



## **TRABALHO FINAL COM BASE EM METODOLOGIAS ATIVAS DE APRENDIZAGEM**

### **BANCO DE DADOS II**

Engenharia de Software

Alunos:

Juliano da S Silva - @ojulianos

Alexandre Hermes - @Alexandre200319

Vinicius Raccolto - @vraccolto


Jefferson Pacheco - @jzpacheco





Anderson Pizzolo - @Anderson-Pizzolo




Gabriel Angelo Kaufmann - @GabrielAKaufmann

# GIT HUB


[https://github.com/ojulianos/banco\\_2\\_seguros\\_imoveis](https://github.com/ojulianos/banco_2_seguros_imoveis)

 **banco\_2\_seguros\_imoveis** Public



 Pin  Unwatch **2**  Fork **0**  Star **2**

 main  2 branches  0 tags

[Go to file](#) [Add file](#) [Code](#)

 **ojulianos** ajuste clientes 8651f83 9 hours ago 🕒 69 commits






📁 back-end	adicionadas instruções no readme do back-end	11 hours ago
📁 ddl	Ajustes consultas SQL	2 days ago
📁 dicionario-dados	fix: Inserting description for imoveis table	last week
📁 dml	Ajustes consultas SQL	2 days ago
📁 front-end	ajuste clientes	9 hours ago
📄 Diagrama ER.png	Ajustes consultas SQL	2 days ago
📄 README.md	atualizado readme, corrigidos links e referencias	2 days ago
📄 db_diagram.md	atualizado readme, corrigidos links e referencias	2 days ago
📄 package-lock.json	cadastro de Clientes e Imoveis	2 days ago
📄 perguntas-negocio.md	Update perguntas-negocio.md	2 days ago

 **README.md** 

## Projeto Sistema de Seguros de Imóveis

### About

No description, website, or topics provided.

-  Readme
-  Activity
-  2 stars
-  2 watching
-  0 forks

### Releases

No releases published

[Create a new release](#)







### Packages

No packages published

[Publish your first package](#)

