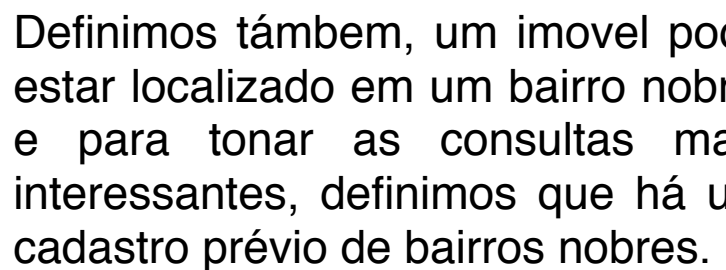


Também definimos que, uma transação, seja ela proveniente de um aluguel ou venda, pode sofrer N atrasos.



Nomes :

Eduardo T. Rocato: 2019190201871

Victor Lelis: 201919040382

Alec Vinicius Albuquerque Vargas: 201919020330

Fabio Henrique Serafim Pessoa: 201719040591

Julio Cezar Rodrigues: 201919020250

Modelo relacional

Para os relacionamento com cardinalidade 1:N foi escolhido o método de adição de coluna, pois torna o esquema um pouco mais simples, com exceção dos relacionamentos Venda e Aluguel.

Cargo(NomeFuncionario(PK)(FK),CpfFuncionario(PK)(FK),IdCargo(PK),
NomeCargo, Descricao, SalarioBase)

Funcionario(NomeFuncionario(PK), CpfFuncionario(PK), IdCargo(FK), Endereco,
DataAdmi, LoginUsuario, LoginSenha)

TelefoneFunc(nomeFunc(PK)(FK), CpfFunc(PK)(FK), NumTelefone(PK))

FormaPagamento(IdPagamento(PK),IdTransacao(PK)(FK),NumParcelas,IdContaRe
cebida)

Transacao(IdTransacao(PK), IdPagamento(FK), ValBruto, ValDesImob,
DataRetirada)

Atraso(IdAtraso(PK), DataPagamento(PK), DataVencimento(PK), ValorAluguel)

TransacaoAtraso(IdTransacao(PK)(FK), IdAtraso(PK)(FK),
DataPagamento(PK)(FK), DataVencimento(PK)(FK))

Nesse caso escolhemos modelar os relacionamentos Venda e aluguel 1:N como tabela própria, assim fica melhor evidenciado que uma transação pode ser uma venda ou um aluguel, e que a forma como um imóvel se relaciona com a transação, indica se ele está para aluguel ou transação.

Venda(IdTransacao(PK)(FK), IdPagamento(PK)(FK), IdImovel(PK)(FK), ValorVenda)

Aluguel(IdTransacao(PK)(FK), IdPagamento(PK)(FK), IdImovel(PK)(FK), ValorAluguel)

Para as especializações foi escolhido o método de uma tabela para cada entidade, porque as subclasses da classe cliente possuem relacionamentos específicos.

Cliente(NomeCliente(PK), CpfCliente(PK), Profissao, Endereco, Email, Sexo, Estado)

ClienteUsuario(NomeCliente(PK)(FK), CpfCliente(PK)(FK))

ClienteProprietario(NomeCliente(PK)(FK), CpfCliente(PK)(FK))

Fiador(NomeFiador(PK), CpfFiador(PK), IdFiador(PK), RG(PK), NomeCliente(PK)(FK), CpfCliente(PK)(FK), Cep)

Indicacao(NomeIndicacao(PK), CpfIndicacao(PK), IdIndicacao(PK), NomeCliente(PK)(FK), CpfCliente(PK)(FK), Cep)

FotosImovel(IdImovel(PK)(FK), Nome_arq_foto(PK))

BairrosNobres(IdBairro(PK), Bairro, Descricao, EscolaLocalidade)

Imovel(IdImovel(PK), IdBairro(FK) Cidade, Estado, Rua, Numero, Bairro, ExcluidoComprador, DataConst, ValorImovel, TipoNegocio)

Terreno(IdImovel(PK)(FK), AcliveDeclive, Comprimento, Largura, Area)

