

# SQL – DML

---

BASE DE DADOS – JAVA.NET

# Conteúdo

---

- **SQL**
- **Comandos DML**
- **INSERT**
  - Sintaxe
  - Exemplos
- **UPDATE - sintaxe**
- **DELETE - sintaxe**
- **SELECT**
  - sintaxe
  - Clausulas WHERE, GROUP BY, HAVING e ORDER BY

# SQL

---

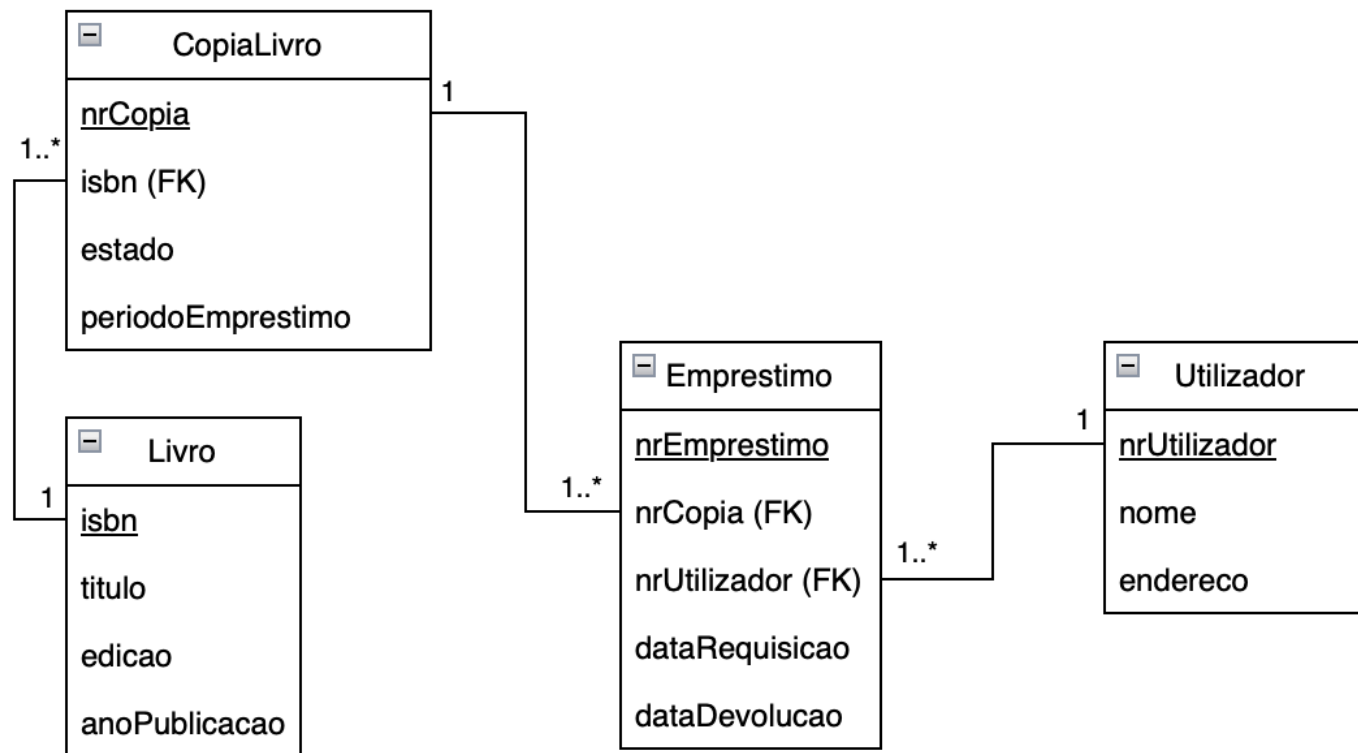
- **Structured Query Language**
- Baseada na álgebra relacional, nomeadamente nos operadores definidos para manipulação das relações
- Operações agrupadas em:
  - ⊖ ~~Data Definition Language (DDL) – definição da estrutura e do controlo de acesso aos dados, e.g. CREATE TABLE, ALTER TABLE~~
  - Data Manipulation Language (DML) – acesso e manipulação de dados – INSERT, UPDATE, DELETE, SELECT
  - Data Control Language (DCL) – definição de permissões de acesso a objetos da base de dados – e.g. GRANT, REVOKE
  - Transaction Control Language (TCL) – manipulação de transações – e.g. COMMIT, ROLLBACK

