



# Estruturas Condicionais

1

## Banco de Dados II



# Banco de Dados II

## Estruturas Condicionais IF-THEN-ELSE

No MySQL, a instrução IF-THEN-ELSE é usada para executar código quando uma condição é TRUE, ou executar código diferente se a condição for avaliada como FALSE.

### Sintaxe

A sintaxe para a instrução IF-THEN-ELSE no MySQL é:

```
IF condition1 THEN  
  {...statements to execute when condition1 is TRUE...}  
  [ ELSEIF condition2  
  THEN {...statements to execute when condition1 is FALSE and  
  condition2 is TRUE...} ]  
  [ ELSE {...statements to execute when both condition1 and  
  condition2 are FALSE...} ]  
END IF;
```





# Banco de Dados II

## Estruturas Condicionais IF-THEN-ELSE

### SE NÃO

Opcional. Você usaria a condição *ELSEIF* quando quisesse executar um conjunto de instruções quando uma segunda condição (por exemplo: *condition2*) fosse TRUE.

OUTRO Opcional.

Você usaria a condição *ELSE* quando quisesse executar um conjunto de instruções quando nenhuma das condições *IF* ou *ELSEIF* fosse avaliada como TRUE.

### Observação

Quando uma condição é considerada VERDADEIRA, a instrução IF-THEN-ELSE executará o código correspondente e não avaliará mais as condições. Se nenhuma condição for atendida, a parte ELSE da instrução IF-THEN-ELSE será executada.

É importante observar que as partes *ELSEIF* e *ELSE* são opcionais.





# Banco de Dados II

## Estruturas Condicionais

### Mysql - Declaração IF-THEN-ELSE

O seguinte é um exemplo usando a instrução IF-THEN-ELSE em uma função MySQL:

```
DELIMITER //
CREATE FUNCTION IncomeLevel ( monthly_value INT )
RETURNS varchar(20)
BEGIN
    DECLARE income_level varchar(20);
    IF monthly_value <= 4000 THEN
        SET income_level = 'Low Income';
    ELSEIF monthly_value > 4000 AND monthly_value <= 7000 THEN
        SET income_level = 'Avg Income';
    ELSE
        SET income_level = 'High Income';
    END IF;
    RETURN income_level;
END; //
```





# Banco de Dados II

## Estruturas Condicionais Mysql - Declaração CASE WHEN no SQL

Trabalhar com condições faz parte do nosso dia a dia. A todo momento, estamos avaliando condições para tomar uma determinada ação ou outra.

Por exemplo, imagine que você está planejando sair no final de semana. Se fizer sol, você irá à praia; caso contrário, você irá ao cinema. Esse é um exemplo cotidiano apenas para ilustrar como as condições fazem parte do nosso dia a dia, nesse caso, a condição é fazer sol.

Dentro do SQL, também precisamos trabalhar e lidar com condições. Para esses casos, vamos utilizar as estruturas condicionais CASE WHEN no SQL.





# Banco de Dados II

## Estruturas Condicionais

### Mysql - Declaração CASE WHEN no SQL

Essas estruturas condicionais serão usadas dentro de uma consulta SELECT. A sintaxe dela funcionará da seguinte forma:

```
SELECT  
coluna1 , coluna2 ,  
CASE  
    WHEN condition1 THEN result1  
    WHEN condition2 THEN result2  
    WHEN conditionN THEN resultN  
    ELSE result  
END  
from tabela;
```





# Banco de Dados II

## Estruturas Condicionais

### Mysql - Declaração CASE WHEN no SQL

A estrutura segue este formato padrão, iniciando com CASE e encerrando com END. Entre essas palavras-chave, inserimos as condições utilizando WHEN. Cada condição WHEN é seguida por uma expressão que será avaliada.

Se a expressão for verdadeira, a instrução associada a essa condição será executada. Podemos ter várias condições encadeadas, e a primeira que for verdadeira determinará a ação a ser tomada.

No exemplo acima, quando (WHEN) a condição 1 (condition1) for verdadeira, então (THEN) o SQL retornará o resultado 1 (result1) para essa consulta.

Por fim temos o ELSE no SQL que é o "caso contrário". Ou seja, se nenhuma das condições for satisfeita o ELSE que será executado.





