

FIAP GRADUAÇÃO

ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

ESTRATÉGIA E IMPLEMENTAÇÃO DE ESTRUTURAS DE DADOS

Profa. Rita de Cássia Rodrigues (DRS → LINGUAGEM PARA RECUPERAÇÃO DE DADOS)
rita@fiap.com.br

AULA 22 – COMANDOS DQL/DRS (SQL)
(DQL → LINGUAGEM PARA CONSULTA DE DADOS)
OPERADORES/EXPRESSÕES/ CLÁUSULAS

- ✓ Objetivo
- ✓ Conceitos referentes a linguagem de manipulação de dados
- ✓ Revisão dos Conceitos
- ✓ Exercícios

- ❑ Aplicar os conceitos da linguagem SQL durante a implementação do banco de dados

Conteúdo Programático referente a esta aula

FIAP

- ☐ **Linguagem de definição de dados**

- ☐ DRS/DQL (SELECT)

- ☐ Operadores

- ☐ Expressões

- ☐ Cláusulas

- ☐ Exercícios

Linguagem SQL

SQL - Structured Query Language

(Linguagem Estruturada de Consulta)

FIAP



É uma **linguagem de definição, manipulação e controle de banco de dados.**

DRS – Linguagem para recuperação de dados ou

DQL – Linguagem para consulta de dados

Utilizada pesquisar e apresentar dados já armazenados no banco de dados.

Exemplo: SELECT .

Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

```
SELECT [ DISTINCT | ALL ] { * | coluna [, coluna, ... ] }  
FROM tabela  
WHERE condição  
ORDER BY coluna [, coluna, ... ]
```



Onde	Descrição
SELECT	Especifica as colunas (campos) desejadas na pesquisa.
DISTINCT	Não mostra eventuais valores repetidos de colunas.
ALL	Mostra todos os valores, mesmo que repetidos. Esse é o padrão se o DISTINCT não for definido. Portanto, não precisamos escrevê-lo.
*	Indica que devem ser mostradas todas as colunas da tabela
FROM	Determina em que tabelas serão efetuadas estas pesquisas.
WHERE	Condição para que se execute a pesquisa (filtra dados).
ORDER BY	Especifica em que ordem deverá ser apresentada a pesquisa desejada – por qual campo estará ordenada (de forma crescente ou decrescente).



Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

Utilizaremos o exercício **IMPLANTAÇÃO DE PROJETOS** na manipulação dos dados utilizando SQL.

Scripts necessários para os exemplos desta aula:

