



BACHARELADO EM CIÊNCIA E TECNOLOGIA

BANCO DE DADOS

João Paulo Paixão Rocha - 156408

João Pedro Andrade Carneiro - 168806

Victor Augusto Reis Marques - 156620

São José dos Campos - SP

Fevereiro - 2025

1.INTRODUÇÃO

Este relatório apresenta a segunda parte do projeto desenvolvido para a disciplina de Banco de Dados do curso de Bacharelado em Ciência da Computação da Universidade Federal de São Paulo. O objetivo do trabalho é a implementação de um banco de dados relacional a partir do modelo conceitual desenvolvido na primeira etapa do projeto. O domínio escolhido para este estudo é um sistema de gestão de atendimentos clínicos, englobando o gerenciamento de pacientes, consultas, profissionais de saúde, setores administrativos e controle de estoque de medicamentos.

Na primeira parte do projeto, foram definidos os requisitos do sistema e elaborado um Modelo Entidade-Relacionamento (ER) detalhado, representando as principais entidades e relacionamentos do domínio. O diagrama ER, apresentado na Figura 1, demonstra a estrutura do banco de dados, incluindo entidades como Funcionário, Paciente, Consulta, Consultório, Setor e Remédios, além de seus atributos e relações.

Nesta segunda etapa, o foco foi a implementação do banco de dados no ambiente de um Sistema Gerenciador de Banco de Dados (SGBD), utilizando a linguagem MariaDB para a criação das tabelas e a inserção de dados. Além disso, foram desenvolvidas consultas para extração de informações relevantes do sistema, contemplando operações básicas, junções e agregações. Por fim, uma interface gráfica utilizando Django, um framework web usado para desenvolvimento de aplicações web usando Python, foi criada para a interação com o banco de dados, possibilitando operações CRUD (Create, Read, Update, Delete) sobre os dados armazenados.

As próximas seções deste relatório detalham as etapas de implementação, incluindo a definição das tabelas e seus atributos, os comandos SQL utilizados para manipulação dos dados e a interface gráfica desenvolvida para o sistema.

2.PROBLEMA

O sistema foi projetado para representar um hospital que possui diversos funcionários que possuem um código único, nome e vínculo (CLT ou CNPJ). Os funcionários são da equipe clínica ou administrativa, e é mantido um controle do histórico desses, armazenando informações como a data de início e fim do status. Na equipe administrativa é mantido o cargo do funcionário, bem como o setor em que ele atua. Assim, a administração é organizada em setores que possuem um código único, nome e localização, e são coordenados por um funcionário administrativo.

A equipe clínica, por sua vez, é composta por médicos e enfermeiros. Os médicos possuem registro no Conselho Regional de Medicina (CRM) e especializações específicas, enquanto os enfermeiros têm seu registro no Conselho Regional de Enfermagem (COREN). Cada clínico realiza atendimentos em consultórios específicos, os quais possuem uma sala, descrição e um funcionário responsável.

As consultas realizadas no hospital envolvem pelo menos um clínico, que pode ser médico ou enfermeiro, e são vinculadas a um consultório, atendendo a um paciente identificado pelo seu código único, nome, CPF, data de nascimento, endereço e telefone. Além disso, as consultas podem incluir a prescrição de remédios. Esses remédios são registrados no sistema com informações como nome, tipo, descrição e indicação se são controlados. As unidades específicas dos remédios possuem detalhes como lote, data de vencimento e laboratório responsável, e estão vinculadas ao estoque, onde é registrada a data de compra e a quantidade disponível.

Esse modelo de dados também permite gerenciar os vínculos entre clínicos e consultórios, possibilitando o registro das unidades de remédios utilizadas em cada consulta e garantindo a rastreabilidade das informações de pacientes, funcionários, setores e medicamentos.

a. Diagrama Entidade-Relacionamento (ER)

Integrando a parte de modelagem conceitual de um banco de dados, foi criado o diagrama entidade-relacionamento para estruturar e visualizar os principais elementos do sistema antes da implementação, exibindo as entidades, atributos e relacionamentos entre elas.

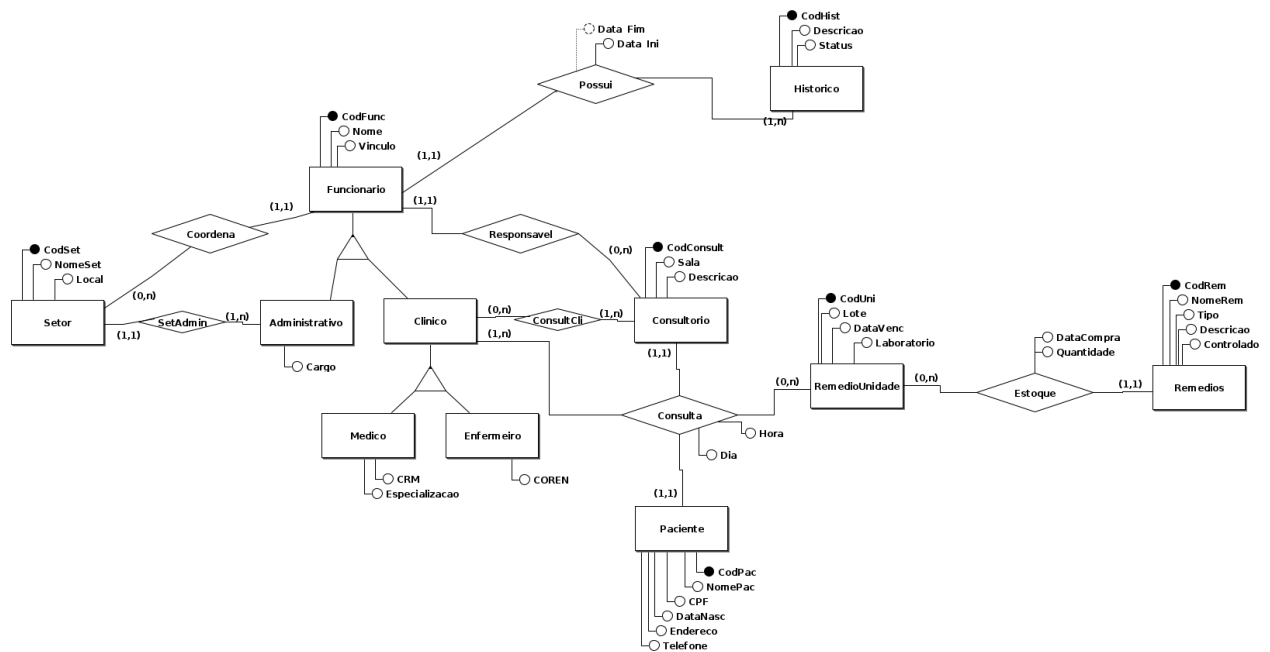


Figura 1: Diagrama Modelo Relacional

b. Lista de tabelas

A partir do diagrama ER (Figura 1) foi desenvolvido a modelagem lógica do bando de dados, sendo representada no formato de esquema relacional. Esse modelo já está mais próximo da implementação:

