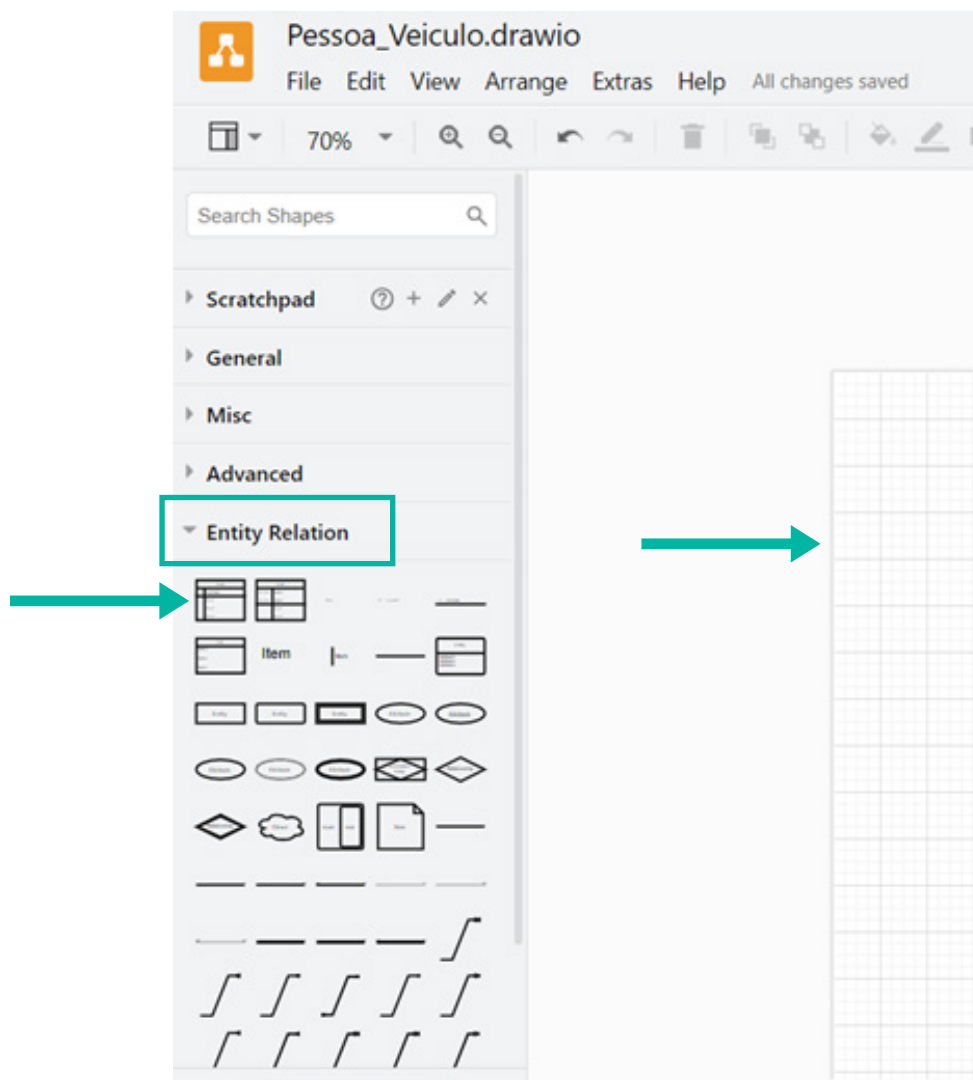


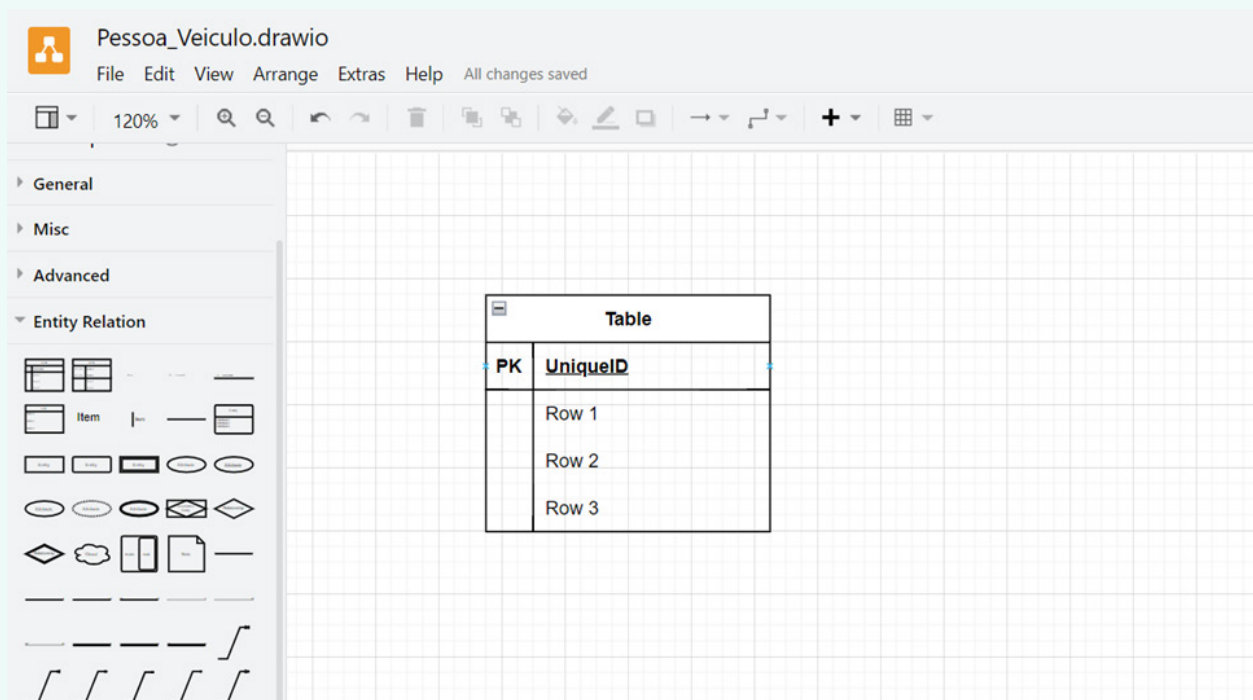
PASSO A PASSO DO DIAGRAMA-ENTIDADE RELACIONAMENTO DO ZERO

MÓDULO I

A. Precisamos, na nossa ferramenta de diagramação, procurar no painel esquerdo a opção ENTITY RELATION. Existem vários tipos itens sobre a complexidade de cada um, nos quais encorajo você, caro aluno, a posteriormente no curso se aprofundar mais. Mas aqui vamos usar apenas as tabelas e os relacionamentos.



B. Clique no primeiro item e o arraste para a tela da direita. Teremos esse resultado:

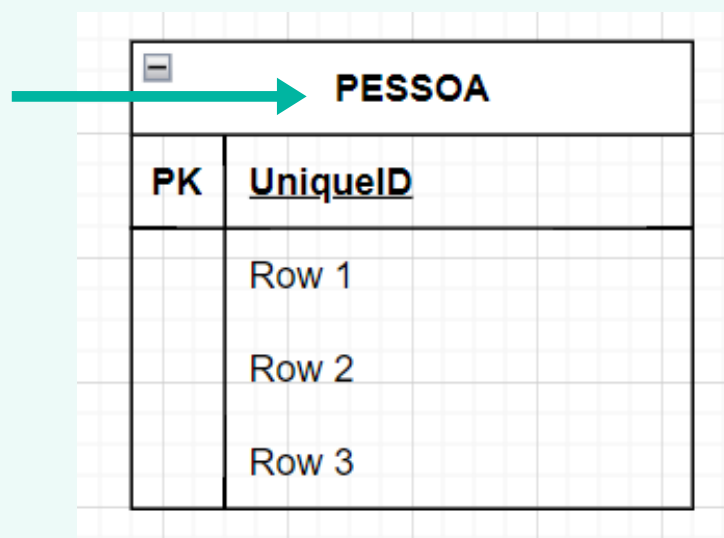