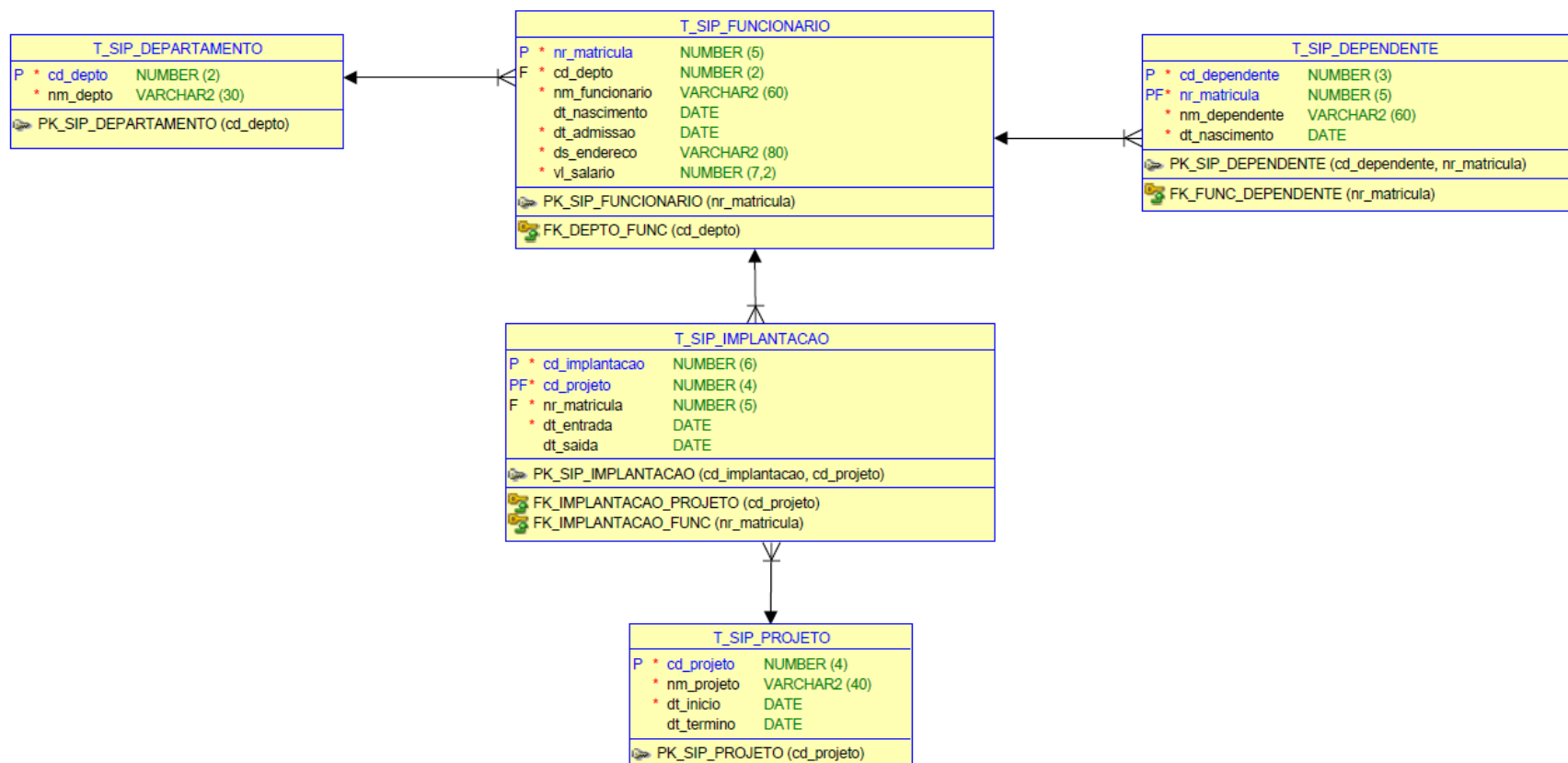
- ✓ [SCRIPT_AULA22_DDL_IMPLANTACAO_PROJETO.sql](#)
- ✓ [SCRIPT_AULA22_DML_IMPLANTACAO_PROJETO.sql](#)



Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

Modelo Relacional ou Físico – IMPLANTAÇÃO DE PROJETOS



Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

FIAP



EXEMPLOS: SELECT

Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

FIAP



1) Retorna todas as colunas e registros da tabela “FUNCIONARIO”:

```
SELECT * FROM T_EX01_FUNCIONARIO;
```

Atenção: O comando acima retornará todas as colunas (*) e linhas da tabela funcionário.

Obs.: Não é uma boa prática utilizar a cláusula Select *, pois todas as colunas são retornadas, independentemente das colunas que serão utilizadas realmente.



Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

FIAP



2) Retorna as colunas especificadas e todos os registros da tabela

“FUNCIONARIO”:

```
SELECT      NR_MATRICULA ,  
            CD_DEPARTAMENTO ,  
            DT_ADMISSAO ,  
            VL_SALARIO  
  
FROM  T_EX01_FUNCIONARIO;
```

Atenção: O comando acima retornará apenas as colunas especificadas na consulta para todas as linhas da tabela funcionário.



Linguagem SQL

FIAP

Consulta dos Dados utilizando a linguagem SQL



3) Retorna as colunas especificadas e todos os registros da tabela “FUNCIONARIO”, conforme a condição especificada:

```
SELECT      NR_MATRICULA ,  
            CD_DEPARTAMENTO ,  
            DT_ADMISSAO ,  
            VL_SALARIO  
  
FROM  T_EX01_FUNCIONARIO  
  
WHERE  CD_DEPARTAMENTO=3;
```



Atenção: O comando acima retornará apenas as colunas especificadas na consulta e, em relação as linhas, somente aquelas que satisfaçam a condição especificada.

Dê preferência a consultas onde as colunas e filtro são especificados, com isso evitamos retornar informações que não serão utilizadas pelo usuário.



Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

4) Exemplo utilizando DISTINCT

```
SELECT    DISTINCT CD_DEPARTAMENTO
FROM      T_EX01_FUNCIONARIO;
```

	CD_DEPARTAMENTO
1	1
2	2
3	4
4	3



Atenção: O comando acima retornará todos os departamentos da tabela funcionário sem repetição do código do cargo.

