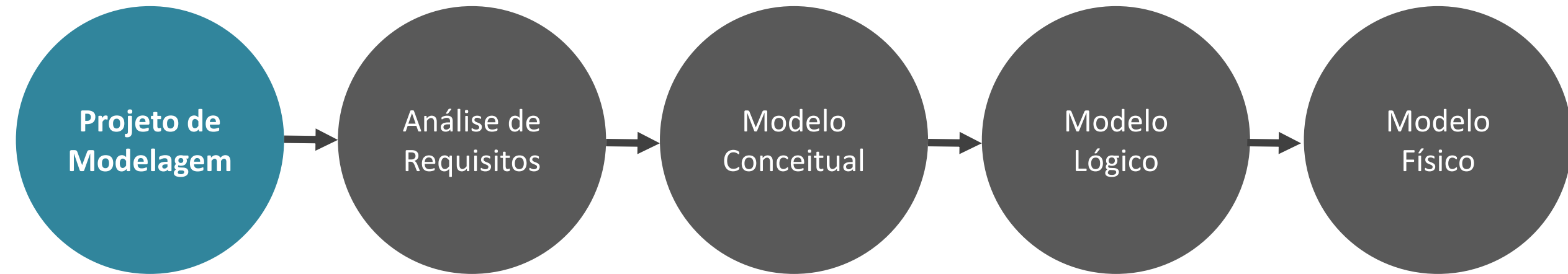


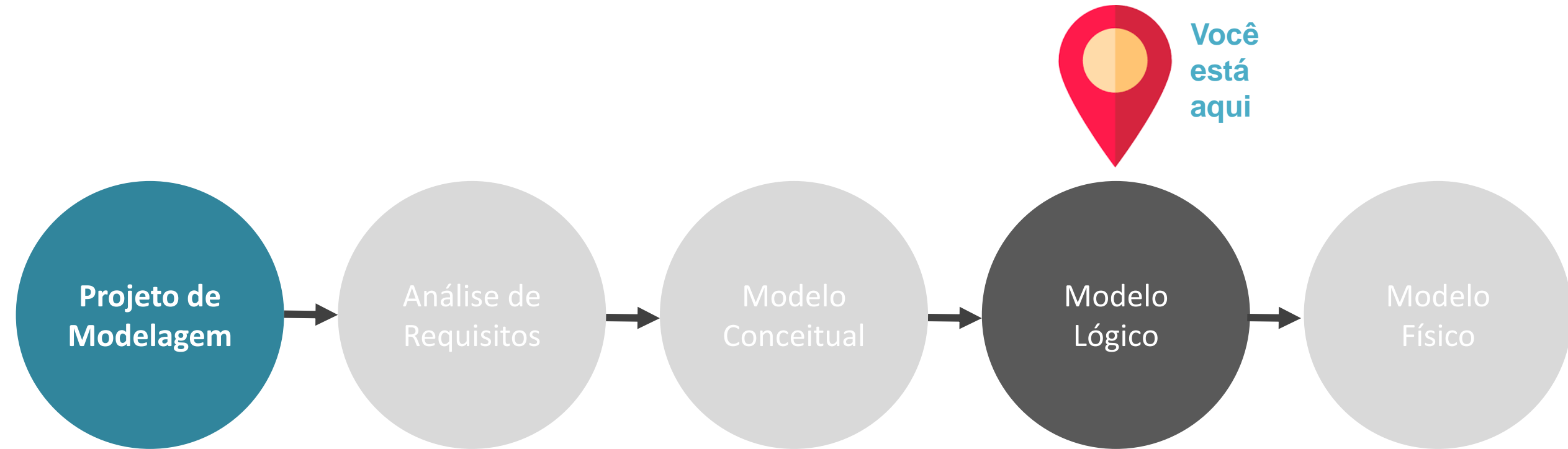
# PROJETO DE BANCO DE DADOS

## Modelo Lógico

# Quais são as etapas de um Projeto de Modelagem?



# Quais são as etapas de um Projeto de Modelagem?



Nesta aula vamos falar sobre a terceira delas, o **Modelo Lógico**!

# Importância da modelagem lógica

A **modelagem lógica** trata da definição da **estrutura do banco de dados** onde serão armazenados os dados, descrevendo detalhadamente suas **tabelas, campos e relacionamentos**. O modelo lógico de banco de dados tem muita importância como documentação técnica independente do SGBD, além de oferecer outras vantagens.