1. Funcionario(CodFunc, Nome, Vinculo)
2. Administrativo(#CodFunc, #CodSet, Cargo)
3. Medico(#CodFunc, CRM, Especializacao)
4. Enfermeiro(#CodFunc, COREN)
5. Setor(CodSet, NomeSet, Local, #CodFunc)
6. Historico(CodHist, Descricao, Status, #CodFunc, Data_Ini, Data_Fim)
7. Consultorio(CodConsult, Sala, Descricao, #CodFunc)
8. Consulta(#CodFunc, #CodUnid, Dia, Hora, #CodConsult, #CodPac)
9. ConsultCli(#CodConsult, #CodFunc)
10. Paciente(CodPac, NomePac, CPF, DataNasc, Endereco, Telefone)
11. Remedios(CodRem, NomeRem, Tipo, Descricao, Controlado)
12. RemedioUnidade(CodUnid, Lote, DataVenc, Laboratorio, #CodRem)
13. Estoque(#CodUnid, #CodRem, DataCompra, Quantidade)

3. CRIAÇÃO DO BANCO DE DADOS

Este tópico apresenta os *scripts* SQL responsáveis pela criação e estruturação do banco de dados deste projeto, incluindo a definição das tabelas, chaves primárias e estrangeiras, além das restrições necessárias para garantir a integridade dos dados.

```
CREATE DATABASE Trabalho;
```

```
CREATE TABLE Funcionario(  
    CodFunc          int,  
    Nome             varchar(60),  
    Vinculo           varchar(20),  
    PRIMARY KEY(CodFunc)  
);
```

```
CREATE TABLE Setor(  
    CodSet           int,  
    NomeSet          varchar(20),  
    Local            varchar(20),  
    CodFunc          int,  
    PRIMARY KEY(CodSet),  
    FOREIGN KEY (CodFunc) REFERENCES Funcionario(CodFunc)  
);
```

```
CREATE TABLE Administrativo(  
    CodFunc          int,  
    CodSet           int,  
    Cargo            varchar(20),  
    PRIMARY KEY (CodFunc, CodSet),  
    FOREIGN KEY (CodFunc) REFERENCES Funcionario(CodFunc),  
    FOREIGN KEY (CodSet) REFERENCES Setor(CodSet)  
);
```

```

CREATE TABLE Medico(
    CodFunc          int PRIMARY KEY,
    CRM              varchar(20) NOT NULL,
    Especializacao   varchar(50),
    FOREIGN KEY (CodFunc) REFERENCES Funcionario(CodFunc)
);

```

```

CREATE TABLE Enfermeiro(
    CodFunc          int PRIMARY KEY,
    COREN            varchar(20) NOT NULL,
    FOREIGN KEY (CodFunc) REFERENCES Funcionario(CodFunc)
);

```

```

CREATE TABLE Historico(
    CodHist          int PRIMARY KEY,
    Descricao        TEXT,
    STATUS            varchar(20) NOT NULL,
    CodFunc           int,
    Data_ini          DATE NOT NULL,
    Data_fim          DATE,
    FOREIGN KEY (CodFunc) REFERENCES Funcionario(CodFunc)
);

```

```

CREATE TABLE Consultorio(
    CodConsult       int PRIMARY KEY,
    Sala              varchar(10),
    Descricao         TEXT,
    CodFunc           int,
    FOREIGN KEY (CodFunc) REFERENCES Funcionario(CodFunc)
);

```

```

CREATE TABLE ConsultCli(

```

```

        CodConsult int,
        CodFunc      int,
        PRIMARY KEY (CodConsult, CodFunc),
        FOREIGN KEY (CodConsult) REFERENCES Consultorio(CodConsult),
        FOREIGN KEY (CodFunc) REFERENCES Funcionario(CodFunc)
    );

```

```

CREATE TABLE Paciente(
    CodPac      int PRIMARY KEY,
    NomePac     varchar(50),
    CPF         varchar(14) NOT NULL UNIQUE,
    DataNasc    DATE,
    Endereco    TEXT,
    Telefone    varchar(15)
);

```

```

CREATE TABLE Remedios(
    CodRem      int PRIMARY KEY,
    NomeRem     varchar(50),
    Tipo        varchar(20),
    Descricao   TEXT,
    Controlado  BOOLEAN NOT NULL
);

```

```

CREATE TABLE RemedioUnidade(
    CodUnid     int,
    Lote        varchar(20),
    DataVenc    DATE NOT NULL,
    Laboratorio varchar(50),
    CodRem      int,
    PRIMARY KEY (CodUnid),
    FOREIGN KEY (CodRem) REFERENCES Remedios(CodRem)
);

```

);

```
CREATE TABLE Estoque(  
    CodUnid          int,  
    CodRem           int,  
    DataCompra DATE,  
    Quantidade int,  
    PRIMARY KEY (CodUnid, CodRem),  
    FOREIGN KEY (CodUnid) REFERENCES RemedioUnidade(CodUnid),  
    FOREIGN KEY (CodRem) REFERENCES Remedios(CodRem)  
);
```

```
CREATE TABLE Consulta(  
    CodFunc          int,  
    CodUnid          int,  
    Dia              DATE,  
    Hora             TIME,  
    CodConsult int,  
    CodPac           int,  
    PRIMARY KEY (CodFunc, CodUnid),  
    FOREIGN KEY (CodFunc) REFERENCES Funcionario(CodFunc),  
    FOREIGN KEY (CodUnid) REFERENCES RemedioUnidade(CodUnid),  
    FOREIGN KEY (CodConsult) REFERENCES Consultorio(CodConsult),  
    FOREIGN KEY (CodPac) REFERENCES Paciente(CodPac)  
);
```


4. INSERÇÃO

Esse tópico aborda os *scripts* SQL utilizados para a população do banco de dados, inserindo as informações para a realização das consultas no próximo tópico.

