

# Técnico em Desenvolvimento de Sistemas

---

## Modelo de entidade e relacionamento: entidade, atributos, chave primária e estrangeira, relacionamentos

Inicialmente, é importante lembrar que a primeira etapa do projeto de banco de dados é a construção de um modelo conceitual e tem por objetivo estabelecer uma descrição abstrata dos dados a serem armazenados, independentemente da implementação ou da tecnologia a ser utilizada.

Segundo Heuser (2001), o **modelo de entidade e relacionamento (modelo ER)** é a técnica mais difundida e utilizada por ser representada graficamente por meio do **diagrama entidade e relacionamento (DER)**, e, portanto, pode ser considerada um padrão de fato para modelagem conceitual. Mesmo as técnicas de modelagem orientada a objetos, que têm surgido nos últimos anos, são baseadas nos conceitos da abordagem ER.

Neste material, você vai estudar os conceitos de modelagem entidade e relacionamento, seus atributos, chave primária, chave estrangeira, generalização/especialização.

Por enquanto, será utilizada a notação original de Peter Chen, criador da abordagem entidade e relacionamento, em 1976, em conjunto com os diagramas do MySQL Workbench 8.0, os quais, no momento da elaboração desse material, são a versão mais atual disponível.

## Conceitos

## Entidade

---

Segundo Heuser (2001, p. 12), é o conjunto de objetos da realidade modelada sobre os quais se deseja manter informações no banco de dados.

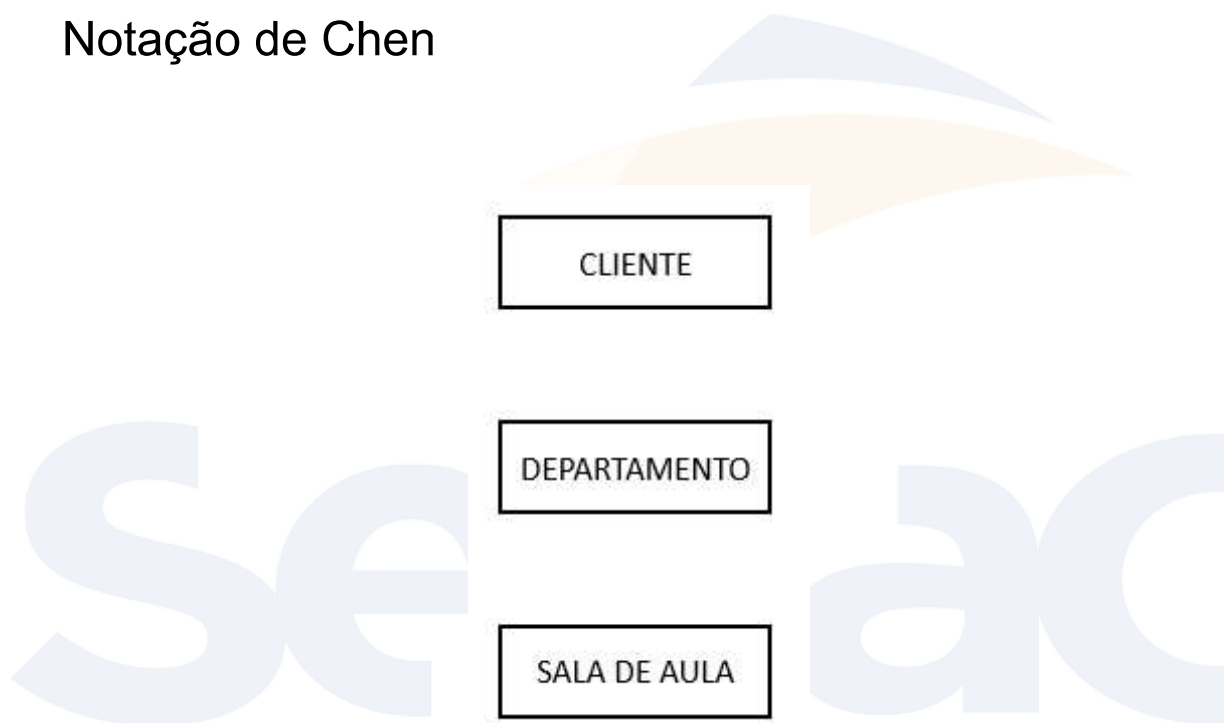
Como o modelo entidade e relacionamento, ou simplesmente modelo ER, é uma abstração do banco de dados (BD), somente os objetos sobre os quais se quer manter informações são importantes e devem constar na modelagem.

