



Funções MySQL



**Certified
Developer**
The Ultimate Tech Degree

DigitalHouse >
Coding School



Índice

- | | | | | | | | |
|----------|----------------|----------|--------------------|----------|-----------------|----------|----------------|
| 1 | CONCAT | 2 | COALESCE | 3 | DATEDIFF | 4 | EXTRACT |
| 5 | REPLACE | 6 | DATE FORMAT | 7 | CASE | | |



1 | CONCAT



Concat

Usamos **CONCAT** para **concatenar** duas ou mais expressões.

```
SQL SELECT CONCAT('Olá', 'mundo.');
```

```
> 'Olá mundo.'
```

```
SQL SELECT CONCAT('A resposta é: ', 24, '.');
```

```
> 'A resposta é: 24.'
```

```
SQL SELECT CONCAT('Nome: ', first_name, ' ', last_name)  
FROM actors;
```

```
> 'Nome: Emilia Clarke'
```

2

COALESCE



Coalesce

Usamos **COALESCE** para obter a **primeira expressão** que **não é nula**.

```
SQL SELECT COALESCE(NULL, 1, 20, 'Digital House');  
> 1
```

```
SQL SELECT COALESCE(NULL, NULL, 'Digital House');  
> 'Digital House'
```





Coalesce

Os três clientes na tabela a seguir têm um ou mais dados nulos:

CLIENTES				
id	nome	celular	casa	trabalho
1	João	124	345	980
2	Roberto		187	243
3	Matheus			428





Coalesce

Usando o **COALESCE**, podemos obter os **primeiros dados não nulos** de cada registro, preenchendo as colunas em branco.

```
SQL SELECT id, nome, COALESCE(celular, casa, trabalho) AS telefone  
FROM clientes;
```

CLIENTES		
id	nome	celular
1	João	124
2	Roberto	187
3	Matheus	428



3

DATEDIFF



Datediff

Usamos o **DATEDIFF** para calcular a diferença entre duas datas. Tomamos o intervalo especificado como granularidade.

```
SQL SELECT DATEDIFF(hour, '2017/08/25 07:00', '2017/08/25 12:45');
```

> 5

Retorna 5 porque é o número de horas de diferença entre 7:00 e 12:45. Esta informação fornece um resultado aproximado.

```
SQL SELECT DATEDIFF(minute, '2017/08/25 07:00', '2017/08/25 12:45');
```

> 345

Retorna 345 porque é o número de minutos que varia de 7:00 a 12:45 (300min + 45min).



4 | EXTRACT



Extract

Usamos **EXTRACT** para **extrair** dados de uma data:

```
SQL SELECT EXTRACT(SECOND FROM '2014-02-13 08:44:21');
```

```
> 21
```

```
SQL SELECT EXTRACT(MINUTE FROM '2014-02-13 08:44:21');
```

```
> 44
```

```
SQL SELECT EXTRACT(HOUR FROM '2014-02-13 08:44:21');
```

```
> 8
```

```
SQL SELECT EXTRACT(DAY FROM '2014-02-13 08:44:21');
```

```
> 13
```



Extract

```
SQL SELECT EXTRACT(WEEK FROM '2014-02-13 08:44:21');  
> 6
```

```
SQL SELECT EXTRACT(MONTH FROM '2014-02-13 08:44:21');  
> 2
```

```
SQL SELECT EXTRACT(QUARTER FROM '2014-02-13 08:44:21');  
> 1
```

```
SQL SELECT EXTRACT(YEAR FROM '2014-02-13 08:44:21');  
> 2014
```

5

REPLACE



Replace

Usamos **REPLACE** para substituir um caractere por outro:

```
SQL SELECT REPLACE('abc abc', 'a', 'B');
```

```
> Bbc Bbc
```

```
SQL SELECT REPLACE('abc abc', 'A', 'B');
```

```
> abc abc
```

```
SQL SELECT REPLACE('123 123', '2', '5');
```

```
> 153 153
```



6

DATE FORMAT



DATE FORMAT

Usamos **DATE FORMAT** para avaliar a sentença e retornar a primeira que for cumprida:

```
SQL SELECT DATE_FORMAT('2017-06-15', '%Y');  
> '2017'
```

```
SQL SELECT DATE_FORMAT('2017-06-15', '%W %M %e %Y');  
> 'Thursday June 15 2017'
```



7 | CASE



CASE

Usamos **CASE** para avaliar as condições e retornar a primeira que for atendida. Neste exemplo, a tabela resultante terá 4 colunas: *id*, *title*, *rating*, *rating_categories*. Este último mostrará 'Ruim', 'Regular', etc, de **acordo com a classificação** do filme.

SQL

```
SELECT id, title, rating
CASE
  WHEN rating < 4 THEN 'Ruim'
  WHEN rating < 6 THEN 'Regular'
  WHEN rating < 8 THEN 'Bom'
  WHEN rating < 9.5 THEN 'Muito bom'
  ELSE 'Excelente'
END AS rating_categories
FROM movies
ORDER BY rating
```

DigitalHouse>
Coding School