INSERT INTO Funcionario (CodFunc, Nome, Vinculo)

VALUES (1,'João Silva', 'CLT'),
(2,'Maria Oliveira', 'CLT'),
(3,'Carlos Souza', 'CNPJ'),
(4,'Yuri Baldi', 'CNPJ')
(5,'Roberson dos Santos', 'CLT'),
(6,'Victor Marques', 'CLT'),
(7,'Daniela Musa', 'CNPJ'),
(8,'João Carneiro', 'CNPJ'),
(9,'João Rocha', 'CLT'),
(10, 'Beatriz Silve', 'CLT'),
(11, 'Fernanda Almeida', 'CLT'),
(12, 'Carlos Mendes', 'CNPJ'),
(13, 'Roberto Farias', 'CLT'),
(14, 'Aline Castro', 'CNPJ'),
(15, 'Gustavo Lima', 'CLT'),
(16, 'Patrícia Fonseca', 'CNPJ'),
(17, 'Lucas Silva', 'CLT'),
(18, 'Ana Paula Santos', 'CNPJ'),
(19, 'Fábio Ribeiro', 'CLT'),
(20, 'Camila Teixeira', 'CNPJ'),
(21, 'Rafael Moreira', 'CLT'),
(22, 'Juliana Alves', 'CLT'),
(23, 'Marcos Vinícius', 'CNPJ'),
(24, 'Cláudia Rocha', 'CLT'),
(25, 'Eduardo Matos', 'CNPJ'),
(26, 'Renata Oliveira', 'CLT'),
(27, 'Thiago Ferreira', 'CNPJ'),

```
(28, 'Letícia Ramos', 'CLT'),
(29, 'Bruno Costa', 'CLT'),
(30, 'Viviane Santos', 'CNPJ');
```

```
INSERT INTO Setor (CodSet, NomeSet, Local, CodFunc)
```

```
VALUES    (1, 'Administrativo', 'Bloco A', 1),
           (2, 'Financeiro', 'Bloco B', 2),
           (3, 'Recursos Humanos', 'Bloco C', 3),
           (4, 'Recepção', 'Hall', 4),
           (5, 'Des. de Disp. Biom.', 'Bloco D', 5),
           (6, 'Tec. da Inf.', 'Bloco E', 6),
           (7, 'Suporte Técnico', 'Bloco F', 7),
           (8, 'Projetos', 'Bloco G', 8),
           (9, 'Manut. de Equip. Méd', 'Bloco H', 9),
           (10, 'Pesq. em Eng. Biom.', 'Bloco I', 10);
```

```
INSERT INTO Administrativo(CodFunc, CodSet, Cargo)
```

```
VALUES    (1, 1, 'Gerente'),
           (2, 2, 'Analista'),
           (3, 3, 'Gerente'),
           (4, 4, 'Recepcionista'),
           (5, 5, 'Analista'),
           (6, 6, 'Supervisor'),
           (7, 7, 'Coordenador'),
           (8, 8, 'Analista'),
           (9, 9, 'Analista'),
           (10, 10, 'Pesquisador');
```

```
INSERT INTO Medico (CodFunc, CRM, Especializacao)
```

```
VALUES    (11, '123456-SP', 'Cardiologia'),
           (12, '789012-SP', 'Pediatria'),
```

```

(13, '345678-RJ', 'Ortopedia'),
(14, '567890-MG', 'Dermatologia'),
(15, '234567-SP', 'Gastroenterologia'),
(16, '890123-RJ', 'Neurologia'),
(17, '456789-MG', 'Psiquiatria'),
(18, '678901-PR', 'Oftalmologia'),
(19, '345123-SP', 'Ginecologia'),
(20, '901234-RS', 'Anestesiologia');

```

INSERT INTO Enfermeiro (CodFunc, COREN)

```

VALUES      (21, 'COREN-12345-SP'),
            (22, 'COREN-67890-SP'),
            (23, 'COREN-23456-MG'),
            (24, 'COREN-78901-RJ'),
            (25, 'COREN-34567-PR'),
            (26, 'COREN-89012-BA'),
            (27, 'COREN-45678-RS'),
            (28, 'COREN-90123-SC'),
            (29, 'COREN-56789-PE'),
            (30, 'COREN-01234-AL');

```

INSERT INTO Consultorio (CodConsult, Sala, Descricao, CodFunc)

```

VALUES      (1, '101', 'Consultório de Cardiologia', 11),
            (2, '102', 'Consultório de Pediatria', 12),
            (3, '103', 'Consultório de Ortopedia', 13),
            (4, '104', 'Consultório de Dermatologia', 14),
            (5, '105', 'Consultório de Gastroenterologia', 15),
            (6, '106', 'Consultório de Neurologia', 16),
            (7, '107', 'Consultório de Psiquiatria', 17),
            (8, '108', 'Consultório de Oftalmologia', 18),
            (9, '109', 'Consultório de Ginecologia', 19),
            (10, '110', 'Consultório de Anestesiologia', 20);

```

```

INSERT INTO Paciente (CodPac, NomePac, CPF, DataNasc, Endereco, Telefone)
VALUES      (1, 'Amanda Silva', '123.456.789-01', '1990-05-12', 'Rua das Flores,
123', '(11) 98765-4321'),

            (2, 'Bruno Costa', '234.567.890-12', '1985-08-19', 'Av. Paulista, 456',
'(11) 91234-5678'),

            (3, 'Carla Mendes', '345.678.901-23', '1993-11-22', 'Rua Afonso Pena,
789', '(21) 99876-5432'),

            (4, 'Daniel Ferreira', '456.789.012-34', '1988-03-15', 'Praça Central,
101', '(31) 92345-6789'),

            (5, 'Elaine Rocha', '567.890.123-45', '1995-07-07', 'Rua dos Lírios,
202', '(41) 93210-9876'),

            (6, 'Fábio Ribeiro', '678.901.234-56', '1991-12-03', 'Av. das Palmeiras,
303', '(51) 94432-1567'),

            (7, 'Gustavo Lima', '789.012.345-67', '1987-09-10', 'Rua das
Orquídeas, 404', '(61) 95543-2678'),

            (8, 'Heloísa Martins', '890.123.456-78', '1994-01-18', 'Av. Principal,
505', '(71) 96654-3789'),

            (9, 'Isabela Fonseca', '901.234.567-89', '1992-06-25', 'Rua Secundária,
606', '(81) 97765-4890'),

            (10, 'João Alves', '012.345.678-90', '1989-04-09', 'Praça Nova, 707',
'(91) 98876-5901'),

            (11, 'Karina Oliveira', '123.654.987-01', '1996-02-28', 'Av. do Sol, 808',
'(31) 99987-1122'),

            (12, 'Leonardo Santos', '234.765.098-12', '1984-10-14', 'Rua Nova
Esperança, 909', '(41) 91123-2233'),

            (13, 'Mariana Costa', '345.876.109-23', '1997-08-21', 'Rua das Rosas,
1010', '(51) 92234-3344'),

            (14, 'Nathália Ribeiro', '456.987.210-34', '1986-11-30', 'Av. da Paz,
1111', '(61) 93345-4455'),

            (15, 'Otávio Souza', '567.098.321-45', '1993-07-04', 'Praça das Águas,
1212', '(71) 94456-5566');

