

### *Exercícios de Revisão para a Prova 1*

#### **Consultas em Álgebra Relacional**

**Exercício 1** - Explique o que são as seguintes funções de um Sistema Gerenciador de Banco de Dados:

1. Garantir a consistência dos dados

#### **Resposta**

Um SGBD tem recursos que possibilitam a definição e a imposição de restrições de integridade sobre os BDs que mantém. Exemplos dessas restrições são as de tipo de dados, de chave e de integridade referencial. Além disso, o SGBD é responsável por garantir que os dados fiquem corretos mesmo quando ocorram falhas de software e hardware durante a execução de operações de atualização (seja desfazendo o efeito de operações iniciadas mas não finalizadas, ou reiniciando a execução de operações interrompidas).

2. Prover isolamento entre programas e dados

#### **Resposta**

Em um SGBD, a estrutura dos arquivos de dados de um BD é armazenada no catálogo do SGBD, separadamente dos programas que o acessam. Com isso, a estrutura do BD pode ser modificada sem que seja necessário modificar os programas que o acessam.

3. Controlar acessos concorrentes aos dados

#### **Resposta**

Um SGBD evita erros que podem ser causados quando mais de um usuário ou aplicação tenta atualizar um determinado item de dado ao mesmo tempo. O SGBD garante que o acesso aos dados ocorra de forma controlada, de modo que o resultado das atualizações seja sempre correto.

**Exercício 2** - Considere as seguintes relações:

Fornecedores(codf : inteiro, nome: string, endereco: string)

Pecas(codp: inteiro, nome: string, cor: string)

Catalogo(codf: inteiro, codp: inteiro, preco: real)

codf: é uma chave estrangeira para Fornecedores

codp: é uma chave estrangeira para Pecas

Os campos-chave encontram-se sublinhados, e o domínio de cada campo encontra-se listado após o nome do campo. Assim, codf é a chave de Fornecedores, codp é a chave de Peças e a composição de codf e codp formam a chave de catálogo. A relação Catalogo lista os preços cobrados por peças pelos Fornecedores. Escreva as seguintes consultas em álgebra relacional e cálculo relacional de tuplas. Obs.: As consultas de (m) a (o) só podem ser respondidas em álgebra relacional

#### **Resposta em Álgebra Relacional:**

- a) Encontre os nomes dos fornecedores que fornecem alguma peça vermelha.

$\pi_{\text{Fornecedores.nome}} ( \text{Fornecedores} * ( \text{Catalogo} * ( \sigma_{\text{cor} = \text{'vermelha'}} ( \text{Pecas} ) ) ) )$

b) Encontre os códigos dos fornecedores que fornecem alguma peça vermelha ou verde.

$$\pi_{\text{codf}}(\text{Catalogo} * (\sigma_{\text{cor} = \text{'vermelha'}} \text{ OR } \sigma_{\text{cor} = \text{'verde'}} (\text{Pecas})))$$

c) Encontre os códigos dos fornecedores que fornecem alguma peça vermelha ou que estão no endereço Av. Paulista, 572.

$$\pi_{\text{codf}}(\text{Catalogo} * (\sigma_{\text{cor} = \text{'vermelha'}} (\text{Pecas}))) \cup \pi_{\text{codf}}(\sigma_{\text{endereco} = \text{'Av. Paulista, 572'}} (\text{Fornecedores}))$$

d) Encontre os códigos dos fornecedores que fornecem alguma peça vermelha e alguma peça verde.

$$\pi_{\text{codf}}(\text{Catalogo} * (\sigma_{\text{cor} = \text{'vermelha'}} (\text{Pecas}))) \cap \pi_{\text{codf}}(\text{Catalogo} * (\sigma_{\text{cor} = \text{'verde'}} (\text{Pecas})))$$

e) Encontre os códigos dos fornecedores que fornecem todas as peças.

$$\pi_{\text{codf,codp}}(\text{Catalogo}) \div \pi_{\text{codp}}(\text{Pecas})$$

f) Encontre os códigos dos fornecedores que fornecem todas as peças vermelhas.

