

Gabriel Dario da Rosa

## SENAI - DESENVOLVIMENTO DE SISTEMAS - 2º FASE

20/02/2023

1 - SELECT - COMANDO PRINCIPAL PARA TODA INICIAÇÃO DE CONSULTA OBRIGATÓRIA  
FROM - COMANDO USADO PARA ESPECIFICAR QUAL TABELA PARA SELECIONAR - DE.  
WHERE - AONDE - CLÁUSULA PARA FILTRAR REGISTROS - USADO PARA FILTRAR DADOS ESPECÍFICOS;

2 - Cláusula WHERE - Usado para especificar dados de alguma de uma tabela, como nome, idade e filtrá-los.

3 - LIKE - Para pesquisar 'Strings' dentro da tabela de banco de dados, filtrar por determinadas palavras e filtrar. Por exemplo, quer fazer uma pesquisa de Produtos onde começam com a letra A.  
NOT LIKE - Mesa função do LIKE, porém ao contrário, em vez trazer os dados com a String que pesquisamos, vai retornar o que não tiver que ele botou, no primeiro caso irá mostrar todos os resultados que não começaram com a letra A.

4- AND - CLÁUSULA CONDICIONAL QUE SÓ RETORNARÁ TRUE SE AS DUAS FOREM VERDADEIRAS.

OR - RETORNA VERDADEIRO SE UMA DAS CONDIÇÕES FOREM TRUE;

NOT - EXIBE REGISTRO SE A CONDIÇÃO NÃO FOR VERDADEIRA

5 - NULL - Falta de informação de uma tabela, ela não é nula nem zero. Pode ser interpretada como desconhecida ou nenhum valor.

6- JOIN - Agrupamento de tabela. Conjunto entre duas tabelas, formando uma só. Serve para combinar informações iguais de ambas as tabelas e juntá-las.

7 - GROUP BY - Agrupar linhas que tenham semelhanças, usados para somatórias, valor máximo, mínimo.

8 - HAVING se aplica em um conjunto de valores, tabelas enquanto WHERE se aplica individualmente.

9 - ORDER BY - Ordena sua tabela do seu jeito, crescente, decrescente, ordem alfabética ou contrário,

10- ALIÁS - São declarações que podem renomear temporariamente uma tabela ou coluna;

## 11 -

```
SELECT cidade_escola, aluno_nome, cidade_aluno FROM EscolaAluno WHERE cidade_escola = cidade_aluno  
AND cidade_aluno = 'sao jose';
```

## 12 -

```
CREATE TABLE Escola (  
  
id INT NOT NULL,  
  
nome_da_escola VARCHAR(50),  
  
cidade VARCHAR(50),  
  
PRIMARY KEY(id)  
  
);
```

```
CREATE TABLE Aluno (  
  
id_aluno INT,  
  
nome VARCHAR(50) NOT NULL,  
  
cidade VARCHAR(50) NOT NULL,  
  
foreign key(id_aluno) references Escola(id)  
  
);
```

## 13 -

```
CREATE TABLE DEPARTAMENTO (  
  
id INT NOT NULL,  
  
nome VARCHAR(50),  
  
gerente VARCHAR(50),  
  
PRIMARY KEY(id)  
  
);
```

```
CREATE TABLE Funcionario (  
departamento_id INT,  
nome VARCHAR(50) NOT NULL,  
foreign key(departamento_id) references DEPARTAMENTO(id)  
);
```

```
INSERT INTO Departamento(id,nome, gerente)  
VALUES(1, 'Logistica', 'Geraldo' );
```

```
INSERT INTO Funcionario (departamento_id, nome)  
VALUES(1, 'Pedro' );
```

```
INSERT INTO Funcionario (departamento_id, nome)  
VALUES(1, 'Luiz' );
```