```

```

INSERT INTO Remedios (CodRem, NomeRem, Tipo, Descricao, Controlado)
VALUES      (1, 'Paracetamol', 'Analgésico', 'Utilizado para alívio de dores e febres.',
FALSE),

            (2, 'Ibuprofeno', 'Anti-inflamatório', 'Usado no tratamento de
inflamações e dores.', FALSE),

```

(3, 'Amoxicilina', 'Antibiótico', 'Antibiótico de amplo espectro para infecções.', TRUE),

(4, 'Diazepam', 'Ansiolítico', 'Utilizado no tratamento de ansiedade.', TRUE),

(5, 'Omeprazol', 'Antiácido', 'Usado no tratamento de gastrite e refluxo.', FALSE),

(6, 'Metformina', 'Antidiabético', 'Tratamento de diabetes tipo 2.', FALSE),

(7, 'Losartana', 'Antihipertensivo', 'Indicado para controle da pressão arterial.', FALSE),

(8, 'Cetirizina', 'Antialérgico', 'Usado no tratamento de alergias.', FALSE),

(9, 'Rivotril', 'Ansiolítico', 'Usado no tratamento de ansiedade e epilepsia.', TRUE),

(10, 'Prednisona', 'Corticosteroide', 'Indicada para tratamento de inflamações graves.', TRUE),

(11, 'Aspirina', 'Analgésico', 'Usado para aliviar dores leves e febres.', 0),

(12, 'Cloroquina', 'Antimalárico', 'Tratamento de malária e doenças autoimunes.', 1),

(13, 'Loratadina', 'Antialérgico', 'Usado no tratamento de alergias e rinite.', 0),

(14, 'Fluoxetina', 'Antidepressivo', 'Utilizado no tratamento de depressão e ansiedade.', 1),

(15, 'Carbamazepina', 'Anticonvulsivante', 'Indicado para o tratamento de epilepsia.', 1);

INSERT INTO RemedioUnidade (CodUnid, Lote, DataVenc, Laboratorio, CodRem)
VALUES

(1, 'L12345', '2025-12-31', 'Laboratório ABC', 1),

(2, 'L23456', '2024-06-15', 'Laboratório XYZ', 2),

(3, 'L34567', '2025-08-20', 'Farmacêutica Nacional', 3),

(4, 'L45678', '2026-03-10', 'Laboratório Silva', 4),

(5, 'L56789', '2024-12-01', 'Química Moderna', 5),

(6, 'L67890', '2025-11-25', 'Indústria Farma', 6),

(7, 'L78901', '2026-07-30', 'LabVida', 7),

```
(8, 'L89012', '2025-05-18', 'Saúde e Vida', 8),  
(9, 'L90123', '2024-09-14', 'Laboratório Santos', 9),  
(10, 'L01234', '2026-01-08', 'BioMedic', 10);
```

```
INSERT INTO Estoque (CodUnid, CodRem, DataCompra, Quantidade)  
VALUES
```

```
(1, 1, '2025-01-10', 50),  
(2, 2, '2025-01-12', 100),  
(3, 3, '2025-01-15', 75),  
(4, 4, '2025-01-18', 30),  
(5, 5, '2025-01-20', 60),  
(6, 6, '2025-01-22', 40),  
(7, 7, '2025-01-25', 90),  
(8, 8, '2025-01-28', 110),  
(9, 9, '2025-01-30', 20),  
(10, 10, '2025-02-01', 35);
```

```
INSERT INTO ConsultCli (CodConsult, CodFunc)  
VALUES
```

```
(1, 11),  
(2, 12),  
(3, 13),  
(4, 14),  
(5, 15),  
(6, 16),  
(7, 17),  
(8, 18),  
(9, 19),  
(10, 20);
```

```
INSERT INTO Historico (CodHist, Descricao, STATUS, CodFunc, Data_ini,  
Data_fim)
```

VALUES

(1, 'Funcionário ativo no setor administrativo.', 'Ativo', 1, '2023-01-01', NULL),
(2, 'Funcionário inativo durante período de férias.', 'Inativo', 2, '2025-01-10', '2025-01-25'),
(3, 'Funcionário inativo por licença médica.', 'Inativo', 3, '2024-12-01', '2025-01-15'),
(4, 'Funcionário inativo após demissão.', 'Inativo', 4, '2020-05-01', '2020-05-01'),
(5, 'Funcionário ativo no setor de trauma.', 'Ativo', 5, '2022-01-01', NULL),
(6, 'Funcionário inativo durante período de férias.', 'Inativo', 6, '2024-12-20', '2025-01-10'),
(7, 'Funcionário inativo por licença maternidade.', 'Inativo', 7, '2024-09-01', '2025-03-01'),
(8, 'Funcionário ativo no setor do berçário.', 'Ativo', 8, '2023-06-01', NULL),
(9, 'Funcionário inativo após demissão.', 'Inativo', 9, '2021-03-15', '2021-03-15'),
(10, 'Funcionário ativo no setor de oncologia.', 'Ativo', 10, '2022-07-01', NULL);

INSERT INTO Historico (CodHist, Descricao, STATUS, CodFunc, Data_ini, Data_fim)

VALUES

(11, 'Funcionário ativo no setor financeiro.', 'Ativo', 11, '2023-01-15', NULL),
(12, 'Funcionário inativo após projeto concluído.', 'Inativo', 12, '2022-06-01', '2023-06-01'),
(13, 'Funcionário ativo na recepção.', 'Ativo', 13, '2024-02-01', NULL),
(14, 'Funcionário inativo após término do contrato.', 'Inativo', 14, '2021-05-01', '2021-05-31'),
(15, 'Funcionário ativo no setor de TI.', 'Ativo', 15, '2023-09-01', NULL),
(16, 'Funcionário inativo por licença médica.', 'Inativo', 16, '2024-03-01', '2024-06-01'),
(17, 'Funcionário ativo no setor jurídico.', 'Ativo', 17, '2023-11-01', NULL),
(18, 'Funcionário inativo após término do contrato.', 'Inativo', 18, '2021-07-01', '2021-07-31'),
(19, 'Funcionário ativo no setor de marketing.', 'Ativo', 19, '2022-10-01', NULL),

```

(20, 'Funcionário ativo no setor de vendas.', 'Ativo', 20, '2023-05-01', NULL),
(21, 'Funcionário ativo no setor de logística.', 'Ativo', 21, '2022-08-01',
NULL),
(22, 'Funcionário ativo no setor administrativo.', 'Ativo', 22, '2023-04-01',
NULL),
(23, 'Funcionário inativo após demissão.', 'Inativo', 23, '2021-03-01',
'2021-03-01'),
(24, 'Funcionário ativo no setor financeiro.', 'Ativo', 24, '2023-02-01', NULL),
(25, 'Funcionário inativo por licença médica.', 'Inativo', 25, '2024-05-01',
'2024-07-01'),
(26, 'Funcionário ativo no setor de RH.', 'Ativo', 26, '2023-07-01', NULL),
(27, 'Funcionário inativo após demissão.', 'Inativo', 27, '2022-09-01',
'2022-09-01'),
(28, 'Funcionário ativo no setor de projetos.', 'Ativo', 28, '2023-08-01',
NULL),
(29, 'Funcionário ativo no setor de suporte técnico.', 'Ativo', 29, '2023-10-01',
NULL),
(30, 'Funcionário inativo após término do contrato.', 'Inativo', 30, '2024-01-01',
'2024-03-01');