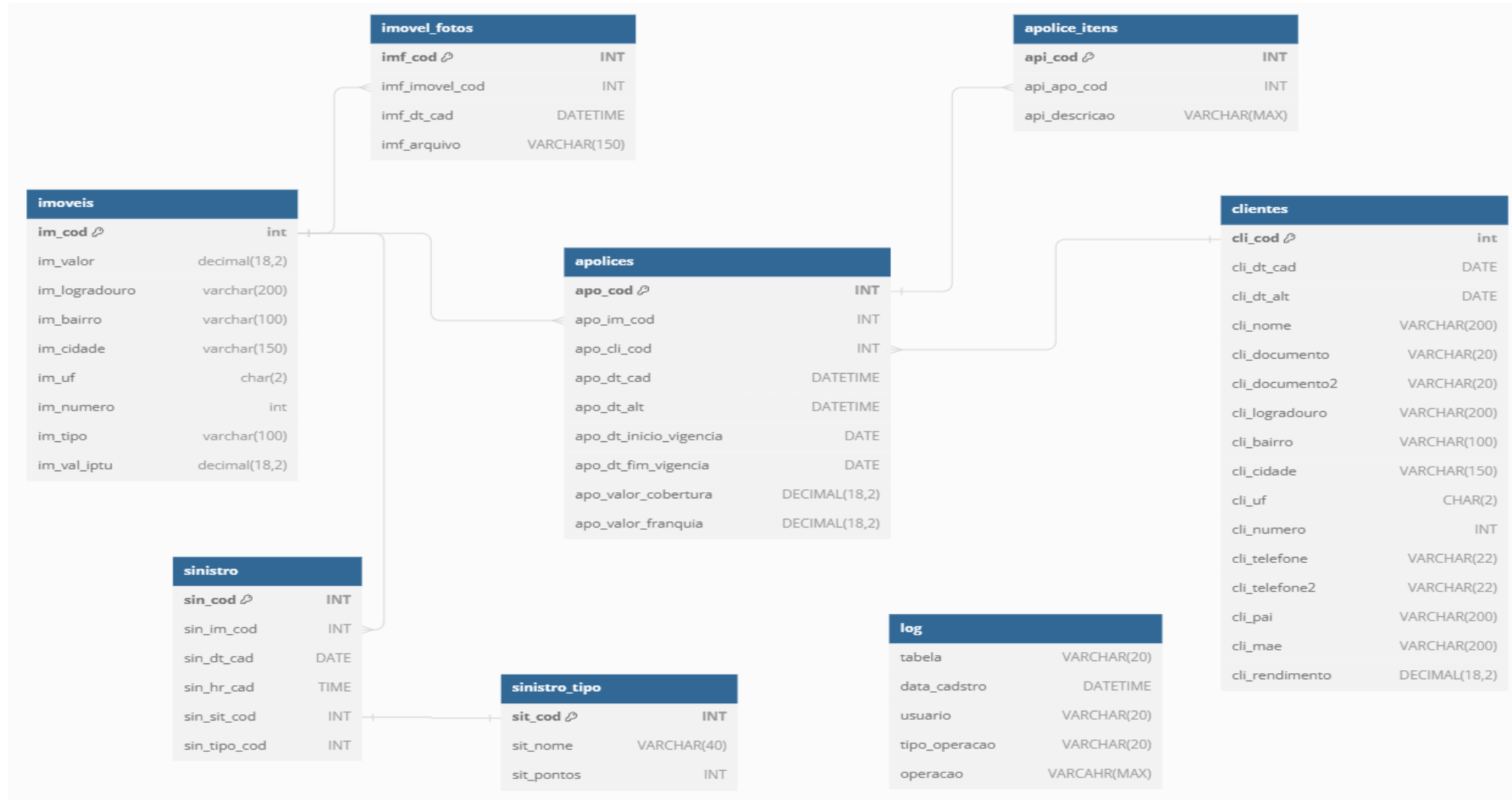
### Contributors

**6**

# MODELO ER FÍSICO

[https://github.com/ojulianos/banco\\_2\\_seguros\\_imoveis/blob/main/db\\_diagram.md](https://github.com/ojulianos/banco_2_seguros_imoveis/blob/main/db_diagram.md)



# DICIONÁRIO DE DADOS

[https://github.com/ojulianos/banco\\_2\\_seguros\\_imoveis/blob/main/dicionario-dados/dicionario-dados.ods](https://github.com/ojulianos/banco_2_seguros_imoveis/blob/main/dicionario-dados/dicionario-dados.ods)

Tabela	apolices					
Descrição	Tabela responsável pelo armazenamento das apólices					
Atributos						
Nome da Coluna	Tipo do Dado	Valor min e max	Nulidade	PK	FK	Descrição
apo_cod	INT		NOT NULL	X		Código da apólice
apo_im_cod	INT		NOT NULL		X	Código do imóvel participante
apo_cli_cod	INT		NOT NULL		X	Código do cliente participante
apo_dt_cad	DATE		NOT NULL			Data de cadastro da apólice
apo_dt_alt	DATE					Data de alteração da apólice
apo_dt_inicio_vigencia	DATE					Data de início de vigência
apo_dt_fim_vigencia	DATE					Data final da vigência
apo_valor_cobertura	FLOAT					Valor da coberta da apólice
apo_valor_franquia	FLOAT					Valor da franquia
Índice						
Nome do índice	Clustered	NonClustered	Unique	Colunas		
[PK_apolices__13764B3B5913410E]	X		X	apo_cod		
idx_apolice_imovel_cliente		X		apo_cod, apo_im_cod, apo_cli_cod		

# SCRIPTS DDL

[https://github.com/ojulianos/banco\\_2\\_seguros\\_imóveis/blob/main/ddl/01-ScriptDDL.sql](https://github.com/ojulianos/banco_2_seguros_imóveis/blob/main/ddl/01-ScriptDDL.sql)

```
-- REMOVE INDICES CASO EXISTAM
ALTER TABLE [apolices] DROP CONSTRAINT IF EXISTS fk_apolices__cliente_cod
GO
ALTER TABLE [apolices] DROP CONSTRAINT IF EXISTS fk_apolices__imovel_cod
GO
ALTER TABLE [apolice_itens] DROP CONSTRAINT IF EXISTS fk_apolice_itens__apolice_cod
GO
ALTER TABLE [imovel_fotos] DROP CONSTRAINT IF EXISTS fk_imovel_fotos__imovel
GO
ALTER TABLE [sinistro] DROP CONSTRAINT IF EXISTS fk_sinistro__imovel_cod
GO
ALTER TABLE [sinistro] DROP CONSTRAINT IF EXISTS fk_sinistro__sinistro_tipo_cod
GO
```

```
-- TABELA DE IMOVEIS
DROP TABLE IF EXISTS [imoveis]
GO
CREATE TABLE [imoveis] (
    [im_cod] int IDENTITY NOT NULL PRIMARY KEY,
    [im_valor] decimal(18,2),
    [im_logradouro] varchar(200),
    [im_bairro] varchar(100),
    [im_cidade] varchar(150),
    [im_uf] char(2),
    [im_numero] int,
    [im_tipo] varchar(100),
    [im_val_ipitu] decimal(18,2)
)
GO

-- TABELA DE FOTOS DO IMOVEL
DROP TABLE IF EXISTS [imovel_fotos]
GO
CREATE TABLE [imovel_fotos] (
    [imf_cod] INT IDENTITY NOT NULL PRIMARY KEY,
    [imf_imovel_cod] INT,
    [imf_dt_cad] DATETIME DEFAULT GETDATE(),
    [imf_arquivo] VARCHAR(150)
)
GO
```

