSEXO FROM DEPTE;
```

TPPARENTESCO	CODSEXO
esposa	f
filha	f
filho	m

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - RESTRINGINDO O NÚMERO DE LINHAS

- EMPREGAR A CLÁUSULA *WHERE*
- POSSIBILIDADE DE APLICAR MÚLTIPLAS CONDIÇÕES SOBRE AS LINHAS DAS TABELAS MENCIONADAS NO *FROM*

```
SELECT      * | { [DISTINCT] column | expression [alias] , ... }  
FROM        table  
[WHERE      condition(s) ] ;
```

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - RESTRINGINDO O NÚMERO DE LINHAS - OPER. COMPARAÇÃO

Operador Comparação	Significado
=	Igual a
>	Maior que
>=	Maior ou igual a
<	Menor que
<=	Menor ou igual a
<>	Diferente

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - RESTRINGINDO O NÚMERO DE LINHAS - OPER. COMPARAÇÃO

- RETORNA TODAS AS COLUNAS DOS EMPREGADOS CUJOS NOMES NÃO SEJAM “JOSE”

```
SELECT * FROM EMP WHERE upper(NMEMP) <> 'JOSE'
```

CODEMP	CODEMPGERENTE	CODDEPTO	NMEMP	SNEMP	DTNASC	ENDER	CODSEXO	VLSALARIO
55	<null>	2	PAulo	Shazan	19/4/1952	R. Prof. Castilho, 2003	M	21000
39	55	2	Ana	Paula	19/1/1976	R. Prof. Isídio Cunha, 1	F	4000
29	55	3	Tereza	Kohn	22/4/1963	R. Prof. Mario Sá, 198	F	7500
13	55	1	Gilberto	Vurg	22/1/1960	R. Prof. Faustino, 111	M	8000
17	29	3	Daniela	Smith	10/12/1980	Av. Ibirapuera, 1200	F	2600
16	29	3	Fabiana	Schultz	30/3/1977	Av. Ibirapuera, 91	F	4500
2	13	1	Mariana	Campos	22/9/1973	R. Cerro Cora, 190	F	4500

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - RESTRINGINDO O NÚMERO DE LINHAS - OPER. LÓGICO

Operador	Significado
AND	Retorna VERDADEIRO se todos os componentes da condição forem verdadeiros
OR	Retorna verdadeiro se uma das condições for verdadeira
NOT	Retorna verdadeiro se a condição for falsa

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - RESTRINGINDO O NÚMERO DE LINHAS - OPER. LÓGICO

- **SELECIONE O NOME E SOBRENOME DOS EMPREGADOS QUE POSSUAM SALÁRIO MENOR OU IGUAL A 4000 E SEJAM DO SEXO FEMININO**

```
SELECT NMEMP, SNEMP FROM EMP
WHERE   CODSEXO      = 'F'
AND     VLSALARIO    <= 4000
```

NMEMP	SNEMP
Ana	Paula
Daniela	Smith

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - RESTRINGINDO O NÚMERO DE LINHAS - OUTROS OPER.

Operador	Significado
BETWEEN ...AND...	Entre dois valores, inclusive
IN(lista)	Igualdade a item de lista de valores
LIKE	Igualdade a padrão de caracteres
IS NULL	Valor igual a nulo

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - RESTRINGINDO O NÚMERO DE LINHAS - OUTROS OPER.

```
SELECT NMEMP, SNEMP, VLSALARIO FROM EMP  
WHERE VLSALARIO BETWEEN 4000 and 10000
```

NMEMP	SNEMP	VLSALARIO
Ana	Paula	4000
Tereza	Kohn	7500
Gilberto	Vurg	8000
Fabiana	Schultz	4500
Mariana	Campos	4500
Jose	Santos	5000

Seleciona o Nome,
Sobrenome e Salário
dos Empregados
cujo a faixa salarial encontra-se
entre 4000 e 10000

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - RESTRINGINDO O NÚMERO DE LINHAS - OUTROS OPER.

Seleciona todas as colunas dos Dependentes que possuam 'SANTOS' em alguma parte do seu nome

```
SELECT      * FROM DEPTE
WHERE       upper (NMDEPTE) LIKE  '%SANTOS%';
```

NROSEQUENCIA	CODEMP	NMDEPTE	TPPARENTESCO	DTNASC	CODSEXO
1	1	Ramiro Santos	filho	22/1/1988	m
2	1	Samuel Santos	filho	1/9/1990	m

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - RESTRINGINDO O NÚMERO DE LINHAS - OUTROS OPER.

Seleciona todas as colunas dos Empregados onde a unidade do dia de Nascimento seja igual a 2

```
SELECT      * FROM      EMP
WHERE       cast (dtnasc as varchar) LIKE '%_2' ;
```

Codemp	Codempgerente	Coddepto	Nmemp	Snemp	Dtnasc	Ender	Codsexo	Vlsalario
29	55	3	Tereza	Kohn	22/6/1963	R. Prof. Mario Sá, 198	F	7500
13	55	1	Gilberto	Vurg	22/1/1960	R. Prof. Faustino, 111	M	8000
17	29	3	Daniela	Smith	12/9/1980	Av. Ibirapuera, 1200	F	2600
2	13	1	Mariana	Campos	22/5/1973	R. Cerro Cora, 190	F	4500

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - RESTRINGINDO O NÚMERO DE LINHAS - OUTROS OPER.