$$\pi_{\text{codf,codp}}(\text{Catalogo}) \div \pi_{\text{codp}}(\sigma_{\text{cor} = \text{'vermelha'}} (\text{Pecas}))$$

g) Encontre os códigos dos fornecedores que fornecem todas as peças vermelhas ou verdes.

$$\pi_{\text{codf,codp}}(\text{Catalogo}) \div \pi_{\text{codp}}(\sigma_{\text{cor} = \text{'vermelha'}} \text{ OR } \sigma_{\text{cor} = \text{'verde'}} (\text{Pecas}))$$

h) Encontre os códigos dos fornecedores que fornecem todas as peças vermelhas ou fornecem todas as peças verdes.

$$\pi_{\text{codf,codp}}(\text{Catalogo}) \div \pi_{\text{codp}}(\sigma_{\text{cor} = \text{'vermelha'}} (\text{Pecas})) \cup \pi_{\text{codf,codp}}(\text{Catalogo}) \div \pi_{\text{codp}}(\sigma_{\text{cor} = \text{'verde'}} (\text{Pecas}))$$

i) Encontre os pares de códigos de fornecedor em que o primeiro fornecedor do par cobre mais por alguma peça que o segundo fornecedor do par.

$$\pi_{\text{codf, codf2}}(\sigma_{\text{codp} = \text{codp2 AND } \text{preco} > \text{preco2}} (\text{Catalogo} \times (\rho_{\text{Catalogo2}(\text{codf2,codp2,preco2})} (\text{Catalogo}))))$$

j) Encontre os códigos das peças fornecidas por pelo menos dois fornecedores diferentes.

$$\pi_{\text{codp}}(\sigma_{\text{codp} = \text{codp2 AND } \text{codf} \neq \text{codf2}} (\text{Catalogo} \times (\rho_{\text{Catalogo2}(\text{codf2,codp2,preco2})} (\text{Catalogo}))))$$

k) Encontre os códigos das peças mais caras fornecidas pelo fornecedor chamado Bill Gates.

$$\pi_{\text{codp}}(\sigma_{\text{codf} = \text{codf\_bill AND } \text{preco} = \text{maior\_preco}} (\text{Catalogo} \times (\rho_{(\text{codf\_bill, maior\_preco})} (\text{codf } \mathcal{F}_{\text{MAX}} \text{ preco} (\text{Catalogo} * (\sigma_{\text{nome} = \text{'Bill Gates'}} (\text{Fornecedores}))))))$$

l) Encontre os códigos das peças fornecidas por todos os fornecedores por menos de R\$200,00. (Se algum fornecedor não fornece a peça ou cobra mais do R\$200,00 por ela, a peça não é selecionada).

$$(\pi_{\text{codp,codf}}(\sigma_{\text{preco} < 200} (\text{Catalogo})) \div \pi_{\text{codf}}(\text{Fornecedores}))$$

m) Encontre o número total de peças no catálogo.

$$\mathcal{F}_{\text{CONT}}_{\text{codp}}(\text{Catalogo})$$

- n) Para cada fornecedor, encontre a quantidade de peças vermelhas fornecidas e o preço médio dessas peças.

$$\text{codf } \mathcal{F}_{\text{CONT codp, MEDIA preco}} ( \text{Catalogo} * ( \sigma_{\text{cor} = \text{'vermelha'}} ( \text{Pecas} ) ) )$$

- o) Liste todos os pares [nome\_peça, nome\_fornecedor] tais que nome\_peça é o nome de uma peça fornecida pelo fornecedor de nome nome\_fornecedor. O nome de uma peça deve aparecer na listagem mesmo se não houve um fornecedor que a fornece (nesse caso, o nome do fornecedor deverá aparecer como NULL).

$$\pi_{\text{nome, nomef}}(\text{Peca} \bowtie_{\text{Peca.codp = Forn.codp}} ( \rho_{\text{Forn(codf, codp, nomef)}} ( \pi_{\text{codf, codp, nome}} ( \text{Fornecedores} * \text{Catalogo} ) ) ) )$$

### Resposta em Cálculo de Tuplas:

- a) Encontre os nomes dos fornecedores que fornecem alguma peça vermelha.

