



INSTITUTO FEDERAL
ESPÍRITO SANTO
Campus Colatina

BANCO DE DADOS I

Recuperando Dados

Prof. Gustavo Ludovico Guidoni



Cláusula SELECT

- O comando **select** é utilizado para extrair dados do SGBD
- Palavras – chaves (estrutura do comando)
 - **SELECT** informa as colunas que serão listadas
 - **FROM** informa as tabelas que serão consultadas
 - **WHERE** restringe as linhas consultadas



Criando SELECT

```
SELECT [ ALL | DISTINCT] select_list
[INTO [ new_table_name ]]
[FROM { table_name/view_name } [optimizer_hints)]
    [[, { table_name2/view_name2 } {optimizer_hints}]
    [..., { table_name16/view_name16 } [(optimizer_hints
    )]]]
[WHERE clause]
[GROUP BY clause]
[HAVING clause]
[ORDER BY clause]
[COMPUTE clause]
```



Notação SQL

- Não faz distinção entre maiúsculas e minúsculas
- Podem estar em uma ou mais linhas
- Palavras-chave não podem ser abreviadas ou divididas entre as linhas
- Normalmente, as cláusulas são colocadas em linhas separadas
- Guias e indentações são usadas para aperfeiçoar a legibilidade



Seleccionando Datos

- Seleccionando columnas

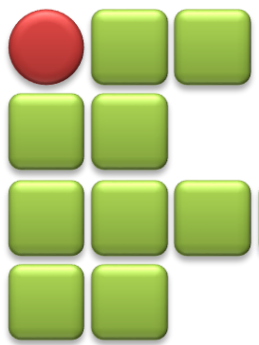
- Sintaxe

```
SELECT column_name[, column_name...]  
FROM   table_name
```

- Exemplo

```
SELECT cod_employado, nom_employado  
FROM   empleado
```

cod_employado	nom_employado
-----	-----
1020	SMITH
1528	ALLEN
2053	WARD



Seleccionando Datos

- Seleccionando todas as columnas

```
SELECT *  
FROM   genero
```

- Seleccionando columnas específicas

```
SELECT cod_genero, dsc_genero  
FROM   genero
```



Expressões Aritméticas

- Crie expressões com dados numéricos e de data usando operadores aritméticos

Operador	Descrição
+	Adição
-	Subtração
*	Multiplicação
/	Divisão



Expressões Aritméticas

```
SELECT nom_empregado  
      ,      val_salario  
      ,      val_salario + 300  
FROM   empregado
```

nom_empregado	val_salario	
-----	-----	-----
SMITH	800.00	1100.00
ALLEN	1600.00	1900.00
WARD	1250.00	1550.00
JONES	2975.00	3275.00
MARTIN	1250.00	1550.00
BLAKE	2850.00	3150.00



Precedência de Operadores



$*$ $/$ $+$ $-$

- A multiplicação e a divisão têm prioridade sobre a adição e a subtração
- Os operadores com a mesma prioridade são avaliados da esquerda para a direita
- Parênteses são usados para forçar a avaliação priorizada e para esclarecer as instruções

Precedência de Operadores

```
SELECT nom_empregado
      , 12 * val_salario + 100
      , 12 * (val_salario + 100)
FROM   empregado
```

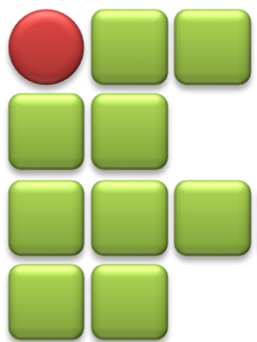
nom_empregado

SMITH	9700.00	10800.00
ALLEN	19300.00	20400.00
WARD	15100.00	16200.00
JONES	35800.00	36900.00
MARTIN	15100.00	16200.00
BLAKE	34300.00	35400.00

Valores nulos

```
SELECT nom_empregado
      , val_comissao
      , 12 * val_salario + val_comissao
FROM   empregado
```

nom_empregado	val_comissao	
-----	-----	-----
SMITH	NULL	NULL
ALLEN	300.00	19500.00
JONES	NULL	NULL
MARTIN	1400.00	16400.00
BLAKE	NULL	NULL
TURNER	.00	14000.00



Definindo um Apelido para Coluna

- Útil para cálculos
- Segue imediatamente o nome da coluna
 - Palavra-chave **AS** é opcional entre o nome da coluna e o apelido
- Necessita de aspas caso contenha espaços ou caracteres especiais

Definindo um Apelido para Coluna

```
SELECT nom_empregado AS Nome_empregado
      , val_salario   Sal_empregado
FROM   empregado
```

Nome_empregado	Sal_empregado
SMITH	800.00

```
SELECT nom_empregado AS 'Nome do empregado'
      , val_salario   AS 'Salário Anual'
FROM   empregado
```

Nome do empregado	Salário Anual
SMITH	800.00



Operador de Concatenação

- Concatena colunas ou *strings* de caracteres a outras colunas
 - Sintaxe: `concat(coluna | string [, coluna | string])`
 - Os valores *strings* devem estar entre aspas
 - Cada *string* de caractere é gerada uma vez para cada linha retornada



Operador de Concatenação

```
SELECT concat (nom_empregado, ' é um ', nom_cargo) AS  
Empregados  
FROM empregado
```

Empregados

SMITH é um BALCONISTA

ALLEN é um VENDEDOR

WARD é um VENDEDOR

Eliminando Linhas Duplicadas

- Dados de uma consulta podem aparecer repetidos no resultado
 - O que fazer quando não queremos que isto aconteça?

```
SELECT cod_departamento  
FROM   empregado
```

```
cod_departamento  
-----  
10  
10  
20  
10  
30  
10  
...
```


Eliminando Linhas Duplicadas

- Elimine **linhas** duplicadas usando a palavra-chave **DISTINCT** na cláusula **SELECT**

```
SELECT DISTINCT  
    cod_departamento  
FROM    empregado
```

```
cod_departamento  
-----  
10  
20  
30
```

As linhas recuperadas
são distintas

```
SELECT DISTINCT  
    cod_departamento  
    , cargo  
FROM    empregado
```

```
cod_departamento  cargo  
-----  
10                BALCONISTA  
10                GEENTE  
10                PRERSIDENTE  
20                ANALISTA  
20                BALCONISTA  
20                GERENTE  
30                BALCONISTA  
30                GERENTE  
30                VENDEDOR
```