```
SELECT    NMPROJ, NMLOCAL
FROM      PROJ
WHERE     NMLOCAL IN ( 'SP' );
```

NMPROJ	NMLOCAL
reengenharia	SP
produto X	SP
produto Y	SP
automatizacao	SP

```
SELECT    NMPROJ, NMLOCAL
FROM      PROJ
WHERE     NMLOCAL NOT IN ( 'SP' );
```

NMPROJ	NMLOCAL
reengenharia	MG

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT -CLASSIFICANDO O RESULTADO

- **CLÁUSULA *ORDER BY***
 - PERMITE ORDENAR AS LINHAS EM UMA SELEÇÃO
 - PODE-SE UTILIZAR UMA OU MAIS COLUNAS PARA ORDENAÇÃO
 - ORDEM DAS COLUNAS É IMPORTANTE. O SGBD IRÁ CLASSIFICAR O RESULTADO DE ACORDO COM A ORDEM INFORMADA

```
SELECT    *|{[DISTINCT] column|expression [alias],...}  
FROM      table  
[WHERE    condition(s)]  
ORDER BY  COLUMN 1 [ASC,DESC], ...;
```

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT -CLASSIFICANDO O RESULTADO

```
SELECT    NMEMP, SNEMP, DTNASC
FROM      EMP
ORDER BY  DTNASC DESC;
```

NMEMP	SNEMP	DTNASC
Daniela	Smith	10/12/1980
Fabiana	Schultz	30/3/1977
Ana	Paula	19/1/1976
Mariana	Campos	22/9/1973
Tereza	Kohn	22/4/1963
Jose	Santos	10/1/1961
Gilberto	Vurg	22/1/1960
PAulo	Shazan	19/4/1952

Seleciona Nome, Sobrenome e Data Nascimento dos Empregados, ordenando-os decendentemente por Data

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - CLASSIFICANDO O RESULTADO

```
-- Ordena por ordem ASCENDENTE de Salário Anual  
SELECT    NMEMP, SNEMP, VLSALARIO * 12 SalarioAnual  
FROM      EMP  
ORDER BY  VLSALARIO * 12;
```

Nmemp	Snemp	Salarioanual
Daniela	Smith	31200
Ana	Paula	48000
Fabiana	Schultz	54000
Mariana	Campos	54000
Jose	Santos	60000
Tereza	Kohn	90000
Gilberto	Vurg	96000
PAulo	Shazan	252000

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - FUNÇÕES DE AGREGAÇÃO

- OPERAM SOBRE UM CONJUNTO DE LINHAS PARA APRESENTAR UM VALOR
 - A TABELA INTEIRA OU UM AGRUPAMENTO DE DADOS

Operador Comparação	Significado
SUM (COLUNA)	Soma todos os valores de um conjunto de linhas
MAX (COLUNA)	Retorna o maior valor em um conjunto de linhas
MIN(COLUNA)	Retorna o menor valor em um conjunto de linhas
COUNT(COLUNA ou *)	Retorna a quantidade de itens no conjunto de linhas
AVG	Retorna a média aritmética de um conjunto de linhas

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - FUNÇÕES DE AGREGAÇÃO

EMP	
Nmemp	Vlsalario
Ana	4000
Tereza	7500
Gilberto	8000
Daniela	2600
Fabiana	4500
Jose	5000
Mariana	4500
PAulo	21000

Qual é o maior Salário da tabela EMP?

Max(vlsalario)
21000

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - FUNÇÕES DE AGREGAÇÃO

```
SELECT    MIN (VLSALARIO) , MAX (VLSALARIO) ,  
          AVG (VLSALARIO) , SUM (VLSALARIO) , COUNT (*)  
FROM      EMP ;
```

Min(vlsalario)	Max(vlsalario)	Avg(vlsalario)	Sum(vlsalario)	Count(*)
2600	21000	7137,5	57100	8

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - AGRUPAMENTO DE DADOS

- Divide as linhas de linha de uma seleção com base em UM ou MAIS colunas
- Ordem das colunas indica como o agrupamento será realizado

Codsexo	Vlsalario
F	4000
F	7500
F	2600
F	4500
F	4500
M	5000
M	21000
M	8000

Qual a
média de
salário
para o sexo
Masculino e
Feminino?.

Codsexo	Avg(vlsalario)
M	11333,3333333333
F	4620

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - AGRUPAMENTO DE DADOS

```
SELECT  CODSEXO, MIN(VLSALARIO) MenorSal,
        MAX(VLSALARIO) MaiorSal, AVG(VLSALARIO),
        SUM(VLSALARIO) TotalSal, COUNT(*) QtdeEmp
FROM    EMP
GROUP BY CODSEXO
ORDER BY CODSEXO;
```

Observe que as colunas do SELECT ou fazem parte do critério de agrupamento no GROUP BY ou são utilizadas em Funções de Agregação

CODSEXO	MENORSAL	MAIORSAL	MEDIASAL	TOTALSAL	QTDEEMP
F	2600	7500	4620	23100	5
M	5000	21000	11333,3333333333	34000	3

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - FILTRANDO AGRUPAMENTOS

- PERMITE SELECIONAR QUAIS GRUPOS DE LINHAS, DENTRE AQUELES FORMADOS PELO GROUP BY, SERÃO CONSIDERADOS

Coluna1	Coluna3	Qtde
A	X	2
A	Y	1
B	D	2
C	X	1

Após critério
sobre Agrup.

Coluna1	Coluna3	Qtde
A	X	2
B	D	2

●
Having Qtde > 1

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - FILTRANDO AGRUPAMENTOS

```
SELECT coddepto, codsexo, min(vlsalario) as MenorSal,  
max(vlsalario) as MaiorSal, avg(vlsalario) as MediaSal,  
sum(vlsalario) as TotalSal, count(*) as QtdeEmp  
FROM EMP  
GROUP BY coddepto, codsexo  
HAVING count(*) > 1  
ORDER BY coddepto, codsexo Desc
```

Emprega as funções de agregação na seleção agrupada por Código do Depto e Sexo, exibindo somente aqueles com **MAIS QUE** um empregado

CODDEPTO	CODSEXO	MENORSAL	MAIORSAL	MEDIASAL	TOTALSAL	QTDEEMP
1	M	5000	8000	6500	13000	2
3	F	2600	7500	6666666667	14600	3

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - MACRO PASSOS DE EXECUÇÃO

Select ... Where..

Group By ...

Having ...

Order by ...

Resultado Seleção

Passo1:

Seleciona as colunas desejadas e aplica os critérios, se houverem

Passo2:

Sob o resultado anterior aplica os critérios de agrupamento

Passo3:

Sob o resultado do agrupamento aplica os critérios de seleção

Passo4:

Ordena o resultado da seleção pós-agrupamento

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - TRATAMENTO DE NULOS

- COLUNAS COM VALORES NULOS SIGNIFICAM DESCONHECIDOS
- NULO NÃO É IGUAL A BRANCO OU A ZERO!!!
- TENDO A COL1 E COL2 OS VALORES NULOS, A COMPARAÇÃO COL1 = COL2 OU COL1 <> COL2 É SEMPRE FALSA.
- NULO NÃO É IGUAL A OUTRO NULO!!!
- SELEÇÃO OU EXTRAÇÃO DE NULOS DEVE-SE UTILIZAR:
 - *IS NULL OU IS NOT NULL*

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - TRATAMENTO DE NULOS