C. Agora está na hora de avançarmos mais um passo crucial na diagramação e modelagem de um banco de dados. Você se recorda sobre as chaves primárias e estrangeiras? Vou lhe ajudar a relembrar:

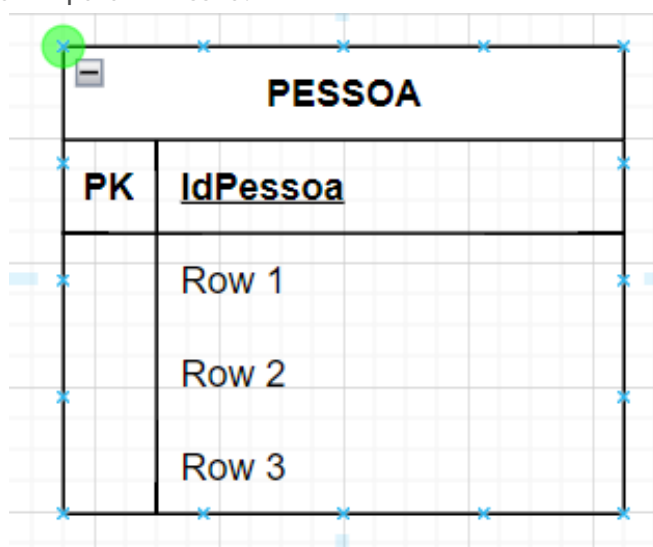
- A **chave primária** identifica exclusivamente cada registro em uma tabela, enquanto a **chave estrangeira** estabelece uma relação com a chave primária de outra tabela. A chave primária garante a unicidade dos dados, enquanto a chave estrangeira mantém a consistência entre tabelas relacionadas. Esses conceitos são fundamentais para garantir a integridade e a eficiência de um banco de dados.