Conteúdo completo da tabela “FUNCIONARIO”.

	CD_DEPARTAMENTO
1	1
2	2
3	1
4	3
5	3
6	4





Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

5) Exemplo utilizando DISTINCT

```
SELECT DISTINCT CD_DEPARTAMENTO,
                DT_ADMISSAO
FROM T_EX01_FUNCIONARIO;
```

	CD_DEPARTAMENTO	DT_ADMISSAO
1	3	05/04/2010 12:00
2	1	11/02/2003 12:00
3	4	17/03/2009 12:00
4	1	09/08/2010 12:00
5	2	07/06/2005 12:00
6	3	08/09/2011 12:00

Atenção: Quando utilizamos 2 ou mais colunas no select, será considerado o conjuntos de colunas e não apenas uma das colunas.

Conteúdo da tabela “FUNCIONARIO” sem aplicação do DISTINCT.

	CD_DEPARTAMENTO	DT_ADMISSAO
1	1	09/08/2010 12:00
2	2	07/06/2005 12:00
3	1	11/02/2003 12:00
4	3	08/09/2011 12:00
5	3	05/04/2010 12:00
6	4	17/03/2009 12:00
7	4	17/03/2009 12:00



Linguagem SQL

FIAP

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – Operadores Relacionais



OPERADOR	SIGNIFICADO	EXEMPLO
=	Igual	codigo=2
<	Menor que	preco<10
<=	Menor ou igual a	preco<=10
>	Maior que	preco>10
>=	Maior ou igual a	preco>=10
<> Ou !=	diferente	codigo<>2





Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – Operadores Relacionais

EXEMPLOS utilizando operadores RELACIONAIS

-- FILTRANDO POR NÚMEROS

```
SELECT NR_MATRICULA, NM_NOME, VL_SALARIO  
FROM T_EX01_FUNCIONARIO  
WHERE VL_SALARIO > 1500.50;
```

-- FILTRANDO POR TEXTO

```
SELECT NR_MATRICULA, NM_NOME  
FROM T_EX01_FUNCIONARIO  
WHERE NM_NOME='ROSA MARIA';
```

-- FILTRANDO POR DATA

```
SELECT NR_MATRICULA, NM_NOME, DT_ADMISSAO  
FROM T_EX01_FUNCIONARIO  
WHERE DT_ADMISSAO > TO_DATE('01/08/2010', 'DD/MM/YYYY');
```

Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – Operadores Lógicos



OPERADOR	DESCRIÇÃO
AND	Retorna TRUE se ambas as condições forem verdadeiras
OR	Retorna TRUE se ao menos uma das condições for verdadeira
NOT	Retorna TRUE se a condição for falsa

Atenção no momento de montar uma condição composta, pois existe ordem de precedência:

- ✓ NOT, AND e OR, que pode ser alterada com a inclusão de parênteses.



Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – Operadores Lógicos

EXEMPLOS utilizando operadores LÓGICOS

-- OPERADOR AND

```
SELECT NR_MATRICULA, NM_NOME, CD_DEPARTAMENTO, DT_ADMISSAO
FROM T_EX01_FUNCIONARIO
WHERE DT_ADMISSAO > TO_DATE('01/08/2010', 'DD/MM/YYYY') AND
      CD_DEPARTAMENTO=3;
```

-- OPERADOR OR

```
SELECT NR_MATRICULA, NM_NOME, CD_DEPARTAMENTO, VL_SALARIO
FROM T_EX01_FUNCIONARIO
WHERE CD_DEPARTAMENTO=4 OR VL_SALARIO>3000;
```

-- OPERADOR NOT

```
SELECT NR_MATRICULA, NM_NOME, CD_DEPARTAMENTO, VL_SALARIO
FROM T_EX01_FUNCIONARIO
WHERE NOT CD_DEPARTAMENTO=4;
```

Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – Operadores Especiais



OPERADOR	DESCRIÇÃO
BETWEEN... AND ...	Entre dois valores (inclusive)
IN (<i>lista</i>)	Compara o valor de uma coluna com um conjunto de valores.
LIKE <i>valor</i>	Compara cadeia de caracteres
IS NULL / IS NOT NULL	É um valor nulo. Não nulo.