$$\{ \mathbf{f.nome} \mid \mathbf{Fornecedores(f)} \mathbf{AND} ( (\exists c)(\exists p)(\mathbf{Catalogo(c)} \mathbf{AND} \mathbf{Pecas(p)} \mathbf{AND} \mathbf{c.codf = f.codf} \mathbf{AND} \mathbf{c.codp = p.codp} \mathbf{AND} \mathbf{p.cor = 'vermelho'}) ) \}$$

- b) Encontre os códigos dos fornecedores que fornecem alguma peça vermelha ou verde.

$$\{ \mathbf{c.codf} \mid \mathbf{Catalogo(c)} \mathbf{AND} ((\exists p)(\mathbf{Pecas(p)} \mathbf{AND} \mathbf{c.codp = p.codp} \mathbf{AND} (\mathbf{p.cor = 'vermelho'} \mathbf{OR} \mathbf{p.cor = 'verde'})) ) \}$$

- c) Encontre os códigos dos fornecedores que fornecem alguma peça vermelha ou que estão no endereço Av. Paulista, 572.

$$\{ \mathbf{f.nome} \mid \mathbf{Fornecedores(f)} \mathbf{AND} ( \mathbf{f.endereco = 'Av. Paulista, 572'} \mathbf{OR} ( (\exists c)(\exists p)(\mathbf{Catalogo(c)} \mathbf{AND} \mathbf{Pecas(p)} \mathbf{AND} \mathbf{c.codf = f.codf} \mathbf{AND} \mathbf{c.codp = p.codp} \mathbf{AND} \mathbf{p.cor = 'vermelho'}) ) ) \}$$

- d) Encontre os códigos dos fornecedores que fornecem alguma peça vermelha e alguma peça verde.

$$\{ \mathbf{c1.codf} \mid \mathbf{Catalogo(c1)} \mathbf{AND} ((\exists p1)(\mathbf{Pecas(p1)} \mathbf{AND} \mathbf{c1.codp = p1.codp} \mathbf{AND} \mathbf{p1.cor = 'vermelho'}) \mathbf{AND} ((\exists c2)(\exists p2)(\mathbf{Catalogo(c2)} \mathbf{AND} \mathbf{Pecas(p2)} \mathbf{AND} \mathbf{c2.codf = c1.codf} \mathbf{AND} \mathbf{c2.codp = p2.codp} \mathbf{AND} \mathbf{p2.cor = 'verde'})) ) \}$$

- e) Encontre os códigos dos fornecedores que fornecem todas as peças.

$$\{ \mathbf{f.codf} \mid \mathbf{Fornecedores(f)} \mathbf{AND} ( (\forall p)( \mathbf{NOT} \mathbf{Pecas(p)} \mathbf{OR} ((\exists c)( \mathbf{Catalogo(c)} \mathbf{AND} \mathbf{c.codf = f.codf} \mathbf{AND} \mathbf{c.codp = p.codp} ) ) ) ) \}$$

ou

$$\{ \mathbf{f.codf} \mid \mathbf{Fornecedores(f)} \mathbf{AND} ( \mathbf{NOT}(\exists p)( \mathbf{Pecas(p)} \mathbf{AND} \mathbf{NOT}((\exists c)( \mathbf{Catalogo(c)} \mathbf{AND} \mathbf{c.codf = f.codf} \mathbf{AND} \mathbf{c.codp = p.codp} ) ) ) ) \}$$

- f) Encontre os códigos dos fornecedores que fornecem todas as peças vermelhas.

$\{ f.codf \mid \text{Fornecedores}(f) \text{ AND } ( (\forall p)( \text{NOT } \text{Pecas}(p) \text{ OR } \text{NOT } (p.cor = 'vermelha') \text{ OR } ( (\exists c)( \text{Catalogo}(c) \text{ AND } c.codf = f.codf \text{ AND } c.codp = p.codp) ) ) ) \}$

ou

$\{ f.codf \mid \text{Fornecedores}(f) \text{ AND } ( \text{NOT}(\exists p)( \text{Pecas}(p) \text{ AND } p.cor = 'vermelha' \text{ AND } \text{NOT} ( (\exists c)( \text{Catalogo}(c) \text{ AND } c.codf = f.codf \text{ AND } c.codp = p.codp) ) ) ) \}$

g) Encontre os códigos dos fornecedores que fornecem todas as peças vermelhas ou verdes.

