

Aula 023

Professores:

Geraldo Xexéo
Geraldo Zimbrão

Conteúdo:

Exercício - Conversão OO
Relacional e SQL

⇒ Sistema de Registro de Multas

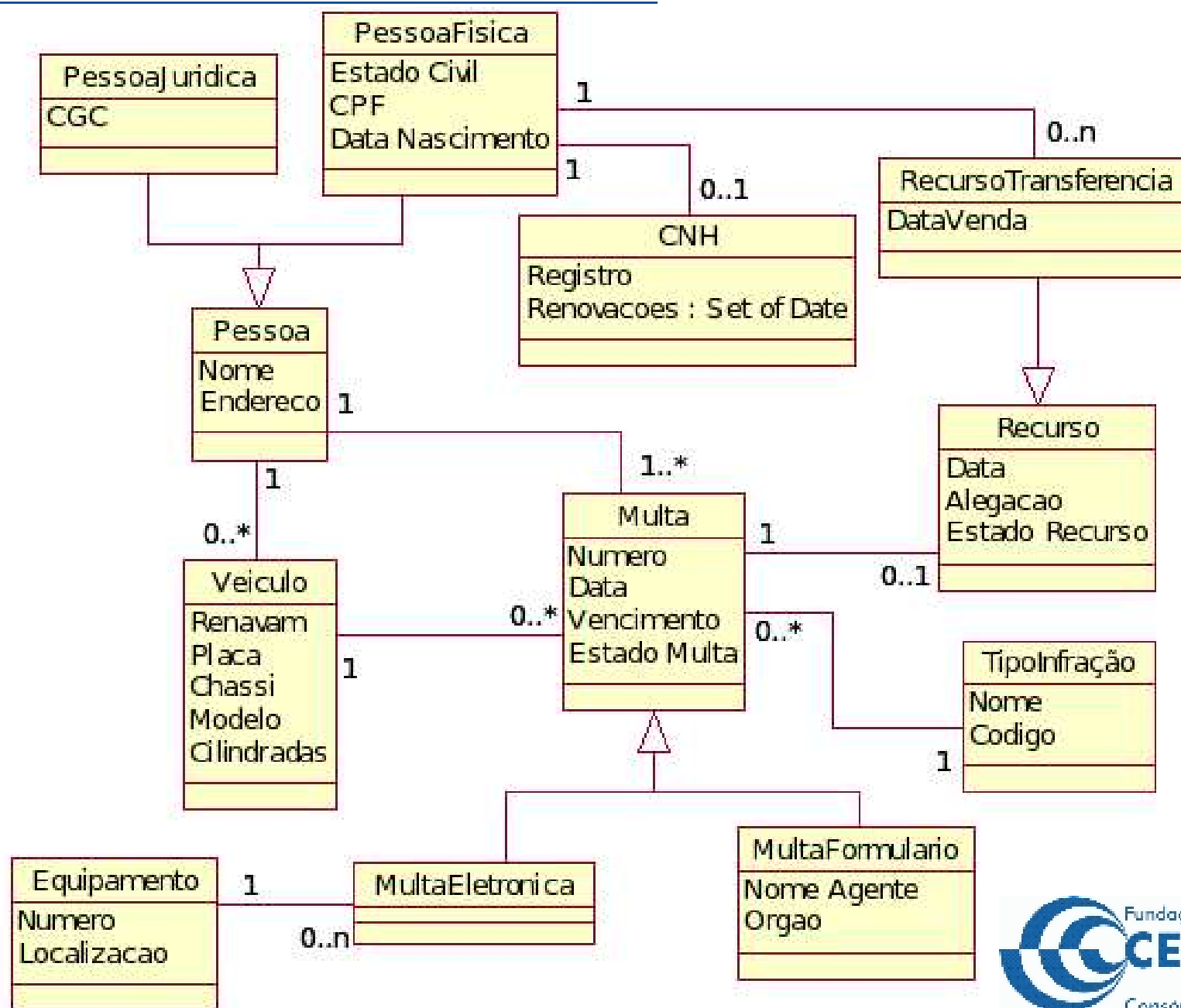
Descrição do Modelo de Classes

- ➡ O sistema deve controlar multas aplicadas por formulário ou por meios eletrônicos
 - ➡ As multas aplicadas eletronicamente devem ter a indicação do aparelho que a aplicou
- ➡ Há um cadastro de pessoas físicas e jurídicas com os respectivos veículos
- ➡ Uma multa é aplicada a um veículo e a um condutor
 - ➡ As pessoas jurídicas devem indicar quem era o condutor do veículo na hora da multa
- ➡ Há a possibilidade de dois tipos de recursos
 - ➡ Transferência de responsabilidade
 - ➡ Contestação da multa

