



EDUARDO AKIRA DA ROSA MIYAMOTO

**MODELAGEM DE DADOS:**  
MODELOS DE BANCO DE DADOS

TOYOKAWA - JAPÃO  
2025



EDUARDO AKIRA DA ROSA MIYAMOTO - RA2025\*\*\*\*

## **MODELAGEM DE DADOS**

MODELOS DE BANCO DE DADOS -UNIDADE 2

Professor da Matéria: Kariston Stevan Luiz

Curso: CyberSecurity

Relatório apresentado à disciplina de  
MODELAGEM DE DADOS, do curso de  
Cibersecurity da Faculdade Anhanguera  
polo Guarulhos, como requisito parcial  
para aprovação na disciplina da Unidade  
2.

Toyokawa - Japão

## **INTRODUÇÃO**

Neste trabalho, estamos explorando a Modelagem de Dados, etapa essencial para a construção de um Banco de Dados bem estruturado e organizado. O foco principal será nos elementos do modelo Entidade-Relacionamento, fundamentais para garantir que o sistema seja funcional, consistente e livre de ambiguidades.

### Aula 3: Elementos do modelo Entidade-Relacionamento(ER) - II

**Nome do Projeto:** Diagrama Entidade-Relacionamento(DER)

**Linguagem usada:** SQL

**Programa usado:** Mysql Workbench

O objetivo é desenvolver um Diagrama Entidade-Relacionamento (DER), que represente uma base de dados possível de ser implementado em um sistema de gerenciamento de banco de dados utilizando o software Workbench MySQL.

A empresa MoveRent necessita controlar os Ciclomotores locados às pessoas que querem se movimentar pela cidade, por causa do trabalho, passeio, lazer etc.

- Pessoa: indivíduo que possuem registro no Brasil e são autorizadas a fazer locação de um ciclomotor.
- Ciclomotor: equipamento para locomover uma pessoa;
- Trajeto: banco de rastreamento do trajeto realizado pelo Ciclomotor;
- Locação: registro da locação um ciclomotor realizada pela pessoa.

Cada entidade acima possuem os seguintes atributos:

- Pessoa (CPF, nome, email, telefone)
- Ciclomotor (IDCM, nome, ano, cor)
- Locação (IDLOC, data, hora, local)
- Trajeto (IDTRAJETO, data, hora, local)

Defina as chaves primárias e secundárias, defina também os tipos de dados dos atributos e elabore o DER utilizando o Workbench MySQL.



Todo o desenvolvimento deste projeto será realizado usando o MySQL Workbench, o que seria essa ferramenta ?

O MySQL Workbench é uma ferramenta visual unificada para arquitetos de banco de dados, desenvolvedores e administradores de banco de dados. O MySQL Workbench oferece modelagem de dados, desenvolvimento em SQL e ferramentas de administração abrangentes para configuração de servidores, administração de usuários, backup e muito mais. O MySQL Workbench está disponível para Windows, Linux e Mac OS X.

É uma ferramenta visual que nos permite modelar de forma mais fácil o banco de dados.

Você pode encontrar mais informações diretamente no site oficial:

“ <https://www.mysql.com/products/workbench/> ”

Todos os arquivos aqui disponibilizados também estarão em nosso GitHub:

“ <https://github.com/akiramiyamoto-dev/> ”



## Desenvolvimento

### Compreensão

Para realizarmos o Diagrama Entidade - Relacionamento de forma organizada, primeiramente teremos que analisar todos os dados que nos foram passados, vamos identificar as entidades, definir os atributos, escolher as chaves primárias, identificar os relacionamentos e também por final a Cardinalidade dos relacionamentos. Vamos estar explicando cada fase desse projeto.

#### 1. Identificando as entidades:

Uma entidade é qualquer objeto, conceito ou evento do mundo real que tenha importância para a organização e que precise ser representado em um banco de dados.

No nosso caso são eles:

- Pessoa
- Ciclomotor
- Locação
- Trajeto

#### 2. Definição dos atributos:

Atributos são as características ou propriedades que descrevem uma entidade. Eles representam as informações que você deseja armazenar sobre cada ocorrência da entidade em seu banco de dados.

Os atributos de cada entidade são:

- **Pessoa:** Atributos (CPF, nome, email, telefone)
- **Ciclomotor:** Atributos (IDCM, nome, ano, cor)
- **Locação:** Atributos (IDLOC, data, hora, local)
- **Trajeto:** Atributos (IDTRAJETO, data, hora, local)

#### 3. Escolha da chave primária(PK) e da chave estrangeira(FK):

A chave primária (Primary Key) é um atributo ou conjunto de atributos de uma entidade que identifica unicamente cada instância dessa entidade.

Não pode haver dois registros com o mesmo valor na chave primária.

Serve como referência para relacionamentos com outras tabelas (através de chaves estrangeiras).

