

Centro Universitário UNISATC

Engenharia de Software 3a fase – Banco de Dados II – Prof. Jorge Luiz da Silva

TRABALHO FINAL COM BASE EM METODOLOGIAS ATIVAS DE APRENDIZAGEM

Projeto de banco de dados para um sistema de gerenciamento de seguros de imóvel.

Gabriel Rona Guzzatti - <https://github.com/Guzzatti>

João Vitor Rodrigues Rocha - <https://github.com/JoaoVitorRodriguesRocha>

João Pedro Darabas - <https://github.com/jpdarabas>

Lucas Fortunato Martins - <https://github.com/lucasfm20>

Criciúma, 08/12/2023

URL do projeto no GitHub

<https://github.com/Guzzatti/seguro-imoveis-satc-2023>

The screenshot shows the GitHub repository page for 'seguro-imoveis-satc-2023'. The repository is public and was forked from 'jlsilva01/projeto_final_bd2_satc_2023'. The main branch is selected, and the repository is 65 commits ahead and 52 commits behind the upstream branch. The repository contains a README file and several folders: CRUD, Imagens, and Scripts. The README file was updated 3 hours ago, while the other folders were updated 4, 12, and 3 hours ago respectively.

seguro-imoveis-satc-2023 Public

forked from [jlsilva01/projeto_final_bd2_satc_2023](#)

main

Go to file Add file <> Code

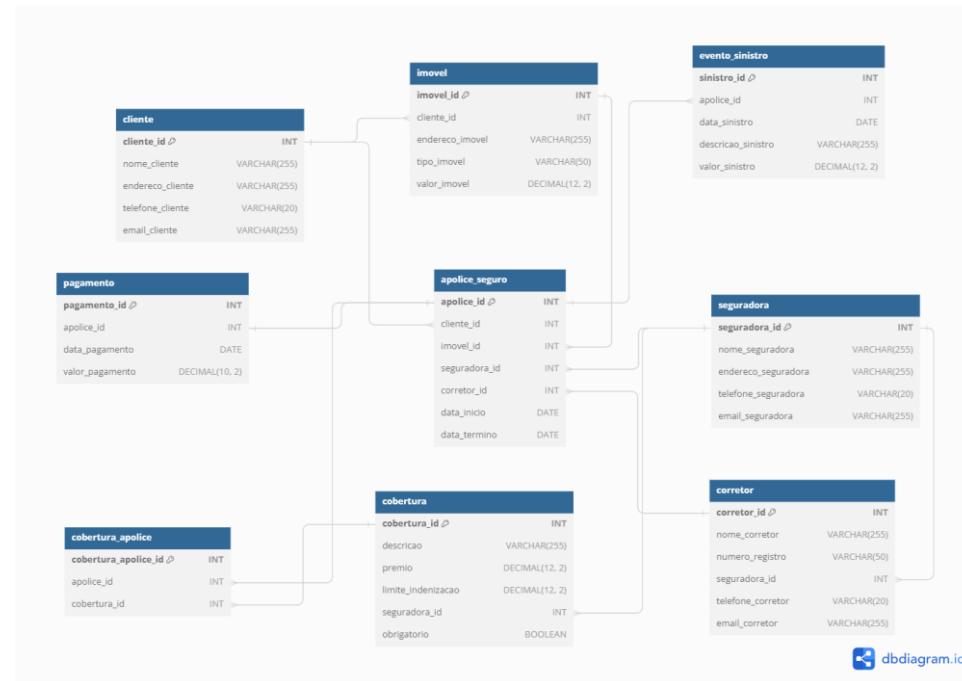
Branches Tags

This branch is 65 commits ahead, 52 commits behind jlsilva01:main. Contribute Sync fork

Guzzatti readme ... 3 hours ago 75

CRUD	phpinfo	4 hours ago
Imagens	unique change	12 hours ago
Scripts	FIX functions	3 hours ago
README.md	readme	3 hours ago

