

# Modelo relacional e funções SQL

<LAB365>

**SENAI**

# OBJETIVOS

- Entender o que é e como utilizar funções SQL
- Entender o que é e como fazer um CRUD

# COMANDOS DDL (DATA DEFINITION LANGUAGE)

- Os comandos DDL são usados para definir e gerenciar estruturas de banco de dados, como tabelas e índices.
- CREATE**: Cria uma nova tabela, índice ou outro objeto no banco de dados.

```
CREATE TABLE usuarios (  
    id SERIAL PRIMARY KEY,  
    nome VARCHAR(50) NOT NULL,  
    cidade VARCHAR(50) NOT NULL,  
    email VARCHAR(100) UNIQUE NOT NULL,  
    data_nascimento DATE,  
    ativo BOOLEAN DEFAULT true  
);
```

```
CREATE TABLE pedidos (  
    id SERIAL PRIMARY KEY,  
    usuario_id INT,  
    data_pedido DATE,  
    valor DECIMAL(10, 2),  
    FOREIGN KEY (usuario_id) REFERENCES  
    usuarios(id)  
);
```

# COMANDOS DDL (DATA DEFINITION LANGUAGE)

- **ALTER**: Modifica a estrutura de uma tabela existente.

- Adicionar Coluna a uma Tabela

```
ALTER TABLE usuarios ADD COLUMN telefone VARCHAR(20);
```

- Alterar o Tipo de Dados de uma Coluna

```
ALTER TABLE usuarios
```

```
ALTER COLUMN telefone type TEXT;
```

- Renomear uma Tabela

```
ALTER TABLE usuarios
```

```
RENAME TO users;
```

# COMANDOS DDL (DATA DEFINITION LANGUAGE)

- **DROP**: Remove uma tabela ou outro objeto do banco de dados.

```
DROP TABLE usuarios;
```

# COMANDOS DML/CRUD (DATA MANIPULATION LANGUAGE)

- As operações CRUD representam as quatro operações básicas que podem ser realizadas em dados armazenados em um banco de dados.
  - CREATE: Criação
  - READ: Leitura
  - UPDATE: Atualização
  - DELETE: Deleção

# COMANDOS DML/CRUD (DATA MANIPULATION LANGUAGE)

- **INSERT**: Adiciona novos registros a uma tabela.

```
INSERT INTO usuarios (nome, email, cidade, data_nascimento)
VALUES ('Alice', 'alice@example.com', 'Florianópolis', '1990-01-15');
```

- **SELECT**: Consulta dados em uma tabela.

```
SELECT * FROM usuarios
```

- **UPDATE**: Atualiza dados existentes em uma tabela.

```
UPDATE usuarios SET email = 'alice@gmail.com' WHERE id = 1;
```

- **DELETE**: Remove registros de uma tabela.

```
DELETE FROM usuarios WHERE id = 1;
```

# FUNÇÕES SQL

- As funções SQL são usadas para realizar operações nos dados, como cálculos e transformações. Elas podem ser divididas em várias categorias



# FUNÇÕES AGREGADAS

- Funções agregadas realizam cálculos em um conjunto de valores e retornam um único valor.

- **SUM**: Calcula a soma de um conjunto de valores.

```
SELECT SUM(valor) FROM pedidos;
```

- **AVG**: Calcula a média de um conjunto de valores.

```
SELECT AVG(valor) FROM pedidos;
```

- **COUNT**: Conta o número de registros em uma tabela.

```
SELECT COUNT(*) FROM pedidos;
```

# FUNÇÕES AGREGADAS

- **MAX**: Retorna o valor máximo de um conjunto de valores.

```
SELECT MAX(valor) FROM pedidos;
```

- **MIN**: Retorna o valor mínimo de um conjunto de valores.

```
SELECT MIN(valor) FROM pedidos;
```

# FUNÇÕES DE TEXTO

- Funções de texto manipulam strings.