$\{ f.codf \mid \text{Fornecedores}(f) \text{ AND } ( (\forall p)( \text{NOT } \text{Pecas}(p) \text{ OR } \text{NOT } (p.cor = 'vermelha' \text{ OR } p.cor = 'verde') \text{ OR } ( (\exists c)( \text{Catalogo}(c) \text{ AND } c.codf = f.codf \text{ AND } c.codp = p.codp) ) ) ) \}$

ou

$\{ f.codf \mid \text{Fornecedores}(f) \text{ AND } ( \text{NOT}(\exists p)( \text{Pecas}(p) \text{ AND } (p.cor = 'vermelha' \text{ OR } p.cor = 'verde') \text{ AND } \text{NOT} ( (\exists c)( \text{Catalogo}(c) \text{ AND } c.codf = f.codf \text{ AND } c.codp = p.codp) ) ) ) \}$

h) Encontre os códigos dos fornecedores que fornecem todas as peças vermelhas ou fornecem todas as peças verdes.

$\{ f.codf \mid \text{Fornecedores}(f) \text{ AND } ( ( (\forall p1)( \text{NOT } \text{Pecas}(p1) \text{ OR } \text{NOT } (p1.cor = 'vermelha') \text{ OR } ( (\exists c1)( \text{Catalogo}(c1) \text{ AND } c1.codf = f.codf \text{ AND } c1.codp = p1.codp) ) ) ) \text{ OR } ( (\forall p2)( \text{NOT } \text{Pecas}(p2) \text{ OR } \text{NOT } (p2.cor = 'verde') \text{ OR } ( (\exists c2)( \text{Catalogo}(c2) \text{ AND } c2.codf = f.codf \text{ AND } c2.codp = p2.codp) ) ) ) ) \}$

ou

$\{ f.codf \mid \text{Fornecedores}(f) \text{ AND } ( ( \text{NOT}(\exists p1)( \text{Pecas}(p1) \text{ AND } p1.cor = 'vermelha' \text{ AND } \text{NOT} ( (\exists c1)( \text{Catalogo}(c1) \text{ AND } c1.codf = f.codf \text{ AND } c1.codp = p1.codp) ) ) ) \text{ OR } ( \text{NOT}(\exists p2)( \text{Pecas}(p2) \text{ AND } p2.cor = 'verde' \text{ AND } \text{NOT} ( (\exists c2)( \text{Catalogo}(c2) \text{ AND } c2.codf = f.codf \text{ AND } c2.codp = p2.codp) ) ) ) ) \}$

i) Encontre os pares de códigos de fornecedor em que o primeiro fornecedor do par cobre mais por alguma peça que o segundo fornecedor do par.

$\{ c1.codf, c2.codf \mid \text{Catalogo}(c1) \text{ AND } \text{Catalogo}(c2) \text{ AND } c1.codf \neq c2.codf \text{ AND } c1.codp = c2.codp \text{ AND } c1.preco > c2.preco \}$

j) Encontre os códigos das peças fornecidas por pelo menos dois fornecedores diferentes.

$\{ c1.codp \mid \text{Catalogo}(c1) \text{ AND } ( (\exists c2)( \text{Catalogo}(c2) \text{ AND } c1.codf \neq c2.codf \text{ AND } c1.codp = c2.codp ) ) \}$

$c1.codp = c2.codp ) )}$

k) Encontre os códigos das peças mais caras fornecidas pelo fornecedor chamado Bill Gates.