SalaComercial(IdImovel(PK)(FK), QtdBanheiro, QtdComodo, Area)

Moradia(IdImovel(PK)(FK), QtdEstar, QtdSuites, QtdQuartos, VagasGaragem, Area, Descricao, ArmarioEmbutido, QtdJantar)

Casa(IdImovel(PK)(FK), NumCasa)

Apto(IdImovel(PK)(FK), ValorCondo, Andar, Portaria24)

HistoricoImovel(IdHistorico(PK),Idmovel(FK),DataConstru,ExcluidoComprador, Bairro, Rua, Numero, Estado, Cidade, ValorImovel, TipoNegocio)

FotosHistoricoImovel(IdHistorico(PK), Nome_arq_foto(PK))

HistoricoMoradia(IdHistorico(FK)(PK), QtdJantar, ArmarioEmbutido, Descricao, Area, VagasGaragem, QtdQuartos, QtdSuites, QtdEstar)

HistoricoTerreno(IdHistorico(FK)(PK), AcliveDeclive, Comprimento, Area, largura)

HistoricoSalaComercial(IdHistorico(FK)(PK), Area, QtdBanheiro, QtdComodo)

HistoricoCasa(IdHistorico(FK)(PK), NumCasa)

HistoricoApto(IdHistorico(FK)(PK), Portaria24, ValorCondo, Andar)

ClienteImovel(NomeCliente(PK)(FK), CpfCliente(PK)(FK), IdImovel(PK)(FK))

ClienteFuncionarioTransacao(NomeFuncionario(PK)(FK),CpfFuncionario(PK)(FK),Id Transacao(PK)(FK),NomeCliente(PK)(FK),CpfCliente(PK)(FK), comissao_func)

ClienteFiadorIndicacao(NomeCliente(PK)(FK),CpfCliente(PK)(FK),NomeFiador(PK)(FK),CpfFiador(PK)(FK),RGFiador(PK)(FK),IdFiador(PK)(FK),NomeIndicacao(PK)(FK),CpfIndicacao(PK)(FK), IdIndicacao(PK)(FK))

Consultas em Álgebra Relacional e SQL:

Álgebra Relacional:

1) soma de todas as vendas realizadas no mes X do ano y

$A \leftarrow \text{Venda} * \text{Transacao}$

$B \leftarrow \sigma \text{ DataRetirada} \geq 01/x/y \text{ AND } \text{DataRetirada} \leq 31/x/y (A)$

$C \leftarrow \pi \text{ ValorVenda} (B)$

$R \leftarrow \mathcal{F} \text{ SUM } \text{ValorVenda}$

2) soma de todos os alugueis realizados no mes x do ano y

$A \leftarrow \text{Aluguel} * \text{Transacao}$

$B \leftarrow \sigma \text{ DataRetirada} \geq 01/x/y \text{ AND } \text{DataRetirada} \leq 31/x/y (A)$

$C \leftarrow \pi \text{ ValorAluguel} (B)$

$R \leftarrow \mathcal{F} \text{ SUM } \text{ValorAluguel} (C)$

3) lista de imoveis em bairros nobres disponivel para venda

$R \leftarrow \text{Imovel} * \text{BairrosNobres}$

4) listar imoveis na faixa de preço do cliente sendo x o piso e y o teto

$R \leftarrow \sigma \text{ ValorImovel} \geq x \text{ AND } \text{ValorImovel} \leq y (\text{Imovel})$

5) ordenar funcionarios em um ranking de vendas

$A \leftarrow \text{CpfFuncionario } \mathcal{F} \text{ COUNT } \text{IdTransacao} (\text{ClienteFuncionarioTransacao})$

$B \leftarrow \pi \text{ NomeFuncionario, CpfFuncionario, IdTransacao} (A)$

$R \leftarrow \tau \text{ IdTransacao DESC} (B)$

6) lista de imoveis com alugueis atrasados, a quantidade de alugueis atrasados e o valor total dos alugueis atrasados.