```
SELECT      * FROM CONSULT
WHERE      VLSALHORA IS NOT NULL;
```

Selecione os Consultores
que **NÃO** apresentam
NULOS
em seus respectivos
Valor por Horas

CODCONSULT	CODDEPTO	NMCONSULT	SNCONSULT	VLSALHORA
1	2	Carlos	Figueroa	32
3	2	Cristina	Fernandes	28
4	3	Humberto	Osvaldo	31
6	3	Paula	Gilt	41
8	2	Francisca	da Silva	33

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - TRATAMENTO DE NULOS

```
SELECT    Nmconsult, SnConsult, Vlsalhora * 160
FROM      CONSULT;
```

Toda operação com
NULO resulta sempre
em **NULO**

NMCONSULT	SNCONSULT	F_1
Carlos	Figuroa	5120
Sebastiao	Sa	<null>
Cristina	Fernandes	4480
Humberto	Osvaldo	4960
Gustavo	Bianco	<null>
Paula	Gilt	6560
Fernanda	Dantas	<null>
Francisca	da Silva	5280

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - TRATAMENTO DE NULOS

- VALORES NULOS UTILIZADAS EM FUNÇÕES AGREGADAS SÃO IGNORADOS
- A FUNÇÃO COUNT PERMITE DUAS SITUAÇÕES:
 - *COUNT (COLUNA): SE A COLUNA APRESENTAR NULOS, ESTES NÃO FARÃO PARTE DA CONTAGEM FINAL*
 - *COUNT (*): TODAS AS LINHAS SERÃO CONTADAS, INDEPENDENTE DE APRESENTAREM OU NÃO NULOS*

SQL - DML (DATA MANIPULATION LANGUAGE)

SELECT - TRATAMENTO DE NULOS

```
SELECT      COUNT (*) , COUNT (VLSALHORA) ,  
            AVG (VLSALHORA)  MEDIAFUNCAO ,  
            SUM (VLSALHORA) / COUNT (*)  MEDIACALCULADA  
FROM        CONSULT ;
```

Count(*)	Count(vlsalhora)	Mediafuncao	Mediacalculada
8	5	33	20,625

SQL - DML (DATA MANIPULATION LANGUAGE)

JUNÇÃO INTERNA DE TABELAS - EQUIJOIN

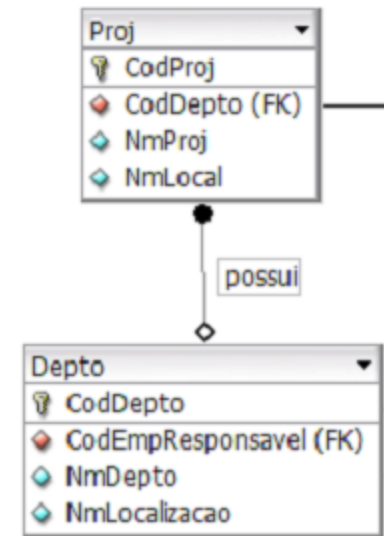
- OPERAÇÃO QUE APRESENTA UM RESULTADO DA ASSOCIAÇÃO DAS LINHAS PROVENIENTES DE 2 OU MAIS TABELAS
- ASSOCIAÇÃO É POSSÍVEL PELA CHAVE ESTRANGEIRA

Deppto

CodDeppto	NmDeppto	NmLocalização	CodEmpResponsavel
1	Pesquisa	São Paulo	13
2	Administração	Campinas	55
3	Financeiro	Campinas	29

Proj

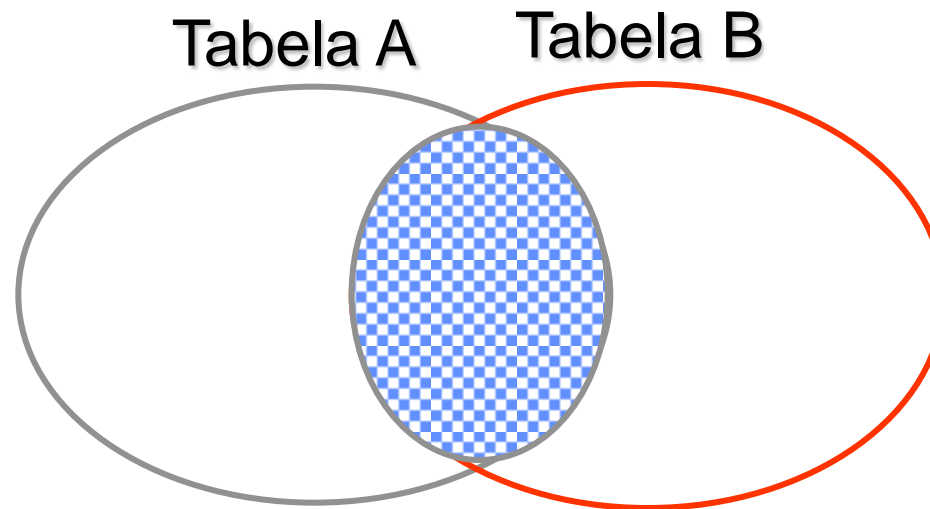
CodProj	NmLocal	NmProj	CodDeppto
1	SP	Reengenharia	2
2	SP	Produto X	1
3	MG	Reengenharia	2
4	SP	Produto Y	1
5	SP	Automatização	3



SQL - DML (DATA MANIPULATION LANGUAGE)

JUNÇÃO INTERNA DE TABELAS - EQUIJOIN

- **ÁREA HACHURADA INDICA AS LINHAS QUE APRESENTAM ASSOCIAÇÕES (RELACIONAMENTOS) GARANTIDAS POR UMA CHAVE ESTRANGEIRA**
- **ELEMENTOS FORA DESTA ÁREA NÃO SÃO APRESENTADOS**



SQL - DML (DATA MANIPULATION LANGUAGE)

JUNÇÃO INTERNA DE TABELAS - EQUIJOIN

Observe que o vínculo entre
As tabelas ocorre pelo uso da
Chave Primária e a Estrangeira

