

DISCIPLINA

# Banco de Dados I



Prof. Jackson Henrique

## AULA 11

---

- **Comandos DML (Linguagem de Definição de Dados):  
SELECT**

# Selecionar Registros - SELECT

- Uma das principais funções de um banco de dados é possibilitar a **consulta dos registros** armazenados nas **tabelas**;
- Para consultar um ou muitos registros utilizamos o comando **SELECT**;
- O comando SELECT **não modifica nenhum registro**, apenas mostra para o usuário os **registros armazenados** na tabela de **acordo com a condição**;

# Selecionar Registros - SELECT

- **Sintaxe:**

- **SELECT** atributo1, atributo2, ... **FROM** nome\_da\_tabela **WHERE** (condição);

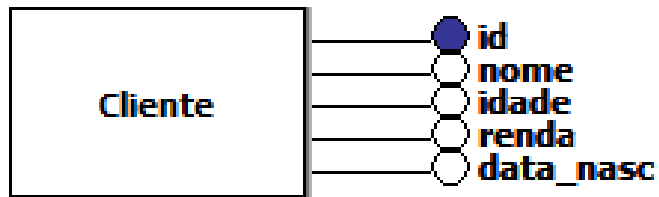
- Também é possível **omitir os nomes dos atributos** usando o símbolo asterisco \*. Com o \* no lugar atributos **todos os atributos** serão selecionados;

- **Sintaxe com \*:**

- **SELECT \* FROM** nome\_da\_tabela **WHERE** (condição);

# Uso do Select com Condição

- **Importante** destacar que o SELECT **pode ser usando sem condição**, neste caso, **todos os registros** da tabela serão selecionados;
- Vamos aos **exemplos**. Considere o seguinte banco de dados:



| Cliente |                |           |           |               |
|---------|----------------|-----------|-----------|---------------|
| id_cli  | nome_cli       | idade_cli | renda_cli | data_nasc_cli |
| 1       | José da Silva  | 33        | 1500.50   | 1987-01-30    |
| 2       | Ana Maria      | 30        | 2500.00   | 1990-02-20    |
| 3       | Gustavo Silva  | 20        | 5000.00   | 2000-01-31    |
| 4       | Marcos Pereira | 27        | 1020.00   | 1993-06-21    |
| 5       | Thiago Souza   | 33        | 5000.00   | 1987-06-30    |

# Uso do Select com Condição

- **Objetivo:** Selecione o id, o nome e a renda dos clientes com renda superior a 1600 reais;

- **Comando:**

➤ ***SELECT** id\_cli, nome\_cli, renda\_cli **FROM** Cliente **WHERE** (renda\_cli > 1600);*

| TABELA CLIENTE |                  |           |           |               |
|----------------|------------------|-----------|-----------|---------------|
| id_cli         | nome_cli         | idade_cli | renda_cli | data_nasc_cli |
| 1              | José da Silva    | 33        | 1500.50   | 1987-01-30    |
| 2              | Ana Maria        | 30        | 2500.00   | 1990-02-20    |
| 3              | Gustavo H. Silva | 20        | 5000.00   | 2000-01-31    |
| 4              | Marcos Pereira   | 27        | 1020.00   | 1993-06-21    |
| 5              | Thiago Souza     | 33        | 5000.00   | 1987-06-30    |

| RESULTADO DA CONSULTA |                  |           |
|-----------------------|------------------|-----------|
| id_cli                | nome_cli         | renda_cli |
| 2                     | Ana Maria        | 2500.00   |
| 3                     | Gustavo H. Silva | 5000.00   |
| 5                     | Thiago Souza     | 5000.00   |

# Uso do Select com Condição

- **Objetivo:** Selecione os clientes com renda superior a 1600 reais;