A chave estrangeira(Foreign Key) é um atributo (ou conjunto de atributos) em uma tabela que referencia a chave primária de outra tabela.

Aqui vamos definir as chaves primárias e estrangeiras:

- **Pessoa:**
  - CPF: Chave primária(PK)
- **Ciclomotor:**
  - IDCM: chave primária(PK)
- **Locação:**
  - IDLOC: Chave primária(PK)
  - CPF: Chave estrangeira(FK) referenciando a entidade Pessoa(CPF)
  - IDCM: Chave estrangeira(FK) referenciando a entidade Ciclomotor(IDCM)
- **Trajeto:**
  - IDTRAJETO: Chave primária(PK)
  - IDLOC: Chave estrangeira(FK) referenciando a entidade Locação(IDLOC)

#### 4. Identificação dos relacionamentos e Cardinalidades

Em modelagem de dados, um relacionamento é uma associação lógica entre duas ou mais entidades, que define como elas interagem e se conectam. Ele descreve a relação de uma ocorrência de uma entidade com a(s) ocorrência(s) de outra, evitando a redundância de dados e garantindo a integridade referencial do sistema.

##### Como os Relacionamentos Funcionam?

- **Conexão Lógica:**

Os relacionamentos estabelecem ligações entre entidades (conceitos do mundo real, como "Aluno" ou "Curso"), refletindo as conexões entre elas no sistema.

- **Chaves Primárias e Estrangeiras:**

São utilizados para construir essas conexões; a chave primária de uma tabela (entidade) é inserida na outra tabela como chave estrangeira para estabelecer o vínculo.

- **Cardinalidade**

A cardinalidade define quantas instâncias de uma entidade podem se relacionar com instâncias de outra entidade. Ela indica restrições quantitativas do relacionamento. É fundamental para modelar o DER corretamente e evitar inconsistências no banco de dados.

- **Cardinalidade:**

Define a quantidade de instâncias de uma entidade associadas a instâncias de outra.

Os tipos mais comuns incluem:

**1:1** (Um para Um): Cada ocorrência de uma entidade se associa a no máximo uma ocorrência da outra.

**1:N** (Um para Muitos): Uma única ocorrência de uma entidade se relaciona com várias ocorrências da outra.

**N:M** (Muitos para Muitos): Ocorrem quando múltiplas instâncias de uma entidade se relacionam com múltiplas instâncias de outra. Para gerenciar N:M, é criada uma tabela intermediária com chaves estrangeiras de ambas as entidades.

Neste projeto os relacionamentos são:

1. Pessoa --- Locação: 1:N (um para muitos)
  - a) Uma pessoa pode fazer várias locações.
  - b) Cada Locação pertence a uma pessoa.
2. Ciclomotor --- Locação: 1:N (um para muitos)
  - a) Um ciclomotor poder ser alugado várias vezes(diferentes locações).
  - b) Cada locação tem exatamente um ciclomotor.
3. Locação --- Trajeto: 1:N (um para muitos)
  - a) Cada locação pode ter vários trajetos registrados.
  - b) Cada trajeto pertence a uma locação.



## 5. Definição dos tipos de dados(Data Type)

### Pessoa

CPF: Chave primária (PK), CHAR(11)  
nome: VARCHAR(100)  
email: VARCHAR(100)  
telefone: VARCHAR(15)

### Ciclomotor

IDCM: Chave primária (PK), INT AUTO\_INCREMENT  
nome: VARCHAR(50)  
ano: YEAR  
cor: VARCHAR(20)

### Locação

IDLOC: Chave primária (PK), INT AUTO\_INCREMENT  
data: DATE  
hora: TIME  
local: VARCHAR(100)  
CPF: Chave estrangeira (FK) referenciando Pessoa(CPF)  
IDCM: Chave estrangeira (FK) referenciando Ciclomotor(IDCM)

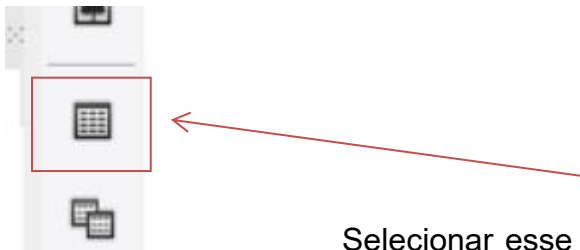
### Trajetos

IDTRAJETO: Chave primária (PK), INT AUTO\_INCREMENT  
data: DATE  
hora: TIME  
local: VARCHAR(100)  
IDLOC: Chave estrangeira (FK) referenciando Locação(IDLOC)

