



Adianti

# SQL

## CONTATO:



@adiantisolution



/adiantisolutions



/adiantisolutions

FOLHA DE DICAS  
COM COMANDOS SQL

## CAST

Converte um valor recebido para um tipo de dado desejado.

```
SELECT CAST(1.4 AS INT);      --1
```

## NULLIF

Recebe dois valores e retornar o primeiro, se forem iguais retorna nulo

```
SELECT NULLIF(1,1);      --NULL
SELECT NULLIF(1,2);      --1;
SELECT NULLIF(NULL,2);   --NULL;
```

## SUBSTR

Extrair parte de uma string

```
--PostgreSQL, MySQL e Sql Server
SELECT SUBSTRING('abc', 1, 2);  --ab
```

```
--Oracle
SELECT SUBSTR('abc',1,2) FROM DUAL; --ab
```

## LETRAS MAIÚSCULAS E MINÚSCULAS

```
--PostgreSQL, MySQL e Sql Server
SELECT LOWER('ABC');           --abc
SELECT UPPER('abc');           --ABC
```

```
--Oracle
SELECT LOWER('ABC') FROM DUAL;  --abc
SELECT UPPER('abc') FROM DUAL;  --ABC
```

## OPERADOR LIKE

```
--PostgreSQL, MySQL, Sql Server e Oracle
SELECT coluna FROM tabela
WHERE coluna like 'padrao%';
```

O caractere % indica que após o padrão a string pode ter qualquer conteúdo. O % também pode ser usado no início do padrão indicando que a string pode iniciar com qualquer conteúdo.

## COALESCE

Recebe uma lista de valores e retorna o primeiro valor não nulo

```
SELECT coalesce(null, null, 1, null);  //1
```

## CONVERTER STRING PARA TIPO DATE

```
--postgreSQL
SELECT TO_DATE('25-12-2019', 'DD-MM-YYYY');
```

```
--MySQL
SELECT STR_TO_DATE('25-12-2019', '%d-%m-%Y');
```

```
--Oracle
SELECT TO_DATE('25-12-2019', 'DD-MM-YYYY') FROM dual;
```

```
--Sql Server
SELECT CONVERT(DATETIME, '25-12-2019', 105);
```

## DATA ATUAL

```
--postgreSQL e MySQL
SELECT CURRENT_DATE;
```

```
--SQL Server
SELECT CAST(GETDATE() AS DATE)
```

```
--Oracle
SELECT TO_CHAR (SYSDATE, 'YYYY-MM-DD') "NOW" FROM DUAL;
```

## EXTRAIR PARTE DA DATA

```
--postgreSQL e MySQL
SELECT EXTRACT(YEAR FROM CURRENT_DATE);
```

```
--SQL Server
SELECT DATEPART(yyyy, GETDATE())
```

```
--Oracle
SELECT TO_CHAR(SYSDATE, 'YYYY') FROM DUAL;
```

## PAGINAÇÃO

Pular 10 registros e obter os próximos 10

```
--postgreSQL e MySQL
SELECT * FROM tabela LIMIT 10 OFFSET 10;
```

```
-- Sql Server
SELECT coluna
FROM (SELECT ROW_NUMBER() OVER (
ORDER BY coluna ASC) AS __RN__, coluna FROM tabela) AS TAB
WHERE __RN__ BETWEEN 11 AND 20
```

```
-- Oracle
SELECT coluna
FROM ( SELECT rownum AS "__NL__", coluna
FROM (SELECT coluna
FROM tabela
ORDER BY coluna) SUB) TAB
WHERE "__NL__" BETWEEN 1 AND 2
```