# Banco de Dados II

## Estruturas Condicionais Mysql - Declaração CASE WHEN no SQL

### Exemplo CASE WHEN no SQL com Uma Condição

A estrutura segue este formato padrão, iniciando com CASE e encerrando com END. Entre essas palavras-chave, inserimos as condições utilizando WHEN. Cada condição WHEN é seguida por uma expressão que será avaliada.

Se a expressão for verdadeira, a instrução associada a essa condição será executada. Podemos ter várias condições encadeadas, e a primeira que for verdadeira determinará a ação a ser tomada.

No exemplo acima, quando (WHEN) a condição 1 (condition1) for verdadeira, então (THEN) o SQL retornará o resultado 1 (result1) para essa consulta.

Por fim temos o ELSE no SQL que é o "caso contrário". Ou seja, se nenhuma das condições for satisfeita o ELSE que será executado.







# Banco de Dados II

## Estruturas Condicionais Mysql - Declaração CASE WHEN no SQL

### Exemplo CASE WHEN no SQL com Uma Condição

Vamos ver um exemplo prático de como utilizar as estruturas condicionais para facilitar o entendimento.

Nesse primeiro exemplo, vamos fazer uma consulta utilizando apenas 1 condição dentro da estrutura CASE WHEN no SQL. Você pode usar qualquer banco de dados ou tabela que você tenha disponível para fazer sua consulta.

No meu caso utilizarei esta tabela de produtos:





# Banco de Dados II

## Estruturas Condicionais

### Mysql - Declaração CASE WHEN no SQL

### Exemplo CASE WHEN no SQL com Uma Condição

```
SELECT * FROM produtos;
```

ID_Produto	Nome_Produto	ID_Categoria	Marca_Produto	Num_Serie	Preco_Unit	Custo_Unit
1	Monitor LED 19,5" Full HD HDMI	1	DELL	MNT-DL-831923	2300	966
2	Monitor Curvo 24" 144Hz HDMI	1	SAMSUNG	MNT-SS-001939	2800	980
3	Webcam Full HD 1080p	1	LOGITECH	WBC-LT-934GG4	450	90
4	Kit Teclado + Mouse sem fio Wireless	2	DELL	KTM-DL-041039	350	129,5
5	Kit Teclado + Mouse Slim Bluetooth	2	DELL	KTM-DL-111924	280	109,2
6	Cadeira Gamer reclinável Azul/Laranja	3	ALTURA	CGM-AL-9N914J	1800	540
7	Cadeira Gamer PC Racer Vermelha	3	ALTURA	CGM-AL-0147FI	3100	1395
8	Headphone Bluetooth 2000	4	SONY	HDP-SN-194821	600	258
9	Fone de Ouvido Tune T5000	4	JBL	HDP-JB-091934	780	327,6
10	Microfone Condensador MC1000	5	AKG	MIC-AK-237591	1100	275
11	Microfone Condensador com Tripé	5	BLUE	MIC-BL-819455	800	344
12	Microfone de mesa com fio condensa...	5	BLUE	MIC-BL-761411	650	214,5
13	Notebook LC2100 Intel Core i5 8GB	6	SAMSUNG	NOT-SS-918457	3400	850
14	Notebook Inspiron 15 5000 4GB	6	DELL	NOT-DL-000012	3100	1209
15	Notebook IdeaPad RF32000	6	DELL	NOT-DL-77164I	4200	1176
16	Notebook Motion Ultra 2	6	SAMSUNG	NOT-SS-13139U	2900	1160





# Banco de Dados II

## Estruturas Condicionais

### Mysql - Declaração CASE WHEN no SQL

### Exemplo CASE WHEN no SQL com Uma Condição

Vamos supor que, a partir dessa base de dados, queremos classificar os produtos de acordo com o preço unitário (Preco\_Unit). Dividindo-os entre CLASSE A caso o preço unitário seja maior do que 3000 ou, caso contrário, CLASSE B.