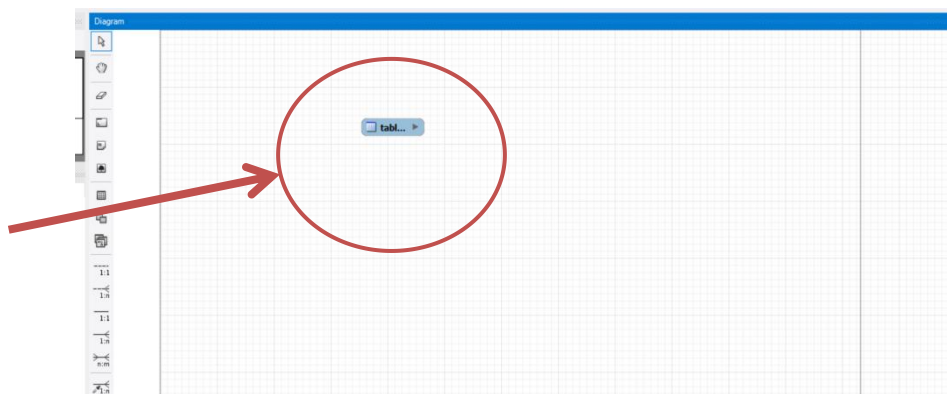
## 6. Modelando no Workbench MySQL

Primeiramente vamos ter que baixar e instalar o MySQL Workbench no computador. Após instalar, siga os seguintes passos:

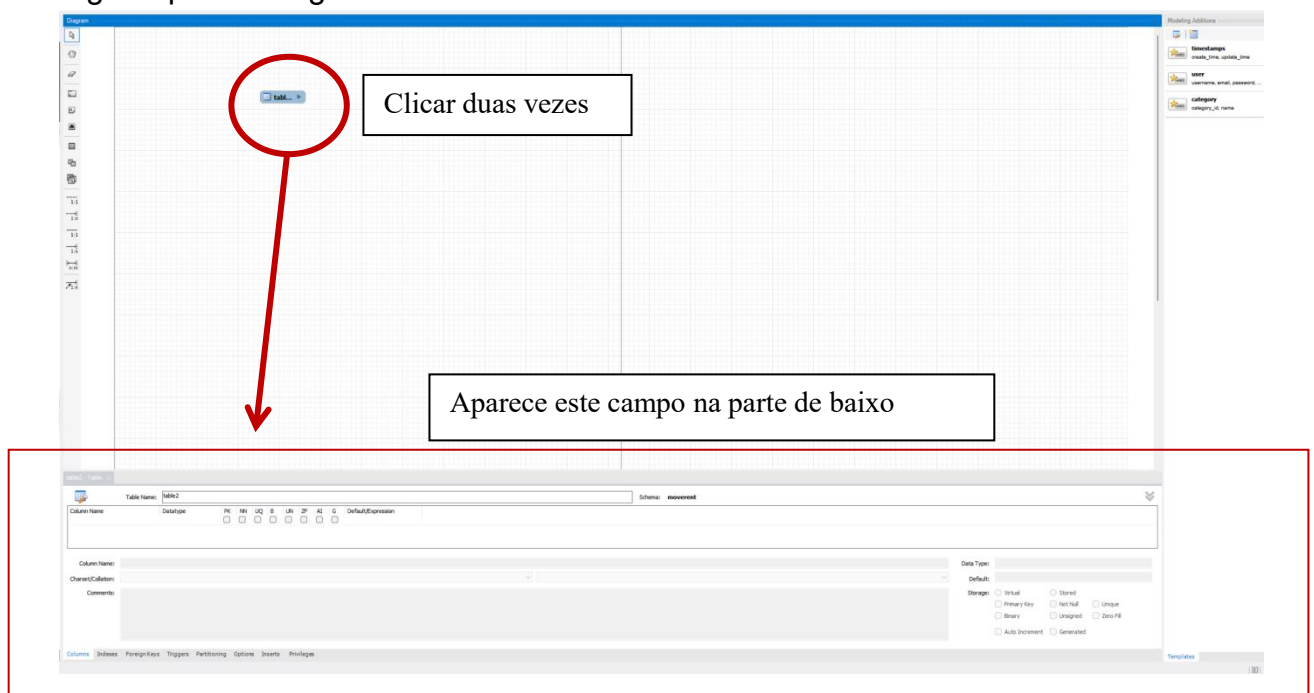
- Inicie o MySQL Workbench e crie um novo modelo em branco, File-> New-> Model Overview -> Add Diagram, será aberta uma área de edição chamada de EER Diagram.
- Agora, vamos criar as entidades:
  - Pessoa, Ciclomotor, Locação e Trajeto



Selecionar esse ícone para criar as entidades. Depois clicar em alguma parte da tela com o lado direito do mouse. Vai aparecer um novo elemento na tela, e esse será a entidade.



O próximo passo é clicar duas vezes em cima desse elemento, na parte de baixo vai aparecer um novo campo para poder editar o nome, as colunas, o tipo de dados e as regras que vão reger esse elemento.