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – Operadores Especiais (LIKE)



EXPRESSÃO	DESCRIÇÃO
LIKE 'A%'	Todas as palavras que iniciem com A.
LIKE '%A'	Todas as palavras que terminem com a letra A.
LIKE '%A%'	Todas as palavras que tenham a letra A em qualquer posição.
LIKE 'A_'	String de dois caracteres que tenham a primeira letra A e o segundo caractere seja qualquer outro.
LIKE '_A'	String de dois caracteres cujo primeiro caractere seja qualquer um e a última letra seja A.
LIKE '_A_'	String de três caracteres cuja segunda letra seja A independentemente do primeiro ou do último caractere.
LIKE '%A_'	Todas as palavras que tenham a letra A na penúltima posição e a última seja qualquer outro caractere.
LIKE '_A%'	Todos que tenham a letra A na segunda posição e o primeiro caractere seja qualquer um.



Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – Operadores Especiais

EXEMPLOS utilizando operadores ESPECIAIS

-- OPERADOR BETWEEN

```
SELECT NR_MATRICULA, NM_NOME, CD_DEPARTAMENTO, VL_SALARIO  
FROM   T_EX01_FUNCIONARIO  
WHERE  VL_SALARIO BETWEEN 600 AND 1200;
```

```
SELECT NR_MATRICULA, NM_NOME, CD_DEPARTAMENTO, DT_ADMISSAO  
FROM   T_EX01_FUNCIONARIO  
WHERE  DT_ADMISSAO BETWEEN TO_DATE('01/01/2011', 'DD/MM/YYYY') AND  
                                TO_DATE('30/12/2011', 'DD/MM/YYYY');
```

```
SELECT NR_MATRICULA, NM_NOME, CD_DEPARTAMENTO, DT_ADMISSAO  
FROM   T_EX01_FUNCIONARIO  
WHERE  NM_NOME BETWEEN 'J' AND 'RU';
```



Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – Operadores Especiais

EXEMPLOS utilizando operadores ESPECIAIS

-- OPERADOR IN

```
SELECT NR_MATRICULA, NM_NOME, CD_DEPARTAMENTO
FROM   T_EX01_FUNCIONARIO
WHERE  CD_DEPARTAMENTO IN (1,2,3);
```

-- OPERADOR IN

```
SELECT      NR_MATRICULA ,
            DT_ADMISSAO ,
            VL_SALARIO
FROM        T_EX01_FUNCIONARIO
WHERE       TO_CHAR(DT_ADMISSAO, 'YYYY') IN ('2010', '2011');
```

Nota: TO_CHAR (<conteudo>,<formato>): Converte o conteúdo informado para uma string. Podemos utilizar converter uma data/horário em uma string.



Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – Operadores Especiais

EXEMPLOS utilizando operadores ESPECIAIS

-- OPERADOR NOT IN

```
SELECT NR_MATRICULA, NM_NOME, CD_DEPARTAMENTO
FROM   T_EX01_FUNCIONARIO
WHERE  CD_DEPARTAMENTO NOT IN (1,2,3);
```

-- OPERADOR IS NOT NULL

```
SELECT NR_MATRICULA, NM_NOME, DT_NASCIMENTO
FROM   T_EX01_FUNCIONARIO
WHERE  DT_NASCIMENTO IS NOT NULL;
```

-- OPERADOR IS NULL

```
SELECT NR_MATRICULA, NM_NOME, DT_NASCIMENTO
FROM   T_EX01_FUNCIONARIO
WHERE  DT_NASCIMENTO IS NULL;
```




Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – Operadores Especiais

EXEMPLOS utilizando operadores ESPECIAIS

-- OPERADOR LIKE

```
SELECT CD_DEPARTAMENTO, NM_DEPARTAMENTO
FROM   T_EX01_DEPARTAMENTO
WHERE  NM_DEPARTAMENTO LIKE 'FINA%';
```

```
SELECT CD_DEPARTAMENTO, NM_DEPARTAMENTO
FROM   T_EX01_DEPARTAMENTO
WHERE  NM_DEPARTAMENTO LIKE '_A%';
```



Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – Operadores Especiais

EXEMPLOS utilizando operadores ESPECIAIS

-- OPERADOR LIKE

```
SELECT CD_DEPARTAMENTO, NM_DEPARTAMENTO
FROM   T_EX01_DEPARTAMENTO
WHERE  NM_DEPARTAMENTO LIKE '____UR%';
```

```
SELECT CD_DEPARTAMENTO, NM_DEPARTAMENTO
FROM   T_EX01_DEPARTAMENTO
WHERE  NM_DEPARTAMENTO LIKE '%MER%';
```

Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

FIAP



Pesquisa Básica em Tabelas – CLAÚSULA ORDER BY

Esta cláusula vem por último na declaração SELECT e pode ordenar o resultado em ordem ascendente (ASC, que é o default) e descendente (DESC).