# SCRIPTS DDL

[https://github.com/ojulianos/banco\\_2\\_seguros\\_imóveis/blob/main/ddl/01-ScriptDDL.sql](https://github.com/ojulianos/banco_2_seguros_imóveis/blob/main/ddl/01-ScriptDDL.sql)

```
-- TABELA DE CLIENTES
DROP TABLE IF EXISTS [clientes]
GO
CREATE TABLE [clientes] (
    [cli_cod] int IDENTITY NOT NULL PRIMARY KEY,
    [cli_dt_cad] DATE DEFAULT GETDATE(),
    [cli_dt_alt] DATE,
    [cli_nome] VARCHAR(200),
    [cli_documento] VARCHAR(20),
    [cli_documento2] VARCHAR(20),
    [cli_logradouro] VARCHAR(200),
    [cli_bairro] VARCHAR(100),
    [cli_cidade] VARCHAR(150),
    [cli_uf] CHAR(2),
    [cli_numero] INT,
    [cli_telefone] VARCHAR(22),
    [cli_telefone2] VARCHAR(22),
    [cli_pai] VARCHAR(200),
    [cli_mae] VARCHAR(200),
    [cli_rendimento] DECIMAL(18,2)
)
GO
```

```
-- TABELA DE APOLICES
DROP TABLE IF EXISTS [apolices]
GO
CREATE TABLE [apolices] (
    [apo_cod] INT IDENTITY NOT NULL PRIMARY KEY,
    [apo_im_cod] INT,
    [apo_cli_cod] INT,
    [apo_dt_cad] DATETIME,
    [apo_dt_alt] DATETIME,
    [apo_dt_inicio_vigencia] DATE,
    [apo_dt_fim_vigencia] DATE,
    [apo_valor_cobertura] DECIMAL(18,2),
    [apo_valor_franquia] DECIMAL(18,2)
)
GO
```

```
-- TABELA DE ITENS/CONTRATOS DA APOLICE
DROP TABLE IF EXISTS [apolice_itens]
GO
CREATE TABLE [apolice_itens] (
    [api_cod] INT IDENTITY NOT NULL PRIMARY KEY,
    [api_apo_cod] INT,
    [api_descricao] VARCHAR(MAX)
)
GO

-- TABELA DE SINISTROS
DROP TABLE IF EXISTS [sinistro]
GO
CREATE TABLE [sinistro] (
    [sin_cod] INT IDENTITY NOT NULL PRIMARY KEY,
    [sin_im_cod] INT,
    [sin_dt_cad] DATE,
    [sin_hr_cad] TIME,
    [sin_sit_cod] INT,
    [sin_tipo_cod] INT
)
GO
```

# SCRIPTS DDL

[https://github.com/ojulianos/banco\\_2\\_seguros\\_imóveis/blob/main/ddl/01-ScriptDDL.sql](https://github.com/ojulianos/banco_2_seguros_imóveis/blob/main/ddl/01-ScriptDDL.sql)

```
-- TABELA DE TIPOS DE SINISTRO
DROP TABLE IF EXISTS [sinistro_tipo]
GO
CREATE TABLE [sinistro_tipo] (
    [sit_cod] INT IDENTITY NOT NULL PRIMARY KEY,
    [sit_nome] VARCHAR(40),
    [sit_pontos] INT
)
GO

-- TABELA DE LOGS
DROP TABLE IF EXISTS [log]
GO
CREATE TABLE [log] (
    [tabela] VARCHAR(20),
    [data_cadastro] DATETIME,
    [usuario] VARCHAR(20),
    [tipo_operacao] VARCHAR(20),
    [operacao] VARCHAR(MAX)
)
GO
```

```
---- CHAVES ESTRANGEIRAS ----
-- TABELA DE APOLICES
ALTER TABLE [apolices]
ADD CONSTRAINT fk_apolices__cliente_cod FOREIGN KEY ([apo_cli_cod])
REFERENCES [clientes] ([cli_cod])
GO

ALTER TABLE [apolices]
ADD CONSTRAINT fk_apolices__imovel_cod FOREIGN KEY ([apo_im_cod])
REFERENCES [imoveis] ([im_cod])
GO

-- TABELA DE ITENS/CONTRATOS APOLICE
ALTER TABLE [apolice_itens]
ADD CONSTRAINT fk_apolice_itens__apolice_cod FOREIGN KEY ([api_apo_cod])
REFERENCES [apolices] ([apo_cod])
GO

-- TABELA DE FOTOS DO IMOVEL
ALTER TABLE [imovel_fotos]
ADD CONSTRAINT fk_imovel_fotos__imovel FOREIGN KEY ([imf_imovel_cod])
REFERENCES [imoveis] ([im_cod])
GO
```