Um uma estrutura CASE WHEN no SQL essa condição pode ser escrita da seguinte forma:

```
SELECT  
    Nome_Produto,Preco_Unit,  
    CASE  
        WHEN Preco_Unit >= 3000 THEN 'CLASSE A'  
        ELSE 'CLASSE B'  
    END AS Classe_Produto  
FROM produtos;
```





# Banco de Dados II

## Estruturas Condicionais

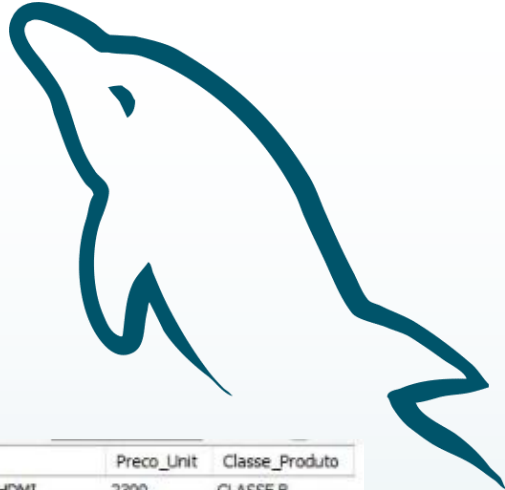
### Mysql - Declaração CASE WHEN no SQL

#### Exemplo CASE WHEN no SQL com Uma Condição

Estamos selecionando da tabela produtos as colunas Nome\_Produto e Preco\_Unit e comparando o valor atribuído ao preço unitário. Quando o preço unitário for maior do que 3000, o produto é classificado como CLASSE A; caso contrário, como CLASSE B.

Ao final, criamos um alias, um nome para a coluna que será gerada com o resultado das condições que estamos analisando, nesse caso, Classe\_Produto.

Executando esse código, teremos como resultado a tabela ao lado:



Nome_Produto	Preco_Unit	Classe_Produto
Monitor LED 19,5" Full HD HDMI	2300	CLASSE B
Monitor Curvo 24" 144Hz HDMI	2800	CLASSE B
Webcam Full HD 1080p	450	CLASSE B
Kit Teclado + Mouse sem fio Wireless	350	CLASSE B
Kit Teclado + Mouse Slim Bluetooth	280	CLASSE B
Cadeira Gamer reclinável Azul/Laranja	1800	CLASSE B
Cadeira Gamer PC Racer Vermelha	3100	CLASSE A
Headphone Bluetooth 2000	600	CLASSE B
Fone de Ouvido Tune T5000	780	CLASSE B
Microfone Condensador MC1000	1100	CLASSE B
Microfone Condensador com Tripé	800	CLASSE B
Microfone de mesa com fio condensa...	650	CLASSE B
Notebook LC2100 Intel Core i5 8GB	3400	CLASSE A
Notebook Inspiron 15 5000 4GB	3100	CLASSE A
Notebook IdeaPad RF32000	4200	CLASSE A
Notebook Motion Ultra 2	2900	CLASSE B

*Repare que ela apresenta as duas colunas que selecionamos, mais a terceira coluna criada a partir da condição que estabelecemos.*



# Banco de Dados II

## Estruturas Condicionais

### Mysql - Declaração CASE WHEN no SQL

### CASE WHEN no SQL com Mais de Uma Condição

Vamos considerar um cenário agora em que queremos classificar os produtos em 3 classes diferentes. Classe A quando o preço unitário for maior do que 3000, Classe B quando o preço estiver entre 1000 e 3000 e Classe C quando o preço for menor do que 1000.

Utilizando CASE WHEN SQL nosso código ficará assim:

```
SELECT
  Nome_Produto,Preco_Unit,
  CASE
    WHEN Preco_Unit >= 3000 THEN 'CLASSE A'
    WHEN Preco_Unit >= 1000 THEN 'CLASSE B'
    ELSE 'CLASSE C'
  END AS Classe_Produto
FROM produtos;
```





# Banco de Dados II

## Estruturas Condicionais

### Mysql - Declaração CASE WHEN no SQL

### CASE WHEN no SQL com Mais de Uma Condição

Como resultado desse código teremos a seguinte tabela:

Nome_Produto	Preco_Unit	Classe_Produto
▶ Notebook IdeaPad RF32000	4200	CLASSE A
Notebook LC2100 Intel Core i5 8GB	3400	CLASSE A
Cadeira Gamer PC Racer Vermelha	3100	CLASSE A
Notebook Inspiron 15 5000 4GB	3100	CLASSE A
Notebook Motion Ultra 2	2900	CLASSE B
Monitor Curvo 24" 144Hz HDMI	2800	CLASSE B
Monitor LED 19,5" Full HD HDMI	2300	CLASSE B
Cadeira Gamer reclinável Azul/Laranja	1800	CLASSE B
Microfone Condensador MC1000	1100	CLASSE B
Microfone Condensador com Tripé	800	CLASSE C
Fone de Ouvido Tune T5000	780	CLASSE C
Microfone de mesa com fio condensa...	650	CLASSE C
Headphone Bluetooth 2000	600	CLASSE C
Webcam Full HD 1080p	450	CLASSE C
Kit Teclado + Mouse sem fio Wireless	350	CLASSE C
Kit Teclado + Mouse Slim Bluetooth	280	CLASSE C

Importante se atentar para o fato de que a avaliação das condições é feita de cima para baixo. Logo, quando uma é satisfeita, as condições abaixo são ignoradas. Por isso, os produtos acima de 3000, que por consequência também são acima de 1000, aparecem apenas como Classe A.

Uma dica para saber quantas estruturas condicionais no SQL você precisará criar é verificar o número de classificações que você deseja fazer menos 1. Nesse caso, tínhamos 3 condições, logo precisamos criar 2 CASE WHEN no SQL.





# Banco de Dados II

## Estruturas Condicionais Especiais

## Estruturas Condicionais Especiais para **Valores Nulos**

Um valor nulo é a ausência de valor. Ou seja, um valor nulo não é a mesma coisa que um espaço vazio ou zero.

Com as funções que iremos ver ao longo dessa aula, fica fácil e prático identificar e lidar com os valores nulos, permitindo que você trabalhe e realize suas análises de dados sem problemas.





# Banco de Dados II

## Estruturas Condicionais Especiais para **Valores Nulos**

### Como Visualizar Valores Nulos

Vamos começar visualizando os valores nulos do banco de dados. Para isso, vou selecionar toda a tabela de clientes presente nele.

```
SELECT *  
FROM clientes;
```

ID_Cliente	Nome	Sobrenome	Data_Nascimento	Estado_Civil	Sexo	Email	Telefone	Renda_Anual	Qtd_Filhos	Escolaridade
52	Deanna	Munoz	1972-03-05	C	F	deanna33@yahoo.com.br	(85) 92440-1858	40000	2	Parcial
53	Gilbert	Raje	1972-02-29	C	M	gilbert35@yahoo.com.br	(41) 95730-5711	40000	2	Parcial
54	Michele	Nath	1973-03-29	C	F	michele19@yahoo.com.br	(62) 98188-6208	40000	3	Parcial
55	Carl	Andersen	1973-10-07	C	M	carl12@gmail.com	(11) 97180-1453	70000	2	Graduação
56	Marc	Diaz	1974-04-22	C	M	marc6@gmail.com	(41) 92126-6598	80000	2	Parcial
57	Ashlee	Andersen	1974-03-27	S	F	ashlee19@gmail.com	(85) 97355-1991	80000	2	Parcial
58	Jon	Zhou	1974-03-12	C	M	jon28@hotmail.com	(85) 96675-4855	80000	2	Parcial
59	Todd	Gao	1974-02-22	C	M	todd14@gmail.com	(11) 96585-5605	80000	2	Parcial
60	Noah	Powell	1995-08-28	C	M	noah5@yahoo.com.br	(85) 98162-6253	40000	0	Ensino médio
61	Angela	Murphy	1995-04-02	S	F	angela41@yahoo.com.br		40000	0	Ensino médio
62	Chase	Reed	1995-12-02	C	M	chase21@hotmail.com		40000	0	Ensino médio
63	Jessica	Henderson	1993-10-04	C	F	jessica29@gmail.com	(21) 93673-6903	60000	0	Parcial

Visualizando essa tabela, podemos ver que na coluna Telefone temos alguns valores nulos. Nesse pedaço exibido na imagem, podemos ver dois valores nulos e um espaço em branco logo abaixo do primeiro valor nulo.