## DICAS ÚTEIS

Table Name: <input type="text" value="Pessoa"/>		Schema: <b>moverent</b>									
Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression	
CPF	CHAR(11)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
nome	VARCHAR(100)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
email	VARCHAR(100)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
telefone	VARCHAR(15)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		

Entendendo cada campo:

**PK = Primary Key** [chave primária]

Identificador único (não pode repetir nem ser nulo)

**NN = Not Null** [Não Nulo]

O campo é obrigatório, não aceita valores nulos. (a pessoa vai ter que registrar os dados no campo que houver essa opção).

**UQ = Unique** [Único]

O valor não pode se repetir em nenhuma linha da tabela.

**BIN = Binary** [Binário]

Guarda dados em formato binário(sensível a maiúsculas/minúsculas).

**UN = Unsigned** [Sem sinal]

Número sem sinal(não aceita valores negativos).

**ZF = Zero Fill** [Preenchimento automático]

Preenche números com zeros à esquerda até o tamanho definido.

**AI = Auto Increment** [Incremento automático]

Incrementa automaticamente o valor.

**G = Generated** [Gerado]

Coluna gerada automaticamente pelo banco(calculada a partir de outras).



### **Tipos de dados de caractere**

Os tipos de dados de caractere do MySQL são usados para armazenar dados de texto e caracteres.

**CHAR:** O tipo de dados CHAR é usado para armazenar dados de texto fixos com um comprimento específico. Nesse sentido, ele é usado ao criar tabelas para armazenar valores de texto com o mesmo tamanho em todas as colunas. Veja o exemplo que cria uma tabela “meus\_valores” com uma coluna “nome” e armazena textos de no máximo 10 caracteres de comprimento.

```
CREATE TABLE meus_valores (  
    nome CHAR(10)  
);
```

**VARCHAR:** O tipo de dados VARCHAR é semelhante ao tipo CHAR, mas permite que os valores de texto sejam variáveis, com comprimentos diferentes em cada coluna. Dessa forma, ele é usado ao criar tabelas para armazenar valores de texto com comprimentos variáveis. Veja o exemplo abaixo que cria uma tabela “meus\_valores” com uma coluna “email” que pode armazenar textos de até 200 caracteres de comprimento, sendo flexível quanto ao tamanho dos dados armazenados :

```
CREATE TABLE meus_valores (  
    email VARCHAR(200)  
);
```

### **Tipos de dados de data e hora do MySQL**

Os tipos de dados de data e hora do MySQL são usados para armazenar valores de data e hora.

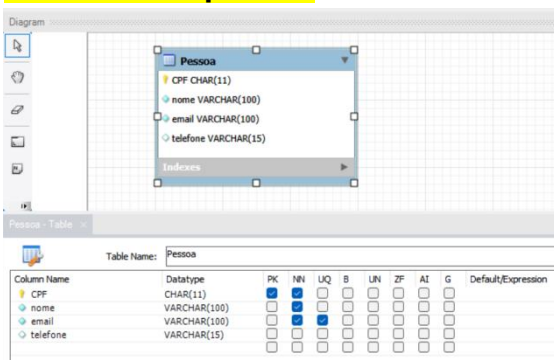
**DATE:** Utiliza-se o tipo de dados DATE para armazenar apenas a data sem a hora. Portanto, está composto por 8 bytes e pode armazenar valores de data no intervalo de 1000 a 9999 para o ano e de 0 a 65535 para o dia do mês. Veja o exemplo:

```
INSERT INTO meus_valores (data) VALUES ('2022-03-14');
```

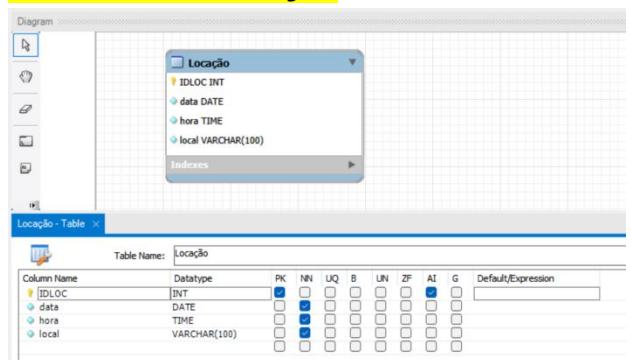
Listamos aqui neste documento apenas os Tipos de Dados (Data Type) que são usados no projeto, no site oficial do MySQL Workbench

Conforme já vimos sobre as entidades e seus tipos de dados, agora é só inserirmos cada um com suas devidas características.(Página 9)

