

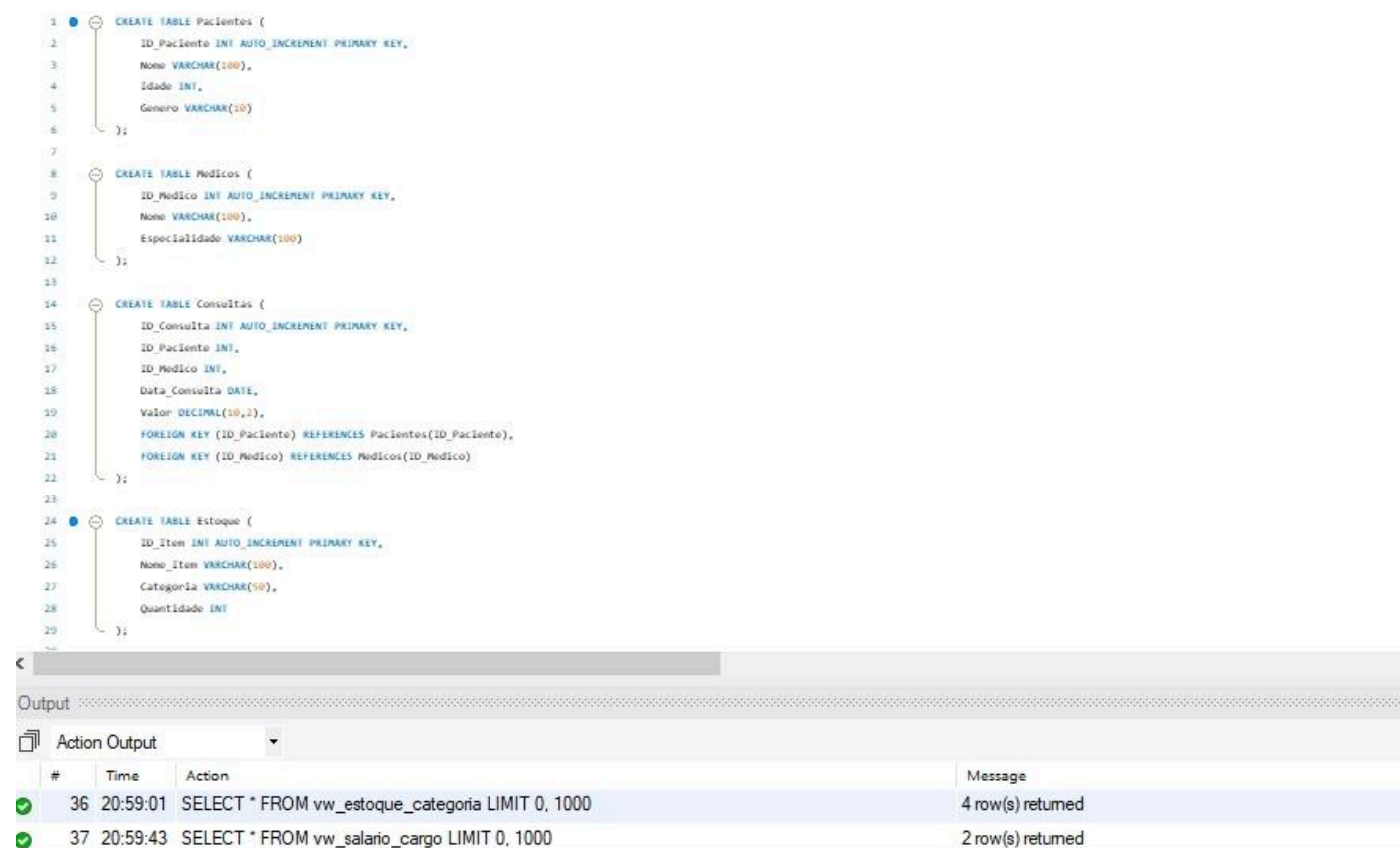
Davi Resende
Rian Salin

Estrutura, Pesquisa e Ordenação de Dados;

Criação de 3 Views simples, com prints da execução, dos códigos da criação de tabelas e da inserção de dados como exemplo:

Baseado no excel e relatórios da gestão clínica hospitalar, ainda não importado para o PowerBI e feito/rodado no MySQL. Dados inseridos fictícios, desde a concepção da última versão final.