- **CONCAT**: Concatena duas ou mais strings.

```
SELECT CONCAT('O email de ', nome, ' é ', email) FROM usuarios;
```

- **UPPER**: Converte uma string para letras maiúsculas.

```
SELECT UPPER(nome) FROM usuarios;
```

- **LOWER**: Converte uma string para letras minúsculas.

```
SELECT LOWER(nome) FROM usuarios;
```

- **SUBSTRING**: Extrai uma substring de uma string.

```
SELECT SUBSTRING(nome FROM 1 FOR 3) FROM usuarios;
```

# FUNÇÕES DE DATA E HORA

- Funções de data e hora manipulam valores de data e hora.
- **NOW**: Retorna a data e hora atuais.

```
SELECT NOW();
```

- **DATE PART**: Extrai uma subparte de uma data.

```
SELECT DATE_PART('year', data_nascimento) FROM usuarios;
```

- **AGE**: Calcula a diferença entre duas datas.

```
SELECT AGE(NOW(), data_nascimento) FROM usuarios;
```

# INTERVALO

DEV!

Finalizamos o nosso primeiro período de hoje.  
Que tal descansar um pouco?!

Nos vemos em 20 minutos.

Início: 20:20

Retorno: 20:40

# OPERADORES SQL

- **AND**: Retorna true se ambas as condições forem verdadeiras.

```
SELECT * FROM usuarios WHERE cidade = 'São Paulo' AND nome = 'Alice';
```

- **OR**: Retorna true se pelo menos uma das condições for verdadeira.

```
SELECT * FROM usuarios WHERE cidade = 'Rio de Janeiro' OR cidade = 'São Paulo';
```

- **NOT**: Inverte o resultado de uma condição, retornando true se a condição for falsa e vice-versa.

```
SELECT * FROM usuarios WHERE NOT cidade = 'São Paulo';
```

- **IN**: É usado para verificar se um valor está presente em uma lista de valores.

```
SELECT * FROM usuarios WHERE cidade IN ('São Paulo', 'Rio de Janeiro', 'Belo Horizonte');
```

- **LIMIT**: É usado para limitar o número de linhas retornadas por uma consulta.

```
SELECT * FROM usuarios LIMIT 10;
```

- **OFFSET**: Especifica o número de linhas a serem ignoradas a partir do início do conjunto de resultados.

```
SELECT * FROM usuarios LIMIT 10 OFFSET 0;
```

- **BETWEEN**: É usado para selecionar valores dentro de um intervalo específico.

```
SELECT * FROM pedidos WHERE data_pedido BETWEEN '2023-01-01' AND '2023-12-31';
```

- **ORDER BY**: É usado para classificar os resultados de uma consulta em ordem crescente ou decrescente com base em uma ou mais colunas.

```
SELECT * FROM pedidos ORDER BY preco DESC;
```

```
SELECT * FROM pedidos ORDER BY preco ASC;
```

- **LIKE**: É usado para realizar correspondência de padrões em valores de texto. Ele é comumente usado com o caractere de curinga %, que representa zero ou mais caracteres, e o caractere \_, que representa um único caractere.

```
SELECT * FROM pedidos WHERE nome LIKE 'note%';
```

```
SELECT * FROM usuarios WHERE email LIKE '%@example.com';
```

```
SELECT * FROM usuarios WHERE nome LIKE '_lice';
```



- Discord: Pedro Henrique - phbs#2006
- Email: pedro.barroso@edu.sc.senai.br
- Linkedin: <https://www.linkedin.com/in/pedro-h-b-da-silva/>
- Github: <https://github.com/pedrohbsilva/>

## AVALIAÇÃO DOCENTE

O que você está achando das minhas aulas neste conteúdo?

[Clique aqui](#) ou escaneie o QRCode ao lado para avaliar minha aula.

Sinta-se à vontade para fornecer uma avaliação sempre que achar necessário.





**OBRIGADO!**

<LAB365>