# SCRIPTS DDL

[https://github.com/ojulianos/banco\\_2\\_seguros\\_imóveis/blob/main/ddl/01-ScriptDDL.sql](https://github.com/ojulianos/banco_2_seguros_imóveis/blob/main/ddl/01-ScriptDDL.sql)

```
-- TABELA DE SINISTROS
ALTER TABLE [sinistro]
ADD CONSTRAINT fk_sinistro_imovel_cod FOREIGN KEY ([sin_im_cod])
REFERENCES [imoveis] ([im_cod])
GO

ALTER TABLE [sinistro]
ADD CONSTRAINT fk_sinistro_sinistro_tipo_cod FOREIGN KEY ([sin_tipo_cod])
REFERENCES [sinistro_tipo] ([sit_cod])
GO
```

```
-- INDICES
DROP INDEX IF EXISTS idx_apolice_imovel_cliente ON apolices
GO

CREATE INDEX idx_apolice_imovel_cliente ON apolices (apo_cod, apo_im_cod, apo_cli_cod)
GO

DROP INDEX IF EXISTS idx_cliente_documento ON clientes
GO

CREATE INDEX idx_cliente_documento ON clientes (cli_documento)
GO

DROP INDEX IF EXISTS idx_sinistro_imovel ON sinistro
GO

CREATE INDEX idx_sinistro_imovel ON sinistro (sin_cod, sin_im_cod)
GO
```