Modelo ER Físico



Dicionário de Dados

Tabela	cliente					
Descrição	Tabela responsável por armazenar os dados dos clientes das seguradoras.					
Atributos						
Nome da Coluna	Tipo do Dado	Valor min e max	Nulidade	PK	FK	Descrição
cliente_id	int	1 – 2147483647	NOT NULL	X		Código de identificador do cliente
nome_cliente	varchar(255)	1 – 255	NULL			Nome da avaliação
endereco_cliente	varchar(255)	1 – 255	NULL			Endereço do cliente
telefone_cliente	varchar(20)	8 – 20	NULL			Telefone do cliente
email_cliente	varchar(255)	1 – 255	NULL			E-mail do cliente
Índice						
Nome do índice	Clustered	NonClustered	Unique	Colunas		
PRIMARY	X		X	cliente_id		

Tabela	imovel					
Descrição	Tabela responsável por armazenar os dados dos imóveis dos clientes.					
Atributos						
Nome da Coluna	Tipo do Dado	Valor min e max	Nulidade	PK	FK	Descrição
imovel_id	int	1 – 2147483647	NOT NULL	X		Código de identificador do imóvel
cliente_id	int	1 – 2147483647	NULL		X	Código de identificador do cliente
endereco_imovel	varchar(255)	1 – 255	NULL			Endereço do cliente
tipo_imovel	varchar(50)	1 – 50	NULL			Tipo do imóvel

valor_imovel	decimal(12,2)	1 – 9999999999,99	NULL			Valor do imóvel
Índice						
Nome do índice	Clustered	NonClustered	Unique	Colunas		
PRIMARY	X		X	imovel_id		
cliente_id		X		cliente_id		

Tabela	evento_sinistro					
Descrição	Tabela responsável por armazenar os dados dos sinistros.					
Atributos						
Nome da Coluna	Tipo do Dado	Valor min e max	Nulidade	PK	FK	Descrição
sinistro_id	int	1 – 2147483647	NOT NULL	X		Código de identificação do sinistro
apolice_id	int	1 – 2147483647	NULL		X	Código de identificação da apólice
data_sinistro	varchar(255)	1 – 255	NULL			Data do sinistro
descricao_sinistro	varchar(255)	1 – 255	NULL			Descrição do sinistro
valor_sinistro	decimal(12,2)	1 – 9999999999,99	NULL			Valor do sinistro
Índice						
Nome do índice	Clustered	NonClustered	Unique	Colunas		
PRIMARY	X		X	sinistro_id		
apolice_id		X		apolice_id		
idx_evento_sinistro		X		data_sinistro, descricao_sinistro, valor_sinistro		

Tabela	pagamento					
Descrição	Tabela responsável por armazenar os dados dos pagamentos dos apólices					
Atributos						
Nome da Coluna	Tipo do Dado	Valor min e max	Nulidade	PK	FK	Descrição
pagamento_id	int	1 – 2147483647	NOT NULL	X		Código de identificação do pagamento
apolice_id	int	1 – 2147483647	NULL		X	Código de identificação da apólice
data_pagamento	date	01/01/1923 - (data atual)	NULL			Data do pagamento
valor_pagamento	decimal(10,2)	1 – 99999999,99	NULL			Valor do pagamento
Índice						
Nome do índice	Clustered	NonClustered	Unique	Colunas		
PRIMARY	X		X	sinistro_id		
apolice_id		X		apolice_id		
idx_evento_sinistro		X		data_sinistro, descricao_sinistro, valor_sinistro		

Tabela	apolice_seguro					
Descrição	Tabela responsável por armazenar os dados das apólices de seguro.					
Atributos						
Nome da Coluna	Tipo do Dado	Valor min e max	Nulidade	PK	FK	Descrição
apolice_id	int	1 – 2147483647	NOT NULL	X		Código de identificação da apólice
cliente_id	int	1 – 2147483647	NULL		X	Código de identificação do cliente
imovel_id	int	1 – 2147483647	NULL		X	Código de identificação do imóvel
seguradora_id	int	1 – 2147483647	NULL		X	Código de identificação da seguradora

corretor_id	int	1 – 2147483647	NULL		X	Código de identificação do corretor
data_inicio	DATE	01/01/1923 - (data atual)	NULL			Data de início do seguro
data_termino	DATE	01/01/1923 - 01/01/2123	NULL			Data do fim do seguro
Índice						
Nome do índice	Clustered	NonClustered	Unique	Colunas		
PRIMARY	X		X	apolice_id		
cliente_id		X		cliente_id		
imovel_id		X		imovel_id		
seguradora_id		X		seguradora_id		
corretor_id		X		corretor_id		
idx_apolice_seguro		X		data_termino		