Podemos ordenar por mais de uma coluna.

A cláusula ORDER BY aceita números representando a coluna na ordem em que foi declarada na cláusula SELECT, além de aceitar o alias dado a alguma coluna.



Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – CLÁUSULA ORDER BY

EXEMPLOS utilizando a cláusula ORDER BY

-- ORDER BY ASC

```
SELECT      CD_DEPARTAMENTO, NM_DEPARTAMENTO
FROM        T_EX01_DEPARTAMENTO
ORDER BY    NM_DEPARTAMENTO;
```

-- ORDER BY DESC

```
SELECT      CD_DEPARTAMENTO, NM_DEPARTAMENTO
FROM        T_EX01_DEPARTAMENTO
ORDER BY    NM_DEPARTAMENTO DESC;
```



Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – CLÁUSULA ORDER BY

EXEMPLOS utilizando a cláusula ORDER BY

-- ORDER BY ASC E DESC

```
SELECT      NR_MATRICULA, CD_DEPARTAMENTO, NM_NOME, VL_SALARIO
FROM        T_EX01_FUNCIONARIO
WHERE       CD_DEPARTAMENTO >1
ORDER BY    CD_DEPARTAMENTO ASC, NM_NOME ASC, VL_SALARIO DESC;
```

-- ORDER BY UTILIZANDO NÚMEROS QUE REPRESENTAM AS COLUNAS

```
SELECT      NR_MATRICULA, CD_DEPARTAMENTO, NM_NOME, VL_SALARIO
FROM        T_EX01_FUNCIONARIO
WHERE       CD_DEPARTAMENTO >1
ORDER BY    2 ASC, 3 ASC, 4 DESC;
```

Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL



Qualificadores de Nome

Consiste no nome da tabela seguido de um ponto e o nome da coluna da tabela, por exemplo:

```
SELECT  T_EX01_DEPARTAMENTO.NM_DEPARTAMENTO  
        Nome da tabela      Nome do campo  
FROM    T_EX01_DEPARTAMENTO;
```

Colocar a identificação (qualificador) é opcional, porém é uma prática recomendada para facilitar o entendimento do comando.

Porém quando estivermos recuperando colunas com mesmo nome, em tabelas diferentes, se faz necessário informar a tabela que se deseja recuperar a informação.

Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – QUALIFICADORES DE NOME



Qualificadores de Nome – EXEMPLO

```
SELECT T_EX01_FUNCIONARIO.NR_MATRICULA ,  
       T_EX01_FUNCIONARIO.CD_DEPARTAMENTO ,  
       T_EX01_FUNCIONARIO.DT_ADMISSAO ,  
       T_EX01_FUNCIONARIO.VL_SALARIO  
FROM T_EX01_FUNCIONARIO;
```

Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

FIAP



ALIAS – APELIDO PARA COLUNAS

Às vezes, ao retornar dados de um select, o nome da coluna (campo) não é muito claro ou apropriado para a apresentação dos dados.

Por isso, podemos usar um “apelido” para a coluna, que chamamos de “ALIAS”.

É possível renomear o cabeçalho da coluna.

Muito útil quando queremos retornar um nome adequado para o resultado de um cálculo.

Deve ser escrito após o nome da coluna/cálculo, se for uma palavra composta deve estar entre aspas duplas.



Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – ALIAS

ALIAS (APELIDO) PARA COLUNAS – EXEMPLO

-- ALIAS (APELIDO) PARA COLUNAS

```
SELECT      NR_MATRICULA AS MATRICULA ,  
            CD_DEPARTAMENTO DEPARTAMENTO ,  
            DT_ADMISSAO  "DATA ADMISSÃO",  
            VL_SALARIO   SALARIO  
  
FROM T_EX01_FUNCIONARIO;
```



	 MATRICULA	 DEPARTAMENTO	 DATA ADMISSÃO	 SALARIO
1	1234		1 09/08/2010 12:00	1234,56
2	1235		2 07/06/2005 12:00	2345
3	3245		1 11/02/2003 12:00	7654,99
4	2233		3 08/09/2011 12:00	3452,12
5	7866		3 05/04/2010 12:00	980
6	9876		4 17/03/2009 12:00	1789
7	9899		4 17/03/2009 12:00	1780

Linguagem SQL

FIAP

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – ALIAS

ALIAS (APELIDO) PARA TABELAS E QUALIFICADOR_



EXEMPLO

Podemos utilizar apelidos para as tabelas, assim não teremos uma digitação redundante em relação aos nomes das tabelas.