Considere, por exemplo, um sistema de informações de uma clínica veterinária. Algumas entidades podem ser caracterizadas por tutores dos *pets*, os *pets*, a espécie etc. Em outro exemplo, um sistema de informações bancárias pode conter os clientes, as contas-correntes, as contas-poupança, as contas-salário, os cartões de crédito etc.

Uma entidade pode ser um objeto concreto (cliente, *pet*, automóvel), mas também podem ser mapeados objetos abstratos (departamento, sala de aula, conta-corrente).

Nos diagramas ER, as entidades são representadas por **retângulos**, que contêm o nome da entidade. Observe a seguir o quadro comparativo entre a notação de Chen original e os diagramas ER do MySQL Workbench.

## Notação de Chen



## Diagrama ER no MySQL Workbench



Quadro 1 – Quadro comparativo entre entidades na notação de Chen e no diagrama ER no MySQL Workbench

Fonte: Adaptado de Heuser (2001)

Nos materiais complementares na sala de aula você encontrará vídeos orientadores sobre o uso da ferramenta de diagramas do MySQL Workbench.



## Relacionamento

---

Para Heuser (2001, p. 13), relacionamento é o conjunto de associações entre as entidades.

Além de especificar as entidades que devem ser armazenadas no banco de dados, o modelo ER deve permitir a especificação das propriedades dos objetos. A associação entre esses objetos é uma das propriedades das quais se pode querer manter informações.

Nos diagramas ER de Chen, os relacionamentos são representados por **losangos**, contendo os nomes dos relacionamentos ligados por linhas aos retângulos e, no MySQL Workbench, transformam-se em linhas e, em alguns casos, em entidades (o que você verá em seguida).

Agora, considere um DER contendo duas entidades, Aluno e Laboratório de Informática, e a necessidade de manter a informação de qual laboratório de informática em que cada aluno está lotado. Observe a seguir o quadro comparativo entre a notação de Chen original e os diagramas ER do MySQL Workbench (quadro 2).

## Notação de Chen



## Diagrama ER no MySQL Workbench



Quadro 2 – Quadro comparativo entre relacionamentos na notação de Chen e no diagrama ER no MySQL Workbench

Fonte: Adaptado de Heuser (2001)

O DER no quadro 2 representa graficamente que é preciso manter no banco de dados informações sobre:

- ◆ Um conjunto de objetos identificados como alunos (entidade “alunos”)
- ◆ Um conjunto de objetos identificados como laboratórios de informática (entidade “laboratório de informática”)
- ◆ Um conjunto de associações entre “aluno” e “laboratório de informática”, ligando um aluno a um laboratório (relacionamento “lotação”)

## Autorrelacionamento

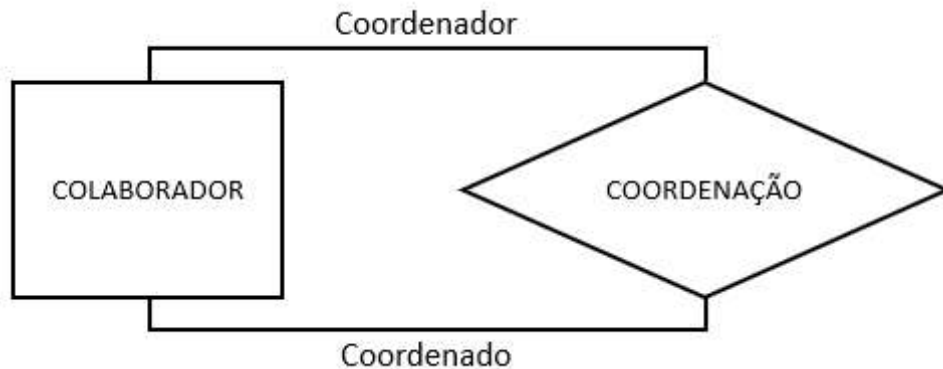
---

O autorrelacionamento nada mais é do que um relacionamento entre ocorrências de uma mesma entidade.