# SQL- Comandos DML

---

- **Inserção**
  - INSERT
  - Inserção de tuplos na tabela
- **Alteração**
  - UPDATE
  - Atualização de tuplos da tabela
- **Eliminação**
  - DELETE
  - Eliminação de tuplos da tabela
- **Consulta**
  - SELECT
  - Obtenção de dados/informação da BD

## ■ Relembrar o MR



# Inserção Tuplos

---

# SQL - INSERT

## INSERT INTO

```
nome_tabela [(coluna_1, coluna_2, ... coluna_n)]  
VALUES (valor_1, valor_2, ..., valor_n);
```

- Insere em *nome\_tabela* os valores especificados em *valor\_1* a *valor\_n*
- Os nomes das colunas, *coluna\_1* a *coluna\_n*, devem ser especificados
- Os campos com valor por omissão ou gerados podem ser omitidos
- Caso alguma RI seja violada o comando falha com a indicação da RI
- Em caso de sucesso é adicionado o tuplo



# SQL - INSERT

---

INSERT INTO livro (isbn, titulo, edicao, ano)

VALUES (9780140449136, 'Crime e Castigo', 4, TO\_DATE('1866-01-01', 'YYYY-MM-DD'));

INSERT INTO livro (isbn, titulo, edicao, ano)

VALUES (9780451524935, '1984', 6, TO\_DATE('1949-06-08', 'YYYY-MM-DD'));

INSERT INTO livro (isbn, titulo, edicao, ano)

VALUES (9780061122415, 'O Sol é para Todos', 8, TO\_DATE('1960-07-11', 'YYYY-MM-DD'));

INSERT INTO livro (isbn, titulo, edicao, ano)

VALUES (9780062315007, 'Orgulho e Preconceito', 5, TO\_DATE('1813-01-28', 'YYYY-MM-DD'));

INSERT INTO livro (isbn, titulo, edicao, ano)

VALUES (9780547249643, 'Mataram a Cotovia', 3, TO\_DATE('1962-07-11', 'YYYY-MM-DD'));

INSERT INTO livro (isbn, titulo, edicao, ano)

VALUES (9780385472579, 'O Grande Gatsby', 10, TO\_DATE('1925-04-10', 'YYYY-MM-DD'));

# Alteração Tuplos

---

# SQL - UPDATE

---

UPDATE *nome\_tabela*

SET     *coluna\_1=valor\_1,*  
          *coluna\_2=valor\_2,*  
          ....  
          *coluna\_n=valor\_n*

[WHERE *condição*];

- Altera todos os campos dos tuplos de *nome\_tabela*, cuja *condição* se verifique
- Os valores dos *campos\_1* a *campo\_n* são atualizados com os respetivos valores (*valor\_1* a *valor\_n*)

# SQL - UPDATE

---

**UPDATE** *nome\_tabela*

**SET**     *coluna\_1=valor\_1,*  
          *coluna\_2=valor\_2,*  
          ....  
          *coluna\_n=valor\_n*

**[WHERE** *condição*];

■ ***valor\_1 a valor\_n*** pode ser:

- um valor discreto ...
- invocação de uma funções que existam no SGBD, ...
- .. desde que seja do mesmo tipo de dados da coluna e não violem nenhuma RI