# Sobre o que vamos falar?

- **Modelo lógico** em modelagem de banco de dados.
- **Chaves** no modelo relacional (**chaves candidatas, chaves primárias, chaves alternadas e chaves estrangeiras**).
- **Restrições de integridade**
- **Dicionário de dados**
- **Anomalias em bancos de dados**
- **Formas normais**

# Introdução

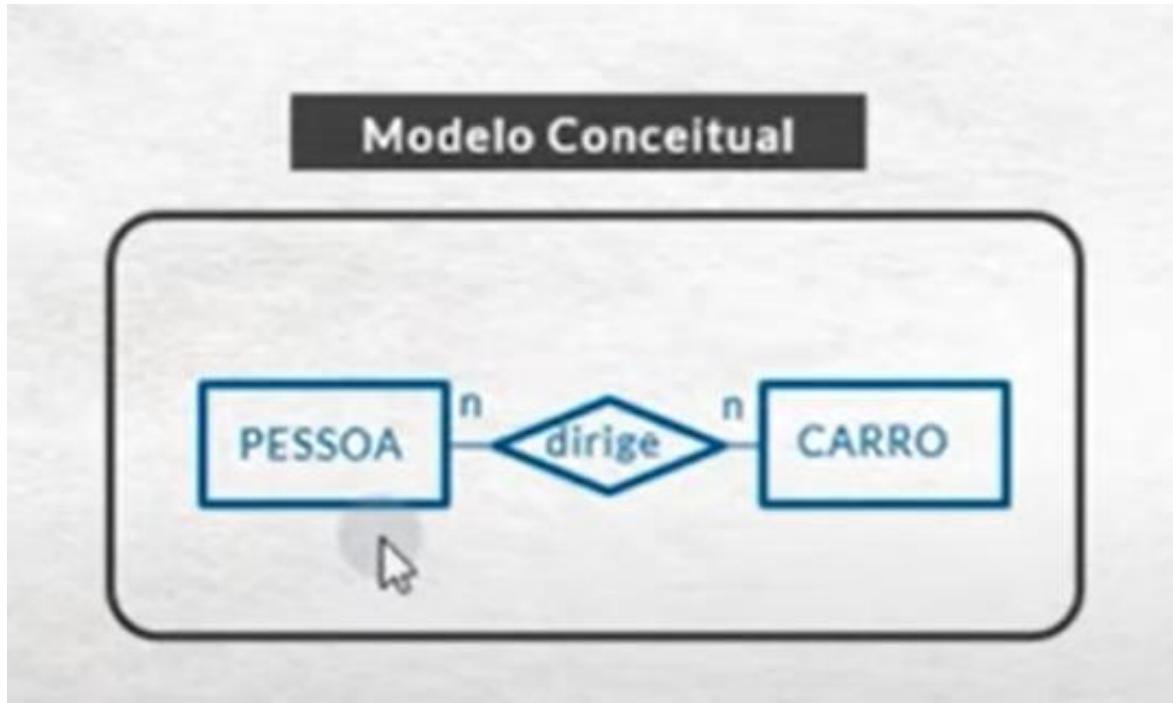
Uma vez que o **analista** observa o mundo real, ele faz uma representação do observado através dos três modelos:

- **Conceitual** -> Onde definimos entidades e relacionamentos
- **Lógico** -> Definição de Como serão as tabelas.
- **Físico** -> Etapa de criação das tabelas dentro do software de banco de dados.

O modelo conceitual foi explicado na última aula, agora vamos nos aprofundar no **modelo lógico**:

