

Prática em Banco de Dados

Banco de Dados

UNIP

Email: saulo.silva@docente.unip.br

Antes de prosseguirmos...

structured query language

- O SQL (Linguagem de Consulta Estruturada) é a linguagem padrão para os bancos de dados **relacionais**.
- Ela foi desenvolvida e criada pela IBM e após diversas variações que começaram a surgir da linguagem, foi necessária à sua adaptação e padronização.
- Embora padronizada pela ANSI e ISO, diferentes bancos de dados relacionais possuem **adaptações da linguagem** com variações para diferentes tipos de bancos de dados.

Antes de prosseguirmos...

- Os comandos SQL do Banco de Dados Oracle são divididos em 3 categorias:
- **DML – Data Manipulation Language**
 - **Select** – Selecionar as colunas de uma tabela **Insert** – Inserir dados em uma tabela
 - **Update** – Atualiza os dados de uma tabela
 - **Delete** – Exclui registros de uma tabela

X

?

Antes de prosseguirmos...

- **DCL – Data **Control** Language**
 - **Grant** – atribuição de privilégios para um usuário acessar/manipular objetos de um outro usuário
 - **Revoke** – revogar os privilégios de acesso/manipulação dos objetos de um usuário
- **DDL – Data **Definition** Language**
 - **Alter** – Alterar estrutura de objetos do banco de dados
 - **Create** – Criação de objetos no banco de dados
 - **Drop** – Excluir objetos do banco de dados
 - **Truncate** – Excluir todos os registros de uma tabela

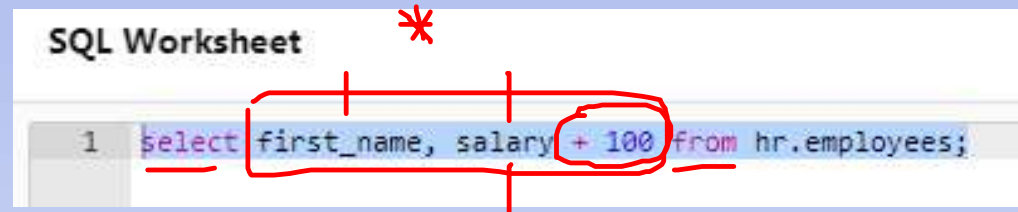
Utilizando expressões aritméticas!

- As consultas SQL (queries) podem ser utilizadas combinadas com expressões aritméticas.

Operadores	Descrição
+	Adição
-	Subtração
/	Divisão
*	Multiplicação

Utilizando expressões aritméticas!

- Se fizermos um select da maneira descrita abaixo, estaremos atribuindo um aumento de \$100 ao campo salary na tabela hr.employees:



The screenshot shows an SQL Worksheet with the following query: `select first_name, salary + 100 from hr.employees;`. Red annotations highlight the expression `salary + 100` with a red box and a red asterisk above it. A red line also underlines the `from hr.employees;` part of the query.

- Observe os resultados...



FIRST_NAME	SALARY+100
Steven	24100
Neena	17100
Lex	17100
Alexander	9100
Bruce	6100
David	4900
Valli	4900
Diana	4300

Utilizando expressões aritméticas!

- Da mesma forma, se fizermos o select multiplicando a coluna salary por 12, conseguiremos saber qual o salário de um ano, já com o aumento:

SQL Worksheet

1 `select first_name, 12 * (salary + 100) from hr.employees;`

EA



FIRST_NAME	<u>12*(SALARY+100)</u>
Steven	289200
Neena	205200
Lex	205200
Alexander	109200
Bruce	73200
David	58800
Valli	58800
Diana	51600

Restrição e Classificação de dados

- Para restringir as linhas retornadas utilizamos a cláusula **WHERE** que é sempre utilizada após a cláusula FROM:

Operadores	Descrição
▪ =	Igual a
▪ >	Maior que
▪ >=	Maior que ou igual a
▪ <	Menor que
▪ <=	Menor que ou igual a
▪ <>	Diferente de
▪ between...and	Entre dois valores
▪ IN(valores)	Sequência de busca
▪ LIKE	Busca por padrão
▪ IS NULL	É um valor nulo

! =

Restrição e Classificação de dados

- Para exibir todos os registros cujo campo DEPARTMENT_NAME tenha valor igual a 'Marketing':

```
select * from hr.departments  
where department_name='Marketing';
```



DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
20	Marketing	201	1800


[Download CSV](#)

Restrição e Classificação de dados

- Para exibir todos os registros da tabela employees, filtrando todos os funcionários com salário menos que \$10000:

SQL Worksheet

```
1 select * from hr.employees where salary <10000;
```



EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
103	Alexander	Hunold	AHUNOLD	590.423.4567	03-JAN-06	IT_PROG	9000	-	102	60
104	Bruce	Ernst	BERNST	590.423.4568	21-MAY-07	IT_PROG	6000	-	103	60
105	David	Austin	DAUSTIN	590.423.4569	25-JUN-05	IT_PROG	4800	-	103	60
106	Valli	Pataballa	VPATABAL	590.423.4560	05-FEB-06	IT_PROG	4800	-	103	60
107	Diana	Lorentz	DLORENTZ	590.423.5567	07-FEB-07	IT_PROG	4200	-	103	60
109	Daniel	Faviet	DFAVIET	515.124.4169	16-AUG-02	FI_ACCOUNT	9000	-	108	100
110	John	Chen	JCHEN	515.124.4269	28-SEP-05	FI_ACCOUNT	8200	-	108	100
111	Ismael	Sciarra	ISCIARRA	515.124.4369	30-SEP-05	FI_ACCOUNT	7700	-	108	100

Restrição e Classificação de dados

- Da mesma forma, podemos selecionar e exibir todos os departamentos da localidade 1700:

```
1 select * from hr.departments where location_id = 1700;
```



DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	Administration	200	1700
30	Purchasing	114	1700
90	Executive	100	1700
100	Finance	108	1700
110	Accounting	205	1700
120	Treasury	-	1700
130	Corporate Tax	-	1700
140	Control And Credit	-	1700

Restrição e Classificação de dados

- Para definirmos faixas de valores a serem exibidas, devemos usar o IN e o BETWEEN:
- A query abaixo utiliza o BETWEEN para retornar todos os departamentos com código entre 10 e 30:

```
select * from hr.departments  
where department_id between 10 and 30;
```

Restrição e Classificação de dados

- A query abaixo utiliza o IN para apresentar todos os departamentos que sejam referentes a Administração e Marketing:

```
select * from hr.departments  
where department_name in ('Administration', 'Marketing');
```

- Da mesma forma, o NOT IN permite que se se apresente todos os departamentos, exceto os de Administração e Marketing:

```
select * from hr.departments  
where department_name not in ('Administration', 'Marketing');
```

Restrição e Classificação de dados

- Também é possível procurar segmentos de caracteres dentro dos campos, mediante uso da cláusula LIKE:
 - '%a%' – Qualquer termo que contenha a letra a (minúscula).
 - '%m' – Qualquer termo que termine com a letra m (minúscula).
 - 'N%' – Qualquer termo que comece com a letra N (maiúscula).

```
select * from hr.departments  
where department_name like 'M%';
```

Restrição e Classificação de dados

- Os comandos **IS NULL** e **IS NOT NULL** permitem pesquisar os campos em branco da tabela:
- O **IS NULL** permite retornar todos os registros que contenham a coluna `manager_id` como nula:

```
select * from hr.employees  
where manager_id is null;
```

- O **IS NOT NULL** permite retornar somente os registros da coluna `manager_id` que não sejam nulos.

```
select * from hr.employees  
where manager_id is not null;
```

Exercícios:

24h

- 1. Apresentar, da tabela EMPLOYEES, todos os dados dos funcionários que recebam entre \$6000 e \$9000.
- 2. Apresentar, da tabela EMPLOYEES, todos os dados dos funcionários cujo cargo seja FI_ACCOUNT e FI_MGR.
- 3. Apresentar, da tabela EMPLOYEES, todos os dados dos funcionários que recebam abaixo de \$4000, aplicando-lhes um aumento de 20%.
- 4. Apresentar, da tabela DEPARTMENTS, todos os dados dos departamentos cujo nome se inicie com **A**.
- 5. Apresentar, da tabela DEPARTMENTS, todos os dados dos departamentos cujo nome termine com a letra **g**.
- 6. Apresentar, da tabela DEPARTMENTS, todos os dados dos departamentos cujo nome contenha as letras '**atio**'.