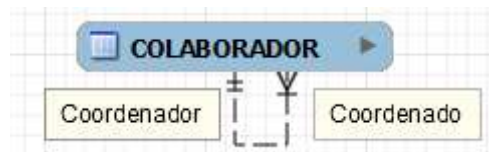
Quando há essa ocorrência, é necessário um conceito adicional que é o papel de cada entidade no relacionamento. Esse papel deve expressar a função que uma instância de entidade cumpre dentro da instância de um relacionamento.

Para facilitar o entendimento desse conceito, imagine a entidade “colaborador” e o relacionamento em que um colaborador “coordena” um grupo de colaboradores. Como se representa essa situação? Observe o quadro 3.

## Notação de Chen



## Diagrama ER no MySQL Workbench



Quadro 3 – Quadro comparativo entre relacionamentos na notação de Chen e no diagrama ER no MySQL Workbench

Fonte: Adaptado de Heuser (2001)

## Cardinalidade em relacionamentos

Uma propriedade importante dos relacionamentos é a quantidade de ocorrências de uma entidade que pode estar associada a uma ocorrência por meio do relacionamento. Essa propriedade é conhecida por cardinalidade.

Portanto, cardinalidade é “**o número de ocorrências de entidade associadas a uma ocorrência da entidade em questão por meio do relacionamento**” (HEUSER, 2001).

## Cardinalidade máxima



A melhor maneira de compreender o conceito de cardinalidade é por meio de exemplos. Logo, considere as seguintes cardinalidades máximas:



Na construção de diagramas ER não há necessidade de representar cardinalidades máximas maiores que 1, ou seja, para cardinalidades máximas, são necessárias apenas duas representações: cardinalidade máxima **1** e a cardinalidade máxima **n**, para representar qualquer cardinalidade maior que 1 e também pode ser lida como **muitos(as)**.

Agora, você pode deduzir que, no exemplo anterior, a cardinalidade máxima de "laboratório de informática" é **n** no relacionamento "lotação".

Veja no quadro 4 a representação das cardinalidades na notação de Chen e no MySQL Workbench.

## Notação de Chen



## Diagrama ER no MySQL Workbench



Quadro 4 – Quadro comparativo entre cardinalidade máxima na notação de Chen e no diagrama ER no MySQL Workbench (lê-se que “cada laboratório de informática lota muitos alunos; cada aluno é lotado em um laboratório de informática”)

Fonte: Adaptado de Heuser (2001)

A cardinalidade de uma entidade, por convenção, vai anotada no lado oposto do relacionamento ao qual se refere. No exemplo do quadro 4, percebe-se que a cardinalidade da máxima entidade “aluno” (1) está anotada ao lado da entidade “laboratório de informática” e a cardinalidade máxima da entidade “laboratório de informática” (n) está anotada ao lado da entidade “aluno”.

## Relacionamento binário

Relacionamentos binários são aqueles que contêm **duas ocorrências de entidade**.

## Classificação de relacionamentos binários

A classificação dos relacionamentos binários é definida por conta das cardinalidades máximas das entidades envolvidas no relacionamento.

## Um para um (1:1)

Ocorre quando a instância de uma entidade se relaciona com no máximo uma instância de outra entidade. Veja o exemplo:

### Notação de Chen



### Diagrama ER no MySQL Workbench



Quadro 5 – Quadro comparativo entre classificação “um para um” na notação de Chen e no diagrama ER no MySQL Workbench (lê-se “cada computador está alocado a um aluno; cada aluno está alocado a um computador”)

Fonte: Adaptado de Heuser (2001)

## Um para muitos (1:n)

Ocorre quando a instância de uma entidade se relaciona com muitas instâncias de outra entidade. Veja o exemplo:

### Notação de Chen



## Diagrama ER no MySQL Workbench



Quadro 6 – Quadro comparativo entre classificação “um para muitos” na notação de Chen e no diagrama ER no MySQL Workbench (lê-se “cada categoria contém vários produtos; cada produto tem uma categoria”)

Fonte: Adaptado de Heuser (2001)

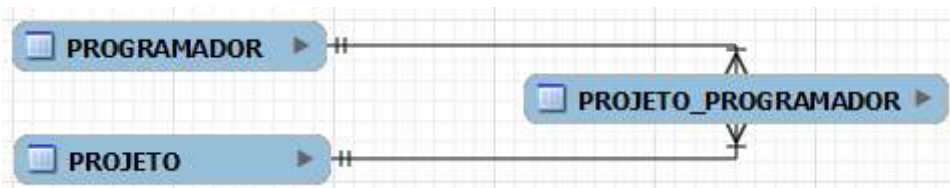
## Muitos para muitos (n:n)