```
SELECT  NEMP, SEMP, NMDEPTO
FROM EMP, DEPTO
WHERE   upper(NMDEPTO) = 'PESQUISA'
AND     DEPTO.CODDEPTO = EMP.CODDEPTO;
```

Nmemp	Snemp	Nmdepto
Gilberto	Vurg	pesquisa
Jose	Santos	pesquisa
Mariana	Campos	pesquisa

SQL - DML (DATA MANIPULATION LANGUAGE)

JUNÇÃO INTERNA DE TABELAS – EQUIJOIN (VARIANTES)

```
SELECT    NMEMP, SEMP, NMDEPTO
FROM EMP INNER JOIN DEPTO on DEPTO.CODDEPTO = EMP.CODDEPTO
WHERE     upper(NMDEPTO) = 'PESQUISA';
```

```
-- Válido para o PostgreSQL
SELECT    NMEMP, SEMP, NMDEPTO
FROM EMP INNER JOIN DEPTO using(CODDEPTO)
WHERE     upper(NMDEPTO) = 'PESQUISA'
```

```
-- Válido para o PostgreSQL
SELECT    NMEMP, SEMP, NMDEPTO
FROM EMP NATURAL INNER JOIN DEPTO
WHERE     upper(NMDEPTO) = 'PESQUISA'
```

SQL - DML (DATA MANIPULATION LANGUAGE)

JUNÇÃO INTERNA DE TABELAS - EQUIJOIN

```
-- Efetua o auto-relacionamento entre a tabela EMP
-- para obter o Empregado GERENTE de um
-- Empregado COMUM que vive em um Endereço que
-- possua a string PROF.
```

```
SELECT      E.NMEMP || E.SNEMP EMPREGADO,
            GER.NMEMP || GER.SNEMP GERENTE
FROM        EMP E, EMP GER
WHERE       upper(E.ENDER) LIKE '%PROF%'
AND         E.CODEMPGERENTE = GER.CODEMP;
```

Empregado	Gerente
AnaPaula	PAuloShazan
TerezaKohn	PAuloShazan
GilbertoVurg	PAuloShazan
JoseSantos	GilbertoVurg

SQL - DML (DATA MANIPULATION LANGUAGE)

JUNÇÃO INTERNA DE TABELAS - EQUIJOIN

```
-- Efetua a seleção dos gestores e a quantidade
-- de empregados sob sua supervisão
select    e.NmEmp, e.SnEmp, count(*) as TotEmp
from      Emp e, Depto d, Emp b
where     d.CodEmpResponsavel = e.CodEmp
and       d.CodEmpResponsavel = b.CodEmpGerente
group    by e.NmEmp, e.SnEmp
order     by e.NmEmp, e.SnEmp;
```

Nmemp	Snemp	Totemp
Gilberto	Vurg	2
PAulo	Shazan	3
Tereza	Kohn	2

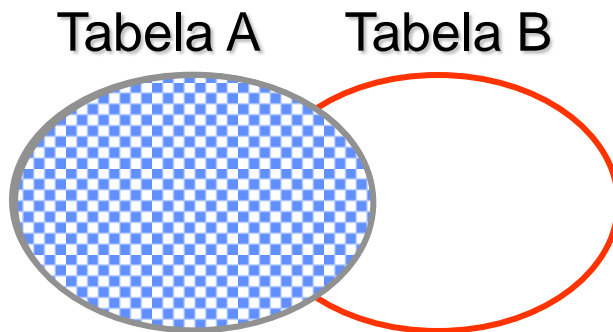
SQL - DML (DATA MANIPULATION LANGUAGE)

JUNÇÃO EXTERNA DE TABELAS - OUTERJOIN

- MESMO CONCEITO APRESENTADO PARA EQUIJOIN
- PORÉM PERMITE SELECIONAR QUAL TABELA VOCÊ DESEJAR VER OS ELEMENTOS QUE NÃO POSSUEM RELACIONAMENTO

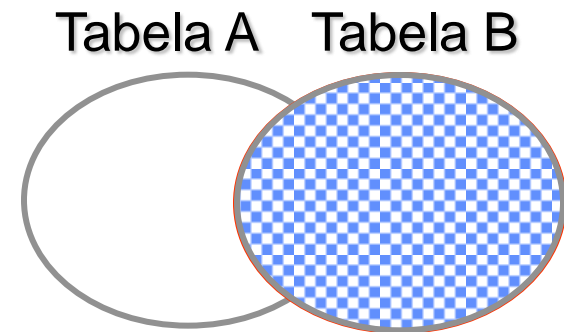
LEFT OUTER JOIN:

Selecionar os elementos da TABELA A mesmo que estes NÃO se relacionem com a TABELA B



RIGHT OUTER JOIN:

Selecionar os elementos da TABELA B mesmo que estes NÃO se relacionem com a TABELA A



SQL - DML (DATA MANIPULATION LANGUAGE)

JUNÇÃO EXTERNA DE TABELAS - OUTERJOIN

```
-- Efetua a seleção dos empregados e a quantidade  
-- de dependentes  
  
select    e.NmEmp, e.SnEmp, count(d.codemp)  
from      Emp e LEFT JOIN Depte d using(codemp)  
group     by e.nmemp, e.snemp;
```

NMEMP	SNEMP	COUNT
Ana	Paula	1
Daniela	Smith	0
Fabiana	Schultz	0
Gilberto	Vurg	1
Jose	Santos	2
Mariana	Campos	0
PAulo	Shazan	0
Tereza	Kohn	2

Observe que indicamos a tabela que NÃO conterá elementos para associação com o sinal (+)

SQL - DML (DATA MANIPULATION LANGUAGE)

JUNÇÃO EXTERNA DE TABELAS - OUTERJOIN

-- Efetua a seleção dos empregados e seus subordinados