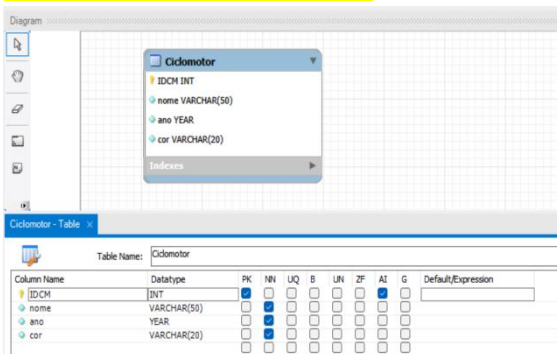
## ● Entidade pessoa



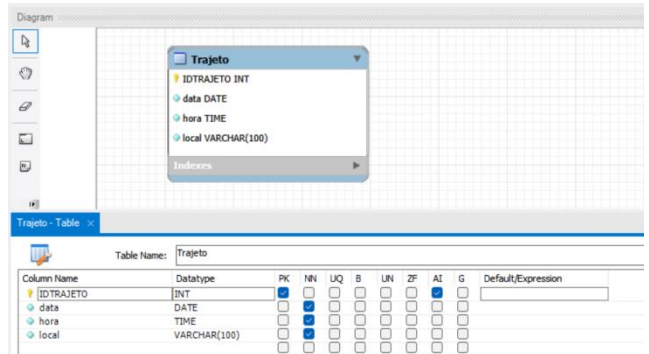
## ● Entidade Locação



## ● Entidade Ciclomotor

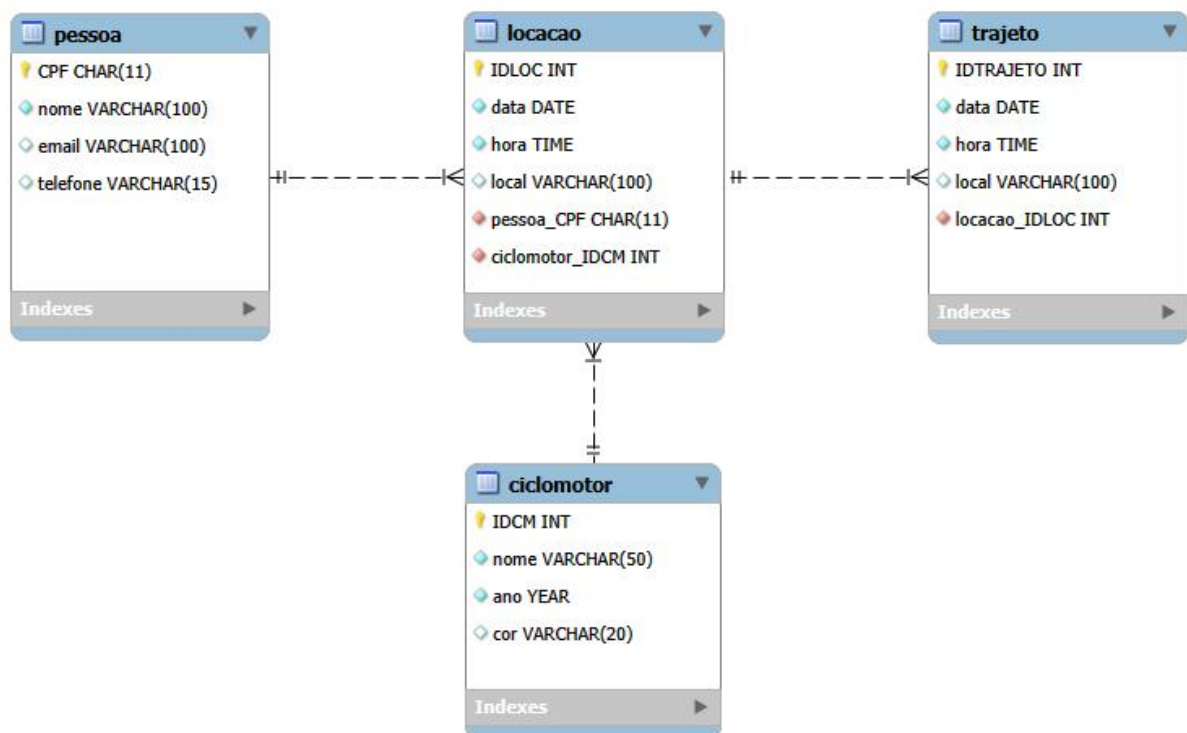


## ● Entidade Trajeto



## Relacionamentos

Agora chegamos ao ponto final que é definir os relacionamentos entre as entidades conforme suas cardinalidades.



Os relacionamentos já foram determinados nas páginas 7 e 8, mas vamos abordar o assunto aqui com mais detalhes.