# SQL - UPDATE

---

**UPDATE** *nome\_tabela*

**SET**     *coluna\_1=valor\_1,*  
          *coluna\_2=valor\_2,*  
          ....  
          *coluna\_n=valor\_n*

**[WHERE** *condição*];

■ ***condição* pode ser:**

- simples, i.e. apenas uma expressão ...
- composta, expressões associadas através de operadores relacionais AND e OR ...
- ..., onde expressão é algo que envolva operadores relacionais, e.g. =, !=, >, ou funções do SGBD

# SQL - UPDATE

---

**UPDATE livro**

**SET ano = TO\_DATE('1961-05-01',  
'YYYY-MM-DD')**

**WHERE isbn = 9780061122415;**

**UPDATE livro**

**SET titulo = 'Orgulho e Preconceito:  
Edição Especial', edicao = 6**

**WHERE isbn = 9780062315007;**

**UPDATE livro**

**SET edicao = 5, ano = TO\_DATE('1870-  
01-01', 'YYYY-MM-DD')**

**WHERE titulo = 'Crime e Castigo';**

**UPDATE livro**

**SET edicao = edicao+1, ano =  
TO\_DATE('1951-01-01', 'YYYY-MM-DD')**

**WHERE titulo = '1984';**

# Eliminação Tuplos

---

# SQL – DELETE

---

**DELETE FROM nome\_tabela**

**[WHERE condição];**

- **Elimina todos os tuplos de nome\_tabela que verifiquem a condição, onde condição pode ser:**
  - simples, i.e. apenas uma expressão ...
  - composta, expressões associadas através de operadores relacionais AND e OR ...
  - ..., onde expressão é algo que envolva operadores relacionais, e.g. =, !=, >, ou funções do SGBD



# SQL – DELETE

---

**DELETE FROM livro**

**WHERE edicao < 5;**

**DELETE FROM livro**

**WHERE ano < TO\_DATE('1900-01-01', 'YYYY-MM-DD');**

**DELETE FROM livro**

**WHERE titulo LIKE '%Gatsby%';**

**DELETE FROM livros;**

# Perguntas



A imagem [Esta Fotografia](#) de Autor Desconhecido está licenciada ao abrigo da [CC BY-SA](#)

# SELECT

---

# SQL - SELECT

```
SELECT [DISTINCT|ALL] {coluna_1, coluna_2, ..., coluna_n }| *  
FROM nome_tabela  
  
[WHERE condição_seleção]  
  
[GROUP BY {coluna_x }  
[HAVING condição_grupos]]  
  
[ORDER BY {coluna método_ordenação}];
```

- Retorna todos os tuplos de *nome\_tabela* que se enquadrem na *condição\_seleção*, ordenados pelo método especificado na cláusula ORDER BY
- A cláusula GROUP BY agrupa os tuplos retornados em grupos, e permite realizar operações sobre os grupos, *e.g.* calcular média, somar. Através da cláusula HAVING pode-se selecionar apenas alguns desses grupos. O resultado de um SELECT com estas cláusulas são tuplos dos grupos e respetivos valores das operações
- O resultado de um SELECT é uma nova tabela!

# SQL - SELECT

---

SELECT [DISTINCT|ALL] {coluna\_1, coluna\_2, ..., coluna\_n }| \*

Projeção

FROM nome\_tabela

Relação

[WHERE condição\_seleção]

Seleção

[GROUP BY {coluna\_x, }

Agrupar

[HAVING condição\_grupos]]

Seleção  
(com funções  
de agregação)

[ORDER BY {coluna método\_ordenação}];

Ordenação

# SQL - SELECT

## ■ Clausula WHERE, permite aplicação de:

- operadores lógicos: AND, OR, NOT
- operadores relacionais: >, <, >=, <=, <>, !=
- operadores associadas a valores contínuos: BETWEEN, NOT BETWEEN, *e.g.* WHERE salario BETWEEN 10 AND 20
- operadores associados a valores discretos: IN, NOT IN, *e.g.* WHERE tamanho IN ('S', 'L')
- +info:
  - <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/Conditions.html>
- funções do SGBD
  - +info: <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/Functions.html>