Tabela	seguradora					
Descrição	Tabela responsável por armazenar os dados das seguradoras.					
Atributos						
Nome da Coluna	Tipo do Dado	Valor min e max	Nulidade	PK	FK	Descrição
seguradora_id	int	1 – 2147483647	NOT NULL	X		Código de identificador da seguradora
nome_seguradora	varchar(255)	1 – 255	NULL			Nome da seguradora
endereco_seguradora	varchar(255)	1 – 255	NULL			Endereço da seguradora
telefone_seguradora	varchar(20)	8 – 20	NULL			Telefone da seguradora
email_seguradora	varchar(255)	1 – 255	NULL			E-mail da seguradora

Índice				
Nome do índice	Clustered	NonClustered	Unique	Colunas
PRIMARY	X		X	seguradora_id

Tabela	corretor					
Descrição	Tabela responsável por armazenar os dados dos corretores de seguro.					
Atributos						
Nome da Coluna	Tipo do Dado	Valor min e max	Nulidade	PK	FK	Descrição
corretor_id	int	1 – 2147483647	NOT NULL	X		Código de identificador do corretor
nome_corretor	varchar(255)	1 – 255	NULL			Nome do corretor
numero_registro	varchar(50)	1 – 255	NULL		X	Número de registro do corretor
seguradora_id	INT	1 – 2147483647	NULL			Código de identificador da seguradora
telefone_corretor	varchar(20)	8 – 20	NULL			Telefone do corretor
email_corretor	varchar(255)	1 – 255	NULL			E-mail do corretor
Índice						
Nome do índice	Clustered	NonClustered	Unique	Colunas		
PRIMARY	X		X	corretor_id		
seguradora_id		X		seguradora_id		

Tabela	cobertura_apolice					
Descrição	Tabela intermediária entre as tabelas apolice_seguro e cobertura					
Atributos						
Nome da Coluna	Tipo do Dado	Valor min e max	Nulidade	PK	FK	Descrição
cobertura_apolice_jd	int	1 – 2147483647	NOT NULL	X		Código de identificação da cobertura_apolice

cobertura_jd	int	1 – 2147483647	NULL		X	Código de identificação da cobertura_apolice
apolice_id	int	1 – 2147483647	NULL		X	Código de identificação do apólice
Índice						
Nome do índice	Clustered	NonClustered	Unique	Colunas		
PRIMARY	X		X	cobertura_apolice_id		
apolice_id		X		apolice_id		
cobertura_id		X		cobertura_id		

Tabela	cobertura					
Descrição	Tabela responsável por armazenar os dados das coberturas de seguro das seguradoras.					
Atributos						
Nome da Coluna	Tipo do Dado	Valor min e max	Nulidade	PK	FK	Descrição
cobertura_id	int	1 – 2147483647	NOT NULL	X		Código de identificação da cobertura
descricao	varchar(255)	1 – 255	NULL			Descrição da cobertura
premio	decimal(12,2)	1 – 9999999999,99	NULL			Prêmio da cobertura
limite_indenizacao	decimal(12,2)	1 – 9999999999,99	NULL			Limite de indenização da cobertura
seguradora_id	varchar(255)	1 – 255	NULL		X	Código de identificação da seguradora
obrigatorio	boolean	0 – 1	NULL			Se a cobertura é obrigatória ou não
Índice						
Nome do índice	Clustered	NonClustered	Unique	Colunas		
PRIMARY	X		X	cobertura_id		

seguradora_id		X		seguradora_id
idx_cobertura		X		obrigatorio, descricao, premio, limite_indenizacao