Print de Criação de Tabelas, com execução correta:



```
1 CREATE TABLE Pacientes (  
2     ID_Paciente INT AUTO_INCREMENT PRIMARY KEY,  
3     Nome VARCHAR(100),  
4     Idade INT,  
5     Genero VARCHAR(10)  
6 );  
7  
8 CREATE TABLE Medicos (  
9     ID_Medico INT AUTO_INCREMENT PRIMARY KEY,  
10    Nome VARCHAR(100),  
11    Especialidade VARCHAR(100)  
12 );  
13  
14 CREATE TABLE Consultas (  
15     ID_Consulta INT AUTO_INCREMENT PRIMARY KEY,  
16     ID_Paciente INT,  
17     ID_Medico INT,  
18     Data_Consulta DATE,  
19     Valor DECIMAL(10,2),  
20     FOREIGN KEY (ID_Paciente) REFERENCES Pacientes(ID_Paciente),  
21     FOREIGN KEY (ID_Medico) REFERENCES Medicos(ID_Medico)  
22 );  
23  
24 CREATE TABLE Estoque (  
25     ID_Item INT AUTO_INCREMENT PRIMARY KEY,  
26     Nome_Item VARCHAR(100),  
27     Categoria VARCHAR(50),  
28     Quantidade INT  
29 );
```

Output

Action Output

#	Time	Action	Message
✓ 36	20:59:01	SELECT * FROM vw_estoque_categoria LIMIT 0, 1000	4 row(s) returned
✓ 37	20:59:43	SELECT * FROM vw_salario_cargo LIMIT 0, 1000	2 row(s) returned

```

9      ID_Medico INT AUTO_INCREMENT PRIMARY KEY,
10     Nome VARCHAR(100),
11     Especialidade VARCHAR(100)
12 );
13
14 CREATE TABLE Consultas (
15     ID_Consulta INT AUTO_INCREMENT PRIMARY KEY,
16     ID_Paciente INT,
17     ID_Medico INT,
18     Data_Consulta DATE,
19     Valor DECIMAL(10,2),
20     FOREIGN KEY (ID_Paciente) REFERENCES Pacientes(ID_Paciente),
21     FOREIGN KEY (ID_Medico) REFERENCES Medicos(ID_Medico)
22 );
23
24 CREATE TABLE Estoque (
25     ID_Item INT AUTO_INCREMENT PRIMARY KEY,
26     Nome_Item VARCHAR(100),
27     Categoria VARCHAR(50),
28     Quantidade INT
29 );
30
31 CREATE TABLE Funcionarios (
32     ID_Funcionario INT AUTO_INCREMENT PRIMARY KEY,
33     Nome VARCHAR(100),
34     Cargo VARCHAR(50),
35     Salario DECIMAL(10,2)
36 );
37

```

Output			
Action Output			
#	Time	Action	Message
✓ 36	20:59:01	SELECT * FROM vw_estoque_categoria LIMIT 0, 1000	4 row(s) returned
✓ 37	20:59:43	SELECT * FROM vw_salario_cargo LIMIT 0, 1000	2 row(s) returned

Código Criação de Tabelas no MYSQL, baseado no Excel;

```

CREATE TABLE Pacientes (
    ID_Paciente INT AUTO_INCREMENT PRIMARY KEY,
    Nome VARCHAR(100),
    Idade INT,
    Genero VARCHAR(10)
);