- **Comando:**

➤ *SELECT \* FROM Cliente WHERE (renda\_cli > 1600);*

| TABELA CLIENTE |                  |           |           |               |
|----------------|------------------|-----------|-----------|---------------|
| id_cli         | nome_cli         | idade_cli | renda_cli | data_nasc_cli |
| 1              | José da Silva    | 33        | 1500.50   | 1987-01-30    |
| 2              | Ana Maria        | 30        | 2500.00   | 1990-02-20    |
| 3              | Gustavo H. Silva | 20        | 5000.00   | 2000-01-31    |
| 4              | Marcos Pereira   | 27        | 1020.00   | 1993-06-21    |
| 5              | Thiago Souza     | 33        | 5000.00   | 1987-06-30    |

| RESULTADO DA CONSULTA |                  |           |           |               |
|-----------------------|------------------|-----------|-----------|---------------|
| id_cli                | nome_cli         | idade_cli | renda_cli | data_nasc_cli |
| 2                     | Ana Maria        | 30        | 2500.00   | 1990-02-20    |
| 3                     | Gustavo H. Silva | 20        | 5000.00   | 2000-01-31    |
| 5                     | Thiago Souza     | 33        | 5000.00   | 1987-06-30    |

# Organizar Resultado – ORDER BY

- Podemos utilizar a palavra chave **ORDER BY** para ordenar os registros selecionadas em uma consulta a partir de um atributo;
- O ORDER BY é usado sempre no **final do comando** SELECT;
- Caso queira ordena os valores de forma **decrescente** basta adicionar a palavra **DESC** após o nome do atributo ordenador;
- **Sintaxe:**
- **SELECT** atributo1, atributo2, ... **FROM** nome\_tabela **WHERE** (condição) **ORDER BY** atributo;



# Uso do Select com ORDER BY

- **Objetivo:** Selecione os clientes com renda maior do que 1000 e ordenados pelo nome;
- **Comando:**

➤ *SELECT \* FROM Cliente WHERE (renda\_cli > 1000) ORDER BY nome\_cli;*

| TABELA CLIENTE |                  |           |           |               |
|----------------|------------------|-----------|-----------|---------------|
| id_cli         | nome_cli         | idade_cli | renda_cli | data_nasc_cli |
| 1              | José da Silva    | 33        | 1500.50   | 1987-01-30    |
| 2              | Ana Maria        | 30        | 2500.00   | 1990-02-20    |
| 3              | Gustavo H. Silva | 20        | 5000.00   | 2000-01-31    |
| 4              | Marcos Pereira   | 27        | 1020.00   | 1993-06-21    |
| 5              | Thiago Souza     | 33        | 5000.00   | 1987-06-30    |

| RESULTADO DA CONSULTA |                  |           |           |               |
|-----------------------|------------------|-----------|-----------|---------------|
| id_cli                | nome_cli         | idade_cli | renda_cli | data_nasc_cli |
| 2                     | Ana Maria        | 30        | 2500.00   | 20/02/1990    |
| 3                     | Gustavo H. Silva | 20        | 5000.00   | 31/01/2000    |
| 1                     | José da Silva    | 33        | 1500.50   | 30/01/1987    |
| 4                     | Marcos Pereira   | 27        | 1020.00   | 21/06/1993    |
| 5                     | Thiago Souza     | 33        | 5000.00   | 30/06/1987    |

# Uso do Select com ORDER BY com DESC

- **Objetivo:** Selecione os clientes com renda maior do que 1000 e ordenados pelo nome de forma decrescente;

- **Comando:**

➤ *SELECT \* FROM Cliente WHERE (renda\_cli > 1000) ORDER BY nome\_cli DESC;*

| TABELA CLIENTE |                  |           |           |               |
|----------------|------------------|-----------|-----------|---------------|
| id_cli         | nome_cli         | idade_cli | renda_cli | data_nasc_cli |
| 1              | José da Silva    | 33        | 1500.50   | 1987-01-30    |
| 2              | Ana Maria        | 30        | 2500.00   | 1990-02-20    |
| 3              | Gustavo H. Silva | 20        | 5000.00   | 2000-01-31    |
| 4              | Marcos Pereira   | 27        | 1020.00   | 1993-06-21    |
| 5              | Thiago Souza     | 33        | 5000.00   | 1987-06-30    |

| RESULTADO DA CONSULTA |                  |           |           |               |
|-----------------------|------------------|-----------|-----------|---------------|
| id_cli                | nome_cli         | idade_cli | renda_cli | data_nasc_cli |
| 5                     | Thiago Souza     | 33        | 5000.00   | 30/06/1987    |
| 4                     | Marcos Pereira   | 27        | 1020.00   | 21/06/1993    |
| 1                     | José da Silva    | 33        | 1500.50   | 30/01/1987    |
| 3                     | Gustavo H. Silva | 20        | 5000.00   | 31/01/2000    |
| 2                     | Ana Maria        | 30        | 2500.00   | 20/02/1990    |

