

Esquema Relacional

CLIENTES(Cliente_ID (PK), Nome, NIF, Morada, Telefone, Email)

AGENCIA(Agencia_ID (PK), Nome, Localizacao, Contacto)

FUNCIONARIOS(Funcionario_ID (PK), Nome, Telefone, Email, Agencia_ID (FK))

VEICULOS(Veiculo_ID (PK), Matricula, Marca, Modelo, Ano, Quilometragem, Cor, PrecoAluguer, Estado, Categoria, Agencia_ID (FK))

ALUGUERES(Aluguer_ID (PK), DataInicio, DataFim, PrecoTotal, Cliente_ID (FK), Veiculo_ID (FK), Funcionario_ID (FK))

NORMALIZAÇÃO

✓ Etapa 1 — Verificação da 1ª Forma Normal (1FN)

📖 Definição da 1FN:

Um esquema relacional está na **Primeira Forma Normal (1FN)** se:

- Todos os atributos contêm **valores atômicos** (não listas, nem campos compostos).
- Não há grupos repetidos nem multivalorados.

🔍 Vamos analisar cada entidade:

1. CLIENTES(Cliente_ID, Nome, NIF, Morada, Telefone, Email)

✓ Todos os atributos são **atômicos**.

♦ **Cumpre a 1FN**

2. AGENCIA(Agencia_ID, Nome, Localizacao, Contacto)

✓ Tudo atômico.

- ♦ **Cumpra a 1FN**
-

3. FUNCIONARIOS(Funcionario_ID, Nome, Telefone, Email, Agencia_ID)

✓ Tudo atômico.

- ♦ **Cumpra a 1FN**
-

4. VEICULOS(...)

Atributos críticos:

- **Estado**: {"Disponível", "Alugado", "Em manutenção"}
- **Categoria**: {"Citadino", "SUV", etc.}

✗ Problema identificado:

- **Estado** e **Categoria** são **valores multivalorados e repetitivos**, o que:
 - **viola a atomicidade**;
 - **repete informação textual** em várias linhas.

🔧 **Solução**: Criar entidades separadas:

ESTADOS(Estado_ID, Designacao)

CATEGORIAS(Categoria_ID, Designacao)

E no **VEICULOS**, substituir:

Estado_ID (FK), Categoria_ID (FK)

- ♦ **Com isto, VEICULOS passa a cumprir a 1FN**
-

5. ALUGUERES(. . .)

✓ Todos os campos são atômicos (datas, preços, FK simples).

- ♦ **Cumpra a 1FN**
-

CLIENTES(Cliente_ID (PK), Nome, NIF, Morada, Telefone, Email)

AGENCIA(Agencia_ID (PK), Nome, Localizacao, Contacto)

FUNCIONARIOS(Funcionario_ID (PK), Nome, Telefone, Email, Agencia_ID (FK))

ESTADOS(Estado_ID (PK), Designacao)

CATEGORIAS(Categoria_ID (PK), Designacao)

VEICULOS(Veiculo_ID (PK), Matricula, Marca, Modelo, Ano, Quilometragem, Cor, PrecoAluguer, Estado_ID (FK), Categoria_ID (FK), Agencia_ID (FK))

ALUGUERES(Aluguer_ID (PK), DataInicio, DataFim, PrecoTotal, Cliente_ID (FK), Veiculo_ID (FK), Funcionario_ID (FK))

✓ Etapa 2 — Verificação da 2ª Forma Normal (2FN)

■ Definição da 2FN:

Um esquema está na **Segunda Forma Normal** se:

- Está na **1FN**
 - E **todos os atributos não-chave dependem da totalidade da chave primária** (não pode haver dependência parcial)
-

Nota:

- A **dependência parcial** só acontece em **tabelas com chaves compostas**
 - Se a tabela tiver **chave primária simples**, ela **cumpe automaticamente a 2FN**
-

Análise entidade por entidade:

1. CLIENTES(Cliente_ID (PK), Nome, NIF, Morada, Telefone, Email)

- Chave primária simples: **Cliente_ID**
- Todos os outros atributos dependem de **Cliente_ID**

 **Cumpe 2FN**

2. AGENCIA(Agencia_ID (PK), Nome, Localizacao, Contacto)

- Chave simples: **Agencia_ID**
- Tudo depende diretamente da PK

 **Cumpe 2FN**

3. FUNCIONARIOS(Funcionario_ID (PK), Nome, Telefone, Email, Agencia_ID (FK))

- PK simples: **Funcionario_ID**
- Tudo depende da PK

 **Cumpe 2FN**

4. ESTADOS(Estado_ID (PK), Designacao)

5. CATEGORIAS(Categoria_ID (PK), Designacao)

- Tabelas de referência → PK simples

✓ Ambas cumprem 2FN

6. VEICULOS(Veiculo_ID (PK), Matricula, Marca, Modelo, Ano, Quilometragem, Cor, PrecoAluguer, Estado_ID, Categoria_ID, Agencia_ID)

- Chave simples: Veiculo_ID
- Nenhum atributo depende apenas de parte da chave (não há chaves compostas)

✓ Cumpre 2FN

7. ALUGUERES(Aluguer_ID (PK), DataInicio, DataFim, PrecoTotal, Cliente_ID, Veiculo_ID, Funcionario_ID)

- Chave simples: Aluguer_ID
- Tudo depende diretamente da chave

✓ Cumpre 2FN

✓ Etapa 3 — Verificação da 3ª Forma Normal (3FN)

📖 Definição da 3FN:

Um esquema relacional está na Terceira Forma Normal se:

- Está na 2FN, e
 - Todos os atributos não-chave dependem apenas da chave primária, e não de outros atributos não-chave (ou seja, não há dependências transitivas)
-

🔍 Verificação entidade por entidade:

1. CLIENTES(Cliente_ID (PK), Nome, NIF, Morada, Telefone, Email)

- Nenhum atributo depende de outro que não seja **Cliente_ID**.
- **NIF** é único, mas está apenas como atributo (poderia até ser alternativo da PK).

✓ Cumpre 3FN

2. AGENCIA(Agencia_ID (PK), Nome, Localizacao, Contacto)

- Tudo depende diretamente de **Agencia_ID**
- **Localizacao** ou **Contacto** não geram dependências transitivas

✓ Cumpre 3FN

3. FUNCIONARIOS(Funcionario_ID (PK), Nome, Telefone, Email, Agencia_ID (FK))

- **Agencia_ID** é uma FK, e não é transitiva (serve para ligação)
- Tudo depende diretamente de **Funcionario_ID**

✓ Cumpre 3FN

4. ESTADOS(Estado_ID (PK), Designacao)

5. CATEGORIAS(Categoria_ID (PK), Designacao)

- Atributos descritivos simples
- Sem dependências transitivas

✓ Ambas cumprem 3FN

6. VEICULOS(...)

- Todos os atributos (como **Marca**, **Ano**, **PrecoAluguer**) dependem diretamente de **Veiculo_ID**
- **Estado_ID** e **Categoria_ID** são FK (já separados como entidades)

✓ Cumpre 3FN

7. ALUGUERES(...)

- Nenhum atributo depende de outro não-chave
- Tudo está diretamente ligado à PK **Aluguer_ID**

✓ Cumpre 3FN