```
select    e.NmEmp, e.SnEmp, b.NmEmp, b.SnEmp
from      Emp b RIGHT JOIN Emp e on e.codempgerente = b.codemp
where     upper(e.Ender)      like '%PROF%';
```

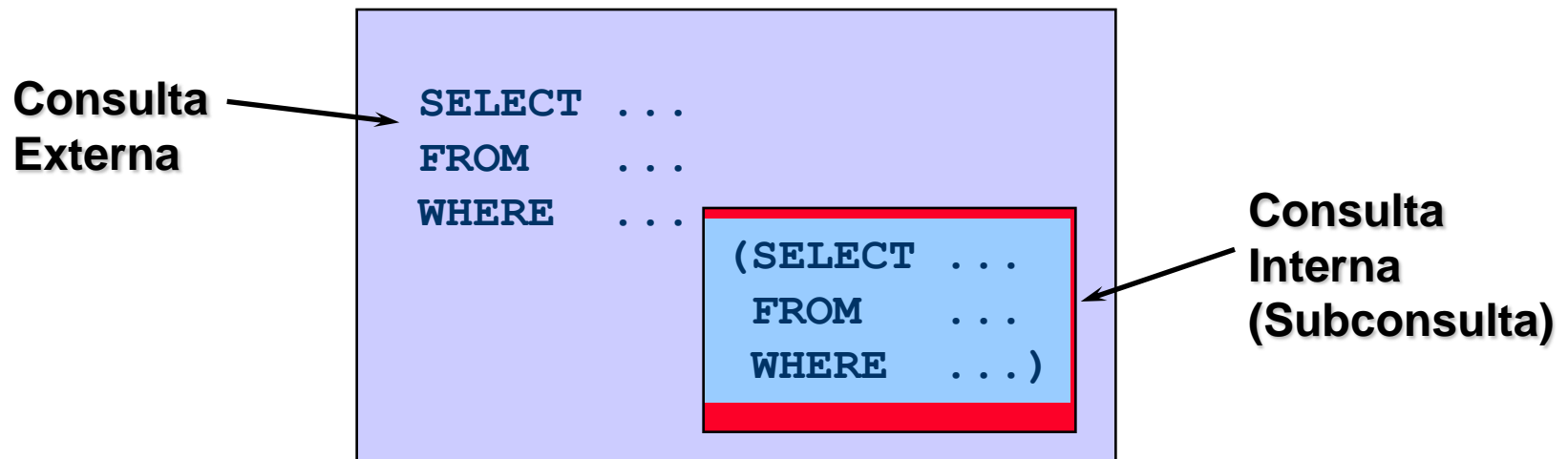
Selecione os empregados
e os nomes gestores
que residiam em ruas com 'Prof'

NMEMP	SNEMP	NMEMP1	SNEMP1
PAulo	Shazan	<null>	<null>
Ana	Paula	PAulo	Shazan
Tereza	Kohn	PAulo	Shazan
Gilberto	Vurg	PAulo	Shazan
Jose	Santos	Gilberto	Vurg

SQL - DML (DATA MANIPULATION LANGUAGE)

SUBCONSULTAS – MÓDULO VIII

- SUBCONSULTAS
 - A CONSULTA INTERNA É EXECUTADA e, ENTÃO, SEU RESULTADO É UTILIZADO PELA CONSULTA EXTERNA
 - DOIS TIPOS:
 - INDEPENDENTES ou CORRELACIONADAS



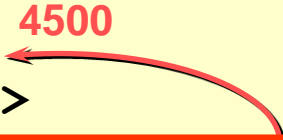
SQL - DML (DATA MANIPULATION LANGUAGE)

SUBCONSULTAS – OPERADORES DE COMPARAÇÃO

- SUBCONSULTAS

- O RESULTADO A CONSULTA INTERNA DEVERIA RETORNAR UM ÚNICO VALOR À CONSULTA EXTERNA

```
SELECT *  
FROM emp  
WHERE vlsalario >  
      (SELECT vlsalario  
       FROM emp  
       WHERE snemp = 'CAMPOS') ;
```



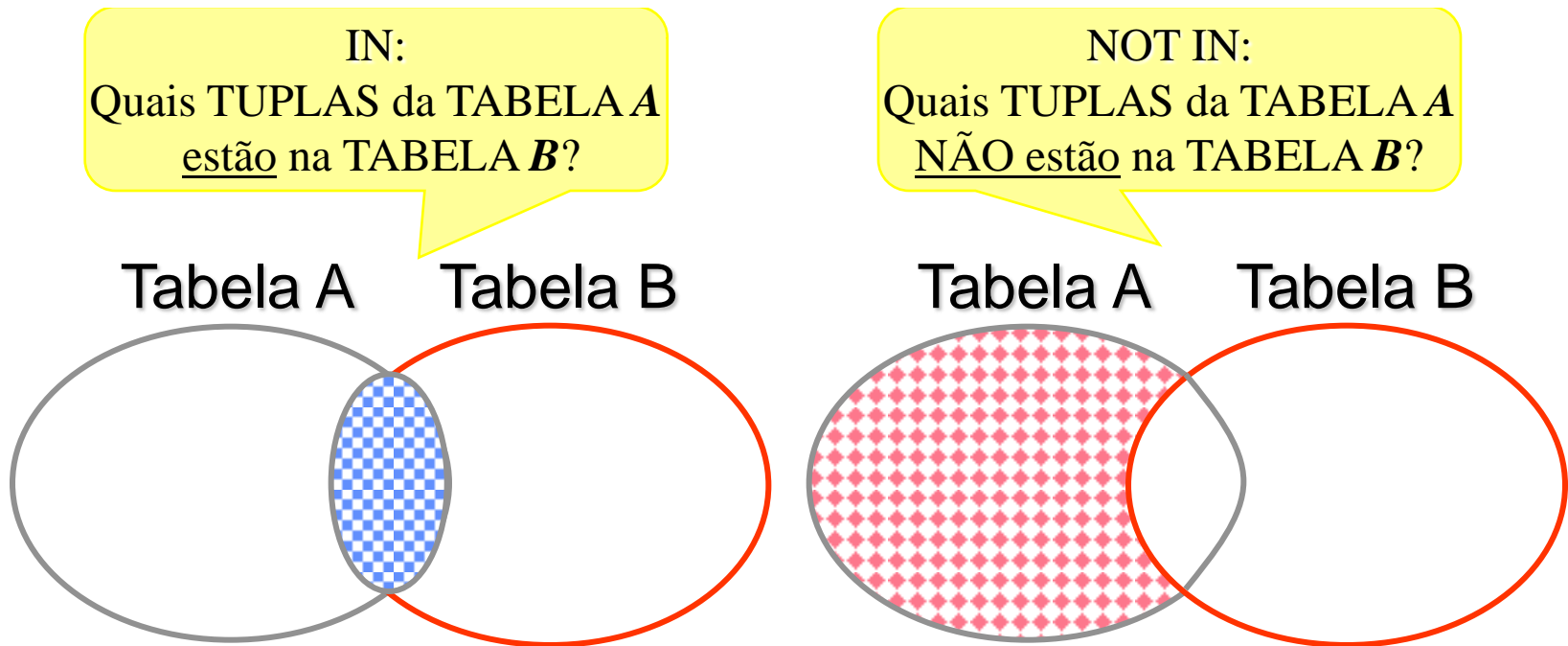
The diagram illustrates a SQL query where a subquery is used in a comparison. The main query selects all rows from the 'emp' table where 'vlsalario' is greater than the result of a subquery. The subquery selects 'vlsalario' from 'emp' where 'snemp' is 'CAMPOS'. A red arrow points from the value '4500' to the comparison operator '>' in the main query, indicating that the subquery returns a single value (4500) that is used for comparison.

SQL - DML (DATA MANIPULATION LANGUAGE)

SUBCONSULTAS – OPERADORES IN ou NOT IN

- SUBCONSULTAS

- PERMITE IDENTIFICAR TUPLAS DE UMA TABELA QUE ESTÃO (OU NÃO ESTÃO) EM UMA TABELA RELACIONADA



SQL - DML (DATA MANIPULATION LANGUAGE)

SUBCONSULTAS – OPERADORES IN ou NOT IN

- SUBCONSULTAS COM OPERADOR IN
 - IDENTIFICAR TODOS OS EMPREGADOS COM DEPENDENTES