# Agrupar Registros Iguais – GROUP BY

- Podemos utilizar a palavra chave **GROUP BY** para agrupar registros com um valores iguais em um atributo;
- Assim o GROUP BY mostra apenas os registros diferentes de um determinado atributo;
- O GROUP BY é usado sempre no **final do comando** SELECT;
- **Sintaxe:**
- **SELECT** atributo1 **FROM** nome\_tabela **WHERE** (condição) **GROUP BY** atributo1;

# Agrupar Registros Iguais – GROUP BY

- **Objetivo:** Mostre as idades diferentes dos clientes cadastrados;

- **Comando:**

➤ *SELECT idade\_cli FROM Cliente GROUP BY idade\_cli;*

| TABELA CLIENTE |                  |           |           |               |
|----------------|------------------|-----------|-----------|---------------|
| id_cli         | nome_cli         | idade_cli | renda_cli | data_nasc_cli |
| 1              | José da Silva    | 33        | 1500.50   | 1987-01-30    |
| 2              | Ana Maria        | 30        | 2500.00   | 1990-02-20    |
| 3              | Gustavo H. Silva | 20        | 5000.00   | 2000-01-31    |
| 4              | Marcos Pereira   | 27        | 1020.00   | 1993-06-21    |
| 5              | Thiago Souza     | 33        | 5000.00   | 1987-06-30    |

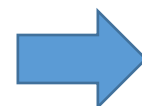


| idade_cli |
|-----------|
| 33        |
| 27        |
| 20        |
| 30        |

# GROUP BY com ORDER BY

- É possível utilizar o ORDER BY junto com o comando GROUP BY, neste caso o ORDER BY fica no final do SELECT;
- **Objetivo:** Mostre as idades diferentes dos clientes cadastrados de forma ordenada;
- **Comando:** *SELECT idade\_cli FROM Cliente GROUP BY idade\_cli ORDER BY idade\_cli;*

| TABELA CLIENTE |                  |           |           |               |
|----------------|------------------|-----------|-----------|---------------|
| id_cli         | nome_cli         | idade_cli | renda_cli | data_nasc_cli |
| 1              | José da Silva    | 33        | 1500.50   | 1987-01-30    |
| 2              | Ana Maria        | 30        | 2500.00   | 1990-02-20    |
| 3              | Gustavo H. Silva | 20        | 5000.00   | 2000-01-31    |
| 4              | Marcos Pereira   | 27        | 1020.00   | 1993-06-21    |
| 5              | Thiago Souza     | 33        | 5000.00   | 1987-06-30    |



| idade_cli |
|-----------|
| 20        |
| 27        |
| 30        |
| 33        |

# Operadores Especiais – IS NULL

- O operador **IS NULL** é usado em uma condição para selecionar valores que sejam **NULL** (nulos) ou **NOT NULL** (não nulos);
- Importante destacar que no MySQL os valores NULL **não podem ser comparados** com o operador = (igual), somente com o **IS**;
- Sintaxe:
  - (atributo IS NULL) ou (atributo IS NOT NULL)
- Exemplo:
  - `SELECT * FROM Cliente WHERE (nome_cli IS NULL);`

# Uso do Select com IS NULL

- **Objetivo:** Selecione o id, nome e renda dos clientes que não possuem renda, ou seja, que estão nulas.

- **Comando:**

➤ *SELECT id\_cli, nome\_cli, idade\_cli FROM Cliente WHERE (renda\_cli IS NULL);*

| TABELA CLIENTE |                  |           |           |               |
|----------------|------------------|-----------|-----------|---------------|
| id_cli         | nome_cli         | idade_cli | renda_cli | data_nasc_cli |
| 1              | José da Silva    | 33        | 1500.50   | 1987-01-30    |
| 2              | Ana Maria        | 30        | 2500.00   | 1990-02-20    |
| 3              | Gustavo H. Silva | 20        | 5000.00   | 2000-01-31    |
| 4              | Marcos Pereira   | 27        | 1020.00   | 1993-06-21    |
| 5              | Thiago Souza     | 33        | null      | 1987-06-30    |

| RESULTADO DA CONSULTA |              |           |
|-----------------------|--------------|-----------|
| id_cli                | nome_cli     | renda_cli |
| 5                     | Thiago Souza | null      |