ChavePrimária(PK) - Ela é simbolizada no diagrama com um ícone de chave:



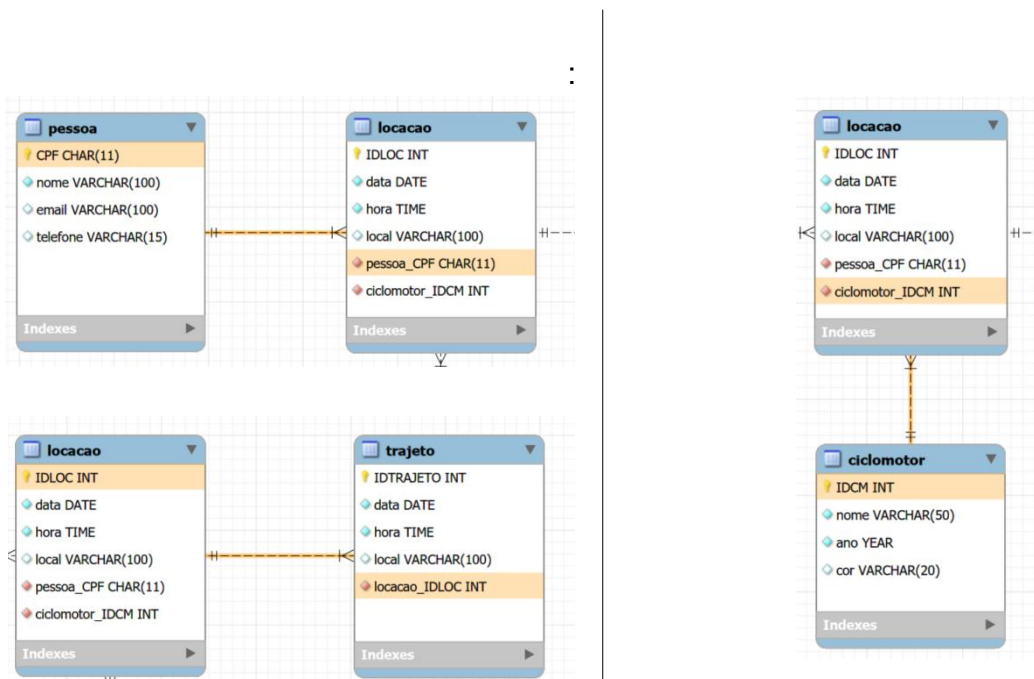
Chave Estrangeira(FK) - é um atributo em uma tabela que referencia a chave primária (PK) de outra tabela, estabelecendo um vínculo entre elas.

Serve para garantir a integridade referencial, ou seja, que os dados da tabela “filha” estejam sempre relacionados a registros válidos da tabela “pai”.

Por exemplo, na relação Pessoa → Locação, a coluna pessoa\_CPF na tabela Locação é uma FK que referencia CPF na tabela Pessoa. Isso impede que uma locação seja registrada sem uma pessoa correspondente.

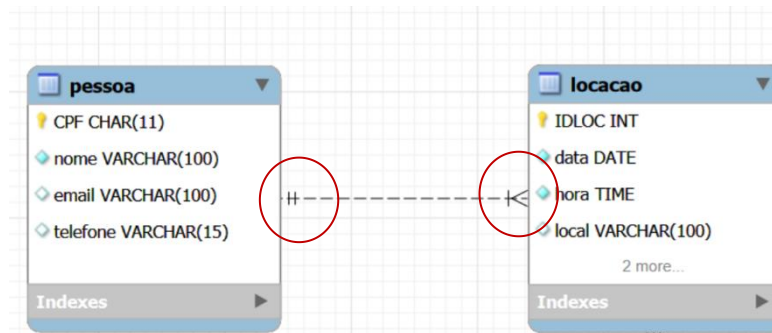
Resumo: FK = ligação entre tabelas, garantindo consistência dos dados.

As imagens abaixo vão mostrar as chaves estrangeiras, com as devidas ligações da entidade Pai.





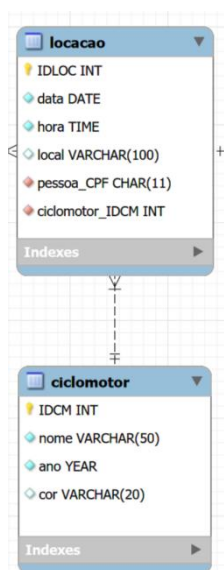
Veja a próxima imagem, no lado da entidade Pessoa, vai ter dois riscos simbolizando a Cardinalidade de 1, e do lado da locação veremos o símbolo que é chamado de “pé de galinha”, pois realmente parece com um pé de galinha, esse simboliza a cardinalidade de Muitos.



Lemos essa imagem da seguinte maneira:

- Uma **pessoa** pode ter zero ou muitas locações. Por exemplo, alguém pode ter alugado vários ciclomotores ou ainda não ter alugado nenhum.
- Uma **locação** deve ter uma e somente uma pessoa associada a ela. Uma locação não pode existir sem um locatário.

Na próxima imagem vemos a Cardinalidade entre Locação e o Ciclomotor



Um ciclomotor pode ser alugado muitas vezes diferentes, por clientes diferentes ou no decorrer do tempo.