Script dos comandos DDL para criação do Banco de dados

```
CREATE TABLE cliente (
    cliente_id INT AUTO_INCREMENT PRIMARY KEY,
    nome_cliente VARCHAR(255),
    endereco_cliente VARCHAR(255),
    telefone_cliente VARCHAR(20),
    email_cliente VARCHAR(255)
);

CREATE TABLE imovel (
    imovel_id INT AUTO_INCREMENT PRIMARY KEY,
    cliente_id INT,
    endereco_imovel VARCHAR(255),
    tipo_imovel VARCHAR(50),
    valor_imovel DECIMAL(12, 2),
    FOREIGN KEY (cliente_id) REFERENCES cliente(cliente_id)
);
```

```
CREATE TABLE cobertura (  
    cobertura_id INT AUTO_INCREMENT PRIMARY KEY,  
    descricao VARCHAR(255),  
    premio DECIMAL(12, 2),  
    limite_indenizacao DECIMAL(12, 2),  
    seguradora_id INT,  
    obrigatorio BOOLEAN,  
    FOREIGN KEY (seguradora_id) REFERENCES seguradora(seguradora_id)  
);
```

```
CREATE TABLE apolice_seguro (  
    apolice_id INT AUTO_INCREMENT PRIMARY KEY,  
    cliente_id INT ,  
    imovel_id INT ,  
    seguradora_id INT,  
    corretor_id INT,  
    data_inicio DATE,  
    data_termino DATE,  
    FOREIGN KEY (cliente_id) REFERENCES cliente(cliente_id),  
    FOREIGN KEY (imovel_id) REFERENCES imovel(imovel_id),  
    FOREIGN KEY (seguradora_id) REFERENCES seguradora(seguradora_id),  
    FOREIGN KEY (corretor_id) REFERENCES corretor(corretor_id)
```

```
);
```

```
CREATE TABLE cobertura_apolice (  
    cobertura_apolice_id INT AUTO_INCREMENT PRIMARY KEY,  
    apolice_id INT,  
    cobertura_id INT,  
    FOREIGN KEY (apolice_id) REFERENCES apolice_seguro(apolice_id),  
    FOREIGN KEY (cobertura_id) REFERENCES cobertura(cobertura_id)  
);
```

```
CREATE TABLE pagamento (  
    pagamento_id INT AUTO_INCREMENT PRIMARY KEY,  
    apolice_id INT,  
    data_pagamento DATE,  
    valor_pagamento DECIMAL(10, 2),  
    FOREIGN KEY (apolice_id) REFERENCES apolice_seguro(apolice_id)  
);
```

```
CREATE TABLE corretor (  
    corretor_id INT AUTO_INCREMENT PRIMARY KEY,  
    nome_corretor VARCHAR(255),
```

```
numero_registro VARCHAR(50),  
seguradora_id INT,  
telefone_corretor VARCHAR(20),  
email_corretor VARCHAR(255),  
FOREIGN KEY (seguradora_id) REFERENCES seguradora(seguradora_id)  
);
```

```
CREATE TABLE seguradora (  
    seguradora_id INT AUTO_INCREMENT PRIMARY KEY,  
    nome_seguradora VARCHAR(255) ,  
    endereco_seguradora VARCHAR(255),  
    telefone_seguradora VARCHAR(20),  
    email_seguradora VARCHAR(255)  
);
```

```
CREATE TABLE evento_sinistro (  
    sinistro_id INT AUTO_INCREMENT PRIMARY KEY,  
    apolice_id INT,  
    data_sinistro DATE,  
    descricao_sinistro VARCHAR(255),  
    valor_sinistro DECIMAL(12, 2),  
    FOREIGN KEY (apolice_id) REFERENCES apolice_seguro(apolice_id)
```

);