# Funções Especiais

- As Funções Especiais no MySQL retornam um ou vários registros de acordo com o atributo de entrada da função;
- Toda função tem um **objetivo específico**. A **forma** como esse objetivo é cumprido é de **responsabilidade do SGBD**;
- Toda função tem **uma entrada**. Essa **entrada** é o **atributo** da tabela que está sendo selecionada;
- As funções são utilizadas **entre** o SELECT e o FROM, substituindo um atributo da tabela foco da seleção;



# Funções Especiais

- Sintaxe do SELECT com Função;

➤ **SELECT** Nome\_Função(atributo) **FROM** nome\_tabela **WHERE** (condição);

- As principais **funções** no MySQL são:

➤ **COUNT**(*atributo*) – Retorna a **quantidade total** de registros não nulos de um atributo;

➤ **SUM**(*atributo*) – Função que retorna a **soma dos valores** de um atributo;

➤ **AVG**(*atributo*) – Função que retorna a **média dos valores** de um atributo;

➤ **MIN**(*atributo*) – Função que retorna o **menor valor** de um atributo;

➤ **MAX**(*atributo*) – Função que retorna o **maior valor** de um atributo.

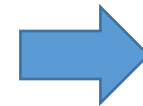
# Uso do Select com Função COUNT

- **Objetivo:** Selecione quantos clientes estão armazenados no banco de dados.

- **Comando:**

➤ ***SELECT COUNT(id\_cli) FROM Cliente;***

| TABELA CLIENTE |                  |           |           |               |
|----------------|------------------|-----------|-----------|---------------|
| id_cli         | nome_cli         | idade_cli | renda_cli | data_nasc_cli |
| 1              | José da Silva    | 33        | 1500.50   | 1987-01-30    |
| 2              | Ana Maria        | 30        | 2500.00   | 1990-02-20    |
| 3              | Gustavo H. Silva | 20        | 5000.00   | 2000-01-31    |
| 4              | Marcos Pereira   | 27        | 1020.00   | 1993-06-21    |
| 5              | Thiago Souza     | 33        | 5000.00   | 1987-06-30    |



## RESULTADO DA CONSULTA

| <i><b>COUNT(id_cli)</b></i> |
|-----------------------------|
| 5                           |

# Uso do Select com Função COUNT

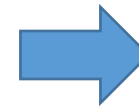
- **Objetivo:** Selecione quantas idades diferentes existem nos clientes armazenados no banco de dados.

- **Comando:**

➤ ***SELECT COUNT(DISTINCT idade\_cli) FROM Cliente;***

O DISTINCT conta os registros DIFERENTES do atributo usado no COUNT()

| TABELA CLIENTE |                  |           |           |               |
|----------------|------------------|-----------|-----------|---------------|
| id_cli         | nome_cli         | idade_cli | renda_cli | data_nasc_cli |
| 1              | José da Silva    | 33        | 1500.50   | 1987-01-30    |
| 2              | Ana Maria        | 30        | 2500.00   | 1990-02-20    |
| 3              | Gustavo H. Silva | 20        | 5000.00   | 2000-01-31    |
| 4              | Marcos Pereira   | 27        | 1020.00   | 1993-06-21    |
| 5              | Thiago Souza     | 33        | 5000.00   | 1987-06-30    |



## RESULTADO DA CONSULTA

| <i><b>COUNT(distinct idade_cli)</b></i> |
|---|
| 4                                       |

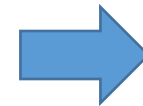
# Uso do Select com Função SUM

- **Objetivo:** Selecione a soma das rendas de todos os clientes armazenados no banco de dados.

- **Comando:**

➤ *SELECT SUM(renda\_cli) FROM Cliente;*

| TABELA CLIENTE |                  |           |           |               |
|----------------|------------------|-----------|-----------|---------------|
| id_cli         | nome_cli         | idade_cli | renda_cli | data_nasc_cli |
| 1              | José da Silva    | 33        | 1500.50   | 1987-01-30    |
| 2              | Ana Maria        | 30        | 2500.00   | 1990-02-20    |
| 3              | Gustavo H. Silva | 20        | 5000.00   | 2000-01-31    |
| 4              | Marcos Pereira   | 27        | 1020.00   | 1993-06-21    |
| 5              | Thiago Souza     | 33        | 5000.00   | 1987-06-30    |



## RESULTADO DA CONSULTA

| <i>SUM(renda_cli)</i> |
|-----------------------|
| 15020.50              |

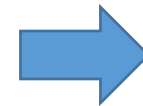
# Uso do Select com Função AVG

- **Objetivo:** Selecione e mostre a idade média dos clientes armazenados no banco de dados.