Descrição do Modelo de Classes

- ➡ De uma multa devemos saber o número, a data, o vencimento, o estado (sem condutor, emitida, paga, em recurso ou cancelada) e o tipo de infração
 - ▢ As multas somente podem ser pagas após a identificação do condutor
- ➡ Das pessoas físicas guardamos a CNH e as datas de sua renovação
- ➡ Os atributos principais estão no modelo

Modelo de Classes



Classes a Serem Mapeadas

Mapeamento Direto

- ▢ Veículo
- ▢ Equipamento
- ▢ Tipo de Infração
- ▢ CNH

Mapeamento Complexo

- ▢ Pessoa (física e jurídica)
- ▢ Multa (eletrônica e com formulário)
- ▢ Recurso (transferência e contestação)
- ▢ Renovações
- ▢ Tipos Enumerados

Tipos Enumerados

```
Create table TiposEnumerados  
(  
  ID integer not null primary key,  
  Tipo varchar(32) not null,  
  Valor varchar(32) not null  
);
```

Classes CNH e Renovações

Create table CNH

```
(  
  ID integer not null primary key,  
  Registro char(20) not null unique  
);
```

Create table RenovacoesCNH

```
(  
  CNH_ID integer not null references CNH(ID)  
    on delete cascade,  
  Data date not null,  
  unique( CNH_ID, Data )  
);
```

Classes Pessoa Física e Jurídica

➡ Uma hierarquia de classes

- ➡ Pessoa, pessoa física e pessoa jurídica
- ➡ Vimos três opções de modelagem:
 - ➡ Uma única tabela
 - ➡ Fragmentação horizontal
 - ➡ Fragmentação vertical
- ➡ Iremos adotar a fragmentação vertical
 - ➡ Mais fácil de controlar os relacionamentos que só existem para pessoa física

Classes Pessoa Física e Jurídica

```
Create table Pessoa (  
  ID integer not null primary key,  
  Nome varchar(50) not null,  
  Endereco varchar(60) not null  
);
```

```
Create table PessoaFisica (  
  ID integer not null primary key references Pessoa on delete cascade,  
  CPF char(8) not null unique,  
  DataNascimento date not null,  
  EstadoCivil integer not null references TiposEnumerados(id),  
  CNH_ID integer references CNH(id)  
);
```

```
Create table PessoaJuridica (  
  ID integer not null primary key references Pessoa on delete cascade,  
  CGC char(12) not null unique  
);
```

Veículo

Create table Veiculo

```
(  
  ID integer not null primary key,  
  Renavam char(9) not null unique,  
  Placa char(7) not null unique,  
  Chassi char(18) not null unique,  
  Modelo varchar(40) not null,  
  Cilindradas integer not null,  
  Proprietario integer not null references Pessoa(id)  
);
```

Equipamento

```
Create table Equipamento  
(  
  ID integer not null primary key,  
  Numero varchar(32) not null unique,  
  Localizacao varchar(60) not null  
);
```

Tipo de Infração

```
Create table TipoInfracao  
(  
  ID integer not null primary key,  
  Nome varchar(50) not null,  
  Codigo varchar(32) not null unique  
);
```

Classes Multa Eletrônica e Multa com Formulário

➡ Nessa hierarquia há pouca diferença entre um tipo de multa e o outro

- ➡ Fácil manter os relacionamentos em qualquer abordagem
 - ➡ Apenas o Equipamento está relacionado com a subclasse Multa Eletrônica
 - ➡ É um relacionamento 1 para n, com a ponta 1 em Multa

Classes Multa Eletrônica e Multa com Formulário

```
Create table Multa (  
  ID integer not null primary key,  
  Tipo char(1) not null,  
  Numero varchar(32) not null unique,  
  Data date not null,  
  Vencimento date not null,  
  Veiculo_ID integer not null references Veiculo(id),  
  Responsavel_ID integer not null references Pessoa(id),  
  TipoInfracao_ID integer not null references TipoInfracao(id),  
  Estado integer not null references TiposEnumerados(id),  
  NomeAgente varchar(50),  
  Orgao varchar(50),  
  Equipamento_ID integer references Equipamento(id),  
  check( Tipo in ('E', 'F') ),  
  check( (Tipo = 'F' and NomeAgente is not null and Orgao is not null  
          and Equipamento_ID is null ) or  
          (Tipo = 'E' and NomeAgente is null and Orgao is null  
          and Equipamento_ID is not null ) )  
);
```