$\{ c1.codp \mid \text{Catalogo}(c1) \text{ AND } ((\exists f1)( \text{Fornecedores}(f1) \text{ AND } c1.codf = f1.codf \text{ AND } f1.nome = 'Bill Gates' ) ) \text{ AND NOT}( (\exists c2) (\exists f2)( \text{Catalogo}(c2) \text{ AND } \text{Fornecedores}(f2) \text{ AND } c2.codf = f2.codf \text{ AND } f2.nome = 'Bill Gates' \text{ AND } c2.preco > c1.preco) ) ) \}$

l) Encontre os códigos das peças fornecidas por todos os fornecedores por menos de R\$200,00 . (Se algum fornecedor não fornece a peça ou cobra mais do R\$200,00 por ela, a peça não é selecionada).

$\{ p.codp \mid \text{Peca}(p) \text{ AND } ( (\forall f)( \text{NOT } \text{Fornecedores}(f) \text{ OR } ( (\exists c)( \text{Catalogo}(c) \text{ AND } c.codf = f.codf \text{ AND } c.codp = p.codp \text{ AND } c.preco \leq 200 ) ) ) ) \}$

ou

$\{ p.codp \mid \text{Peca}(p) \text{ AND } ( \text{NOT}(\exists f)( \text{Fornecedores}(f) \text{ AND } \text{NOT} ( (\exists c)( \text{Catalogo}(c) \text{ AND } c.codf = f.codf \text{ AND } c.codp = p.codp \text{ AND } c.preco \leq 200 ) ) ) ) \}$

m) Encontre o número total de peças no catálogo.

**Não pode ser resolvido no cálculo de tuplas.**

n) Para cada fornecedor, encontre a quantidade de peças vermelhas fornecidas e o preço médio dessas peças.

**Não pode ser resolvido no cálculo de tuplas.**

o) Liste todos os pares [nome\_peça, nome\_fornecedor] tais que nome\_peça é o nome de uma peça fornecida pelo fornecedor de nome nome\_fornecedor. O nome de uma peça deve aparecer na listagem mesmo se não houve um fornecedor que a fornece (nesse caso, o nome do fornecedor deverá aparecer como NULL).

**Não pode ser resolvido no cálculo de tuplas.**

---

## Projeto Conceitual de BD

**Exercício 3** - Usando o modelo Entidade-Relacionamento Estendido, construa um diagrama de esquema conceitual para o BD de uma empresa locadora de veículos.

A empresa aluga automóveis e camionetas de carga. Um veículo de locação é identificado por sua placa, mas possui outras informações importantes como número do chassi, cor, modelo e marca. Além disso, todo veículo é classificado de acordo com uma tabela de tipos da locadora. Por exemplo, o tipo A3 corresponde a automóveis pequenos, de quatro portas e com ar-condicionado, enquanto o tipo C4 corresponde a camionetas com capacidade de carga de até 1 tonelada.