```

```

INSERT INTO Consulta (CodFunc, CodUnid, Dia, Hora, CodConsult, CodPac)
VALUES

```

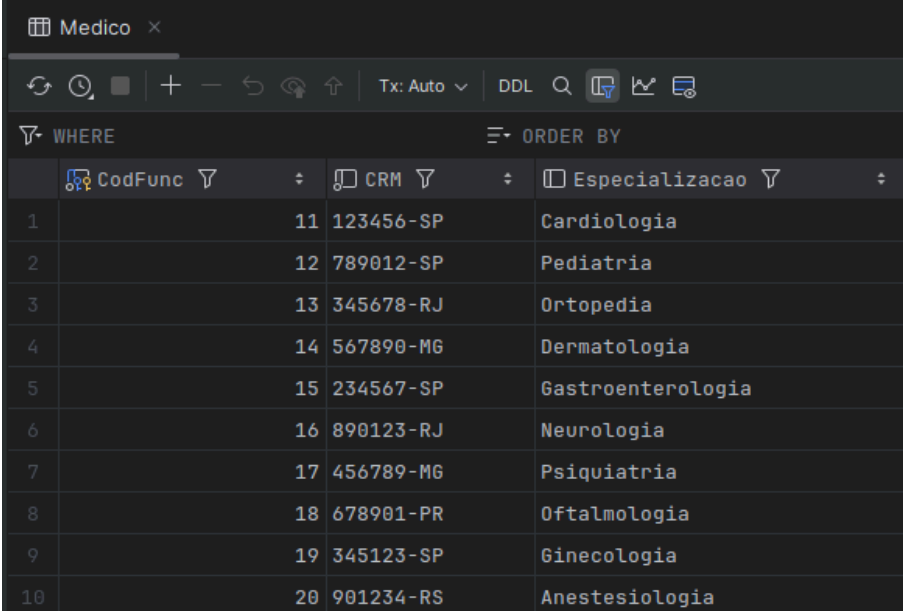
```

(11, 1, '2025-01-15', '09:00:00', 1, 1),
(12, 2, '2025-01-16', '10:30:00', 2, 2),
(13, 3, '2025-01-17', '11:15:00', 3, 3),
(14, 4, '2025-01-18', '08:45:00', 4, 4),
(15, 5, '2025-01-19', '14:00:00', 5, 5),
(16, 6, '2025-01-20', '13:30:00', 6, 6),
(17, 7, '2025-01-21', '15:00:00', 7, 7),
(18, 8, '2025-01-22', '09:30:00', 8, 8),
(19, 9, '2025-01-23', '10:00:00', 9, 9),
(20, 10, '2025-01-24', '11:45:00', 10, 10);

```


a. Listagem

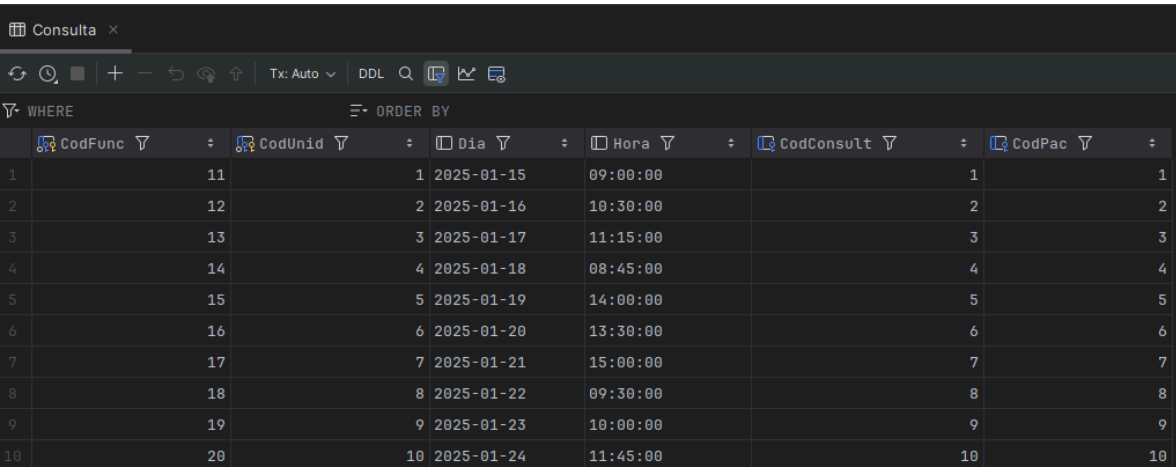
Nessa subetapa, será exibido alguns exemplos dos resultados obtidos com a inserção dos dados nas tabelas



The screenshot shows a database application window titled "Medico". It features a toolbar with icons for refresh, undo, redo, and other functions. Below the toolbar, there are tabs for "WHERE" and "ORDER BY". The main area displays a table with 10 rows and 3 columns: "CodFunc", "CRM", and "Especializacao".

	CodFunc	CRM	Especializacao
1	11	123456-SP	Cardiologia
2	12	789012-SP	Pediatria
3	13	345678-RJ	Ortopedia
4	14	567890-MG	Dermatologia
5	15	234567-SP	Gastroenterologia
6	16	890123-RJ	Neurologia
7	17	456789-MG	Psiquiatria
8	18	678901-PR	Oftalmologia
9	19	345123-SP	Ginecologia
10	20	901234-RS	Anestesiologia

Figura 2: Tabela Medico



The screenshot shows a database application window titled "Consulta". It features a toolbar with icons for refresh, undo, redo, and other functions. Below the toolbar, there are tabs for "WHERE" and "ORDER BY". The main area displays a table with 10 rows and 7 columns: "CodFunc", "CodUnid", "Dia", "Hora", "CodConsult", and "CodPac".