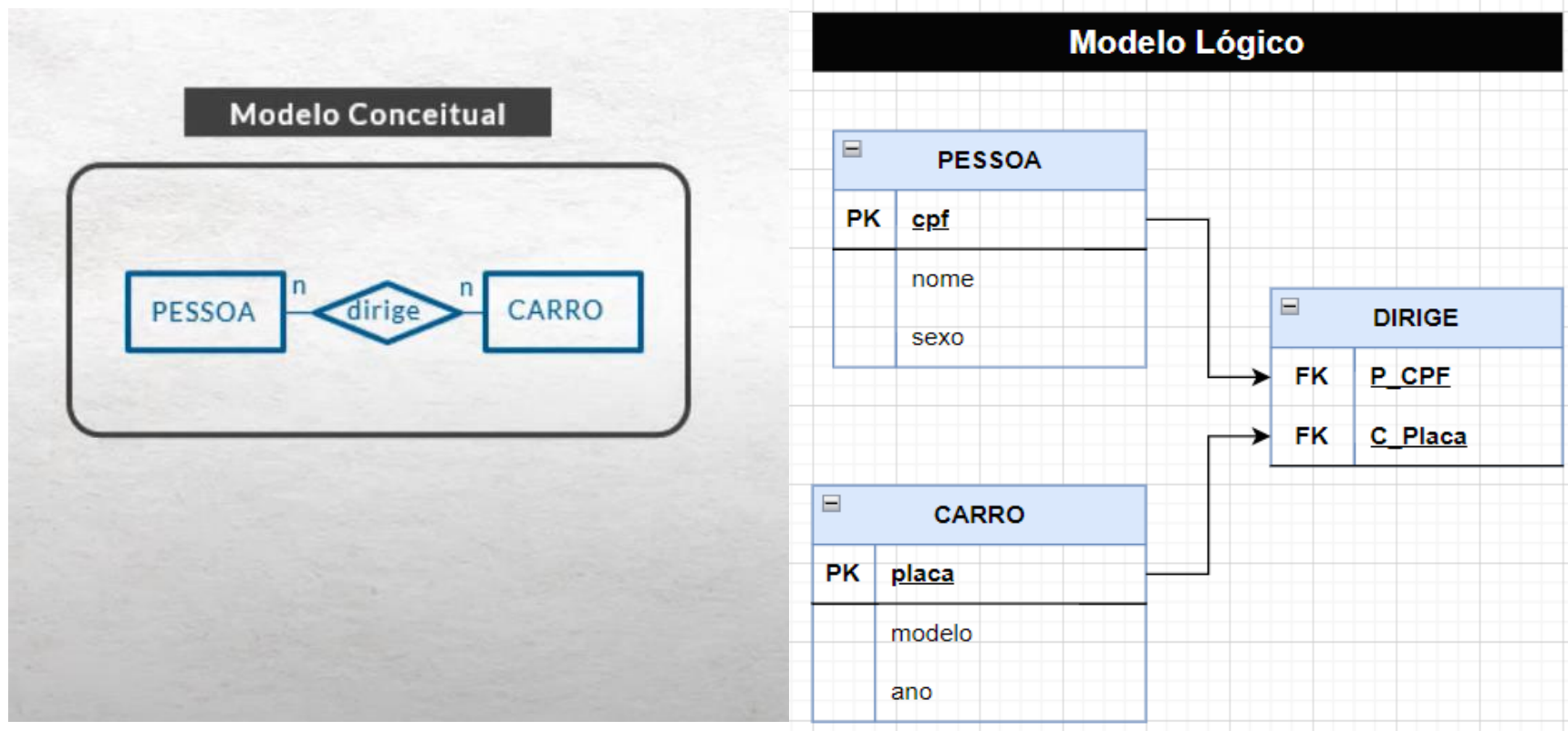
# Introdução

Exemplos de cada uma das 3 etapas da modelagem de dados



# Introdução

Exemplos de cada uma das 3 etapas da modelagem de dados



Modelo conceitual e lógico



# Introdução

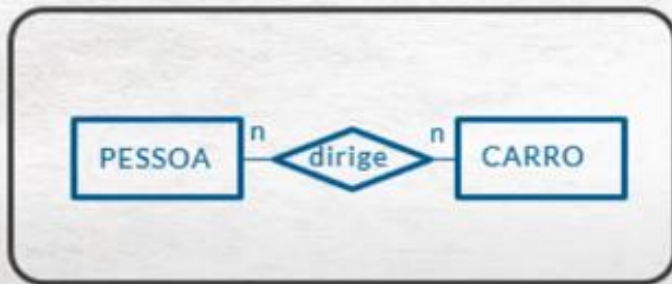


Observe na imagem acima como esses modelos se **relacionam**, no primeiro modelo temos as entidades e relacionamentos e no segundo, sendo nosso **modelo lógico**, temos as tabelas que detém as informações sobre as nossas entidades e relacionamentos.