Classes Recurso e Recurso de Transferência

- ➡ Hierarquia muito semelhante entre si
 - ➡ Atributos fáceis de implementar
 - ➡ Relacionamentos idem
- ➡ Usaremos uma tabela única também

Classes Recurso e Recurso de Transferência

Create table Recurso

```
(  
  ID integer not null primary key,  
  Tipo char(1) not null,  
  Data date not null,  
  Alegacao varchar(4000) not null,  
  Estado integer not null references TiposEnumerados(id),  
  Multa_ID integer not null references Multa(id),  
  PessoaFisica_ID integer references PessoaFisica(id),  
  DataVenda date,  
  check( Tipo in ( 'R', 'T' ) ),  
  check( (Tipo = 'R' and DataVenda is null  
          and PessoaFisica_ID is null ) or  
          (Tipo = 'T' and DataVenda is not null  
          and PessoaFisica_ID is not null ) )  
);
```


Exemplo de Consulta - 1

➡ Dado um CPF, listar todas as multas onde essa pessoa é o responsável

➡ Iremos chamar o CPF dado de ":PARAM"

```
SELECT M.NUMERO  
FROM PESSOAFISICA PF, PESSOA P, MULTA M  
WHERE PF.CPF = :PARAM  
AND PF.ID = P.ID  
AND M.RESPONSAVEL_ID = P.ID
```

Exemplo de Consulta - 2

- ➡ Dado um CGC, listar os números das multas pagas, o tipo da infração, as placas dos veículos e os nomes dos responsáveis, ordenando pelo responsável
- ▢ Iremos chamar o CGC dado de ":PARAM"

Por partes - 1



Listar todas as multas de um dado CGC

- ⇒ Há um relacionamento entre Pessoa Juridica e Pessoa, e de Multa para Veiculo, e de Veiculo para Pessoa
- ⇒ O relacionamento entre Multa e Pessoa é o responsável pela Multa

```
SELECT M.NUMERO, V.PLACA  
FROM PESSOAJURIDICA PJ, PESSOA P, VEICULO V, MULTA M  
WHERE PJ.CGC = :PARAM  
AND PJ.ID = P.ID  
AND V.PROPRIETARIO = P.ID  
AND M.VEICULO_ID = V.ID
```

Por partes - 2

➡ Listar todas as multas pagas de um dado CGC

⇒ Usar a tabela de Tipos enumerados

```
SELECT M.NUMERO, V.PLACA  
FROM PESSOA JURIDICA PJ, PESSOA P, VEICULO V, MULTA M,  
    TIPOSENUMERADOS TE  
WHERE PJ.CGC = :PARAM  
    AND PJ.ID = P.ID  
    AND V.PROPRIETARIO = P.ID  
    AND M.VEICULO_ID = V.ID  
    AND M.ESTADO = TE.ID  
    AND TE.VALOR = 'PAGA'
```

Por partes - 3

➡ Incluir o tipo da infração

⇒ Usar o relacionamento entre Multa e Tipo de Infração

```
SELECT M.NUMERO, TI.CODIGO, V.PLACA  
FROM PESSOA JURIDICA PJ, PESSOA P, VEICULO V, MULTA M,  
    TIPOSENUMERADOS TE, TIPOINFRACAO TI  
WHERE PJ.CGC = :PARAM  
    AND PJ.ID = P.ID  
    AND V.PROPRIETARIO = P.ID  
    AND M.VEICULO_ID = V.ID  
    AND M.ESTADO = TE.ID  
    AND M.TIPOINFRACAO_ID = TI.ID  
    AND TE.VALOR = 'PAGA'
```

Por partes - 4

➡ Incluir o responsável

⇒ Usar o relacionamento entre Multa e Pessoa

```
SELECT M.NUMERO, TI.CODIGO, V.PLACA, R.NOME
FROM PESSOA JURIDICA PJ, PESSOA P, VEICULO V, MULTA M,
     TIPOSENUMERADOS TE, TIPOINFRACAO TI, PESSOA R
WHERE PJ.CGC = :PARAM
      AND PJ.ID = P.ID
      AND V.PROPRIETARIO = P.ID
      AND M.VEICULO_ID = V.ID
      AND M.ESTADO = TE.ID
      AND M.TIPOINFRACAO_ID = TI.ID
      AND M.RESPONSAVEL_ID = R.ID
      AND TE.VALOR = 'PAGA'
ORDER BY R.NOME
```