```
SELECT * FROM EMP
WHERE CODEMP
      in (SELECT CODEMP FROM DEPTE);
```

Codemp	Codempgerente	Coddepto	Nmemp	Snemp	Dtnasc	Ender	Codsexo	Vlsalario
1	13	1	Jose	Santos	6/10/1961	R. Prof. Sebastiao, 15	M	5000
13	55	1	Gilberto	Vurg	22/1/1960	R. Prof. Faustino, 111	M	8000
29	55	3	Tereza	Kohn	22/6/1963	R. Prof. Mario Sá, 198	F	7500

SQL - DML (DATA MANIPULATION LANGUAGE)

SUBCONSULTAS – OPERADORES IN ou NOT IN

- SUBCONSULTAS COM OPERADOR *NOT IN*
 - IDENTIFICAR TODOS OS EMPREGADOS QUE NÃO SÃO GESTORES DE DEPARTAMENTO

```
SELECT * FROM EMP
WHERE CODEMP
      NOT IN (SELECT CODEMPRESPONSAVEL FROM DEPTO);
```

Codemp	Codempgerente	Coddepto	Nmemp	Snemp	Dtnasc	Ender	Codsexo	Vlsalario
39	55	2	Ana	Paula	19/1/1976	R. Prof. Isidio Cunha, 1	F	4000
17	29	3	Daniela	Smith	12/9/1980	Av. Ibirapuera, 1200	F	2600
16	29	3	Fabiana	Schultz	30/3/1977	Av. Ibirapuera, 91	F	4500
1	13	1	Jose	Santos	6/10/1961	R. Prof. Sebastiao, 15	M	5000
2	13	1	Mariana	Campos	22/5/1973	R. Cerro Cora, 190	F	4500

SQL - DML (DATA MANIPULATION LANGUAGE)

SUBCONSULTAS – OPERADORES IN ou NOT IN

- SUBCONSULTAS COM OPERADOR *EXISTS*

```
SELECT * FROM EMP E WHERE NOT EXISTS  
  (SELECT CODEMPRESPONSAVEL FROM DEPTO D  
   WHERE D.CODEMPRESPONSAVEL = E.CODEMP) ;
```

Temos aqui o mesmo resultado
Da cláusula do slide anterior, mas a
Subconsulta é **CORRELACIONADA**

SQL - DML (DATA MANIPULATION LANGUAGE)

SUBCONSULTAS

- SUBCONSULTAS COM MÚLTIPLOS ATRIBUTOS

```
SELECT * FROM CONSULT
WHERE (CODDEPTO, NMCONSULT) NOT IN
      (SELECT CODDEPTO, NMEMP FROM EMP) ;
```

codconsult integer	coddepto integer	nmconsult character varying (30)	snconsult character varying (40)	vlsalhora real
1	2	Carlos	Figueroa	32
2	2	Sebastiao	Sa	[null]
3	2	Cristina	Fernandes	28
4	3	Humberto	Osvaldo	31
5	3	Gustavo	Bianco	[null]
6	3	Paula	Gilt	41
7	3	Fernanda	Dantas	[null]
8	2	Francisca	da Silva	33

SQL - DML (DATA MANIPULATION LANGUAGE)

OPERADORES DE CONJUNTO

- **UNIÃO DE CONSULTAS**

```
-- sintaxe  
SELECT QUERY 1 UNION [ALL] SELECT QUERY 2 UNION [ALL] ...
```

- **INTERSECÇÃO DE CONSULTAS**

```
-- sintaxe  
SELECT QUERY 1 INTERSECT [ALL] SELECT QUERY
```

- **DIFERENÇA ENTRE CONSULTAS**

```
-- sintaxe  
SELECT QUERY 1 EXCEPT [ALL] SELECT QUERY
```

- **EM TODOS OS CASOS, O NÚMERO DE ATRIBUTOS DEVE SER O MESMO**

SQL - DML (DATA MANIPULATION LANGUAGE)

OPERADORES DE CONJUNTO