- **Comando:**

➤ ***SELECT AVG(idade\_cli) FROM Cliente;***

| TABELA CLIENTE |                  |           |           |               |
|----------------|------------------|-----------|-----------|---------------|
| id_cli         | nome_cli         | idade_cli | renda_cli | data_nasc_cli |
| 1              | José da Silva    | 33        | 1500.50   | 1987-01-30    |
| 2              | Ana Maria        | 30        | 2500.00   | 1990-02-20    |
| 3              | Gustavo H. Silva | 20        | 5000.00   | 2000-01-31    |
| 4              | Marcos Pereira   | 27        | 1020.00   | 1993-06-21    |
| 5              | Thiago Souza     | 33        | 5000.00   | 1987-06-30    |



## RESULTADO DA CONSULTA

| AVG(idade_cli) |
|----------------|
| 28.6           |

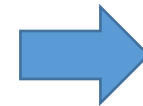
# Uso do Select com Função MAX

- **Objetivo:** Selecione e mostre a maior renda entre os clientes armazenados no banco de dados.

- **Comando:**

➤ *SELECT MAX(renda\_cli) FROM Cliente;*

| TABELA CLIENTE |                  |           |           |               |
|----------------|------------------|-----------|-----------|---------------|
| id_cli         | nome_cli         | idade_cli | renda_cli | data_nasc_cli |
| 1              | José da Silva    | 33        | 1500.50   | 1987-01-30    |
| 2              | Ana Maria        | 30        | 2500.00   | 1990-02-20    |
| 3              | Gustavo H. Silva | 20        | 5000.00   | 2000-01-31    |
| 4              | Marcos Pereira   | 27        | 1020.00   | 1993-06-21    |
| 5              | Thiago Souza     | 33        | 5000.00   | 1987-06-30    |



## RESULTADO DA CONSULTA

| MAX(renda_cli) |
|----------------|
| 5000.00        |

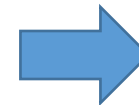
# Uso do Select com Função MIN

- **Objetivo:** Selecione e mostre a menor renda entre os clientes armazenados no banco de dados.

- **Comando:**

➤ *SELECT MIN(renda\_cli) FROM Cliente;*

| TABELA CLIENTE |                  |           |           |               |
|----------------|------------------|-----------|-----------|---------------|
| id_cli         | nome_cli         | idade_cli | renda_cli | data_nasc_cli |
| 1              | José da Silva    | 33        | 1500.50   | 1987-01-30    |
| 2              | Ana Maria        | 30        | 2500.00   | 1990-02-20    |
| 3              | Gustavo H. Silva | 20        | 5000.00   | 2000-01-31    |
| 4              | Marcos Pereira   | 27        | 1020.00   | 1993-06-21    |
| 5              | Thiago Souza     | 33        | 5000.00   | 1987-06-30    |



## RESULTADO DA CONSULTA

| MIN(renda_cli) |
|----------------|
| 1020.00        |

# Regras sobre Funções

1. Funções não podem ser usadas em conjunto com atributos no mesmo SELECT. **Exemplo:** *SELECT nome\_cli, MAX(renda\_cli) FROM Cliente;*
2. Funções não podem ser usadas dentro de condições. **Exemplo:** *SELECT \* FROM Cliente WHERE (renda\_cli > MAX(renda\_cli));*



# Funções Especiais

- **CURTIME():** Retorna o horário atual do sistema operacional no momento da execução.
  - **Exemplo:** *select curtime();*
- **CURDATE():** Retorna a data atual do sistema operacional no momento da execução.
  - **Exemplo:** *select curdate();*
- **DATE\_FORMAT(atributo, '%d/%m/%Y'):** Retorna a data no formato da máscara definida.
  - **Exemplo:** *SELECT DATE\_FORMAT(data\_nascimento\_cli, '%d/%m/%Y') from cliente;*