```

```

CREATE TABLE Medicos (
    ID_Medico INT AUTO_INCREMENT PRIMARY KEY,
    Nome VARCHAR(100),
    Especialidade VARCHAR(100)
);

```

```

CREATE TABLE Consultas (
    ID_Consulta INT AUTO_INCREMENT PRIMARY KEY,
    ID_Paciente INT,
    ID_Medico INT,
    Data_Consulta DATE,
    Valor DECIMAL(10,2),
    FOREIGN KEY (ID_Paciente) REFERENCES Pacientes(ID_Paciente),
    FOREIGN KEY (ID_Medico) REFERENCES Medicos(ID_Medico)
);

```

```

CREATE TABLE Estoque (
    ID_Item INT AUTO_INCREMENT PRIMARY KEY,
    Nome_Item VARCHAR(100),

```

```

    Categoria VARCHAR(50),
    Quantidade INT
);

CREATE TABLE Funcionarios (
    ID_Funcionario INT AUTO_INCREMENT PRIMARY KEY,
    Nome VARCHAR(100),
    Cargo VARCHAR(50),
    Salario DECIMAL(10,2)
);

```

Print da Inserção de Dados de exemplo no MySQL;

```

1 • INSERT INTO Pacientes (Nome, Idade, Genero) VALUES
2   ('Ana Souza', 34, 'Feminino'),
3   ('Carlos Lima', 45, 'Masculino');
4
5 • INSERT INTO Medicos (Nome, Especialidade) VALUES
6   ('Dra. Fernanda', 'Cardiologia'),
7   ('Dr. Paulo', 'Ortopedia');
8
9 • INSERT INTO Consultas (ID_Paciente, ID_Medico, Data_Consulta, Valor) VALUES
10  (1, 1, '2025-05-01', 250.00),
11  (2, 2, '2025-05-02', 300.00);
12
13 • INSERT INTO Estoque (Nome_Item, Categoria, Quantidade) VALUES
14  ('Luvas', 'Equipamentos', 100),
15  ('Seringa', 'Consumíveis', 200);
16
17 • INSERT INTO Funcionarios (Nome, Cargo, Salario) VALUES
18  ('João Silva', 'Recepcionista', 2000.00),
19  ('Maria Oliveira', 'Enfermeira', 3500.00);
20

```

Output

#	Time	Action	Message
36	20:59:01	SELECT * FROM vw_estoque_categoria LIMIT 0, 1000	4 row(s) returned
37	20:59:43	SELECT * FROM vw_salario_cargo LIMIT 0, 1000	2 row(s) returned

Código Inserção de dados de exemplo no MYSQL;

```

INSERT INTO Pacientes (Nome, Idade, Genero) VALUES
('Ana Souza', 34, 'Feminino'),
('Carlos Lima', 45, 'Masculino');

```

```

INSERT INTO Medicos (Nome, Especialidade) VALUES
('Dra. Fernanda', 'Cardiologia'),
('Dr. Paulo', 'Ortopedia');

```

```

INSERT INTO Consultas (ID_Paciente, ID_Medico, Data_Consulta, Valor) VALUES
(1, 1, '2025-05-01', 250.00),

```

(2, 2, '2025-05-02', 300.00);

INSERT INTO Estoque (Nome_Item, Categoria, Quantidade) VALUES
('Luvas', 'Equipamentos', 100),
('Seringa', 'Consumíveis', 200);

INSERT INTO Funcionarios (Nome, Cargo, Salario) VALUES
('João Silva', 'Recepcionista', 2000.00),
('Maria Oliveira', 'Enfermeira', 3500.00);

Print da Criação de 3 Views;

São elas:

View para informações completas das consultas;

View para o total de estoque por categoria;