```
select * from proj  
union  
select * from depto;
```

codproj integer	coddepto integer	nmproj character varying	nmlocal character varying (45)
4	1	produto Y	SP
3	29	financeiro	campinas
5	3	automatizacao	SP
2	1	produto X	SP
1	13	pesquisa	sao paulo
3	2	reengenharia	MG
2	55	administracao	campinas
1	2	reengenharia	SP

Note que
O resultado mantém
os nomes
dos atributos da
1ª. query

SQL - DML (DATA MANIPULATION LANGUAGE)

OPERADORES DE CONJUNTO

```
select codemp, nmemp, snemp from emp where codemp in  
    (select codemp from emp  
     except  
     select codemp from depte)
```

codemp integer	nmemp character varying (30)	snemp character varying (40)
39	Ana	Paula
16	Fabiana	Schultz
17	Daniela	Smith
55	PAulo	Shazan
2	Mariana	Campos

SQL - DML (DATA MANIPULATION LANGUAGE)

ELEMENTOS DA LINGUAGEM

- ***SELECT:***

PERMITE SELECIONAR DADOS DE UMA OU MAIS RELAÇÕES

- ***UPDATE:***

PERMITE ATUALIZAR UMA OU MAIS TUPLAS DE UMA RELAÇÃO

- ***DELETE:***

PERMITE ELIMINAR UMA OU MAIS TUPLAS DE UMA RELAÇÃO

- ***INSERT:***

PERMITE ACRESCENTAR UMA OU MAIS TUPLAS EM UMA RELAÇÃO

SQL - DML (DATA MANIPULATION LANGUAGE)

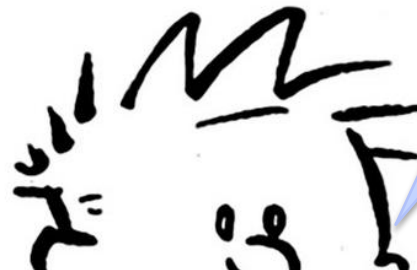
INSERT

- ACRESCEMENTA UMA TUPLA EM UMA RELAÇÃO

```
INSERT INTO nome_tabela [(lista de colunas)]  
VALUES (lista de valores);
```

```
INSERT INTO nome_tabela [(lista de colunas)]  
SELECT ... FROM ... WHERE ... GROUP BY .. ORDER BY;
```

Pssssiu...
**Regras de Integridade
Domínio e Unicidade**
garantem a
consistência dessa
operação



SQL - DML (DATA MANIPULATION LANGUAGE)

INSERT

-- ACRESCENTANDO UM CONSULTOR

-- 1a. Forma

-- Informa todas as colunas a receberem valores

```
Insert into consult (codconsult, nmconsult, snconsult,  
    vlsalhora, coddepto)  
values (9, 'Joaquim', 'Rha', 34, 3);
```

-- 2a. Forma

-- Informa todas as colunas a receberem valores

-- segundo a ordem com que estas aparecem na tabela

```
Insert into consult  
values (9, 3, 'Joaquim', 'Nabuco', 34)
```

SQL - DML (DATA MANIPULATION LANGUAGE)

INSERT

```
-- Acrescenta um Consultor em um Departamento  
-- que possui um projeto em Minas Gerais
```

```
insert into consult  
select 46, coddepto, 'Airton', 'Senna', (2546 / 160)  
from proj NATURAL INNER JOIN depto  
where nmlocal = 'MG'
```

```
-- No PostgreSQL
```

```
--
```

```
Insert into consult (codconsult, nmconsult, snconsult,  
vlsalhora, coddepto) values  
(9, 'Joaquim', 'Rha', 34, 3),  
(19, 'Carlos', 'Chaplin', 31, 3),  
(29, 'Ana', 'Carmelita', 38, 3);
```

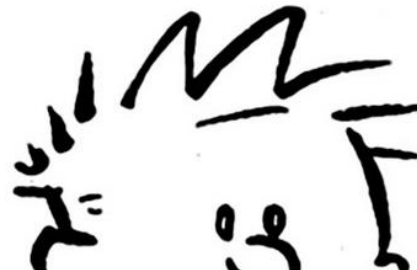
SQL - DML (DATA MANIPULATION LANGUAGE)

DELETE

- REMOVE TUPLAS DE UMA RELAÇÃO

```
DELETE FROM nome_tabela  
[WHERE (lista de condições)];
```

Pssssiu...
Regras de Integridade
garantem a
consistência dessa
operação



SQL - DML (DATA MANIPULATION LANGUAGE)

DELETE

```
-- Remove todas as linha da tabela Dependente
delete from depte;

-- Remove os consultores sem Valor Hora
delete from consult where vlsalhora is null;

-- Remove os empregados que não estão associados
-- a pelo menos UM projeto
delete from emp where codemp not in
    (select codemp from alocado);
```

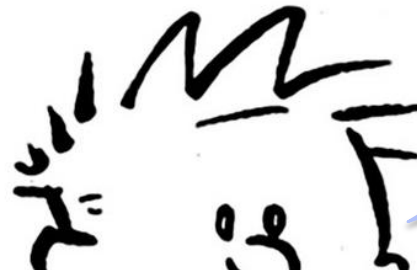
SQL - DML (DATA MANIPULATION LANGUAGE)

UPDATE

- MODIFICA UM OU MAIS ATRIBUTOS DE TUPLAS DE UMA RELAÇÃO

```
UPDATE nome_tabela  
SET  coluna1 = novo_valor, colunaN = novo_valor  
[WHERE  (lista de condições)];
```

Pssssiu...
**Regras de Integridade
Domínio e Unicidade**
garantem a
consistência dessa
operação



SQL - DML (DATA MANIPULATION LANGUAGE)

DELETE

-- Atualiza o nome de Consultor específico

```
update consult set nmconsult = 'Carlitos'
where codconsult = 1;
```

-- 10% de aumento para TODOS os empregados alocados
-- em projetos

```
Update emp set vlsalario = vlsalario * 1.1
where codemp in
    (select codemp from alocado);
```

SQL - DML (DATA MANIPULATION LANGUAGE)

EXTENSÕES DO SQL

- **OBTER VALORES MODIFICADOS ou ACRESCENTADOS**