Repare que o valor nulo aparece identificado como null, diferente do espaço em branco, que é apenas um espaço vazio.







# Banco de Dados II

## Estruturas Condicionais Especiais para **Valores Nulos**

### Função **IFNULL** – Substituir Valor Nulo

A função IFNULL é uma função condicional no SQL que permite verificar se um valor é nulo e, caso seja, substituí-lo por um novo valor.

Essa função recebe dois argumentos: o primeiro é a expressão que será verificada, ou seja, aquela que queremos verificar se é nula; e o segundo argumento é o valor alternativo que irá substituir o valor nulo.

Para substituir os telefones que estão com os valores nulos, precisamos selecionar toda a nossa tabela, passando como primeiro argumento a coluna Telefone e como segundo argumento o valor pelo qual queremos substituir os nulos encontrados.

```
SELECT * ,  
        IFNULL(Telefone, ' (99) 99999-9999' )  
FROM clientes;
```





# Banco de Dados II

## Estruturas Condicionais Especiais para **Valores Nulos**

### Função **IFNULL** – Substituir Valor Nulo

Como resultado, teremos a tabela abaixo com uma nova coluna com os valores substituídos.

ID_Cliente	Nome	Sobrenome	Data_Nascimento	Estado_Civil	Sexo	Email	Telefone	Renda_Anual	Qtd_Filhos	Escolaridade	ifnull(Telefone, '(99) 99999-9999')
58	Jon	Zhou	1974-03-12	C	M	jon28@hotmail.com	(85) 96675-4865	80000	2	Parcial	(85) 96675-4865
59	Todd	Gao	1974-02-22	C	M	todd14@gmail.com	(11) 96585-5605	80000	2	Parcial	(11) 96585-5605
60	Noah	Powell	1995-08-28	C	M	noah5@yahoo.com.br	(85) 98162-6253	40000	0	Ensino médio	(85) 98162-6253
61	Angela	Murphy	1995-04-02	S	F	angela41@yahoo.com.br	NULL	40000	0	Ensino médio	(99) 99999-9999
62	Chase	Reed	1995-12-02	C	M	chase21@hotmail.com	NULL	40000	0	Ensino médio	(99) 99999-9999
63	Jessica	Henderson	1993-10-04	C	F	jessica29@gmail.com	(21) 93673-6903	60000	0	Parcial	(21) 93673-6903
64	Grace	Butler	1993-11-22	C	F	grace62@gmail.com	NULL	70000	0	Parcial	(99) 99999-9999
65	Caleb	Carter	1996-09-20	S	M	caleb40@gmail.com	(31) 97809-1800	60000	0	Parcial	(31) 97809-1800
66	Tiffany	Liang	1975-09-18	S	F	tiffany17@gmail.com	(21) 97231-3503	80000	2	Ensino médio	(21) 97231-3503
67	Carolyn	Navarro	1975-09-16	S	F	carolyn30@yahoo.com.br	(21) 94139-6777	80000	2	Ensino médio	(21) 94139-6777
68	Willie	Raji	1975-03-31	C	M	willie40@yahoo.com.br	(71) 97614-1995	80000	2	Ensino médio	(71) 97614-1995
69	Linda	Serrano	1975-06-21	S	F	linda31@hotmail.com	(21) 92428-2938	80000	2	Ensino médio	(21) 92428-2938
70	Casey	Luo	1975-02-01	S	F	casey6@yahoo.com.br	(31) 96609-9916	80000	2	Ensino médio	(31) 96609-9916
71	Amy	Ye	1976-08-09	S	F	amy16@hotmail.com	(41) 92958-3807	70000	2	Ensino médio	(41) 92958-3807
72	Levi	Arun	1976-08-23	S	M	levi6@gmail.com	(71) 92754-9983	70000	2	Ensino médio	(71) 92754-9983
73	Felicia	Jimenez	1977-11-11	S	F	felicia4@gmail.com	(85) 95647-1806	80000	2	Ensino médio	(85) 95647-1806

Repare que os valores nulos foram alterados, porém o valor vazio permaneceu da mesma forma.