	CodFunc	CodUnid	Dia	Hora	CodConsult	CodPac
1	11	1	2025-01-15	09:00:00	1	1
2	12	2	2025-01-16	10:30:00	2	2
3	13	3	2025-01-17	11:15:00	3	3
4	14	4	2025-01-18	08:45:00	4	4
5	15	5	2025-01-19	14:00:00	5	5
6	16	6	2025-01-20	13:30:00	6	6
7	17	7	2025-01-21	15:00:00	7	7
8	18	8	2025-01-22	09:30:00	8	8
9	19	9	2025-01-23	10:00:00	9	9
10	20	10	2025-01-24	11:45:00	10	10

Figura 3: Tabela Consulta

RemedioUnidade

Tx: Auto DDL

WHERE ORDER BY

	CodUnid	Lote	DataVenc	Laboratorio	CodRem
1	1	L12345	2025-12-31	Laboratório ABC	1
2	2	L23456	2024-06-15	Laboratório XYZ	2
3	3	L34567	2025-08-20	Farmacêutica Nacional	3
4	4	L45678	2026-03-10	Laboratório Silva	4
5	5	L56789	2024-12-01	Química Moderna	5
6	6	L67890	2025-11-25	Indústria Farma	6
7	7	L78901	2026-07-30	LabVida	7
8	8	L89012	2025-05-18	Saúde e Vida	8
9	9	L90123	2024-09-14	Laboratório Santos	9
10	10	L01234	2026-01-08	BioMedic	10

Figura 4: Tabela RemedioUnidade

5. CONSULTAS

a. Consultar com operadores básicos de SQL

Listar o nome dos médicos, sua especialidade e dos pacientes que eles atenderam

```
SELECT      f.Nome AS "Nome do Médico",
            m.Especializacao AS "Especialidade",
            p.NomePac AS "Nome do Paciente"

FROM Consulta c
JOIN Funcionario f ON f.CodFunc = c.CodFunc
JOIN Paciente p ON p.CodPac = c.CodPac
JOIN Medico m ON f.CodFunc = m.CodFunc
ORDER BY f.Nome;
```



The image shows a terminal window with a SQL query result. The query is ordered by the doctor's name. The result is a table with three columns: 'Nome do Médico', 'Especialidade', and 'Nome do Paciente'. There are 12 rows of data.

Nome do Médico	Especialidade	Nome do Paciente
Aline Castro	Dermatologia	Daniel Ferreira
Ana Paula Santos	Oftalmologia	Heloísa Martins
Camila Teixeira	Anestesiologia	João Alves
Carlos Mendes	Pediatria	Bruno Costa
Fábio Ribeiro	Ginecologia	Isabela Fonseca
Fernanda Almeida	Cardiologia	Amanda Silva
Gustavo Lima	Gastroenterologia	Elaine Rocha
Lucas Silva	Psiquiatria	Gustavo Lima
Patrícia Fonseca	Neurologia	Fábio Ribeiro
Roberto Farias	Ortopedia	Carla Mendes

Figura 5: Listagem dos médicos

Listar o nome do médico, o nome do paciente, a data da consulta e o remédio receitado

```
SELECT      f.Nome AS "Nome do Médico",
            p.NomePac AS "Nome do Paciente",
            DATE_FORMAT(c.Dia, '%d/%m/%Y') AS "Data da Consulta",
            r.NomeRem AS "Remédio Receitado"

FROM Consulta c
JOIN Funcionario f ON f.CodFunc = c.CodFunc
JOIN Paciente p ON p.CodPac = c.CodPac
```

JOIN RemedioUnidade ru ON ru.CodRem = c.CodUnid

JOIN Remedios r ON r.CodRem = ru.CodRem

ORDER BY f.Nome;

Nome do Médico	Nome do Paciente	Data da Consulta	Remédio Receitado
Aline Castro	Daniel Ferreira	18/01/2025	Diazepam
Ana Paula Santos	Heloísa Martins	22/01/2025	Cetirizina
Camila Teixeira	João Alves	24/01/2025	Prednisona
Carlos Mendes	Bruno Costa	16/01/2025	Ibuprofeno
Fábio Ribeiro	Isabela Fonseca	23/01/2025	Rivotril
Fernanda Almeida	Amanda Silva	15/01/2025	Paracetamol
Gustavo Lima	Elaine Rocha	19/01/2025	Omeprazol
Lucas Silva	Gustavo Lima	21/01/2025	Losartana
Patrícia Fonseca	Fábio Ribeiro	20/01/2025	Metformina
Roberto Farias	Carla Mendes	17/01/2025	Amoxicilina

Figura 6: Listagem dos médicos e consulta

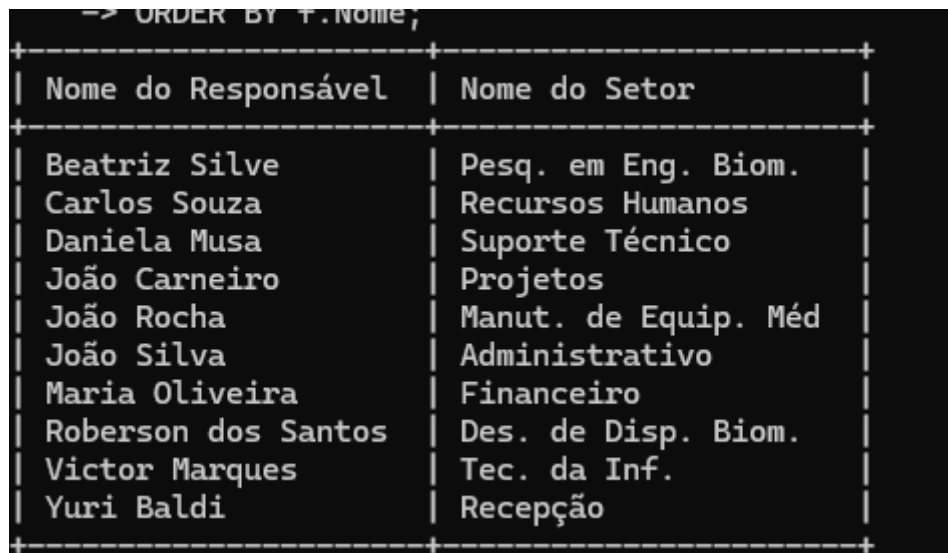
#Listar os nomes dos pacientes que receberam prescrições de medicamentos controlados, indicando o medicamento prescrito e a data prevista de vencimento da receita.

```
SELECT      p.NomePac AS "Nome Paciente",
            r.NomeRem AS "Remédio Controlado",
            DATE_FORMAT DATE_ADD(c.Dia, INTERVAL 30 DAY), '%d/%m/%Y')
AS "Data de vencimento previsto"
FROM Consulta c
JOIN RemedioUnidade ru ON ru.CodUnid = c.CodUnid
JOIN Remedios r ON r.CodRem = ru.CodRem
JOIN Paciente p ON p.CodPac = c.CodPac
WHERE r.Controlado = 1
ORDER BY c.Dia ;
```

Nome Paciente	Remédio Controlado	Data de vencimento previsto
Carla Mendes	Amoxicilina	16/02/2025
Daniel Ferreira	Diazepam	17/02/2025
Isabela Fonseca	Rivotril	22/02/2025
João Alves	Prednisona	23/02/2025