View que informa os salários por cargo.

```
1  -- View com informações completas das consultas
2  CREATE OR REPLACE VIEW vw_consultas_completas AS
3  SELECT
4      c.ID_Conсульта,
5      p.Nome AS Nome_Paciente,
6      p.Idade,
7      p.Genero,
8      m.Nome AS Nome_Medico,
9      m.Especialidade,
10     c.Data_Conсульта,
11     c.Valor
12 FROM Consultas c
13 JOIN Pacientes p ON c.ID_Paciente = p.ID_Paciente
14 JOIN Medicos m ON c.ID_Medico = m.ID_Medico;
15
16 -- View de total de estoque por categoria
17 CREATE OR REPLACE VIEW vw_estoque_categoria AS
18 SELECT
19     Categoria,
20     SUM(Quantidade) AS Total_Items
21 FROM Estoque
22 GROUP BY Categoria;
23
24 -- View de salários por cargo
25 CREATE OR REPLACE VIEW vw_salario_cargo AS
26 SELECT
27     Cargo,
28     COUNT(*) AS Quantidade_Funcionarios,
29     AVG(Salario) AS Salario_Medio
30 FROM Funcionarios
31 GROUP BY Cargo;
```

Output

Action Output

#	Time	Action	Message
36	20:59:01	SELECT * FROM vw_estoque_categoria LIMIT 0, 1000	4 row(s) returned
37	20:59:43	SELECT * FROM vw_salario_cargo LIMIT 0, 1000	2 row(s) returned

Código da Criação das 3 Views no MYSQL;

-- View com informações completas das consultas

```
CREATE OR REPLACE VIEW vw_consultas_completas AS
SELECT
    c.ID_Conсульта,
    p.Nome AS Nome_Paciente,
    p.Idade,
    p.Genero,
    m.Nome AS Nome_Medico,
    m.Especialidade,
    c.Data_Conсульта,
    c.Valor
FROM Consultas c
JOIN Pacientes p ON c.ID_Paciente = p.ID_Paciente
JOIN Medicos m ON c.ID_Medico = m.ID_Medico;
```

-- View de total de estoque por categoria

```
CREATE OR REPLACE VIEW vw_estoque_categoria AS
SELECT
    Categoria,
    SUM(Quantidade) AS Total_Itens
FROM Estoque
GROUP BY Categoria;
```

-- View de salários por cargo




```
CREATE OR REPLACE VIEW vw_salario_cargo AS
SELECT
    Cargo,
    COUNT(*) AS Quantidade_Funcionarios,
    AVG(Salario) AS Salario_Medio
FROM Funcionarios
GROUP BY Cargo;
```

Print da Execução da View de Informações de Consultas Completas;

SQL File 3* SQL File 3* SQL File 4* SQL File 5* x

Limit to 1000 rows




```
1 • SELECT * FROM vw_consultas_completas;
2
```

Result Grid Filter Rows: Export: Wrap Cell Content:  Result Grid  Form Editor  Field Types

ID_Consulta	Nome_Paciente	Idade	Genero	Nome_Medico	Especialidade	Data_Consulta	Valor
1	João da Silva	45	Masculino	Dr. Renato Lopes	Cardiologia	2024-05-01	350.00
3	João da Silva	45	Masculino	Dr. Lucas Braga	Ortopedia	2024-05-07	410.00
5	João da Silva	45	Masculino	Dr. Renato Lopes	Cardiologia	2024-05-01	350.00
7	João da Silva	45	Masculino	Dr. Lucas Braga	Ortopedia	2024-05-07	410.00
9	João da Silva	45	Masculino	Dr. Renato Lopes	Cardiologia	2024-05-01	350.00
11	João da Silva	45	Masculino	Dr. Lucas Braga	Ortopedia	2024-05-07	410.00
13	João da Silva	45	Masculino	Dr. Renato Lopes	Cardiologia	2025-05-01	250.00
2	Maria Souza	33	Feminino	Dra. Fernanda Lima	Dermatologia	2024-05-03	220.00
6	Maria Souza	33	Feminino	Dra. Fernanda Lima	Dermatologia	2024-05-03	220.00
10	Maria Souza	33	Feminino	Dra. Fernanda Lima	Dermatologia	2024-05-03	220.00
14	Maria Souza	33	Feminino	Dra. Fernanda Lima	Dermatologia	2025-05-02	300.00
4	Carlos Alberto	60	Masculino	Dr. Renato Lopes	Cardiologia	2024-05-10	300.00