## Banco de Dados II

Estruturas Condicionais Especiais para **Valores Nulos**

Função **ISNULL** – Verificar se o Valor é Nulo

A função ISNULL verifica se o valor é nulo ou não. Caso seja nulo, ela retorna 1; caso não seja, retorna 0. Essa função recebe como argumento o valor que será verificado.

Seguindo nosso exemplo, podemos utilizar a função ISNULL para verificar os valores da coluna Telefone.

```
SELECT  
    *,  
    ISNULL(Telefone)  
FROM clientes;
```





# Banco de Dados II

## Estruturas Condicionais Especiais para **Valores Nulos**

### Função **ISNULL** – Verificar se o Valor é Nulo

Executando essa consulta, teremos uma nova coluna na nossa tabela, onde os valores não nulos receberam o valor de 0 e os valores nulos receberam 1.

ID_Cliente	Nome	Sobrenome	Data_Nascimento	Estado_Civil	Sexo	Email	Telefone	Renda_Anual	Qtd_Filhos	Escolaridade	isnull(Telefone)
55	Carl	Andersen	1973-10-07	C	M	carl12@gmail.com	(11) 97180-1463	70000	2	Graduação	0
56	Marc	Diaz	1974-04-22	C	M	marc6@gmail.com	(41) 92126-6598	80000	2	Parcial	0
57	Ashlee	Andersen	1974-03-27	S	F	ashlee19@gmail.com	(85) 97355-1991	80000	2	Parcial	0
58	Jon	Zhou	1974-03-12	C	M	jon28@hotmail.com	(85) 96675-4865	80000	2	Parcial	0
59	Todd	Gao	1974-02-22	C	M	todd14@gmail.com	(11) 96585-5605	80000	2	Parcial	0
60	Noah	Powell	1995-08-28	C	M	noah5@yahoo.com.br	(85) 98167-6253	40000	0	Ensino médio	0
61	Angela	Murphy	1995-04-02	S	F	angela41@yahoo.com.br	NULL	40000	0	Ensino médio	1
62	Chase	Reed	1995-12-02	C	M	chase21@hotmail.com	NULL	40000	0	Ensino médio	0
63	Jessica	Henderson	1993-10-04	C	F	jessica29@gmail.com	(21) 93671-8903	60000	0	Parcial	0
64	Grace	Butler	1993-11-22	C	F	grace62@gmail.com	NULL	70000	0	Parcial	1
65	Caleb	Carter	1996-09-20	S	M	caleb40@gmail.com	(31) 97809-1800	60000	0	Parcial	0
66	Tiffany	Liang	1975-09-18	S	F	tiffany17@gmail.com	(21) 97231-3503	80000	2	Ensino médio	0
67	Carolyn	Navarro	1975-09-16	S	F	carolyn30@yahoo.com.br	(21) 94139-6777	80000	2	Ensino médio	0
68	Wilie	Raji	1975-03-31	C	M	wilie40@yahoo.com.br	(71) 97614-1995	80000	2	Ensino médio	0
69	Linda	Serrano	1975-06-21	S	F	linda31@hotmail.com	(21) 92428-2938	80000	2	Ensino médio	0
70	Casey	Luo	1975-02-01	S	F	casey6@yahoo.com.br	(31) 96609-9916	80000	2	Ensino médio	0
71	Amy	Ye	1976-08-09	S	F	amy16@hotmail.com	(41) 92958-3807	70000	2	Ensino médio	0

Observe que assim como ocorreu com a função IFNULL, o espaço vazio não é reconhecido como nulo; apenas os valores realmente nulos recebem o valor de 1.



# Banco de Dados II

## Estruturas Condicionais Especiais para **Valores Nulos**

### Função **ISNULL** – Verificar se o Valor é Nulo

Essa função pode ser utilizada para filtrar a nossa tabela, exibindo apenas os clientes que possuem o telefone nulo.

```
SELECT      *, FROM clientes  
WHERE ISNULL(Telefone) ;
```

O comando WHERE sempre retorna os valores que são verdadeiros. No SQL, o verdadeiro é representado pelo 1 e o falso pelo 0. Então, basicamente, o que estamos fazendo é filtrar a partir da função ISNULL quais são os clientes que têm o telefone com valor nulo.