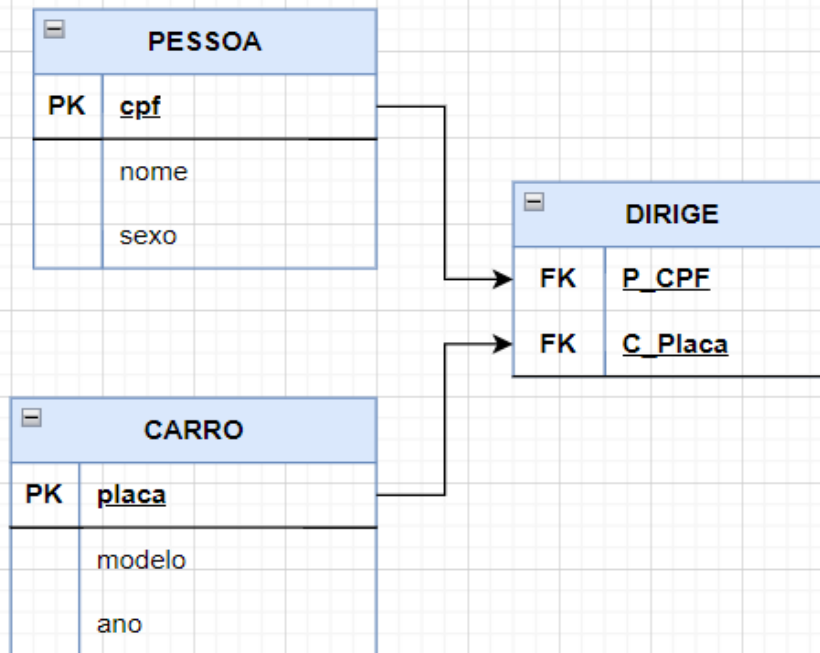
# Introdução

Exemplos de cada uma das **3 etapas** da modelagem de dados

Modelo Conceitual



Modelo Lógico



Modelo Físico

```
CREATE TABLE Pessoa(...  
CREATE TABLE Carro(...  
CREATE TABLE Dirige{...
```

# Modelo Lógico

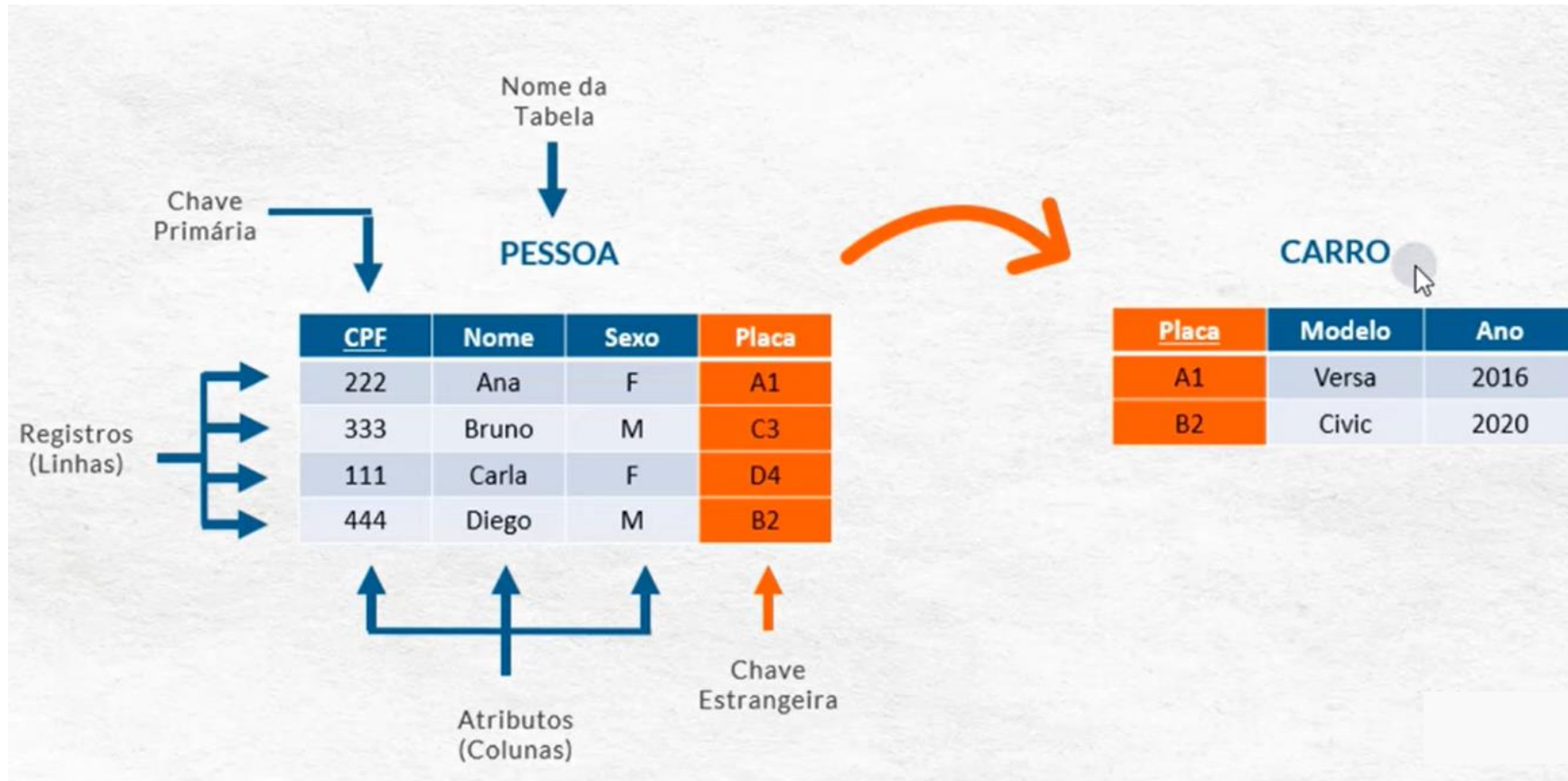
Um **modelo relacional lógico** é composto por **dois** elementos principais que são as **tabelas** e por diferentes tipos de **chaves**:



Composição do modelo lógico

# Tabelas

Uma tabela é um conjunto de **registros exclusivos**. É composto por linhas, colunas e chaves. Diferentes tabelas se relacionam por meio das **Chaves Estrangeiras** como mostra a indicação da seta na imagem.



# Chaves

No **Modelo Relacional**, são consideradas as chaves:

- ☑ Chaves candidatas
- ☑ Chaves Primárias
- ☑ Chaves Alternativas ou Alternadas
- ☑ Chave Estrangeira

# Chaves Candidatas

As **chaves candidatas** são todas as colunas que podem identificar de forma **única** as linhas de uma tabela.

No exemplo abaixo temos uma tabela **PESSOA** com as informações de **CPF**, **CNH**, **Nome** e **Sexo**.

Dentre essas colunas, duas delas poderiam identificar de forma única a pessoa -> **CPF** e **CNH**.

CPF	CNH	Nome	Sexo
111	AAA	Ana	F
222	BBB	Bruno	M
333	CCC	Carla	F
444	DDD	Diego	M

Chaves candidatas

# Chaves Primárias - PK (Primary Key, em inglês)

Dentre as nossas **chaves candidatas**, apenas uma pode ser a **chave primária**.

Uma **chave primária** é uma coluna (ou uma combinação de colunas) cujos valores distinguem uma linha das demais linhas em uma tabela.

Os valores de uma chave primária devem ser únicos e **não** podem ser nulos (not null).

Chave Primária

PESSOA

CPF	CNH	Nome	Sexo
111	AAA	Ana	F
222	BBB	Bruno	M
333	CCC	Carla	F
444	DDD	Diego	M

Chave primária

A **chave primária** de uma tabela se relaciona com a **chave estrangeira** de outra tabela, por este motivo é importante definir qual será a chave primária.



# Chaves Alternadas

As **chaves candidatas** que **não** foram eleitas primárias, serão reconhecidas como **chaves alternadas** (ou **chaves alternativas**).

No exemplo que estamos utilizando a nossa coluna de **CNH** será a nossa chave alternada.