Figura 7: Prescrições dos pacientes

```
# Liste o nome dos funcionarios responsáveis por cada setor e nome do Setor
SELECT      f.Nome AS "Nome do Responsável",
            s.NomeSet AS "Nome do Setor"
FROM Setor s
JOIN Funcionario f ON f.CodFunc = s.CodFunc
ORDER BY f.Nome;
```



Nome do Responsável	Nome do Setor
Beatriz Silve	Pesq. em Eng. Biom.
Carlos Souza	Recursos Humanos
Daniela Musa	Suporte Técnico
João Carneiro	Projetos
João Rocha	Manut. de Equip. Méd
João Silva	Administrativo
Maria Oliveira	Financeiro
Roberson dos Santos	Des. de Disp. Biom.
Victor Marques	Tec. da Inf.
Yuri Baldi	Recepção

Figura 8: Responsáveis pelos setores

```
#Listar todos os funcionarios e suas respectivas datas de entrada no sistema
SELECT      f.Nome AS "Nome do Funcionário",
            DATE_FORMAT(h.Data_ini, "%d/%m/%Y") AS "Data de Entrada",
            h.STATUS AS "Status"
FROM Funcionario f
JOIN Historico h ON f.CodFunc = h.CodFunc
ORDER BY f.Nome
```

Nome do Funcionário	Data de Entrada	Status
Aline Castro	01/05/2021	Inativo
Ana Paula Santos	01/07/2021	Inativo
Beatriz Silve	01/07/2022	Ativo
Bruno Costa	01/10/2023	Ativo
Camila Teixeira	01/05/2023	Ativo
Carlos Mendes	01/06/2022	Inativo
Carlos Souza	01/12/2024	Inativo
Cláudia Rocha	01/02/2023	Ativo
Daniela Musa	01/09/2024	Inativo
Eduardo Matos	01/05/2024	Inativo
Fábio Ribeiro	01/10/2022	Ativo
Fernanda Almeida	15/01/2023	Ativo
Gustavo Lima	01/09/2023	Ativo
João Carneiro	01/06/2023	Ativo
João Rocha	15/03/2021	Inativo
João Silva	01/01/2023	Ativo
Juliana Alves	01/04/2023	Ativo
Letícia Ramos	01/08/2023	Ativo
Lucas Silva	01/11/2023	Ativo
Marcos Vinícius	01/03/2021	Inativo
Maria Oliveira	10/01/2025	Inativo
Patrícia Fonseca	01/03/2024	Inativo
Rafael Moreira	01/08/2022	Ativo
Renata Oliveira	01/07/2023	Ativo
Roberson dos Santos	01/01/2022	Ativo
Roberto Farias	01/02/2024	Ativo
Thiago Ferreira	01/09/2022	Inativo
Victor Marques	20/12/2024	Inativo
Viviane Santos	01/01/2024	Inativo
Yuri Baldi	01/05/2020	Inativo

Figura 9: Entradada dos funcionários

b. Consultas com LEFT JOIN

#Listar todos os nomes dos funcionarios ativos, função e quais setores eles trabalham

```
SELECT      f.Nome AS "Nome do Funcionário",
            CASE
                WHEN m.CodFunc IS NOT NULL THEN "Médico"
                WHEN e.CodFunc IS NOT NULL THEN 'Enfermeiro'
                ELSE "Administrativo"
            END AS "Função",
            CASE
                WHEN m.Especializacao IS NOT NULL THEN
m.Especializacao
                WHEN s.NomeSet IS NOT NULL THEN s.NomeSet
                ELSE "Enfermeiro Geral"
            END AS "Setor/Especialidade"
FROM Funcionario f
JOIN Historico h ON h.CodFunc = f.CodFunc
LEFT JOIN Setor s ON f.CodFunc = s.CodFunc
LEFT JOIN Medico m ON f.CodFunc = m.CodFunc
LEFT JOIN Enfermeiro e ON f.CodFunc = e.CodFunc
WHERE h.STATUS = "Ativo";
```

Nome do Funcionário	Função	Setor/Especialidade
João Silva	Administrativo	Administrativo
Roberson dos Santos	Administrativo	Des. de Disp. Biom.
João Carneiro	Administrativo	Projetos
Beatriz Silve	Administrativo	Pesq. em Eng. Biom.
Fernanda Almeida	Médico	Cardiologia
Roberto Farias	Médico	Ortopedia
Gustavo Lima	Médico	Gastroenterologia
Lucas Silva	Médico	Psiquiatria
Fábio Ribeiro	Médico	Ginecologia
Camila Teixeira	Médico	Anestesiologia
Rafael Moreira	Enfermeiro	Enfermeiro Geral
Juliana Alves	Enfermeiro	Enfermeiro Geral
Cláudia Rocha	Enfermeiro	Enfermeiro Geral
Renata Oliveira	Enfermeiro	Enfermeiro Geral
Letícia Ramos	Enfermeiro	Enfermeiro Geral
Bruno Costa	Enfermeiro	Enfermeiro Geral

Figura 10: Listagem dos funcionários e suas funções

#Listar o nome do Remedio, tipo, se é controlado ou não e quantos há no estoque

SELECT

 r.NomeRem AS "Nome do Remédio",

 r.Tipo AS "Tipo",

 CASE

 WHEN r.controlado = 1 THEN "Sim"

 ELSE "Não"

 END AS "Controlado",

 CASE

 WHEN SUM(e.Quantidade) IS NOT NULL THEN SUM(e.Quantidade)

 ELSE "Sem estoque"

 END AS "Quantidade em Estoque"

FROM Remedios r

LEFT JOIN RemedioUnidade ru ON r.CodRem = ru.CodRem

LEFT JOIN Estoque e ON ru.CodUnid = e.CodUnid

GROUP BY r.NomeRem, r.Tipo, r.Controlado

ORDER BY r.NomeRem;

Nome do Remédio	Tipo	Controlado	Quantidade em Estoque
Amoxicilina	Antibiótico	Sim	75
Aspirina	Analgésico	Não	Sem estoque
Carbamazepina	Anticonvulsivante	Sim	Sem estoque
Cetirizina	Antialérgico	Não	110
Cloroquina	Antimalárico	Sim	Sem estoque
Diazepam	Ansiolítico	Sim	30
Fluoxetina	Antidepressivo	Sim	Sem estoque
Ibuprofeno	Anti-inflamatório	Não	100
Loratadina	Antialérgico	Não	Sem estoque
Losartana	Antihipertensivo	Não	90
Metformina	Antidiabético	Não	40
Omeprazol	Antiácido	Não	60
Paracetamol	Analgésico	Não	50
Prednisona	Corticosteroide	Sim	35
Rivotril	Ansiolítico	Sim	20