Ou seja, para uma pessoa não se repetir dentro do banco de dados, precisamos de um atributo que seja exclusivo. Poderíamos fazer ser o CPF da pessoa, mas devido à indexação de um CPF ser mais demorada, vamos criar um campo chamado **IdPessoa**.

D. Sendo assim, se estamos traduzindo a entidade **PESSOA** para uma tabela, nada mais justo que renomear a tabela de TABLE para PESSOA. Para isso, é só clicar duas vezes em cima do nome TABLE e digitar PESSOA.



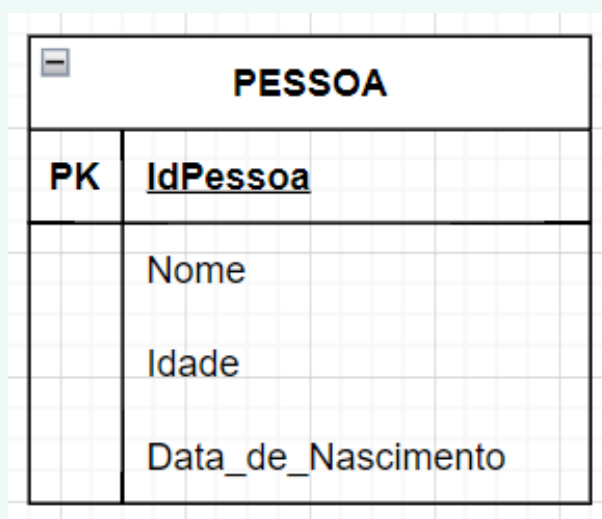
E. Agora, na linha onde está **PK**, que significa **PRIMARY KEY**, ou seja, **CHAVE PRIMÁRIA**, vamos alterar o atributo UniqueID para IdPessoa.



PESSOA	
PK	<u>IdPessoa</u>
	Row 1
	Row 2
	Row 3

F. Agora temos que incluir os outros atributos, que eram: nome, idade, data_de_nascimento.

Importante: existe uma convenção de não utilizar espaços entre nomes compostos. Sempre substitua por *underline* “_”.