# Scripts DDL

[https://github.com/ojulianos/banco\\_2\\_seguros\\_imóveis/blob/main/dml/ScriptDML.sql](https://github.com/ojulianos/banco_2_seguros_imóveis/blob/main/dml/ScriptDML.sql)

```
-- Em caso de problemas para adicionar os dados deve ser executado o script abaixo para cada tabela:
-- SET IDENTITY_INSERT {tabela} ON GO

INSERT INTO clientes ([cli_dt_cad], [cli_dt_alt], [cli_nome], [cli_documento], [cli_documento2], [cli_logradouro], [cli_bairro], [cli_cidade], [cli_uf], [cli_numero], [cli_telefone], [cli_telefone2], [cli_pai], [cli_mae])
VALUES
    ('2023-01-01', '2023-01-05', 'Maria Silva', '12345678901', '98765432101', 'Rua das Flores', 'Boa Viagem', 'Recife', 'PE', 10, '999999999', '888888888', 'Pai 1', 'Mãe 1', 5000),
    ('2023-01-02', '2023-01-06', 'João Santos', '98765432101', '12345678901', 'Avenida dos Girassóis', 'Ponta Negra', 'Natal', 'RN', 15, '777777777', '666666666', 'Pai 2', 'Mãe 2', 6000),
    ('2023-01-03', '2023-01-07', 'Ana Souza', '45678901201', '21098765401', 'Rua das Acácias', 'Ipanema', 'Rio de Janeiro', 'RJ', 20, '555555555', '444444444', 'Pai 3', 'Mãe 3', 7000),
    ('2023-01-04', '2023-01-08', 'Pedro Oliveira', '78901234501', '32109876501', 'Avenida dos Lírios', 'Boa Vista', 'Porto Alegre', 'RS', 25, '333333333', '222222222', 'Pai 4', 'Mãe 4', 8000),
    ('2023-01-05', '2023-01-09', 'Carla Lima', '23456789001', '54321098701', 'Rua das Rosas', 'Alphaville', 'Barueri', 'SP', 30, '111111111', '000000000', 'Pai 5', 'Mãe 5', 9000),
    ('2023-01-06', '2023-01-10', 'Marcos Vieira', '56789012301', '76543210901', 'Avenida dos Cravos', 'Boa Esperança', 'Marabá', 'PA', 35, '222222222', '333333333', 'Pai 6', 'Mãe 6', 10000),
    ('2023-01-07', '2023-01-11', 'Lúcia Ferreira', '89012345601', '10987654301', 'Rua das Orquídeas', 'Praia Grande', 'São Paulo', 'SP', 40, '888888888', '999999999', 'Pai 7', 'Mãe 7', 11000),
    ('2023-01-08', '2023-01-12', 'Rafaela Gomes', '01234567801', '21098765432', 'Avenida das Hortênsias', 'Copacabana', 'Rio de Janeiro', 'RJ', 45, '444444444', '555555555', 'Pai 8', 'Mãe 8', 12000),
    ('2023-01-09', '2023-01-13', 'Fernando Costa', '12345678910', '43210987654', 'Rua dos Jasmins', 'Itaim Bibi', 'São Paulo', 'SP', 50, '999999999', '888888888', 'Pai 9', 'Mãe 9', 13000),
    ('2023-01-10', '2023-01-14', 'Juliana Pereira', '23456789101', '98765432109', 'Avenida das Tulipas', 'Setor Bueno', 'Goiânia', 'GO', 55, '777777777', '666666666', 'Pai 10', 'Mãe 10', 14000),
    ('2023-01-11', '2023-01-15', 'Anderson Rodrigues', '34567891012', '87654321098', 'Rua das Violetas', 'Bessa', 'João Pessoa', 'PB', 60, '666666666', '777777777', 'Pai 11', 'Mãe 11', 15000),
    ('2023-01-15', '2023-01-19', 'Roberto Santos', '78910111234', '87654321098', 'Avenida das Violetas', 'Boa Vista', 'Recife', 'PE', 22, '333333333', '444444444', 'Pai 12', 'Mãe 12', 19000),
    ('2023-01-16', '2023-01-20', 'Carolina Lima', '89101112134', '76543210987', 'Rua dos Cravos', 'Aldeota', 'Fortaleza', 'CE', 45, '555555555', '666666666', 'Pai 13', 'Mãe 13', 20000),
    ('2023-01-17', '2023-01-21', 'Lucas Oliveira', '91011121345', '65432109876', 'Avenida dos Jasmins', 'Boa Esperança', 'Manaus', 'AM', 33, '777777777', '888888888', 'Pai 14', 'Mãe 14', 21000),
    ('2023-01-18', '2023-01-22', 'Marina Costa', '10111213145', '54321098765', 'Rua das Tulipas', 'Centro', 'Goiânia', 'GO', 55, '999999999', '000000000', 'Pai 15', 'Mãe 15', 22000),
    ('2023-01-19', '2023-01-23', 'Henrique Silva', '11121314156', '43210987654', 'Avenida dos Crisântemos', 'São Mateus', 'São Paulo', 'SP', 88, '111111111', '222222222', 'Pai 16', 'Mãe 16', 23000),
    ('2023-01-20', '2023-01-24', 'Tatiane Souza', '12131415167', '32109876543', 'Rua das Orquídeas', 'Boa Viagem', 'Recife', 'PE', 77, '333333333', '444444444', 'Pai 17', 'Mãe 17', 24000),
    ('2023-01-21', '2023-01-25', 'Ricardo Alves', '13141516178', '21098765432', 'Avenida dos Lírios', 'Alphaville', 'Barueri', 'SP', 66, '555555555', '666666666', 'Pai 18', 'Mãe 18', 25000),
    ('2023-01-22', '2023-01-26', 'Fernanda Pereira', '14151617189', '10987654321', 'Rua das Rosas', 'Centro', 'Goiânia', 'GO', 44, '777777777', '888888888', 'Pai 19', 'Mãe 19', 26000),
    ('2023-01-23', '2023-01-27', 'Guilherme Mendes', '15161718190', '98765432109', 'Avenida das Flores', 'Ipanema', 'Rio de Janeiro', 'RJ', 99, '999999999', '000000000', 'Pai 20', 'Mãe 20', 27000),
    ('2023-01-24', '2023-01-28', 'Amanda Ferreira', '16171819200', '87654321098', 'Rua dos Cravos', 'Barra da Tijuca', 'Rio de Janeiro', 'RJ', 11, '111111111', '222222222', 'Pai 21', 'Mãe 21', 28000),
    ('2023-01-25', '2023-01-29', 'Paulo Santos', '17181920011', '76543210987', 'Avenida dos Jasmins', 'Boa Esperança', 'Manaus', 'AM', 22, '333333333', '444444444', 'Pai 22', 'Mãe 22', 29000).
```

# Scripts DDL

[https://github.com/ojulianos/banco\\_2\\_seguros\\_imóveis/blob/main/dml/ScriptDML.sql](https://github.com/ojulianos/banco_2_seguros_imóveis/blob/main/dml/ScriptDML.sql)

TableName	rows
apolice_itens	20
apolices	35
clientes	25
imoveis	29
imovel_fotos	15
log	0
sinistro	30
sinistro_tipo	10

```
]SELECT
    t.name AS TableName,
    p.rows
FROM
    sys.tables t
INNER JOIN
    sys.indexes i ON t.object_id = i.object_id
INNER JOIN
    sys.partitions p ON i.object_id = p.object_id AND i.index_id = p.index_id
INNER JOIN
    sys.allocation_units a ON p.partition_id = a.container_id
LEFT OUTER JOIN
    sys.schemas s ON t.schema_id = s.schema_id
WHERE
    t.name NOT LIKE 'dt%'
    AND t.is_ms_shipped = 0
    AND i.object_id > 255
.
```