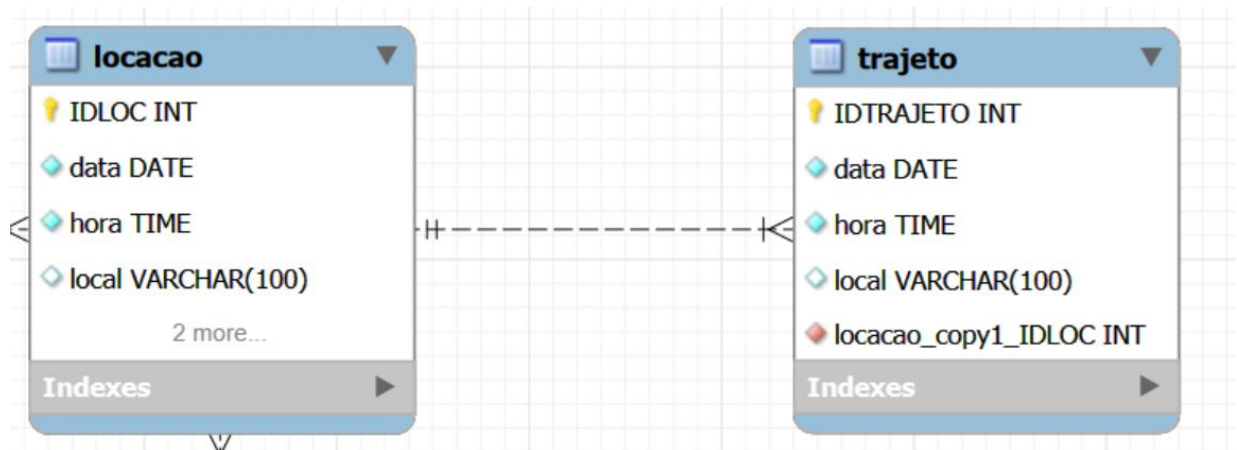
Mas cada locação só pode estar ligada a um ciclomotor de cada vez (quando você aluga, é sempre uma moto específica).

Então, dá pra pensar assim:

Um ciclomotor → várias locações.

Uma locação → um ciclomotor.

No caso da Locação e do Trajeto, a Cardinalidade se define da seguinte maneira:



Uma locação (quando o cliente pega o ciclomotor) pode ter vários trajetos registrados (ex.: saiu da loja → foi ao mercado → voltou).

Cada trajeto pertence sempre a uma única locação.

Ou seja, a relação é também 1 para N:

Uma locação → muitos trajetos.

Um trajeto → uma locação.

## RESUMO

Este trabalho teve como objetivo aplicar os conceitos de modelagem de dados na construção de um Diagrama Entidade-Relacionamento (DER) utilizando a ferramenta MySQL Workbench. A atividade consistiu em representar, por meio de um modelo relacional, a proposta da empresa fictícia MoveRent, que necessita controlar a locação de ciclomotores realizados por pessoas e os trajetos percorridos durante o período de uso.

O modelo desenvolvido contemplou quatro entidades principais: Pessoa (CPF, nome, e-mail, telefone), Ciclomotor (IDCM, nome, ano, cor), Locação (IDLOC, data, hora, local) e Trajeto (IDTRAJETO, data, hora, local). Para cada entidade, foram definidos atributos, chaves primárias e estrangeiras, bem como os tipos de dados adequados. Os relacionamentos identificados foram do tipo um-para-muitos (1:N), sendo: uma pessoa pode realizar várias locações; um ciclomotor pode estar associado a diversas locações ao longo do tempo; e uma locação pode conter diferentes trajetos.

A utilização do MySQL Workbench possibilitou a criação de um DER claro e estruturado, permitindo visualizar a integridade referencial entre as entidades por meio de chaves estrangeiras e da notação “pé de galinha”. Dessa forma, a prática proporcionou não apenas a aplicação dos fundamentos teóricos da disciplina de Modelagem de Dados, mas também o desenvolvimento de habilidades práticas no uso de ferramentas digitais para documentação e análise de banco de dados.

## RESULTADOS DE APRENDIZAGEM

Durante a realização desta atividade, foi possível atingir os seguintes resultados de aprendizagem:

### 1. Compreensão do MySQL Workbench

Aprendi a utilizar o MySQL Workbench como ferramenta de modelagem de dados, entendendo sua interface, menus e funcionalidades básicas para criação de Diagramas Entidade-Relacionamento (DER).

Identifiquei os recursos de criação de tabelas, definição de atributos, tipos de dados, chaves primárias (PK) e estrangeiras (FK), bem como a visualização da cardinalidade dos relacionamentos.

### 2. Criação de um DER funcional

Desenvolvi o Diagrama Entidade-Relacionamento para a empresa fictícia MoveRent, incluindo as entidades Pessoa, Ciclomotor, Locação e Trajeto.

Defini os atributos de cada entidade e associei corretamente as chaves primárias e estrangeiras para garantir a integridade referencial do banco de dados.

Estabeleci os relacionamentos entre as entidades, identificando a cardinalidade 1:N para cada associação, de acordo com a proposta da atividade.