# SQL - SELECT

## ■ Clausula ORDER BY:

- definição dos critérios a aplicar na ordenação dos tuplos obtidos
- um critério é representado pela expressão *nome\_coluna tipo\_ordenação*, onde *tipo\_ordenação* pode ser ASC (ascendente) ou DESC (descendente). ASC é o critério aplicado por omissão.
- o nome da coluna pode ser substituído por um valor numérico, representativo da posição da coluna na projeção
- havendo mais que um critério, a aplicação do segundo ao nésimo critério é feita nos tuplos cujos valores dos campos até nésimo-1 critérios sejam iguais
- e.g.

SELECT codCliente, nome, data, valor, estado

FROM compra

ORDER BY data DESC, estado ASC      -- OU ORDER BY 3 DESC, 5

# SQL - SELECT

## ■ Clausula GROUP BY:

- agrupa os tuplos cujos valores das colunas incluídas na clausula sejam iguais
- permite a aplicação funções de agregação, *e.g. sum, count, avg, max, min*, sobre uma coluna que não estejam faça parte da clausula GROUP BY
- as funções podem ser aplicadas nas clausulas de projeção e HAVING

## ■ Clausula HAVING permite selecionar registos, pela aplicação de funções de agregação

- **+info:** <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/Aggregate-Functions.html>



# SQL - SELECT

```
select *  
from livro;
```

ISBN	TITULO	EDICAO	ANO
9788501070269	Ilíada	(null)	01/01/00
9788520912343	A Eneida	(null)	01/01/29
9788571104522	Divina Comédia	10	01/01/20
9788572322443	Dom Quixote	21	01/01/05
9788535902561	Macbeth	12	01/01/06
9788570413158	Candide	3	01/01/59
9788527417450	Guerra e Paz	19	01/01/69
9788501073642	Moby Dick	43	01/01/51
9788501062493	O Primo Basílio	21	01/01/78
9788535903070	Cem Anos de Solidão	17	05/06/67
9788520917461	1984	19	08/06/49
9788535902381	O Senhor dos Anéis	12	29/07/54
9788501073872	O Cortiço	8	01/01/90
9788535905597	O Grande Gatsby	9	10/04/25
9788572931084	O Caçador de Pipas	14	29/05/03

# SQL - SELECT

```
select isbn, titulo  
from livro;
```

ISBN	TITULO
9788501070269	Ilíada
9788520912343	A Eneida
9788571104522	Divina Comédia
9788572322443	Dom Quixote
9788535902561	Macbeth
9788570413158	Candide
9788527417450	Guerra e Paz
9788501073642	Moby Dick
9788501062493	O Primo Basílio
9788535903070	Cem Anos de Solidão
9788520917461	1984
9788535902381	O Senhor dos Anéis
9788501073872	O Cortiço
9788535905597	O Grande Gatsby
9788572931084	O Caçador de Pipas

# SQL - SELECT

```
select *  
from livro  
order by edicao, titulo desc;
```

ISBN	TITULO	EDICAO	ANO
9788570413158	Candide	3	01/01/59
9788501073872	0 Cortiço	8	01/01/90
9788535905597	0 Grande Gatsby	9	10/04/25
9788571104522	Divina Comédia	10	01/01/20
9788535902381	0 Senhor dos Anéis	12	29/07/54
9788535902561	Macbeth	12	01/01/06
9788572931084	0 Caçador de Pipas	14	29/05/03
9788535903070	Cem Anos de Solidão	17	05/06/67
9788527417450	Guerra e Paz	19	01/01/69
9788520917461	1984	19	08/06/49
9788501062493	0 Primo Basílio	21	01/01/78
9788572322443	Dom Quixote	21	01/01/05
9788501073642	Moby Dick	43	01/01/51
9788501070269	Ilíada	(null)	01/01/00
9788520912343	A Eneida	(null)	01/01/29

# SQL - SELECT

select \*  
from livro  
where lower(titulo) like 'o%';

select isbn, titulo, edicao  
from livro  
where edicao between 3 and 12  
order by edicao;