# PROCEDURES

[https://github.com/ojulianos/banco\\_2\\_seguros\\_imóveis/blob/main/ddl/procedure.sql](https://github.com/ojulianos/banco_2_seguros_imóveis/blob/main/ddl/procedure.sql)

```
-- procedure, gera um relatório com todos os imóveis de um cliente e a quantidade de sinistros por imóvel com sql server
drop procedure if exists pr_imoveis_vs_sinistros
go

CREATE PROCEDURE pr_imoveis_vs_sinistros
    @cod_cliente INT
AS
BEGIN
    with cte as (
        select cli_nome
            , im_tipo
            , im_logradouro
            , im_bairro
            , im_cidade
            , im_uf
            , COALESCE((select count(*) from sinistro where sin_im_cod=im_cod), 0) as sinistros
        from clientes
        left join apolices on apo_cli_cod=cli_cod
        left join imoveis on im_cod=apo_im_cod
        where cli_cod=@cod_cliente
    ) select * from cte
    order by sinistros desc
END
go
```

# PROCEDURES

[https://github.com/ojulianos/banco\\_2\\_seguros\\_imóveis/blob/main/ddl/procedure.sql](https://github.com/ojulianos/banco_2_seguros_imóveis/blob/main/ddl/procedure.sql)

```
-- procedure, gera um relatorio com todos os imoveis com o valor individual de cada apolice e valor total da franquia das apolices agrupado por cliente
drop procedure if exists pr_imoveis_vs_franquia
go

CREATE PROCEDURE pr_imoveis_vs_franquia
AS
BEGIN
    select cli_nome
           , im_tipo
           , im_logradouro
           , im_bairro
           , im_cidade
           , im_uf
           , apo_valor_franquia
           , coalesce(sum(apo_valor_franquia) over(partition by cli_cod), 0) total_franquia_cliente
           , sum(apo_valor_franquia) over() total_franquias
    from clientes cli
    left join apolices apo on apo_cli_cod=cli_cod
    left join imoveis im on im_cod=apo_im_cod
END
go

exec pr_imoveis_vs_franquia
go
```

# FUNCTIONS

[https://github.com/ojulianos/banco\\_2\\_seguros\\_imoveis/blob/main/ddl/function.sql](https://github.com/ojulianos/banco_2_seguros_imoveis/blob/main/ddl/function.sql)

```
drop function if exists fn_valor_apolice
go

-- Função para adicionar porcentagem ao valor da apólice de acordo com a pontuação do cliente
create function fn_valor_apolice(@cod_cliente int, @valor_apolice decimal) returns int
as
begin
    declare @pontos int
    declare @resultado decimal

    select @pontos = sum(sit_pontos)
    from clientes
    inner join apolices on apo_cli_cod = cli_cod
    inner join sinistro on sin_im_cod = apo_im_cod
    inner join sinistro_tipo on sin_tipo_cod = sit_cod
    where cli_cod=@cod_cliente

    BEGIN
        IF @pontos > 45
            select @resultado = (@valor_apolice * 0.5)
        ELSE IF @pontos > 35
            select @resultado = (@valor_apolice * 0.25)
        ELSE IF @pontos > 25
            select @resultado = (@valor_apolice * 0.15)
        ELSE IF @pontos > 15
            select @resultado = (@valor_apolice * 0.10)
        ELSE
            select @resultado = (@valor_apolice * 0.001)
    END

    return @valor_apolice + @resultado
end
go

select [dbo].[fn_valor_apolice](cli_cod, 500) as valor_apolice, cli.*
from clientes cli
go
```

# TRIGGERS

[https://github.com/ojulianos/banco\\_2\\_seguros\\_imóveis/blob/main/ddl/trigger.sql](https://github.com/ojulianos/banco_2_seguros_imóveis/blob/main/ddl/trigger.sql)

```
DROP TRIGGER IF EXISTS
    tr_delete_apolice_item,
    tr_delete_apolice,
    tr_delete_imovel,
    tr_delete_imovel_foto,
    tr_delete_sinistro,
    tr_delete_sinistro_tipo
GO

CREATE TRIGGER tr_delete_apolice_item ON apolice_itens for DELETE AS
BEGIN
    IF (ROWCOUNT_BIG() = 0)
        RETURN;

    DECLARE @dt_exclusao DATE = GETDATE(),
            @usuario_exclusao VARCHAR(100) = SUSER_NAME();

    INSERT INTO dbo.log (tabela, data_cadastro, usuario, tipo_operacao, operacao)
        VALUES ('apolice_itens', @dt_exclusao, @usuario_exclusao, 'delete', (SELECT * FROM deleted FOR JSON PATH));
END
GO
```