Script que popula as tabelas do Banco de dados

```
INSERT INTO cliente (nome_cliente, endereco_cliente, telefone_cliente, email_cliente)
VALUES
('João Vitor', 'Rua A, 123', '+55 11 98765-4321', 'joao.vitor@gmail.com'),
('Maria Oliveira', 'Av. B, 456', '+55 21 99876-5432', 'maria.oliveira@gmail.com'),
('Carlos Santos', 'Rua C, 789', '+55 31 98765-8765', 'carlos.santos@gmail.com'),
('Fernanda de Souza', 'Rua D, 321', '+55 11 12345-6789', 'fernanda@gmail.com'),
('Ricardo Silva', 'Av. E, 654', '+55 21 98765-4321', 'ricardo@gmail.com'),
('Patricia Duarte', 'Rua F, 987', '+55 31 87654-3210', 'patricia@gmail.com'),
('Pedro Pinto', 'Av. G, 789', '+55 11 5555-1111', 'pedro@gmail.com'),
('Isabela Rodrigues', 'Rua H, 987', '+55 21 6666-2222', 'isabela@gmail.com'),
('Felipe Rocha', 'Av. I, 654', '+55 31 7777-3333', 'felipe@gmail.com'),
('Amanda Lima', 'Rua J, 321', '+55 41 8888-4444', 'amanda@gmail.com')
```

```
INSERT INTO imovel (cliente_id, endereco_imovel, tipo_imovel, valor_imovel)
VALUES
(1, 'Rua A, 123', 'Casa', 300000.00),
(2, 'Av. B, 456', 'Apartamento', 200000.00),
(3, 'Rua C, 789', 'Casa', 250000.00),
(4, 'Rua D, 321', 'Apartamento', 180000.00),
(5, 'Av. E, 654', 'Casa', 280000.00),
(6, 'Rua F, 987', 'Apartamento', 220000.00),
(7, 'Av. G, 789', 'Casa', 350000.00),
(8, 'Rua H, 987', 'Apartamento', 180000.00),
(9, 'Av. I, 654', 'Casa', 270000.00),
(10, 'Rua J, 321', 'Apartamento', 200000.00)
```

```
INSERT INTO seguradora (nome_seguradora, endereco_seguradora, telefone_seguradora, email_seguradora)
VALUES
('Seguradora A', 'Av. X, 789', '+55 11 1111-2222', 'contato@seguradoraA.com'),
('Seguradora B', 'Rua Y, 456', '+55 21 2222-3333', 'contato@seguradoraB.com'),
('Seguradora C', 'Av. Z, 123', '+55 31 3333-4444', 'contato@seguradoraC.com'),
('Seguradora D', 'Av. W, 987', '+55 11 5555-7777', 'contato@seguradoraD.com'),
('Seguradora E', 'Rua X, 654', '+55 21 6666-8888', 'contato@seguradoraE.com'),
('Seguradora F', 'Av. Y, 321', '+55 31 7777-9999', 'contato@seguradoraF.com'),
```

```
('Seguradora G', 'Rua K, 123', '+55 11 1111-5555', 'contato@seguradoraG.com'),  
( 'Seguradora H', 'Av. L, 456', '+55 21 2222-6666', 'contato@seguradoraH.com'),  
( 'Seguradora I', 'Rua M, 789', '+55 31 3333-7777', 'contato@seguradoraI.com'),  
( 'Seguradora J', 'Av. N, 987', '+55 41 4444-8888', 'contato@seguradoraJ.com')
```

```
INSERT INTO corretor (nome_corretor, numero_registro, seguradora_id, telefone_corretor, email_corretor)  
VALUES
```

```
('Ana 1', 'CR123', 1, '+55 11 5555-6666', 'ana.1@corretora.com'),  
( 'José 2', 'CR456', 2, '+55 21 6666-7777', 'jose.2@corretora.com'),  
( 'Mariana 3', 'CR789', 3, '+55 31 7777-8888', 'mariana.3@corretora.com'),  
( 'Roberto 4', 'CR234', 1, '+55 11 2222-3333', 'roberto.4@corretora.com'),  
( 'Camila 5', 'CR567', 2, '+55 21 3333-4444', 'camila.5@corretora.com'),  
( 'Lucas Fortunato Martins', 'CR890', 3, '+55 31 4444-5555', 'lucasfm20@corretora.com'),  