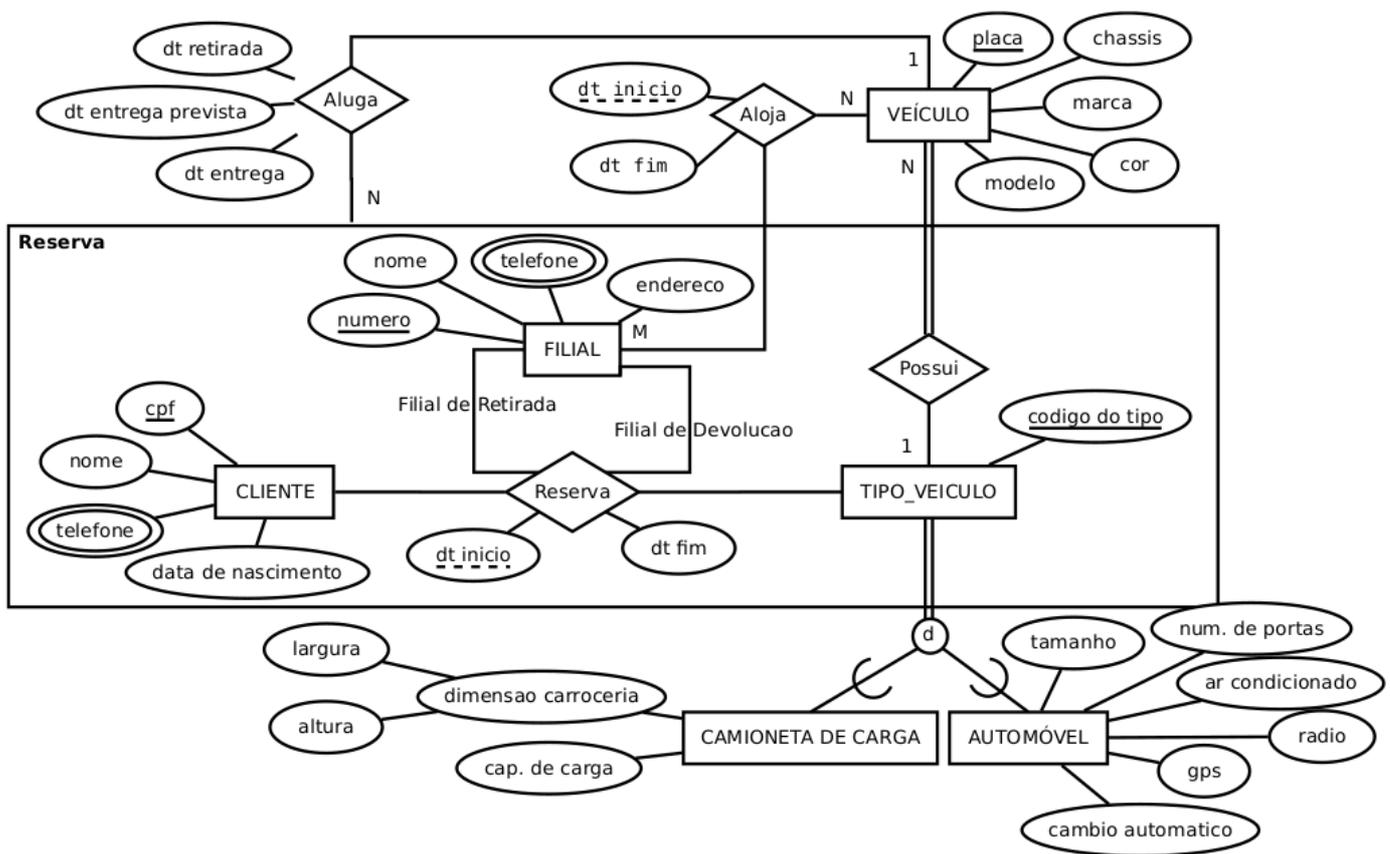
Um tipo de automóvel define o tamanho (pequeno, médio ou grande), o número de portas e os acessórios disponíveis, que podem ser: ar-condicionado, rádio, GPS e câmbio automático. Já um tipo de camioneta de carga define a capacidade de carga da camioneta e a dimensão (largura e altura) da sua carroceria.

Para alugar um veículo, um cliente primeiro faz uma reserva (por telefone ou no *site* da locadora) onde define as datas de início e término da locação, o tipo de veículo que deseja alugar, a filial de retirada e a filial de devolução do veículo. E somente na retirada do veículo o cliente é “associado” a um veículo real, disponível na filial de retirada. A locadora tem uma grande rede de filiais. De suas filiais, a locadora registra nome, número de identificação, endereço e telefones de contato. E de seus clientes, a locadora registra nome, data de nascimento, telefones e CPF.

A partir do banco de dados, devemos poder obter as seguintes informações: (i) a lista dos veículos disponíveis em cada filial na data corrente; (ii) a lista de reservas de locação ainda não atendidas em cada filial; (iii) os veículos presentemente alugados por cada filial, as suas respectivas filiais de entrega (caso sejam diferentes das de locação) e datas de entrega previstas; (iv) os cliente, as datas e locais de retirada e entrega de todas as locações já feitas na locadora.