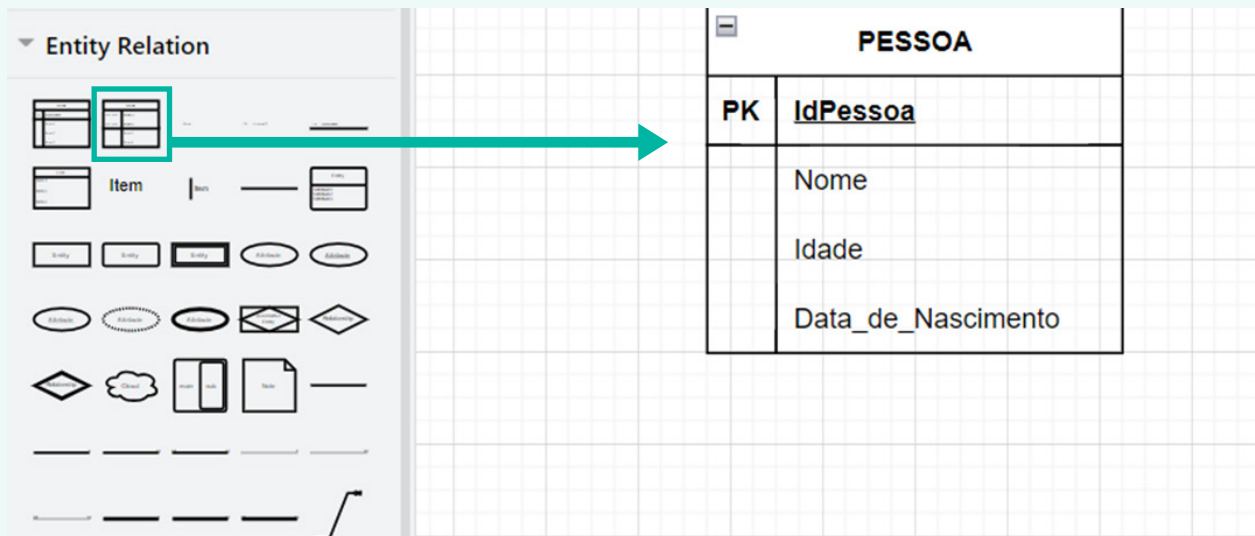


PESSOA	
PK	<u>IdPessoa</u>
	Nome
	Idade
	Data_de_Nascimento

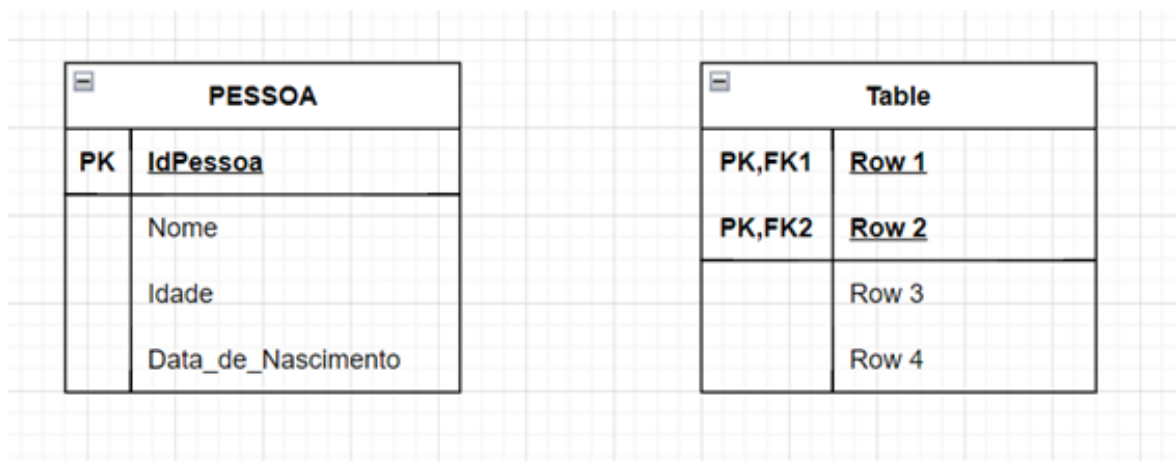
G. O que a tabela anterior demonstra? **A tabela acima demonstra que iremos armazenar entidades do tipo PESSOA.**

- **Cada pessoa será identificada por um ID único.** Ou seja, um atributo de identificação será apenas de uma pessoa. Por exemplo, se cadastrarmos um Id 1 para a Pessoa com nome Maria, esse ID com valor 1 será sempre da Maria, e nunca poderá ser do João – a não ser que a Maria seja deletada do sistema, e o auto-increment não seja ativado nas inserções de Id.
- **É possível termos duas Marias, com a mesma idade, e mesma data_de_nascimento?** Sim! Esses problemas de duplicatas são tratados inicialmente no código antes da inserção no banco de dados. Entretanto, ao fim, nunca teremos duas pessoas diferentes com o mesmo Id. Aqui isso parece confuso, mas na próxima etapa fará sentido.

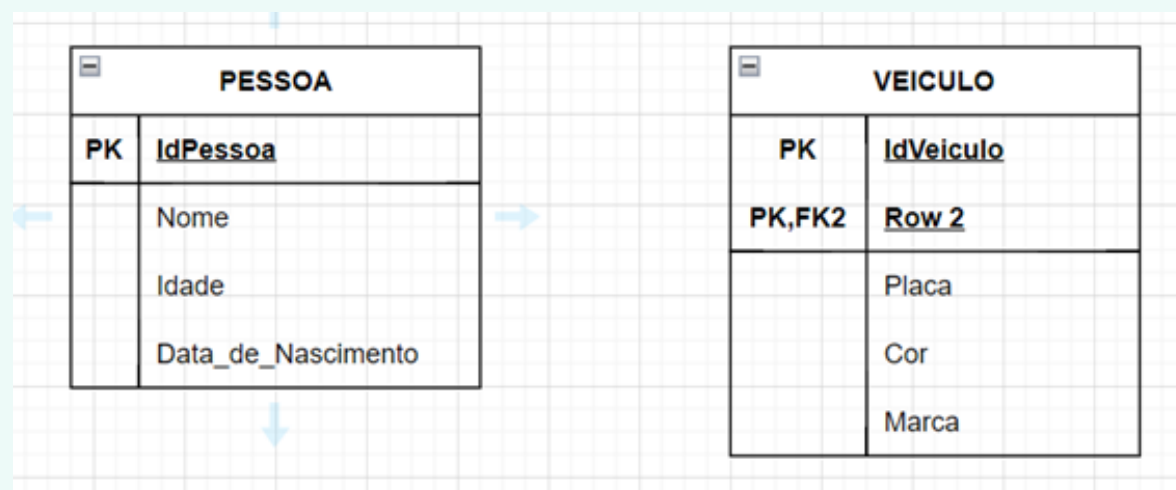
H. Agora vamos para a segunda tabela: VEÍCULO. Para ela, eu sugiro que você arraste para a sua tela o segundo item.