```
Update emp set vlsalario = vlsalario * 1.1
where codemp in
  (select codemp from alocado)
returning nmemp, vlsalario;
```

```
DELETE FROM CONSULT WHERE CODCONSULT = 46
RETURNING *;
```

OBJETOS PROGRAMADOS – A GLIMPSE....

OBJETOS PROGRAMADOS

CARACTERÍSTICAS

- **COMBINA A LINGUAGEM DE MANIPULAÇÃO DE DADOS (SQL) COM AS VANTAGENS DE UMA LINGUAGEM DE PROCEDIMENTOS (PL)**
- **ALTO DESEMPENHO PARA OPERAÇÕES INTENSIVAS EM DADOS**
 - **CÓDIGO PRÉ-COMPILADO**
- **PROPORCIONA O ENCAPSULAMENTO DO BD**
 - **AGILIZA A EVOLUÇÃO DAS APLICAÇÕES**
 - **SIMPLIFICA OS PROCEDIMENTOS DE SEGURANÇA**
 - **CONCENTRA REGRAS SEMÂNTICAS EM UM ÚNICO PONTO**

OBJETOS PROGRAMADOS

TIPOS DE OBJETOS

- **STORED PROCEDURES e USER FUNCTIONS**

- CONJUNTO DE INSTRUÇÕES PL/SQL QUE REALIZAM CERTA ATIVIDADE
- EXECUTADAS DE MODO EXPLÍCITO
- EXEMPLOS:
 - ACRESCENTAR CLIENTES, CHECANDO SEUS DADOS

- **TRIGGERS**

- MESMAS CARACTERÍSTICAS DA SP/FUNCTIONS
- MAS.....SÃO EXECUTADAS DE MODO IMPLÍCITO PELO SGBD

- **PostgreSQL**

- ESCRITAS EM C, PERL, PYTHON, JAVA, TSQL, R, PGSQL

FUNCTIONS NO POSTGRESQL

CRIANDO OU MODIFICANDO

```
-- SINTAXE na criação ou modificação de FUNCTIONS
--
create or replace
FUNCTION nome_função(nome_parâmetro tipo_parâmetro)
RETURNS tipo_retorno AS $$
DECLARE
    -- lista de variáveis
BEGIN
    -- BLOCO SQL e PL
END;
$$ LANGUAGE tipo_linguagem;

-- SINTAXE na remoção de FUNCTIONS
--
DROP FUNCTION nome_função(nome_parâmetro tipo_parâmetro);
```

FUNCTIONS NO POSTGRESQL

EXEMPLO #1

```
create or replace FUNCTION minhafunc(Cod anyelement)
RETURNS text AS $$
DECLARE
    SAL integer;
    res text;
BEGIN
    -- Obtém o salário de um empregado específico
    select vlsalario into sal from emp where emp.codemp = Cod;
    -- Sintaxe IF múltiplas condições e com ELS IF
    IF NOT FOUND THEN
        RAISE EXCEPTION 'Não encontrei o empregado %', $1;
    ELSIF sal between 1000 and 3000 THEN
        res := 'Salario Baixo';
    ELSIF sal between 3001 and 6000 then
        res := 'Salario Medio';
    else
        res := 'Salario Alto';
    end if;
    return res;
END;
$$ LANGUAGE plpgsql;
```

FUNCTIONS NO POSTGRESQL

EXEMPLO #1 (Cont.)

```
26 -- Busca por um empregado PRESENTE na relação EMP
27 select * from minhafunc(1);
28
```

Data Output Explain Messages Notifications Query History

	minhafunc text
1	Salario Alto

```
26 -- Busca por um empregado INEXISTENTE
27 select * from minhafunc(601);
28
```

Data Output Explain Messages Notifications Query History

ERROR: Não encontrei o empregado 601
CONTEXT: PL/pgSQL function minhafunc(anyelement) line 10 at RAISE
SQL state: P0001

FUNCTIONS NO POSTGRESQL

EXEMPLO #2

```
CREATE or replace FUNCTION getemp(integer)
RETURNS SETOF emp AS $$
BEGIN
    RETURN QUERY SELECT * from emp where codempgerente = $1;
    -- Caso não encontre....
    IF NOT FOUND THEN
        RAISE EXCEPTION 'Não encontrei emp. gerenciados por %', $1;
    END IF;
    RETURN;
END
$$
LANGUAGE plpgsql;
```

FUNCTIONS NO POSTGRESQL

EXEMPLO #2 (Cont.)

```
58 -- Gestor COM subordinados
59 SELECT * FROM getemp(29);
60
```

Data Output Explain Messages Notifications Query History									
	codemp integer	codempgerente integer	coddepto integer	nmemp character varying (30)	snemp character varying (40)	dtnasc date	ender character varying (255)	codsexo character (1)	vlsalario real
1	16	29	3	Fabiana	Schultz	1977-03...	Av. Ibirapuera, 91	F	5445
2	17	29	3	Daniela	Smith	1980-07...	Av. Ibirapuera, 1200	F	3146

```
58 -- Gestor SEM subordinados
59 SELECT * FROM getemp(1);
60
```

Data Output Explain Messages Notifications Query History

ERROR: Não encontrei empregados gerenciados por 1
CONTEXT: PL/pgSQL function getemp(integer) line 5 at RAISE
SQL state: P0001

- **BÁSICOS**

- DATE, Christopher John. **SQL and relational theory: how to write accurate SQL code.** " O'Reilly Media, Inc.", 2011.

- **AVANÇADOS**

- CELKO, Joe. **Joe Celko's SQL for smarties: advanced SQL programming.** Elsevier, 2010.
- MOLINARO, Anthony. **SQL Cookbook: Query Solutions and Techniques for Database Developers.** O'Reilly Media, Inc., 2005.
- KARWIN, Bill. **SQL antipatterns: avoiding the pitfalls of database programming.** Pragmatic Bookshelf, 2010.

- **PostgreSQL**

- FONTAINE, D. **Mastering PostgreSQL in Application Development.** lulu.com, 2017