PESSOA

CPF	CNH	Nome	Sexo
111	AAA	Ana	F
222	BBB	Bruno	M
333	CCC	Carla	F
444	DDD	Diego	M

Chave Alternada

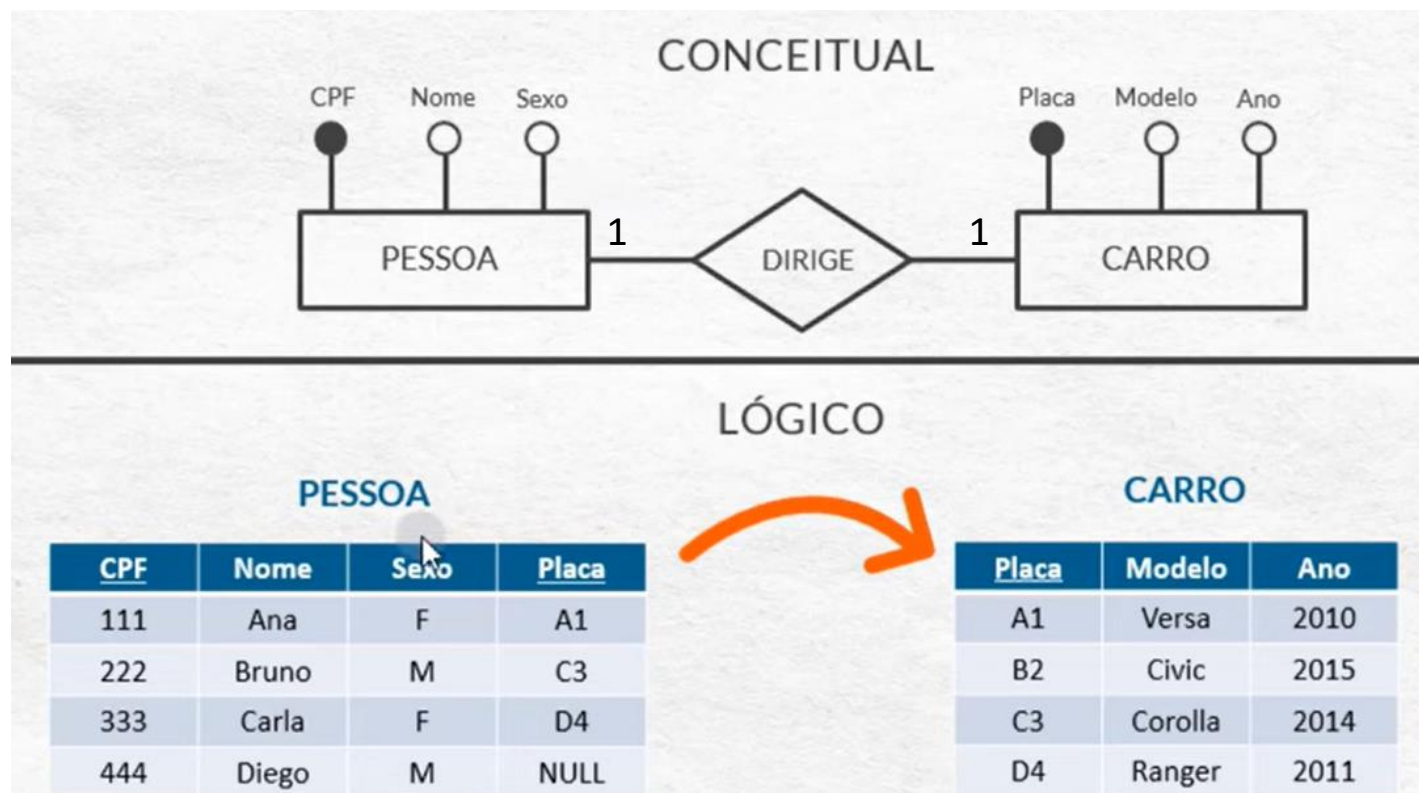


Chave alternativa ou alternada



# Chave Estrangeira - FK (Foreign Key, em inglês)

Uma **chave estrangeira** se trata de uma coluna (ou combinação de colunas), cujos valores aparecem na chave primária da tabela que está relacionada. Por meio da chave estrangeira é possível criar relacionamentos em um banco de dados relacional.



Observe que através da placa conseguimos identificar qual carro cada pessoa dirige, este é o papel da chave estrangeira, promover um relacionamento seguro entre as tabelas.

# Restrição de Integridade

Quando criamos tabelas em bancos de dados, essas tabelas aceitam qualquer valor.

Ou seja, se quisermos adicionar um produto com preço negativo ou um cliente sem nome e sem CPF, nada nos impede.

E é claro que adicionar um produto com preço negativo não faz nenhum sentido, e adicionar o cliente sem as suas informações básicas (nome e cpf) também não é de grande utilidade.

Para garantir que os dados terão algum nível de consistência, vamos precisar criar as **Restrições de Integridade**.

Geralmente, as restrições de integridade ajudam a evitar **erros humanos** no momento de registrar valores em um banco de dados.

# Restrição de Integridade

**Restrições de Integridade** são regras de consistência dos dados, é o que vai garantir a validade dos dados presentes dentro do banco de dados.