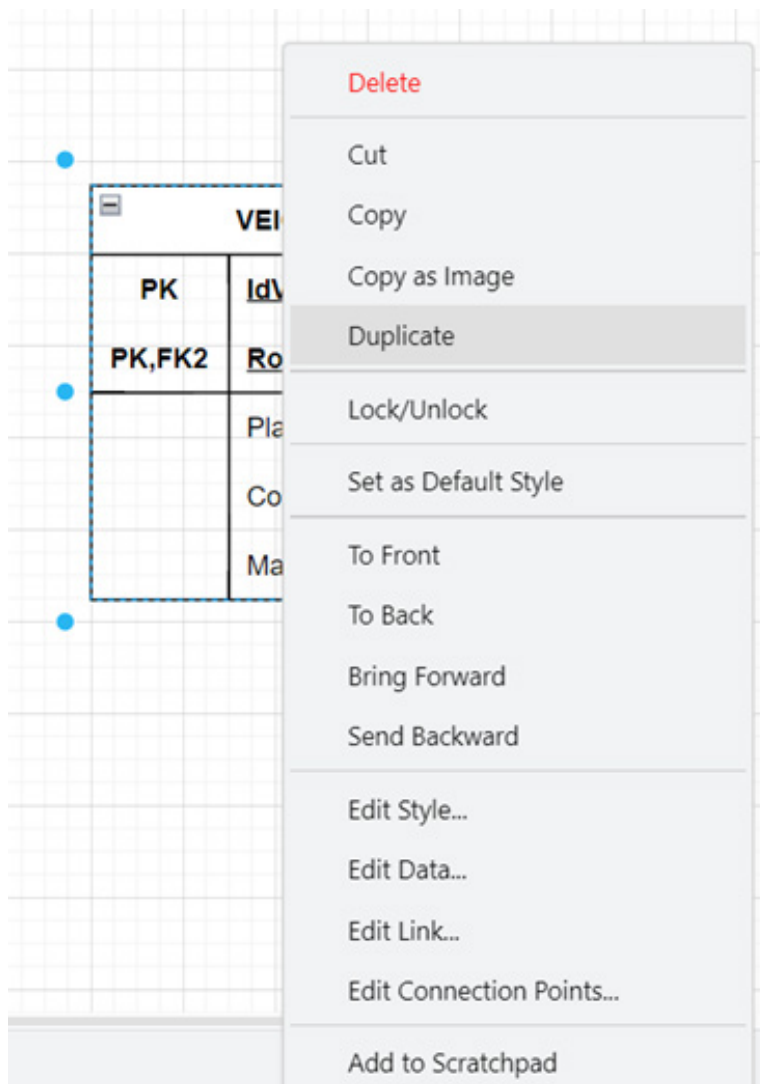
I. Resulta no modelo a seguir. Essa tabela secundária nós vamos alterar para VEICULO. Na primeira linha colocaremos PK – IdVeiculo. Essa será nossa chave de identificação exclusiva de cada veículo. A seguir, insira os atributos que fizemos no nosso fluxograma anteriormente: Cor, Placa e Marca.



J. O resultado será:



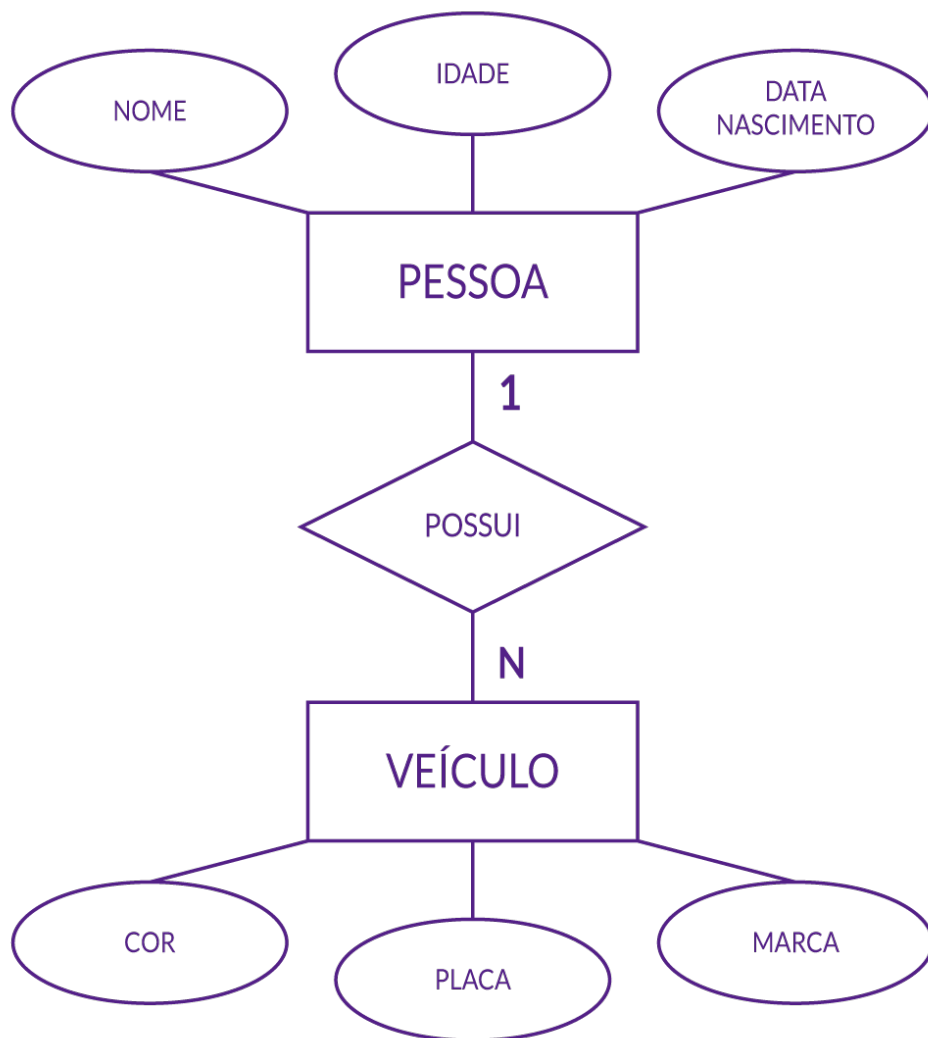
K. Nota: para inserir a linha “Marca”, eu cliquei na linha “Cor” com o botão direito do mouse e apertei “duplicate”.