-- APELIDOS PARA TABELAS E QUALIFICADORES

```
SELECT FUNC.NR_MATRICULA ,  
        FUNC.CD_DEPARTAMENTO ,  
        FUNC.DT_ADMISSAO ,  
        FUNC.VL_SALARIO  
FROM T_EX01_FUNCIONARIO FUNC;
```



Linguagem SQL

FIAP



Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – SUBQUERY (SUBCONSULTA)

CONSULTAS COM SUBQUERY – EXEMPLO

```
SELECT      NR_MATRICULA ,
            CD_DEPARTAMENTO ,
            DT_ADMISSAO ,
            VL_SALARIO

FROM        T_EX01_FUNCIONARIO

WHERE       CD_DEPARTAMENTO IN

            (

                SELECT CD_DEPARTAMENTO

FROM        T_EX01_DEPARTAMENTO

WHERE       UPPER(NM_DEPARTAMENTO) LIKE ' %MER%'

            );
```

Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

FIAP



Pesquisa Básica em Tabelas – CÁLCULOS

CONSULTAS CONTENDO COLUNAS CALCULADAS – EXEMPLO

-- CALCULAR O SALÁRIO ANUAL DE UM FUNCIONARIO E EXIBIR
ATRAVÉS DA CONSULTA

```
SELECT      NR_MATRICULA ,  
            CD_DEPARTAMENTO ,  
            DT_ADMISSAO ,  
            VL_SALARIO ,
```

	NR_MATRICULA	CD_DEPARTAMENTO	DT_ADMISSAO	VL_SALARIO	SALÁRIO ANUAL
1	1234		1 09/08/2010 12:00	1234,56	14814,72
2	1235		2 07/06/2005 12:00	2345	28140
3	3245		1 11/02/2003 12:00	7654,99	91859,88
4	2233		3 08/09/2011 12:00	3452,12	41425,44
5	7866		3 05/04/2010 12:00	980	11760
6	9876		4 17/03/2009 12:00	1789	21468
7	9899		4 17/03/2009 12:00	1780	21360

```
(VL_SALARIO * 12) "SALÁRIO ANUAL"
```

```
FROM      T_EX01_FUNCIONARIO;
```

Atenção:

Podemos utilizar os operadores aritméticos: (+ - * /).



Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

FIAP



OPERADOR DE CONCATENAÇÃO (||)

Este operador permite a concatenação de colunas ou string de caracteres com outras colunas.

É representado por duas colunas verticais ||.

É criada uma coluna resultante que é uma expressão de caracteres.

Linguagem SQL

FIAP

Consulta dos Dados utilizando a linguagem SQL Pesquisa Básica em Tabelas – CONCATENAÇÃO (II)



CONSULTAS CONTENDO COLUNAS CONCATENADAS -

EXEMPLO

-- EXIBIR O TEXTO: "O FUNCIONARIO <NOME> FOI

ADMITIDO EM <DATA ADMISSAO>"

```
SELECT      NR_MATRICULA ,  
            'O FUNCIONARIO ' ||  
            NM_NOME ||  
            ', FOI ADMITIDO EM ' ||  
            DT_ADMISSAO  
FROM        T_EX01_FUNCIONARIO;
```

Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

RESTRINGINDO A QUANTIDADE DE LINHAS DURANTE UMA CONSULTA

UTILIZANDO PSEUDOCOLUNA **ROWNUM**



Utilizada para limitar o retorno de uma consulta (em número de linhas).

É retornada uma amostra dos dados, podemos dizer que o resultado é um subconjunto do total de linhas a ser retornada.

Todas as tabelas possuem a pseudocoluna ROWNUM (retorna um número que indica a ordem em que o Oracle seleciona a linha a partir de uma tabela).

Podemos utilizar, por exemplo, para recuperar um certo número de linhas por vez, para realizar uma paginação em uma página WEB.