ISBN	TITULO	EDICAO	ANO
9788501062493	0 Primo Basílio	21	01/01/78
9788535902381	0 Senhor dos Anéis	12	29/07/54
9788501073872	0 Cortiço	8	01/01/90
9788535905597	0 Grande Gatsby		
9788572931084	0 Caçador de Pipas		

ISBN	TITULO	EDICAO
9788570413158	Candide	3
9788501073872	0 Cortiço	8
9788535905597	0 Grande Gatsby	9
9788571104522	Divina Comédia	10
9788535902561	Macbeth	12
9788535902381	0 Senhor dos Anéis	12

# SQL - SELECT

```
select isbn, periodoemprestimo,
case estado
  when 0 then 'emprestado'
  when 1 then 'disponível'
end estado_descricao
from copialivro;
```

ISBN	PERIODOEMPRESTIMO	ESTADO_DESCRICA0
9788501070269	7	disponível
9788520912343	14	disponível
9788571104522	7	disponível
9788572322443	7	disponível
9788535902561	10	disponível
9788535902561	10	emprestado
9788570413158	14	disponível
9788527417450	7	disponível
9788501073642	7	disponível
9788501073642	7	emprestado
9788535903070	7	disponível
9788535903070	7	emprestado
9788535903070	7	disponível

# SQL - SELECT

```
select edicao, count(*) "nrRegistos"  
from livro  
group by edicao  
order by 2 desc, 1 asc;
```

EDICA0	nrRegistos
12	2
19	2
21	2
(null)	2
3	1
8	1
9	1
10	1
14	1
17	1
43	1

# SQL - SELECT

```
select edicao, count(*) "nrRegistos"
from livro
where edicao is not null
group by edicao
order by 2 desc, 1 asc;
```

EDICA0	nrRegistos
12	2
19	2
21	2
3	1
8	1
9	1
10	1
14	1
17	1
43	1

# SQL - SELECT

`select titulo, ano, to_char(ano, 'yyyy') anolancamento, to_char(ano,'mm')  
meslancamento  
from livro;`

TITULO	ANO	ANOLANÇAMENTO	MESLANCAMENTO
Ilíada	01/01/00	0800	01
A Eneida	01/01/29	0029	01
Divina Comédia	01/01/20	1320	01
Dom Quixote	01/01/05	1605	01
Macbeth	01/01/06	1606	01
Candide	01/01/59	1759	01
Guerra e Paz	01/01/69	1869	01
Moby Dick	01/01/51	1851	01
O Primo Basílio	01/01/78	1878	01
Cem Anos de Solidão	05/06/67	1967	06
1984	08/06/49	1949	06
O Senhor dos Anéis	29/07/54	1954	07
O Cortiço	01/01/90	1890	01
O Grande Gatsby	10/04/25	1925	04
O Caçador de Pipas	29/05/03	2003	05



# SQL - SELECT

```
select titulo, ano, to_char(ano, 'yyyy') anolancamento, to_char(ano,'mm')
meslancamento
from livro
where to_char(ano, 'mm') in ( 4, 6 );
```

TITULO	ANO	ANOLANÇAMENTO	MESLANCAMENTO
Cem Anos de Solidão	05/06/67	1967	06
1984	08/06/49	1949	06
0 Grande Gatsby	10/04/25	1925	04

# SQL - SELECT

```
select to_char(ano, 'mm') mesLançamento, max(ano), min(ano)
from livro
group by to_char(ano, 'mm');
```

MESLANÇAMENTO	MAX(ANO)	MIN(ANO)
01	01/01/90	01/01/29
05	29/05/03	29/05/03
06	05/06/67	08/06/49
04	10/04/25	10/04/25
07	29/07/54	29/07/54

# Perguntas



A imagem [Esta Fotografia](#) de Autor Desconhecido está licenciada ao abrigo da [CC BY-SA](#)