( 'Rafaela 6', 'CR567', 1, '+55 11 5555-9999', 'rafaela.6@corretora.com'),  
( 'Luciano 7', 'CR890', 2, '+55 21 6666-0000', 'luciano.7@corretora.com'),  
( 'Gabriel Guzzati', 'CR123', 3, '+55 31 7777-1111', 'guzzateradasnovinha+18@corretora.com'),  
( 'João Pedro Darabas', 'CR456', 4, '+55 41 8888-2222', 'jpdarabas@corretora.com')
```

```
INSERT INTO cobertura (descricao, premio, limite_indenizacao, seguradora_id, obrigatorio)  
VALUES
```

```
('Roubo', 500.00, 10000.00, 1, true),  
( 'Incêndio', 400.00, 8000.00, 2, true),
```



```
('Danos Elétricos', 300.00, 6000.00, 3, false),  
( 'Vidros Quebrados', 150.00, 3000.00, 1, false),  
( 'Responsabilidade Civil', 200.00, 5000.00, 2, true),  
( 'Furto Qualificado', 120.00, 2000.00, 3, false),  
( 'Assistência 24 horas', 100.00, 2000.00, 1, false),  
( 'Danos Morais', 250.00, 5000.00, 2, true),  
( 'Furto Simples', 80.00, 1000.00, 3, false),  
( 'Riscos Atmosféricos', 180.00, 3000.00, 4, true)
```

```
INSERT INTO apolice_seguro (cliente_id, imovel_id, seguradora_id, corretor_id, data_inicio, data_termino)
```

```
VALUES
```

```
(1, 1, 1, 1, '2023-01-01', '2024-01-01'),  
(2, 2, 2, 2, '2023-02-01', '2024-02-01'),  
(3, 3, 3, 3, '2023-03-01', '2024-03-01'),  
(4, 4, 1, 1, '2023-04-01', '2024-04-01'),  
(5, 5, 2, 2, '2023-05-01', '2024-05-01'),  
(6, 6, 3, 3, '2023-06-01', '2024-06-01'),  
(7, 7, 1, 1, '2023-07-01', '2024-07-01'),  
(8, 8, 2, 2, '2023-08-01', '2024-08-01'),  
(9, 9, 3, 3, '2023-09-01', '2024-09-01'),  
(10, 10, 4, 4, '2023-10-01', '2024-10-01')
```

```
INSERT INTO cobertura_apolice (apolice_id, cobertura_id)
```

```
VALUES
```

```
(1, 1),
```

```
(2, 2),
```

```
(3, 3),
```

```
(4, 4),
```

```
(5, 5),
```

```
(6, 6),
```

```
(7, 7),
```

```
(8, 8),
```

```
(9, 9),
```

```
(10, 10)
```

```
INSERT INTO pagamento (apolice_id, data_pagamento, valor_pagamento)
```

```
VALUES
```

```
(1, '2023-01-15', 1500.00),
```

```
(2, '2023-02-15', 1200.00),
```

```
(3, '2023-03-15', 1300.00),
```

```
(4, '2023-04-15', 1400.00),
```

```
(5, '2023-05-15', 1100.00),
```

```
(6, '2023-06-15', 1200.00),
```

```
(7, '2023-07-15', 1600.00),
```

```
(8, '2023-08-15', 1300.00),  
(9, '2023-09-15', 1400.00),  
(10, '2023-10-15', 1500.00)
```

```
INSERT INTO evento_sinistro (apolice_id, data_sinistro, descricao_sinistro, valor_sinistro)
```

```
VALUES
```

```
(1, '2023-02-01', 'Roubo na residência', 10000.00),  
(2, '2023-03-10', 'Incêndio no apartamento', 20000.00),  
(3, '2023-04-05', 'Danos elétricos na casa', 15000.00),  
(4, '2023-05-01', 'Vandalismo no apartamento', 8000.00),  
(5, '2023-06-10', 'Roubo na residência', 15000.00),  
(6, '2023-07-05', 'Danos causados por tempestade', 10000.00),  
(7, '2023-08-01', 'Roubo na residência', 12000.00),  
(8, '2023-09-10', 'Incêndio no apartamento', 25000.00),  
(9, '2023-10-05', 'Vandalismo na casa', 18000.00),  
(10, '2023-11-20', 'Inundação no apartamento', 20000.00)
```