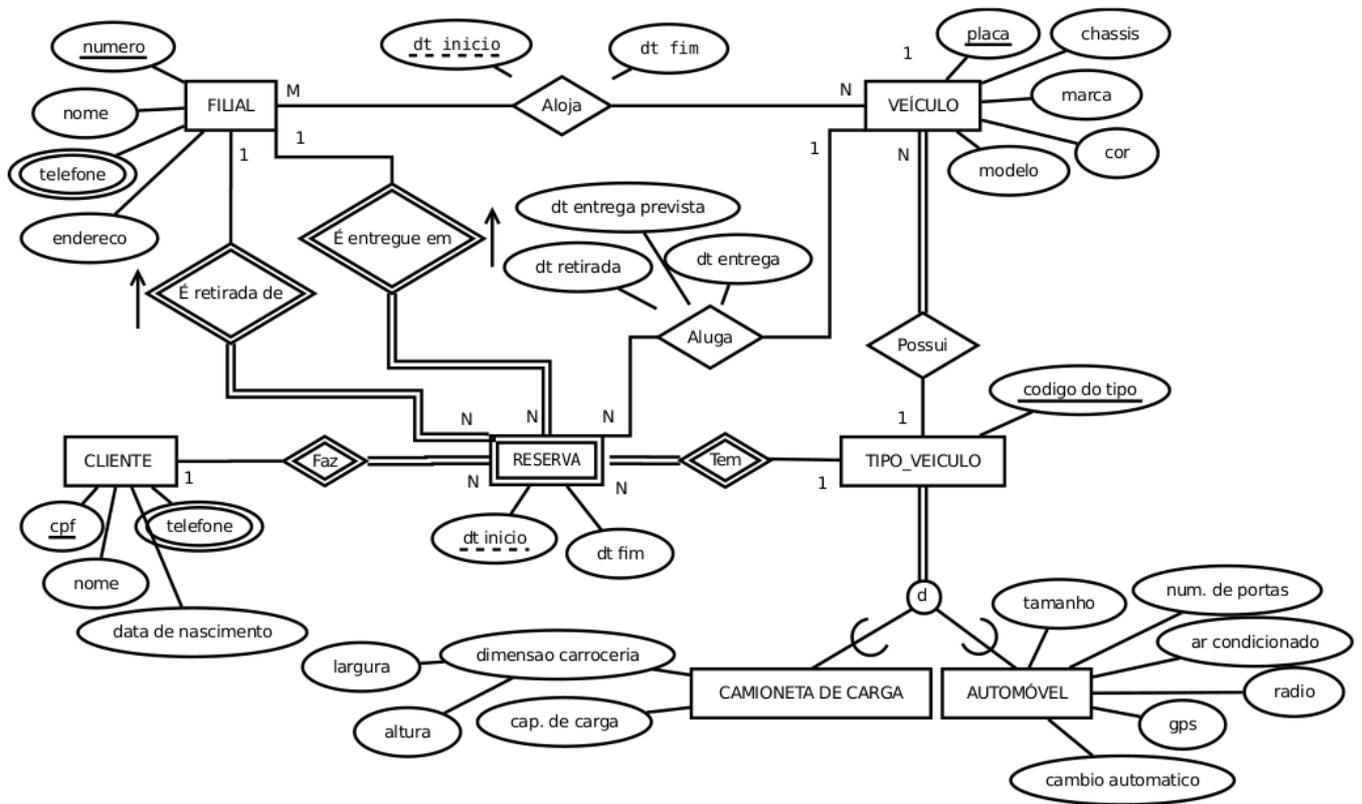
Não se esqueça de incluir no seu modelo as indicações das chaves para as entidades e das restrições de participação e de cardinalidade para os relacionamentos.

### Uma possível resposta:



Obs.: Não foi indicada cardinalidade para o tipo de relacionamento Reserva (entre CLIENTE, TIPO\_VEICULO, FILIAL E FILIAL) porque ele não é binário.