ID_Cliente	Nome	Sobrenome	Data_Nascimento	Estado_Civil	Sexo	Email	Telefone	Renda_Anual	Qtd_Filhos	Escolaridade
22	Russell	Xie	1998-09-12	C	M	russell7@hotmail.com	NULL	60000	0	Parcial
40	Megan	Sanchez	1997-06-08	C	F	megan28@hotmail.com	NULL	70000	0	Parcial
61	Angela	Murphy	1995-04-02	S	F	angela41@yahoo.com.br	NULL	40000	0	Ensino médio
64	Grace	Butler	1993-11-22	C	F	grace62@gmail.com	NULL	70000	0	Parcial
88	Trevor	Bryant	1977-12-12	S	M	trevor18@gmail.com	NULL	90000	2	Parcial
98	Latasha	Navarro	1980-09-10	S	F	latasha10@yahoo.com.br	NULL	60000	1	Parcial

*Repare que aquele valor vazio não consta nessa lista de clientes com o telefone nulo. Para tratar isso, podemos utilizar a função NULLIF.*





## Banco de Dados II

### Estruturas Condicionais Especiais para **Valores Nulos**

#### **NULLIF** – Substituindo Valor por Nulo

A função NULLIF no SQL é usada para comparar duas expressões e retornar null caso as duas expressões sejam iguais. Caso contrário, ela manterá a primeira expressão.

Para isso, precisamos passar dois argumentos para ela. O primeiro argumento é o valor que queremos verificar e o segundo argumento é a expressão com que ele será comparado.

Então, se quisermos transformar os valores vazios da nossa tabela em null, precisamos passar como primeiro argumento a coluna Telefone e como segundo argumento o espaço vazio.

```
SELECT *  
FROM clientes  
WHERE ISNULL(NULLIF(Telefone, ' '));
```





# Banco de Dados II

## Estruturas Condicionais Especiais para **Valores Nulos**

### **NULLIF** – Substituindo Valor por Nulo

Ou seja, dessa vez estamos filtrando os clientes que possuem o valor nulo como Telefone, mas dentro do filtro estamos aplicando também a função NULLIF para que quando algum valor do Telefone for vazio, ela converta esse valor para nulo.

Executando essa consulta, teremos a nossa tabela contendo tanto os telefones nulos quanto os telefones em branco, vazios.

ID_Cliente	Nome	Sobrenome	Data_Nascimento	Estado_Civil	Sexo	Email	Telefone	Renda_Anual	Qtd_Filhos	Escolaridade
22	Russell	Xie	1998-09-12	C	M	russell7@hotmail.com	NULL	60000	0	Parcial
40	Megan	Sanchez	1997-06-08	C	F	megan28@hotmail.com	NULL	70000	0	Parcial
61	Angela	Murphy	1995-04-02	S	F	angela41@yahoo.com.br	NULL	40000	0	Ensino médio
62	Chase	Reed	1995-12-02	C	M	chase21@hotmail.com		40000	0	Ensino médio
64	Grace	Butler	1993-11-22	C	F	grace62@gmail.com	NULL	70000	0	Parcial
88	Trevor	Bryant	1977-12-12	S	M	trevor18@gmail.com	NULL	90000	2	Parcial
98	Latasha	Navarro	1980-09-10	S	F	latasha10@yahoo.com.br	NULL	60000	1	Parcial



# Referências



- **Hashtag Treinamentos**

[https://www.hashtagtreinamentos.com/variaveis-no-mysql-sql?gad\\_source=1&gclid=CjwKCAiAm-67BhBlEiwAEVftNpGzgg\\_Ece0CU2So0u9YRmJVZpZUblhTzqZKxtUU6AmX5bVyRJJc6BoC73oQAvD\\_BwE](https://www.hashtagtreinamentos.com/variaveis-no-mysql-sql?gad_source=1&gclid=CjwKCAiAm-67BhBlEiwAEVftNpGzgg_Ece0CU2So0u9YRmJVZpZUblhTzqZKxtUU6AmX5bVyRJJc6BoC73oQAvD_BwE)

- **Tech on the Net**

[https://www.techonthenet.com/mysql/loops/if\\_then.php](https://www.techonthenet.com/mysql/loops/if_then.php)