# TRIGGERS

[https://github.com/ojulianos/banco\\_2\\_seguros\\_imóveis/blob/main/ddl/trigger.sql](https://github.com/ojulianos/banco_2_seguros_imóveis/blob/main/ddl/trigger.sql)

```
CREATE TRIGGER tr_delete_apolice ON apolices INSTEAD OF DELETE AS
BEGIN
    DECLARE @dt_exclusao DATE = GETDATE(),
            @usuario_exclusao VARCHAR(100) = SUSER_NAME(),
            @itens varchar(max) = '',
            @apolice varchar(max) = '';

    SELECT @apolice = (SELECT * FROM deleted FOR JSON PATH);
    SELECT @itens = (SELECT * FROM apolice_itens WHERE api_apo_cod IN (SELECT api_cod FROM deleted) FOR JSON PATH);
    INSERT INTO dbo.log (tabela, data_cadastro, usuario, tipo_operacao, operacao)
        VALUES ('apolices', @dt_exclusao, @usuario_exclusao, 'delete', @apolice),
            ('apolice_itens', @dt_exclusao, @usuario_exclusao, 'delete', @itens);

    DELETE FROM apolice_itens WHERE api_apo_cod IN (SELECT apo_cod FROM deleted);
    DELETE FROM apolices WHERE apo_cod IN (SELECT apo_cod FROM deleted);

END
GO
```

# TRIGGERS

[https://github.com/ojulianos/banco\\_2\\_seguros\\_imóveis/blob/main/ddl/trigger.sql](https://github.com/ojulianos/banco_2_seguros_imóveis/blob/main/ddl/trigger.sql)

```
CREATE TRIGGER tr_delete_imovel ON moveis INSTEAD OF DELETE AS
BEGIN
    DECLARE @dt_exclusao DATE = GETDATE(),
            @usuario_exclusao VARCHAR(100) = SUSER_NAME(),
            @imovel varchar(max) = '',
            @imovel_fotos varchar(max) = '';

    SELECT @imovel = (SELECT * FROM deleted FOR JSON PATH);
    SELECT @imovel_fotos = (SELECT * FROM imovel_fotos WHERE imf_imovel_cod IN (SELECT im_cod FROM deleted) FOR JSON PATH);
    INSERT INTO dbo.log (tabela, data_cadastro, usuario, tipo_operacao, operacao)
        VALUES ('moveis', @dt_exclusao, @usuario_exclusao, 'delete', @imovel),
            ('imovel_fotos', @dt_exclusao, @usuario_exclusao, 'delete', @imovel_fotos);

    DELETE FROM apolices WHERE apo_im_cod IN (SELECT im_cod FROM deleted);
    DELETE FROM sinistro WHERE sin_im_cod IN (SELECT im_cod FROM deleted);
    DELETE FROM imovel_fotos WHERE imf_imovel_cod IN (SELECT im_cod FROM deleted);
    DELETE FROM moveis WHERE im_cod IN (SELECT im_cod FROM deleted);

END
GO
```



# TRIGGERS

[https://github.com/ojulianos/banco\\_2\\_seguros\\_imóveis/blob/main/ddl/trigger.sql](https://github.com/ojulianos/banco_2_seguros_imóveis/blob/main/ddl/trigger.sql)

```
CREATE TRIGGER tr_delete_imovel_foto ON imovel_fotos for DELETE AS
BEGIN
    IF (ROWCOUNT_BIG() = 0)
        RETURN;

    DECLARE @dt_exclusao DATE = GETDATE(),
            @usuario_exclusao VARCHAR(100) = SUSER_NAME();

    INSERT INTO dbo.log (tabela, data_cadastro, usuario, tipo_operacao, operacao)
        VALUES ('imovel_fotos', @dt_exclusao, @usuario_exclusao, 'delete', (SELECT * FROM deleted FOR JSON PATH));
END
GO

CREATE TRIGGER tr_delete_sinistro ON sinistro for DELETE AS
BEGIN
    IF (ROWCOUNT_BIG() = 0)
        RETURN;

    DECLARE @dt_exclusao DATE = GETDATE(),
            @usuario_exclusao VARCHAR(100) = SUSER_NAME();

    INSERT INTO dbo.log (tabela, data_cadastro, usuario, tipo_operacao, operacao)
        VALUES ('sinistros', @dt_exclusao, @usuario_exclusao, 'delete', (SELECT * FROM deleted FOR JSON PATH));
END
GO
```

# PRINCIPAIS CONSULTAS, BASEADAS EM REGRAS DE NEGÓCIO

[https://github.com/ojulianos/banco\\_2\\_seguros\\_imoveis/blob/main/perguntas-negocio.md](https://github.com/ojulianos/banco_2_seguros_imoveis/blob/main/perguntas-negocio.md)