L. Mas e a linha PK, FK2 ROW_2? Você se recorda sobre as chaves estrangeiras? Elas são a referência que nós temos a uma chave primária de outra tabela.

VEICULO	
PK	<u>IdVeiculo</u>
PK,FK2	<u>Row 2</u>
	Placa
	Cor
	Marca

M. No nosso fluxograma, mencionamos que PESSOA possui vários VEÍCULOS.



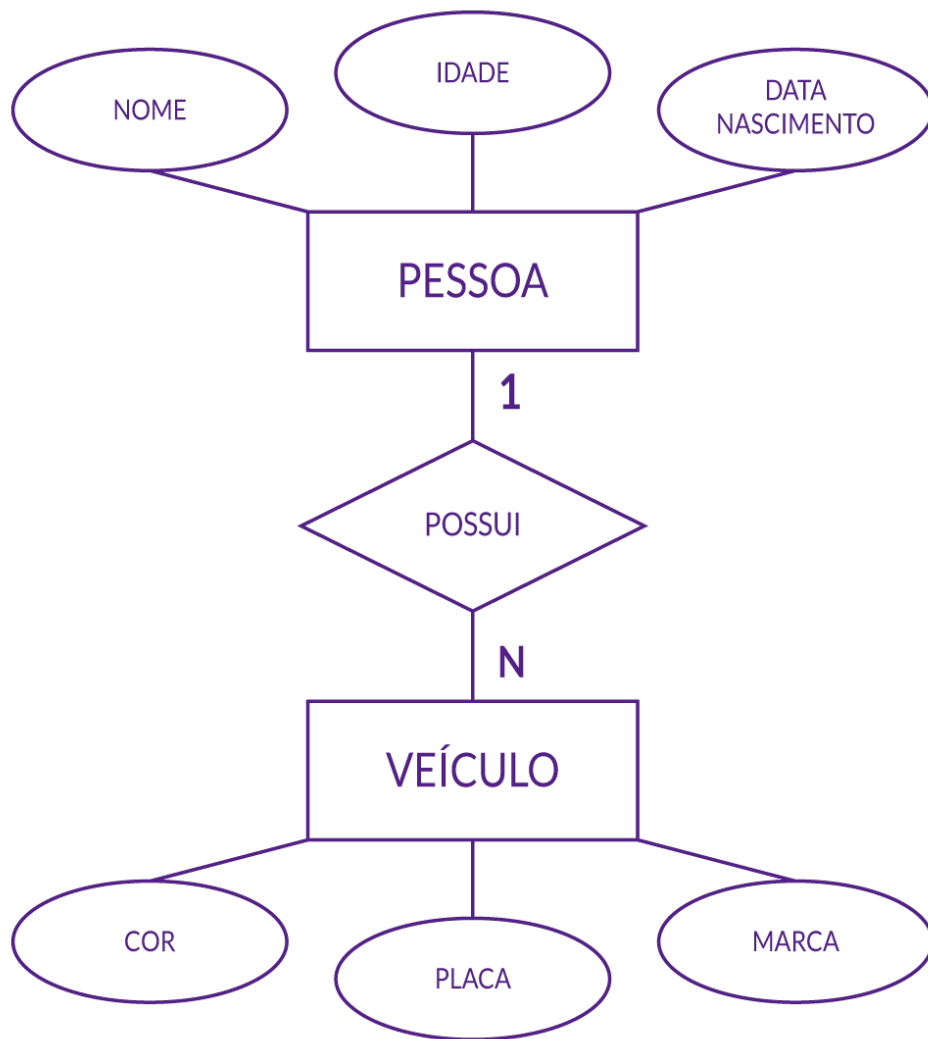
N. Se a pessoa é única, mas os veículos podem ser vários por pessoas, então temos que criar uma referência na tabela VEÍCULO para a tabela CLIENTE, que será PessoaID – costuma-se inverter o Id de posição em DERs para entendermos que é uma chave estrangeira. Também exclua o PK, e deixe só o FK, que significará Foreign Key (chave estrangeira em inglês).

Segue o resultado:

PESSOA	
PK	<u>IdPessoa</u>
	Nome
	Idade
	Data_de_Nascimento

VEICULO	
PK	<u>IdVeiculo</u>
FK	<u>PessoaID</u>
	Placa
	Cor
	Marca

O que está faltando? Olhe novamente seu fluxograma:



Já traduzimos as ENTIDADES para TABELA e os ATRIBUTOS para COLUNAS. O que falta? O relacionamento entre as tabelas ser representado graficamente.