$Relacao \leftarrow \sigma \quad DataPagamento \text{ IS NOT NULL AND } DataPagamento > DataVencimento(TransacaoAtraso)$
 $\pi \text{ ValorAluguel, idImovel}(Relacao)$

7) listar bairros e a quantidade de imoveis no bairro que estão a mais de um tempo y sem vender, sendo $x = dataAtual - y$

$A \leftarrow \sigma \text{ DataAquisicao} < x \text{ (Imovel)}$
 $B \leftarrow \text{Bairro } \Join \text{ COUNT IdImovel (A)}$
 $R \leftarrow \pi \text{ Bairro, IdImovel (B)}$

8) listar funcionarios aptos a se aposentarem sendo $x = dataAtual - 35$ anos

$R \leftarrow \sigma \text{ DataAdmi} \leq x \text{ (Funcionario)}$

9) Quantidade de funcionarios por cargo

$A \leftarrow \text{Cargo} * \text{Funcionario}$
 $B \leftarrow \text{IdCargo } \Join \text{ COUNT CpfFuncionario (A)}$
 $R \leftarrow \pi \text{ NomeCargo, CpfFuncionario (B)}$

10) listar em ordem as quantidades de parcelas mais usadas em pagamentos

$A \leftarrow \pi \text{ IdPagamento, NumParcelas (FormaPagamento)}$
 $B \leftarrow \text{NumParcelas } \Join \text{ COUNT IdPagamento (A)}$
 $R \leftarrow \tau \text{ IdPagamento DESC (B)}$

SQL:

1) soma de todas as vendas realizadas no mes X do ano y

$\text{SELECT SUM(Venda.ValorVenda) FROM Venda INNER JOIN Transacao ON}$
 $\text{Venda.IdTransacao = Transacao.IdTransacao WHERE Transacao.DataRetirada } \geq$
 $01/x/y \text{ AND Transacao.DataRetirada } \leq 31/x/y;$

2) soma de todos os alugueis realizados no mes x do ano y

```
SELECT SUM(Aluguel.ValorAluguel) FROM Aluguel INNER JOIN Transacao ON
Aluguel.IdTransacao=Transacao.IdTransacao WHERE Transacao.DataRetirada >=
01/x/y AND Transacao.DataRetirada <= 31/x/y;
```

3) lista de imoveis em bairros nobres disponivel para venda

```
SELECT * FROM Imovel INNER JOIN BairrosNobres ON
Imovel.Bairro=BairrosNobres.Bairro;
```

4) listar imoveis na faixa de preço do cliente sendo x o piso e y o teto

```
SELECT * FROM Imovel WHERE ValorImovel >= x AND ValorImovel <= y;
```

5) ordenar funcionarios em um ranking de vendas

```
SELECT NomeFuncionario AS Nome, CpfFuncionario AS CPF, COUNT (IdTransacao)
AS QtdVendas FROM ClienteFuncionarioTransacao GROUP BY CpfFuncionario
ORDER BY QtdVendas DESC;
```

6) lista de imoveis com alugueis atrasados, a quantidade de alugueis atrasados e o valor total dos alugueis atrasados.

```
SELECT ValorAluguel, IdImovel FROM TransacaoAtraso WHERE DataPagamento >
DataVencimento.
```

7) listar bairros e a quantidade de imoveis no bairro que estão a mais de um tempo y sem vender, sendo x = dataAtual - y

```
SELECT Bairro, COUNT (IdImovel) FROM Imovel WHERE DataAquisicao < x GROUP
BY Bairro;
```

8) listar funcionarios aptos a se aposentarem sendo x = dataAtual - 35 anos

```
SELECT * FROM Funcionario WHERE DataAdmi <= x;
```

9) Quantidade de funcionarios por cargo

```
SELECT NomeCargo, COUNT (CpfFuncionario) FROM Cargo INNER JOIN
Funcionario ON Cargo.CpfFuncionario = Funcionario.CpfFuncionario AND
Cargo.idCargo = Funcionario.idCargo;
```

10) listar em ordem as quantidades de parcelas mais usadas em pagamentos

```
SELECT COUNT (IdPagamento) AS QtdPagamento, NumParcelas FROM
FormaPagamento GROUP BY NumParcelas ORDER BY QtdPagamento DESC;
```