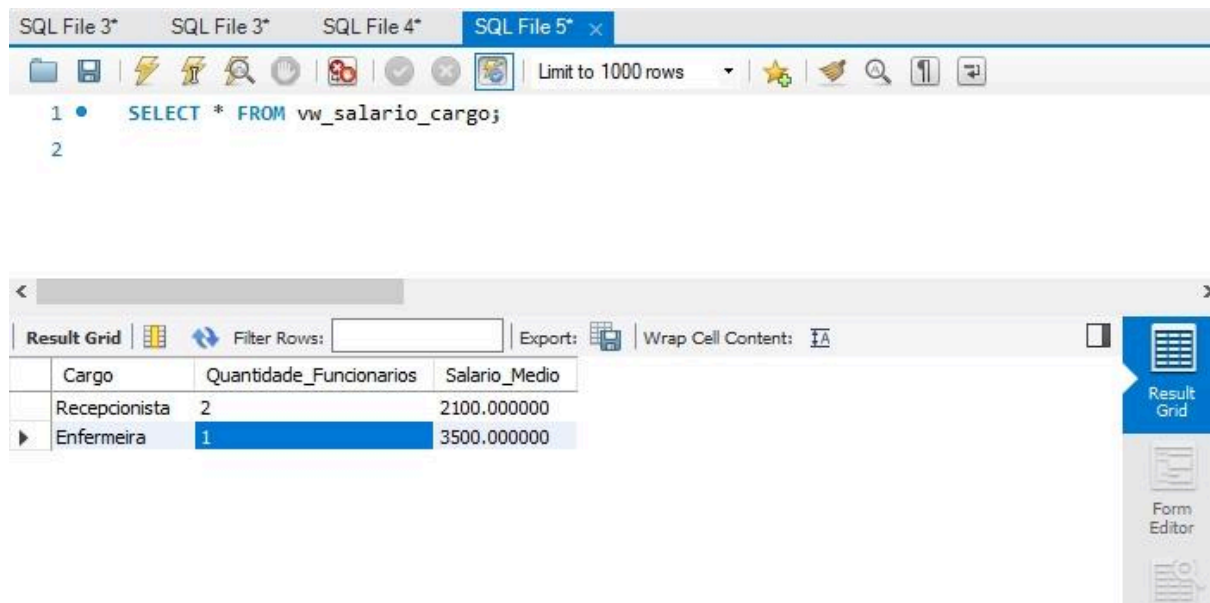
Print da Execução da View de Total de Estoque por Categoria;

```
1 • SELECT * FROM vw_estoque_categoria;
2
```

Result Grid Filter Rows: Export: Wrap Cell Content:  Result Grid  Form Editor  Field Types

Categoria	Total_Itens
EPI	600
Materiais	750
Equipamentos	100
Consumíveis	200

Print da Execução da View de Salários por cargo;



The screenshot displays a SQL IDE interface. The top toolbar includes icons for file operations, execution, and a 'Limit to 1000 rows' dropdown. The SQL editor contains the following query:

```
1 • SELECT * FROM vw_salario_cargo;  
2
```

Below the editor, the 'Result Grid' is visible, showing the execution results in a table format. The table has three columns: 'Cargo', 'Quantidade_Funcionarios', and 'Salario_Medio'. The results are as follows:

Cargo	Quantidade_Funcionarios	Salario_Medio
Recepcionista	2	2100.000000
Enfermeira	1	3500.000000

On the right side of the interface, there are buttons for 'Result Grid' and 'Form Editor'.