Existem duas formas de representar:

- Notação UML: basicamente é a mesma que você usou no fluxograma.
 - O número na primeira entidade, um risco reto, e o número ou N na segunda entidade, ou vice-versa.
- Notação pé-de-corvo: notação específica para diagrama-entidade-relacionamento, que vamos usar agora no site apps.diagrams.

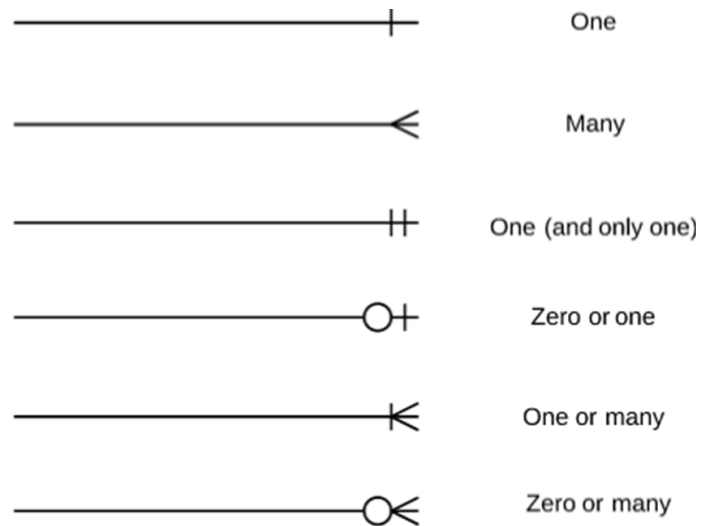
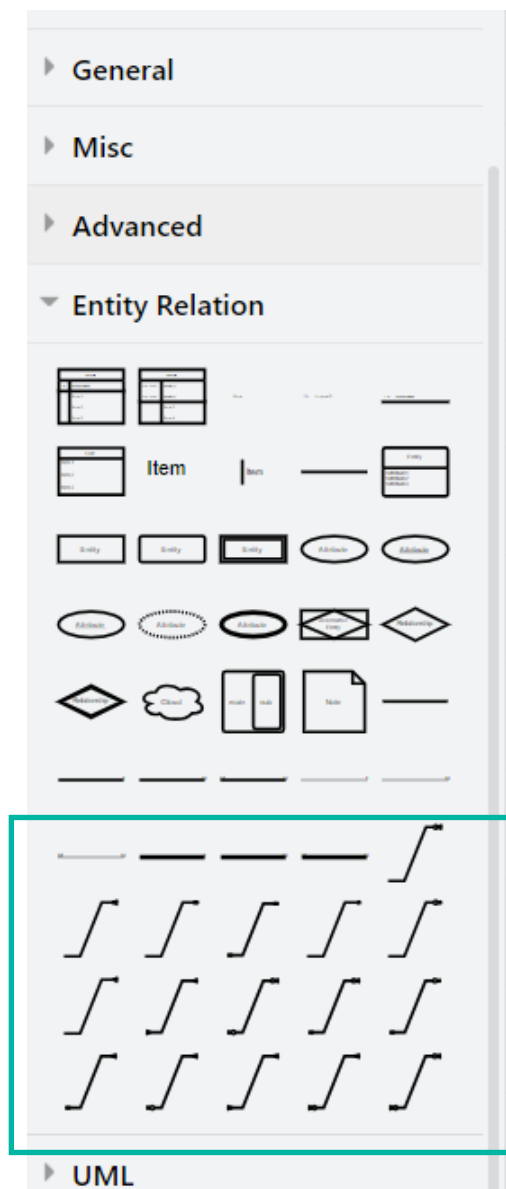


Figura: Notações pé-de-corvo

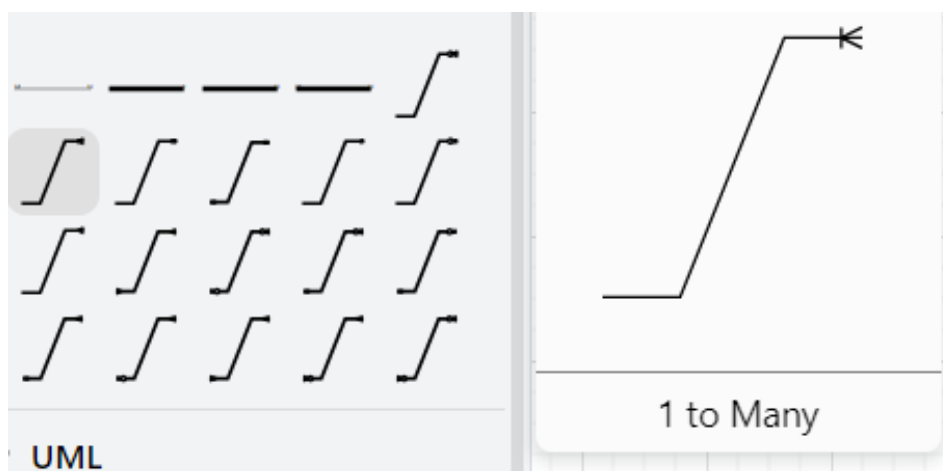
Fonte: <https://www.lucidchart.com/pages/pt/simbolos-de-diagramas-entidade-relacionamento>

Como o apps.diagram utiliza o pé-de-corvo, vamos seguir esse caminho.