**Qual é a distribuição das ocorrências de sinistros por tipo, e quais são os tipos de sinistros mais frequentes**

```
SELECT st.sit_nome, COUNT(s.sin_cod) AS total_ocorrencias
FROM sinistro_tipo st
LEFT JOIN sinistro s ON st.sit_cod = s.sin_tipo_cod
GROUP BY st.sit_nome
ORDER BY total_ocorrencias DESC;
```

**Listar os clientes que possuem apólices de seguros ativas, juntamente com detalhes sobre os imóveis associados a essas apólices**

```
SELECT
    c.cli_nome,
    a.apo_dt_inicio_vigencia,
    a.apo_dt_fim_vigencia,
    im.im_logradouro,
    im.im_bairro,
    im.im_cidade,
    im.im_uf
FROM clientes c
JOIN apolices a ON c.cli_cod = a.apo_cli_cod
JOIN imoveis im ON a.apo_im_cod = im.im_cod
WHERE GETDATE() BETWEEN a.apo_dt_inicio_vigencia AND a.apo_dt_fim_vigencia;
```

# PRINCIPAIS CONSULTAS, BASEADAS EM REGRAS DE NEGÓCIO

[https://github.com/ojulianos/banco\\_2\\_seguros\\_imoveis/blob/main/perguntas-negocio.md](https://github.com/ojulianos/banco_2_seguros_imoveis/blob/main/perguntas-negocio.md)

**Calcular o valor total dos sinistros para cada cliente, considerando apenas os sinistros em imóveis com apólices associadas a esse cliente:**

```
SELECT
    c.cli_nome,
    SUM(st.sit_pontos) AS total_pontos_sinistros
FROM clientes c
JOIN apolices a ON c.cli_cod = a.apo_cli_cod
JOIN imoveis im ON a.apo_im_cod = im.im_cod
JOIN sinistro sin ON sin.sin_im_cod = im.im_cod
JOIN sinistro_tipo st ON sin.sin_tipo_cod = st.sit_cod
GROUP BY c.cli_nome;
```

# PRINCIPAIS CONSULTAS, BASEADAS EM REGRAS DE NEGÓCIO

[https://github.com/ojulianos/banco\\_2\\_seguros\\_imoveis/blob/main/perguntas-negocio.md](https://github.com/ojulianos/banco_2_seguros_imoveis/blob/main/perguntas-negocio.md)

**Quais são os clientes que têm o maior número de sinistros registrados e qual é a média de pontos associados a esses sinistros?**

```
SELECT TOP 5
    c.cli_nome,
    COUNT(s.sin_cod) AS total_sinistros,
    AVG(st.sit_pontos) AS media_pontos
FROM
    clientes c
JOIN
    apolices a ON c.cli_cod = a.apo_cli_cod
JOIN
    sinistro s ON a.apo_cod = s.sin_im_cod
JOIN
    sinistro_tipo st ON s.sin_sit_cod = st.sit_cod
GROUP BY
    c.cli_nome
ORDER BY
    total_sinistros DESC;
```

# API BACK-END

[https://github.com/ojulianos/banco\\_2\\_seguros\\_imoveis/tree/main/back-end](https://github.com/ojulianos/banco_2_seguros_imoveis/tree/main/back-end)

Escrito em java com Maven e Spring Bot

## Lista de Rotas

### •Apolices

- [get] /apolices - Lista todas as apolices
- [post] /apolices - Cadastrar uma nova apolice
- [put] /apolices - Atualiza uma apolice
- [get] /apolices/{id} - Traz o registro de uma apolice
- [delete] /apolices/{id} - Exclui uma apolice


### •Clientes

- [get] /clientes - Lista todas as clientes
- [post] /clientes - Cadastrar uma nova cliente
- [put] /clientes - Atualiza uma cliente
- [get] /clientes/{id} - Traz o registro de uma cliente
- [delete] /clientes/{id} - Exclui uma cliente

# APP FRONT-END

[https://github.com/ojulianos/banco\\_2\\_seguros\\_imoveis/tree/main/front-end](https://github.com/ojulianos/banco_2_seguros_imoveis/tree/main/front-end)

Desenvolvido com React

 Seguradora Lider

Cientes

Imóveis

Apólice

Ajuda


Cadastrar Novo Cliente


NOME	DOCUMENTO	BAIRRO
------	-----------	--------

## Nova Apolice

apo\_im\_cod:

apo\_cli\_cod:

apo\_dt\_inicio\_vigencia:  
dd/mm/aaaa 

apo\_dt\_fim\_vigencia:  
dd/mm/aaaa 

apo\_valor\_cobertura:

apo\_valor\_franquia:

## Novo Cliente

cli\_nome:

cli\_documento:

cli\_documento2:

cli\_logradouro:

cli\_bairro:

cli\_cidade:

PERGUNTAS ?