

# SQL

---

- *SQL - Structured Query Language*
  - DDL - Data Definition Language
  - DML - Data Manipulation Language

# SQL

---

- *DDL - Data Definition Language*
  - Definição da estrutura das tabelas
  - Basicamente:
    - CREATE
    - ALTER
    - DROP

# SQL - DDL

---

*CREATE TABLE nome\_tabela*

*(*

*nome\_coluna data\_type [column\_constraint][,...],*

*[restrições\_tabela]*

*)*

*restrições\_tabela*: basicamente chave primária e estrangeira

# Tipos de Dados/Datatype

---

- Alguns tipos de dados são definidos apenas em alguns SGBD's
- PostgreSQL
  - <https://www.postgresql.org/docs/current/datatype.html>

# SQL - DDL

---

```
CREATE TABLE departamento  
    (coddepto int,  
     nomdepto varchar(50),  
     primary key (coddepto));
```

```
CREATE TABLE funcionario  
    (codfunc int,  
     nomfunc varchar(50),  
     coddepto int,  
     salario numeric (12,2),  
     primary key (codfunc),  
     foreign key (coddepto) references departamento(coddepto));
```

# SQL - DDL

---

```
CREATE TABLE departamento
    (coddepto int,
    nomdepto varchar(50),
    CONSTRAINT pk_depto PRIMARY KEY (coddepto));
```

```
CREATE TABLE funcionario
    (codfunc int,
    nomfunc varchar(50),
    coddepto int,
    salario numeric(12,2),
    CONSTRAINT pk_func PRIMARY KEY (codfunc),
    CONSTRAINT fk_depto_func
        FOREIGN KEY (coddepto) references departamento(coddepto));
```

# SQL - DDL

---

## Adicionar uma coluna

```
ALTER TABLE departamento add column sala char(4);
```

<https://www.postgresql.org/docs/current/sql-altertable.html>

## Apagar uma tabela e os dados

```
DROP TABLE departamento;
```

# SQL - DML

---

INSERT INTO TABELA (COLUNA1,...) VALUES (VALOR1,...);

insert into departamento (coddepto, nomdepto) values (1,'rh');

insert into departamento (coddepto, nomdepto) values (2,'contabilidade');

insert into departamento (coddepto, nomdepto) values (3,'marketing');

insert into departamento (coddepto, nomdepto) values (4,'financeiro');

insert into funcionario (codfunc, nomfunc, coddepto,salario) values (1,'joao',2,'1000.00');

insert into funcionario (codfunc, nomfunc, coddepto,salario) values (2,'ana',1,'1500.00');

insert into funcionario (codfunc, nomfunc, coddepto,salario) values (3,'maria',1,'1300.00');



# SQL - DML

---

SELECT [DISTINCT] \* | coluna(s) | funções  
FROM  
TABELA(S)  
[WHERE expressão]  
[GROUP BY expressão]  
[HAVING expressão]  
[ORDER BY critério]

# SQL - DML

---

Lista os dados de todas as colunas e linhas da tabela

```
SELECT * FROM funcionario;
```

Lista os nomes e salários dos funcionários

```
SELECT nomfunc,salario FROM funcionario;
```

# SQL - DML

---

Lista o número de funcionários na tabela

```
SELECT count(*) FROM funcionario;
```

Lista o número de funcionários na tabela que são do departamento com código = 1

```
SELECT count(*) FROM funcionario  
        WHERE coddepto=1;
```

# SQL - DML

---

Lista o valor do maior salário

```
SELECT max(salario) FROM funcionario;
```

Lista o valor do menor salário

```
SELECT min(salario) FROM funcionario;
```

Lista o valor da soma dos salários

```
SELECT sum(salario) FROM funcionario;
```

Lista o valor da média dos salários

```
SELECT avg(salario) FROM funcionario;
```

# SQL - DML

---

Lista os nomes dos funcionários ordenados pelo salário

```
SELECT nomfunc FROM funcionario  
ORDER BY salario;
```

Lista os nomes dos funcionários ordenados pelo salário. Mas somente os funcionários do departamento cujo código é 1

```
SELECT nomfunc, salario FROM funcionario  
WHERE coddepto = 1  
ORDER BY salario;
```

# SQL - DML

---

Lista os nomes dos funcionários ordenados pelo salário. Mas somente os funcionários do departamento cujo código é 1 e o salário é maior que 1.300,00

```
SELECT nomfunc, salario FROM funcionario  
      WHERE coddepto = 1 AND salario > '1300.00'  
      ORDER BY salario;
```

# SQL - DML

---

Lista os dados dos funcionários cujo nome começa com a 'jo'.

```
SELECT * FROM funcionario  
WHERE nomfunc LIKE 'jo%';
```

# SQL - DML

---

```
SELECT * FROM funcionario,departamento  
      where  
      funcionario.coddepto = departamento.coddepto;
```



# SQL - DML

---

```
SELECT * FROM funcionario,departamento;
```

**ATENÇÃO: PRODUTO CARTESIANO!!!!**

# SQL - DML

---

“ALIAS”

```
SELECT * FROM funcionario AS F ,departamento AS D
      where
      F.coddepto = D.coddepto;
```

# SQL - DML

---

UPDATE tabela

SET coluna = valor

[WHERE condição];

UPDATE departamento

set nomdepto='recursos humanos'

where coddepto=1;

# SQL - DML

---

```
DELETE FROM tabela  
        [WHERE condição];
```

```
DELETE FROM DEPARTAMENTO  
        where coddepto=3;
```