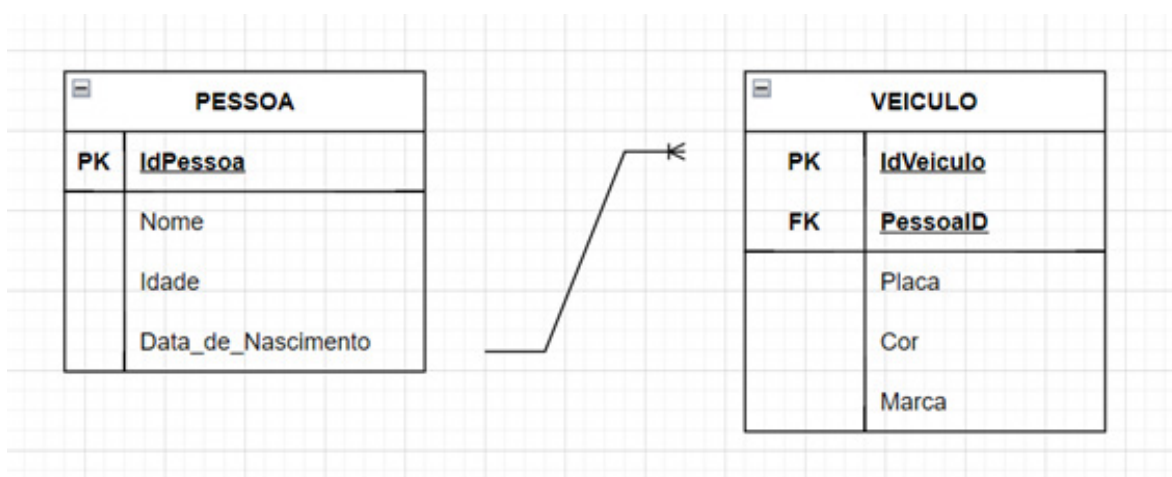
Ao lado esquerdo do painel do site de diagramação que estamos realizando você verá os tipos de relacionamento que podemos utilizar. Vide figura abaixo:



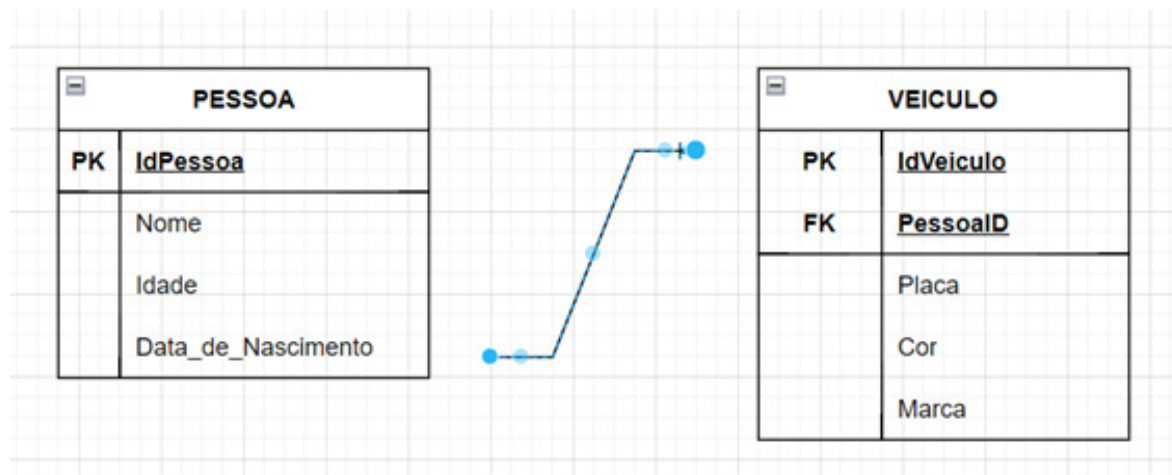
Nesse caso, como estamos fazendo a relação entre PESSOA para vários VEÍCULOS, utilizaremos o 1 to Many:



Arraste o item para sua tela.



Agora precisamos conectar corretamente a nossa chave primária na tabela Pessoa com a chave estrangeira da Pessoa na tabela Veículo. Para isso, clique uma vez no relacionamento criado na tela. Ele ficará azulado e dois pontos extremos aparecerão.



Pronto. Agora puxaremos o do lado esquerdo, que significa relação exclusiva de 1, para o lado da PK IdPessoa, na tabela PESSOA. E a ponta extrema da direita, que significa uma relação de muitos, para a chave estrangeira FK PessoaID, na tabela VEÍCULO.