## Alguns exemplos de Restrição de Integridade:

- ☑ **Integridade de Vazio:** Informa se os valores da coluna são opcionais ou não.
- ☑ **Integridade de Chave:** Valores em uma chave primária não devem ser nulos.
- ☑ **Integridade Referencial:** Valores de uma coluna em uma tabela são válidos baseados nos valores em outra tabela relacionada

# Dicionário de Dados:

Um **dicionário de dados** é um documento usado para armazenar informações sobre o conteúdo, formato e estrutura de um banco de dados, assim como o relacionamentos entre os seus elementos.

Em resumo, se trata de um documento que explica e detalha todas as **entidades, atributos e relacionamentos**.

É importante criar um **dicionário de dados** para minimizar **erros** ao criar a estrutura física do banco de dados e permitir documentar a lógica por trás do projeto.

# Exemplo de Dicionário de Dados:

## Descrição das Tabelas

Começamos listando todas as tabelas do banco de dados, com as suas respectivas descrições e identificação das tabelas relacionadas.

Tabela	Relacionamento	Nome do Relacionamento	Descrição
Funcionario	Departamento	Pertence	Tabela para cadastro dos funcionários de uma empresa
	Localidade	Reside	
Departamento	Funcionario	Pertence	Tabela para cadastro dos departamentos de uma empresa
Localidade	Funcionario	Reside	Tabela para cadastro das localidades de residência

# Anomalias em Banco de Dados

**Anomalias em banco de dados** são mudanças em dados que podem gerar **inconsistência** no banco.

Uma **inconsistência** é geralmente representada por situações em que dados que deveriam ser iguais, apresentam valores diferentes em várias tabelas do banco de dados. Por exemplo, o valor de venda de um produto deve ser o mesmo valor armazenado nas tabelas de **venda** e **nota fiscal**.

Inconsistências geralmente ocorrem quando o banco de dados é projetado de forma inadequada.

**As anomalias são classificadas entre 3 categorias:**

- ☑ Anomalia de inserção
- ☑ Anomalia de exclusão
- ☑ Anomalia de atualização

# Normalização de Banco de Dados

**Normalização** é um conceito, que visa analisar e melhorar a estrutura de um banco de dados por meio de um processo lógico para realizar a divisão de uma tabela complexa em uma ou mais tabelas simples, construído em torno do conceito de **formas normais**.



# Formas Normais

As **formas normais** são um passo a passo de correção nas tabelas do banco de dados, para corrigir as anomalias. Esse passo a passo garante a reestruturação bem-feita da estrutura, assim, garantimos que as tabelas do banco de dados estão bem estruturadas.

**Existem uma série formas normais, mas as principais são:**

- ☑ **Primeira Forma Normal (1FN):** As informações são armazenadas em uma tabela na qual cada coluna contém valores atômicos (valores que não podem ser divididos) e não há grupos repetidos de colunas.



# Formas Normais

- ☑ **Segunda Forma Normal (2FN):** A tabela atende à regra da primeira forma normal e todas as colunas dependem da chave primária da tabela.
- ☑ **Terceira Forma Normal (3FN):** A tabela está na segunda forma normal e nenhuma das suas colunas depende transitivamente da chave primária.

# Resumo

- ✓ Os principais elementos de um modelo lógico são: **Tabelas** e **Chaves**.
- ✓ Para evitar erros humanos na hora de registrar os dados em um banco (exemplo: um preço negativo) utilizamos **Restrições de Integridade**.
- ✓ Para garantir que o projeto de banco de dados estará bem documentado, utilizamos um **Dicionário de Dados**.
- ✓ Corrigimos/evitamos inconsistências nos banco dedados aplicando as **Formas Normais**, que são um passo a passo para garantir que as tabelas do banco estão bem estruturadas.

# Referência

## INTRODUÇÃO A BANCO DE DADOS

Pereira, Paloma Cristina - Introdução a bancos de dados - Editora Senac São Paulo – São Paulo – 2021

E-book. Disponível em:

<https://www.bibliotecadigitalsenac.com.br/?from=busca%3FcontentInfo%3D2915%26term%3DLeite%2525252C%25252520Leonardo%25252520Alexandre%25252520Ferreira%25252520-%25252520Programa%252525C3%252525A7%252525C3%252525A3o%25252520de%25252520banco%25252520de%25252520dados#/legacy/epub/2915>

Acesso em 29/10/2023