# Funções Especiais

- **EXTRACT**(**year** FROM **atributo**): Retorna somente o ano (**year**), mês (**month**) ou dia (**day**) de um atributo data.
  - **Exemplo:** *SELECT EXTRACT(year FROM **datanasc\_cli**) FROM cliente;*
- **ROUND**(**atributo**, N<sup>o</sup> **casas decimais**): Retorna o número com a quantidade de casas decimais definidas na função. Observe a seguir que 2 é a quantidade de números depois da vírgula.
  - **Exemplo:** *SELECT ROUND(**valor\_total\_vend**, 2) FROM venda;*

# Funções Especiais

- **WEEKDAY(atributo)**: Retornar o número do dia da semana para uma data. Conforme a sequência:
  - 0 = segunda, 1 = terça, 2 = quarta, 3 = quinta, 4 = sexta, 5 = sábado, 6 = domingo
  - **Exemplo:** *SELECT WEEKDAY(data\_nascimento\_cli) FROM cliente;*

# Sub Consultas Simples

- **Sintaxe da sub consulta simples:**
  - **SELECT**
  - atributo1,
  - atributo2
  - **FROM**
  - Tabela1
  - **WHERE**
  - (atributo3 = (***SELECT atributo1 FROM Tabela1 WHERE condição***));

# Exemplo de Sub Consultas Simples

- **Objetivo:** Selecione o nome e a renda do cliente que possui a maior renda.

- **SELECT**

- nome\_cli as 'Cliente',

- renda\_cli as 'Maior Renda'

- **FROM**

- Cliente

- **WHERE**

- (renda\_cli = (**SELECT MAX**(renda\_cli) **FROM** cliente));

# Exemplo de Sub Consultas Simples

- **Objetivo:** Selecione o nome e a renda do cliente que possui renda superior a menor renda.

➤ **SELECT**

➤ nome\_cli as 'Cliente',

➤ renda\_cli as 'Maior Renda'

➤ **FROM**

➤ Cliente

➤ **WHERE**

➤ (renda\_cli > (SELECT **MIN**(renda\_cli) **FROM** cliente);

# Exemplo de Sub Consultas Simples

- **Objetivo:** Selecione o nome e a renda do cliente que possui a renda menor do que a média das rendas dos clientes cadastrados.

- **SELECT**

- nome\_cli as 'Cliente',

- renda\_cli as 'Renda'

- **FROM**

- Cliente

- **WHERE**

- (renda\_cli < (SELECT **AVG**(renda\_familiar\_cli) **FROM** cliente));

# Atividade de Fixação

Utilize o **Banco de Dados da Mecânica** e faça os exercícios a seguir:

1. Selecione os clientes pelo nome em ordem crescente;
2. Selecione os clientes pela data de nascimento em ordem decrescente;
3. Selecione os clientes com os sobrenomes Silva ou Souza, do sexo masculino, com renda entre 500 e 10.000 reais e nascidos após 01/01/1980 ;
4. Mostre quantos clientes existem cadastrados;
5. Mostre os dados do cliente que possui a maior renda;
6. Mostre a média das rendas de todos os clientes;



# Atividade de Fixação

7. Mostre o valor médio dos produtos cadastrados;
8. Mostre quais são os estados civis diferentes existentes na tabela cliente;
9. Selecione o dados dos clientes do RG nulo;
10. Mostre quantas cores diferentes existem na tabela carro;
11. Mostre quais são as cores diferentes existem na tabela carro;
12. Mostre todos os dados dos clientes que possuem renda inferior a média ordenados por renda;
13. Mostre todos os dados dos clientes que possuem renda superior a média ordenados por renda decrescente;

# Atividade de Fixação

14. Mostre o menor estoque, maior estoque, estoque médio e soma do estoque dos produtos;
15. Mostre o menor valor, maior valor, valor médio e soma do valor dos produtos;
16. Mostre o nome e o estoque do produto com o menor estoque;
17. Mostre o nome e o estoque do produto com o maior estoque;
18. Mostre o nome, a função e o salario do funcionário com menor salario;
19. Mostre o nome, a função e o salario do funcionário com maior salario;
20. Mostre os dados do serviço do tipo 'conserto' ou 'estética' e com valor superior a 10.00 e que possuem o valor inferior a média;

# Jackson Henrique

Professor

E-mail:

**[Jackson.henrique@ifro.edu.br](mailto:Jackson.henrique@ifro.edu.br)**