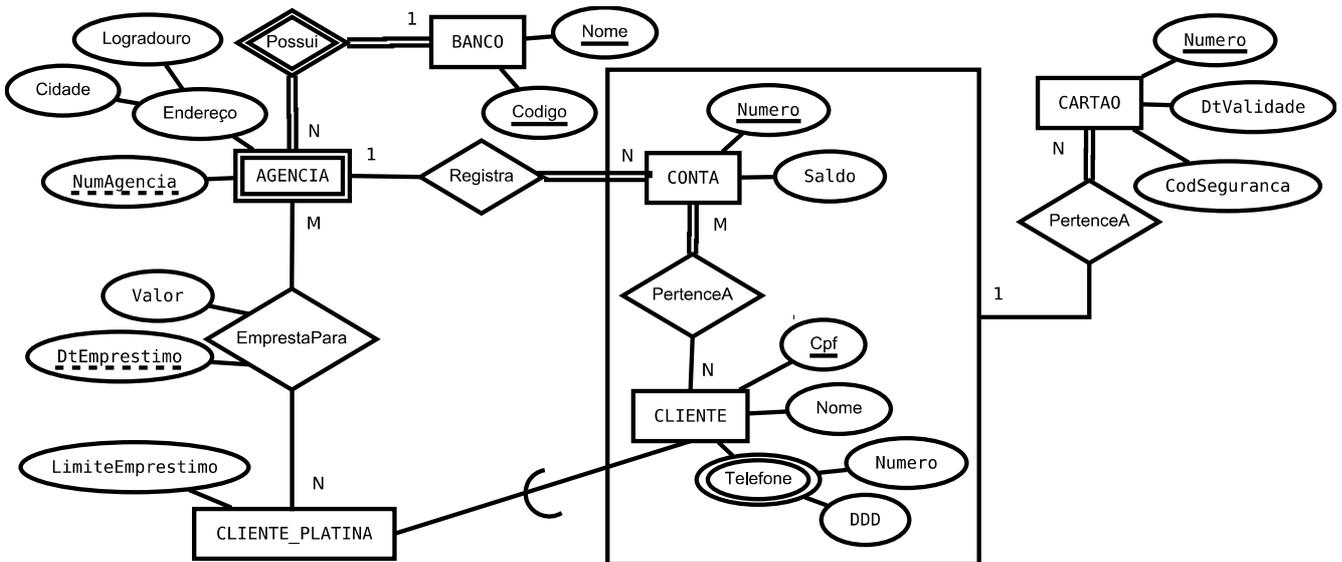
**Outra possível resposta:**



Obs.: Não foi indicada cardinalidade para o tipo de relacionamento Reserva (entre CLIENTE, TIPO\_VEICULO, FILIAL E FILIAL) porque ele não é binário.

**Projeto Lógico de BD**

**Exercício 4** - Crie um esquema de banco de dados relacional para o esquema EER a seguir. Não se esqueça de indicar no seu esquema as chaves primárias, chaves estrangeiras e suas respectivas ações de propagação.



**Resposta:**

Notação simplificada para as ações de propagação das chaves estrangeiras:

- RR: na Remoção, Restrinja (*restrict*)
- RC: na Remoção, Cascadeie (*cascade*)
- AR: na Alteração, Restrinja (*restrict*)
- AC: na Alteração, Cascadeie (*cascade*)

Mapeamento:

Nome(codBanco, nome)

---> nome é uma chave única

Agencia(codBanco, numAgencia, logradouro, cidade)

---> codBanco é uma chave estrangeira para Banco(codigo); RC

Cliente(cpf, nome)

Telefone(cpfCliente, ddd, numero)

---> cpfCliente é uma chave estrangeira para Cliente(cpf); RC

ClientePlatina(cpfCliente, limiteEmprestimo)

---> cpfCliente é uma chave estrangeira para Cliente(cpf); RC

Conta(numero, saldo, codBanco, numAgencia)

---> (codBanco, numAgencia) é uma chave estrangeira para

Agencia(codBanco, numAgencia); RR

PertenceA(numConta, cpfCliente)

---> numConta é uma chave estrangeira para Conta(numero); RC

---> cpfCliente é uma chave estrangeira para Cliente(cpf); RR

EmprestaPara(codBanco, numAgencia, cpfClientePlatina, dtEmprestimo,  
valor)

---> (codBanco, numAgencia) é uma chave estrangeira para

Agencia(codBanco, numAgencia); RR

---> cpfClientePlatina é uma chave estrangeira para

ClientePlatina(cpfCliente); RR

Cartao(numero, dtValidade, codSeguranca, numConta, cpfCliente)

---> (numConta, cpfCliente) é uma chave estrangeira para

PertenceA(numConta, cpfCliente); RC