Figura 11: Listagem dos remédios

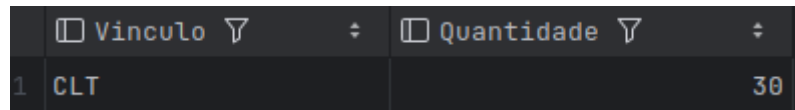
c. Consultas com os operandos matemáticos

Conte a quantidade de funcionários por tipo de vínculo na tabela Funcionario

```
SELECT F.Vinculo, COUNT (*) AS Quantidade
```

```
FROM Funcionarios F;
```

```
GROUP BY F.Vinculo;
```



	Vinculo	Quantidade
1	CLT	30

Figura 12: Listagem da quantidade de vínculos

Calcule a média de médicos por especialização na tabela Medico.

```
SELECT AVG(Quantidade) AS MediaEspecializacao
```

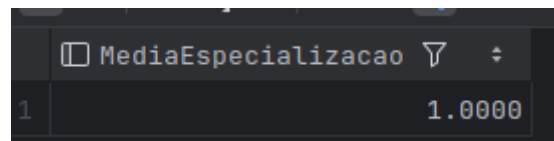
```
FROM (
```

```
    SELECT COUNT(*) AS Quantidade
```

```
    FROM Medico M
```

```
    GROUP BY M.Especializacao
```

```
) AS Subconsulta;
```



	MediaEspecializacao
1	1.0000

Figura 13: Listagem da média de especializações

Agrupe os pacientes por faixa etária e retorne a quantidade de pacientes em cada faixa, ordenando do maior para o menor.

```
SELECT
```

```
    CASE
```

```
        WHEN YEAR(CURDATE()) - YEAR(p.DataNasc) BETWEEN 0 AND 18 THEN  
'0-18 anos'
```

```
        WHEN YEAR(CURDATE()) - YEAR(p.DataNasc) BETWEEN 19 AND 35 THEN  
'19-35 anos'
```

```
        WHEN YEAR(CURDATE()) - YEAR(p.DataNasc) BETWEEN 36 AND 50 THEN  
'36-50 anos'
```

```
        WHEN YEAR(CURDATE()) - YEAR(p.DataNasc) BETWEEN 51 AND 65 THEN  
'51-65 anos'
```

```

ELSE 'Acima de 65 anos'
END AS FaixaEtaria,
COUNT(*) AS QuantidadePacientes
FROM Paciente p
GROUP BY FaixaEtaria
ORDER BY QuantidadePacientes DESC;

```

	FaixaEtaria	QuantidadePacientes
1	19-35 anos	9
2	36-50 anos	6

Figura 14: Listagem da média de idades

```

#Lista nome, CPF e data de nascimento do paciente e ordem por idade
SELECT Nome, CPF, DataNascimento
FROM Pacientes
WHERE DataNascimento = (SELECT MAX(DataNascimento) FROM Pacientes);

```

	NomePac	CPF	DataNasc
4	Daniel Ferreira	456.789.012-34	1988-03-15
5	Elaine Rocha	567.890.123-45	1995-07-07
6	Fábio Ribeiro	678.901.234-56	1991-12-03
7	Gustavo Lima	789.012.345-67	1987-09-10
8	Heloísa Martins	890.123.456-78	1994-01-18
9	Isabela Fonseca	901.234.567-89	1992-06-25
10	João Alves	012.345.678-90	1989-04-09
11	Karina Oliveira	123.654.987-01	1996-02-28
12	Leonardo Santos	234.765.098-12	1984-10-14
13	Mariana Costa	345.876.109-23	1997-08-21
14	Nathália Ribeiro	456.987.210-34	1986-11-30
15	Otávio Souza	567.098.321-45	1993-07-04

Figura 15: Listagem da média de idades

Liste o nome do setor com mais funcionários e o com menos e a quantidade de cada um.

```

SELECT
    S.NomeSet,
    COUNT(*) AS Quantidade
FROM
    Setor S
GROUP BY
    S.NomeSet
HAVING
    COUNT(*) = (SELECT MAX(Quantidade) FROM (SELECT COUNT(*) AS
Quantidade FROM Setor GROUP BY NomeSet) AS Subquery)
    OR COUNT(*) = (SELECT MIN(Quantidade) FROM (SELECT COUNT(*) AS
Quantidade FROM Setor GROUP BY NomeSet) AS Subquery);

```

	NomeSet	Quantidade
1	Administrativo	1
2	Des. de Disp. Biom.	1
3	Financeiro	1
4	Manut. de Equip. Méd	1
5	Pesq. em Eng. Biom.	1
6	Projetos	1
7	Recepção	1
8	Recursos Humanos	1
9	Suporte Técnico	1
10	Tec. da Inf.	1

Figura 16: Listagem dos setores com a maior e menor quantidade de funcionários

6. INTERFACE GRÁFICA

A interface gráfica foi desenvolvida utilizando Django e Python, tendo sido criada uma página que exibe as informações da tabela de consultas.

Consultas					
Código do Funcinário	Código Unidade	Dia	Hora	Código do Consultorio	Código do Paciente
11	1	Jan. 15, 2025	9 a.m.	1	1
12	2	Jan. 16, 2025	10:30 a.m.	2	2
13	3	Jan. 17, 2025	11:15 a.m.	3	3
14	4	Jan. 18, 2025	8:45 a.m.	4	4
15	5	Jan. 19, 2025	2 p.m.	5	5
16	6	Jan. 20, 2025	1:30 p.m.	6	6
17	7	Jan. 21, 2025	3 p.m.	7	7
18	8	Jan. 22, 2025	9:30 a.m.	8	8
19	9	Jan. 23, 2025	10 a.m.	9	9
20	10	Jan. 24, 2025	11:45 a.m.	10	10

Figura 17: Interface exibindo a tabela de consultas

7. BIBLIOGRFIA

1. ELMASRI, R.; NAVATHE, S. B. **Sistemas de Banco de Dados**. 7ª ed. Pearson, 2016.
2. SILBERSCHATZ, A.; KORTH, H.; SUDARSHAN, S. **Sistemas de Banco de Dados**. 6ª ed. Pearson, 2019.
3. DATE, C. J. **Introdução a Sistemas de Bancos de Dados**. 8ª ed. Pearson, 2004.
4. MySQL Documentation. Disponível em: <https://dev.mysql.com/doc/>. Acesso em: 20 fev. 2025.
5. MariaDB Documentation. Disponível em: <https://mariadb.com/kb/en/>. Acesso em: 20 fev. 2025.

6. Django Documentation. Disponível em: <https://docs.djangoproject.com/>. Acesso em: 20 fev. 2025.
7. W3Schools. **SQL Tutorial**. Disponível em: <https://www.w3schools.com/sql/>. Acesso em: 20 fev. 2025.