### 3. Representação clara de um banco de dados

Aprendi a transformar informações do mundo real (locações de ciclomotores) em dados estruturados, organizados em um modelo relacional.

A modelagem permitiu visualizar como os dados se conectam, garantindo consistência e evitando redundâncias.

#### **4. Entrega do DER em arquivo texto**

Ao final da atividade, foi produzido um arquivo .DOCX contendo:

- ✓ Descrição das entidades e atributos;
- ✓ Chaves primárias e estrangeiras;
- ✓ Relacionamentos e cardinalidades;
- ✓ Representação gráfica do DER exportada do MySQL Workbench.

Esse documento serve como registro da modelagem e pode ser utilizado como referência para implementação do banco de dados em ambiente real.

Segue abaixo em anexo o DER completo exportado do MySQL Workbench:

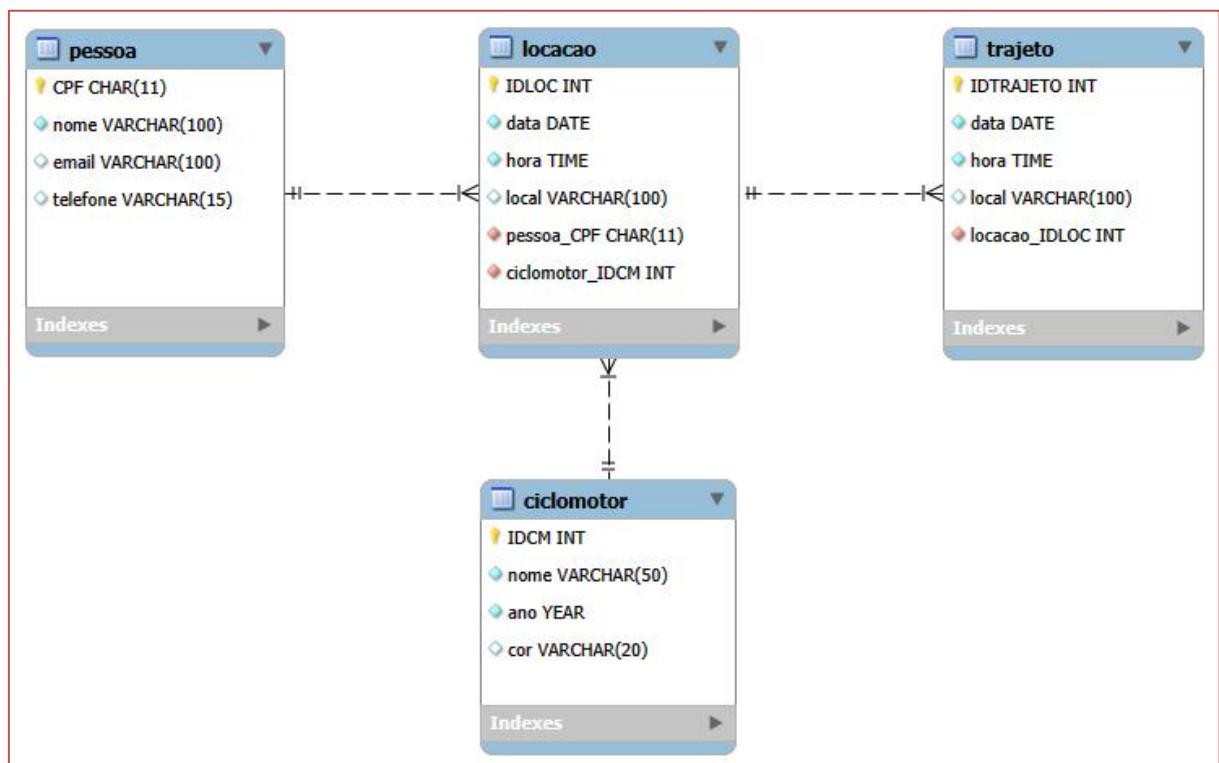


Imagem exportada do MySQL Workbench.

**Conclusão:**

Com esta atividade, desenvolvi habilidades práticas em modelagem de dados e no uso do MySQL Workbench, consolidando os conceitos teóricos da disciplina. Ao representar o sistema da MoveRent de forma estruturada, foi possível compreender a importância da modelagem para garantir um banco de dados funcional, consistente e organizado.

## REFERENCIAS

MACHADO, Felipe Nery Rodrigues. Banco de Dados: Projeto e Implementação. 3. ed. São Paulo: Érica, 2010.

DEVMedia. MER e DER: modelagem de bancos de dados. Disponível em: <https://www.devmedia.com.br/mer-e-der-modelagem-de-bancos-de-dados/14332>. Acesso em: 24 set. 2025.

Pinto, Pedro. Como criar um Diagrama EER com o MySQL Workbench. Pplware, 19 out. 2015. Disponível em: <https://pplware.sapo.pt/software/como-criar-um-diagrama-eer-com-o-mysql-workbench/>. Acesso em: 24 set. 2025.