Índices

```
CREATE INDEX idx_cobertura ON cobertura (obrigatorio, descricao, premio, limite_indenizacao);
```

```
CREATE INDEX idx_pagamento ON pagamento (data_pagamento, valor_pagamento);  
CREATE INDEX idx_apolice_seguro ON apolice_seguro (data_termino);  
CREATE INDEX idx_evento_sinistro ON evento_sinistro (data_sinistro, descricao_sinistro, valor_sinistro);
```

Funções, Procedures e Triggers

```
DELIMITER $$
```

```
CREATE FUNCTION premio_apolice(id_apolice INT)  
RETURNS DECIMAL(12, 2)  
DETERMINISTIC  
BEGIN  
    DECLARE resultado DECIMAL(12, 2);  
    SELECT SUM(c.premio) INTO resultado  
    FROM cobertura_apolice ca  
    INNER JOIN cobertura c  
    ON ca.cobertura_id = c.cobertura_id  
    WHERE ca.apolice_id = id_apolice;  
  
    RETURN resultado;
```

```
END $$
```

```
DELIMITER ;
```

```
-- Listar clientes que ainda não quitaram a dívida do apólice em determinada seguradora
```

```
DELIMITER $$
```

```
CREATE PROCEDURE clientes_com_divida(IN id_seguradora INT)
```

```
BEGIN
```

```
    SELECT nome_cliente, telefone_cliente, email_cliente, premio_apolice(a.apolice_id) - sum(valor_pagamento) as  
'valor a quitar', data_termino
```

```
    from cliente c
```

```
    INNER JOIN apolice_seguro a
```

```
    ON c.cliente_id = a.cliente_id
```

```
    INNER JOIN pagamento p
```

```
    ON a.apolice_id = p.apolice_id
```

```
    WHERE a.seguradora_id = id_seguradora
```

```
    GROUP BY a.apolice_id;
```

```
END $$
```

```
DELIMITER ;
```

DELIMITER \$\$

DELIMITER \$\$

CREATE TRIGGER insert_sinistro_futuro

BEFORE INSERT ON evento_sinistro

FOR EACH ROW

BEGIN

IF NEW.data_sinistro > CURDATE() THEN

SIGNAL SQLSTATE '45000'

SET MESSAGE_TEXT = 'Infelizmente a máquina do tempo ainda não foi inventada, logo não é permitido inserir eventos que ainda não aconteceram.';

END IF;

END \$\$

DELIMITER ;

DELIMITER \$\$

```
CREATE TRIGGER insert_apolice_dadas_invertidas
BEFORE INSERT ON apolice_seguro
FOR EACH ROW
BEGIN
    IF NEW.data_inicio > NEW.data_termino THEN
        SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = 'Não é permitido inserir apólice de seguros que terminam antes de começar.';
    END IF;
END $$

DELIMITER ;
```

Principais consultas mapeadas baseadas em regras de negócio

1- Qual é o prêmio total de uma apólice com o ID igual a 1, considerando as coberturas associadas a essa apólice?

```
SELECT premio_apolice(1) AS premio_total;
```

2- Qual é a cobertura mais cara entre todas as apólices de seguro e qual é o cliente associado a essa cobertura?

```
SELECT co.descricao,co.cobertura_id, c.nome_cliente, MAX(co.premio) AS premio_maximo FROM cobertura co JOIN cobertura_apolice ca ON co.cobertura_id = ca.cobertura_id JOIN apolice_seguro a ON ca.apolice_id = a.apolice_id JOIN cliente c ON a.cliente_id = c.cliente_id GROUP BY co.cobertura_id
```

3- Quais foram os clientes que tiveram a descrição do sinistro como 'Roubo a residência' e qual é o tipo do seu imóvel?

```
select nome_cliente,descricao_sinistro,tipo_imovel from cliente c inner join apolice_seguro a on c.cliente_id = a.cliente_id inner join evento_sinistro e on a.apolice_id = e.apolice_id inner join imovel i on a.imovel_id = i.imovel_id where descricao_sinistro = 'Roubo na residência'
```

4- Quais são as apólices de seguro que vencerão até o dia 30/03/2024?

```
SELECT apolice_id FROM apolice_seguro a WHERE a.data_termino BETWEEN CURDATE() AND '2024-03-30';
```