Linguagem SQL

FIAP



Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – ROWNUM

CONSULTAS UTILIZANDO ROWNUM – EXEMPLO

```
SELECT      ROWNUM ,  
  
            CD_DEPARTAMENTO ,  
  
            NM_DEPARTAMENTO  
  
FROM        T_EX01_DEPARTAMENTO;
```

	ROWNUM	CD_DEPARTAMENTO	NM_DEPARTAMENTO
1	1	1	FINANCEIRO
2	2	2	MARKETING
3	3	3	RECURSOS HUMANOS
4	4	4	COMERCIAL

Linguagem SQL

FIAP

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – ROWNUM



CONSULTAS UTILIZANDO ROWNUM – EXEMPLO

RESTRINGINDO A QUANTIDADE DE LINHAS RECUPERADAS

DURANTE A CONSULTA

```
SELECT      NR_MATRICULA ,  
            CD_DEPARTAMENTO ,  
            DT_ADMISSAO ,  
            VL_SALARIO ,  
            (VL_SALARIO * 12) "SALÁRIO ANUAL"  
  
FROM        T_EX01_FUNCIONARIO  
  
WHERE       ROWNUM <4;
```

R	NR_MATRICULA	R	CD_DEPARTAMENTO	R	DT_ADMISSAO	R	VL_SALARIO	R	SALÁRIO ANUAL
1	1234			1	09/08/2010 12:00		1234,56		14814,72
2	1235			2	07/06/2005 12:00		2345		28140
3	3245			1	11/02/2003 12:00		7654,99		91859,88

Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

FIAP



CONSULTAS UTILIZANDO ROWNUM (ANÁLISE TOP-N)

Esse tipo de consulta é útil em situações onde precisamos encontrar, por exemplo, os N maiores ou menores dentro de determinado grupo, os funcionários mais bem pagos, os que vendem mais, as mercadorias menos vendidas e assim por diante.

Para conseguirmos isso no Oracle precisaremos implementar uma consulta interna, que irá retornar todos os registros necessários, e na consulta externa iremos limitar a exibição num total desejado.

Este processo é conhecido como consulta TOP-N no Oracle.

Linguagem SQL

FIAP

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – ROWNUM



CONSULTAS UTILIZANDO ROWNUM – EXEMPLO

-- PARA EXIBIR OS NOMES E OS SALÁRIOS DOS TRÊS FUNCIONÁRIOS

-- MAIS BEM REMUNERADOS DA TABELA FUNCIONARIO.

```
SELECT  ROWNUM as RANK ,
        NM_NOME ,
        VL_SALARIO
FROM
    (
        SELECT NM_NOME ,
               VL_SALARIO
        FROM T_EX01_FUNCIONARIO
        ORDER BY VL_SALARIO DESC
    )
WHERE ROWNUM <= 3;
```

	RANK	NM_NOME	VL_SALARIO
1	1	ANTONIA CAMARGO	7654,99
2	2	JOÃO DA SILVA	3452,12
3	3	ROSA MARIA	2345

Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

Regras de Precedência



ORDEM DE AVALIAÇÃO	OPERADOR
1	Operadores aritméticos
2	Operador de concatenação
3	Condições de comparação
4	IS [NOT] NULL, LIKE, [NOT] IN
5	[NOT] BETWEEN
6	Condição lógica NOT
7	Condição lógica AND
8	Condição lógica OR

Utilize o parênteses para sobrepor as regras de precedência.

Próxima aula estudaremos

- ❑ Revisão de conceitos através de exercícios

I REFERÊNCIAS



- MACHADO, Felipe Nery R. Banco de Dados - Projeto e Implementação. Érica, 2004.
- Páginas: 330, 331.
- ELMASRI, R.; NAVATHE, S.B. Sistemas de Banco de Dados: Fundamentos e Aplicações. Pearson, 2005. Páginas: 153, 154.
- PRICE, JASON, ORACLE DATABASE 11 g – SQL Domine SQL e PL-SQL no banco de Dados Oracle, Bookman, 2008. Capítulos: 2, 5 e 6.
- **Outros:**
- Manual Oficial Oracle – Introdução ao Oracle 9i (SQL) - **Oracle Corporation, 2000, 2001.**

Copyright © 2016 Profa. Rita de Cássia Rodrigues

Todos direitos reservados. Reprodução ou divulgação total ou parcial deste documento é expressamente proibido sem o consentimento formal, por escrito, do Professor (autor).