Ocorre quando muitas instâncias de uma entidade se relacionam com muitas instâncias de outra entidade. Confira o exemplo:

## Notação de Chen



## Diagrama ER no MySQL Workbench



Quadro 7 – Quadro comparativo entre classificação “um para muitos” na notação de Chen e no diagrama ER no MySQL Workbench (lê-se “cada programador é alocado a vários projetos; cada projeto é alocado a vários programadores”)

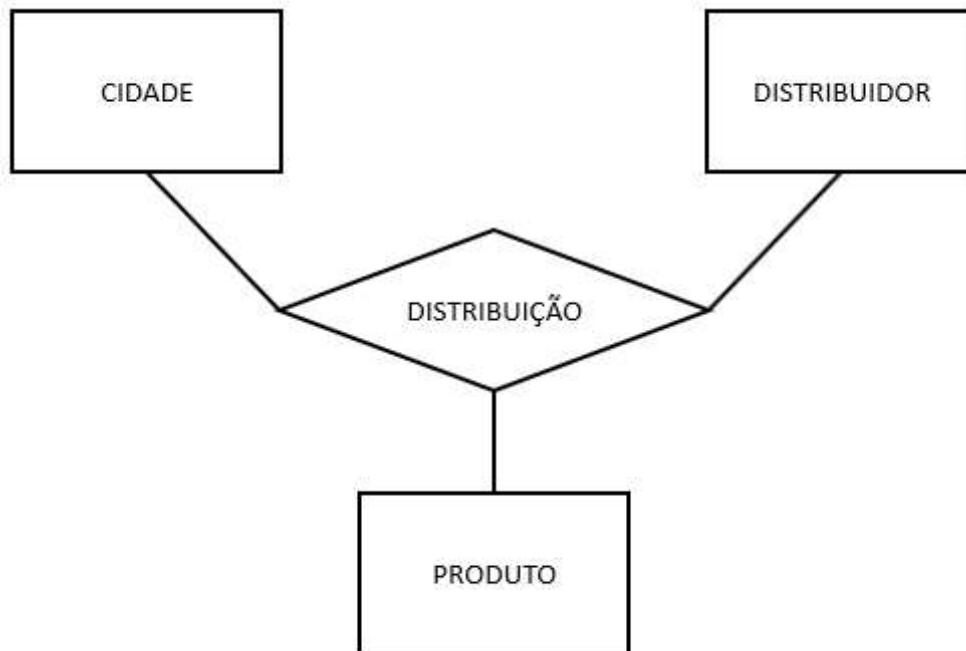
Fonte: Adaptado de Heuser (2001)

No MySQL Workbench, quando se tem um relacionamento “muitos para muitos” (n:n), esse relacionamento se transforma em uma nova entidade.

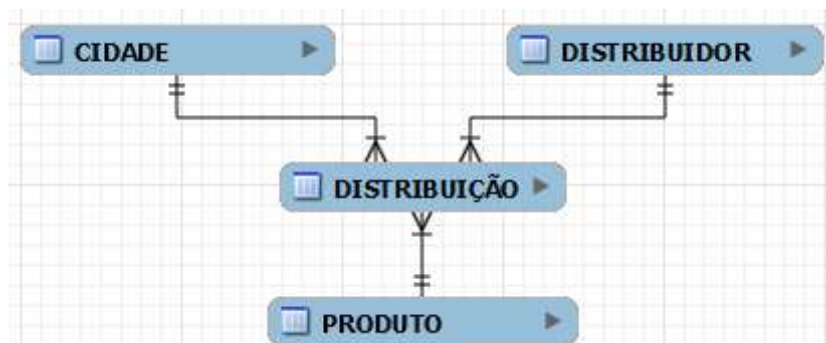
## Relacionamento ternário

Segundo Heuser (2001), os modelos ER permitem a definição de relacionamentos de grau maior que dois, ou seja, pode haver relacionamentos ternários, quaternários. Até o momento, foram utilizados somente relacionamentos binários. Observe, no quadro a seguir, o DER de um relacionamento ternário.

## Notação de Chen



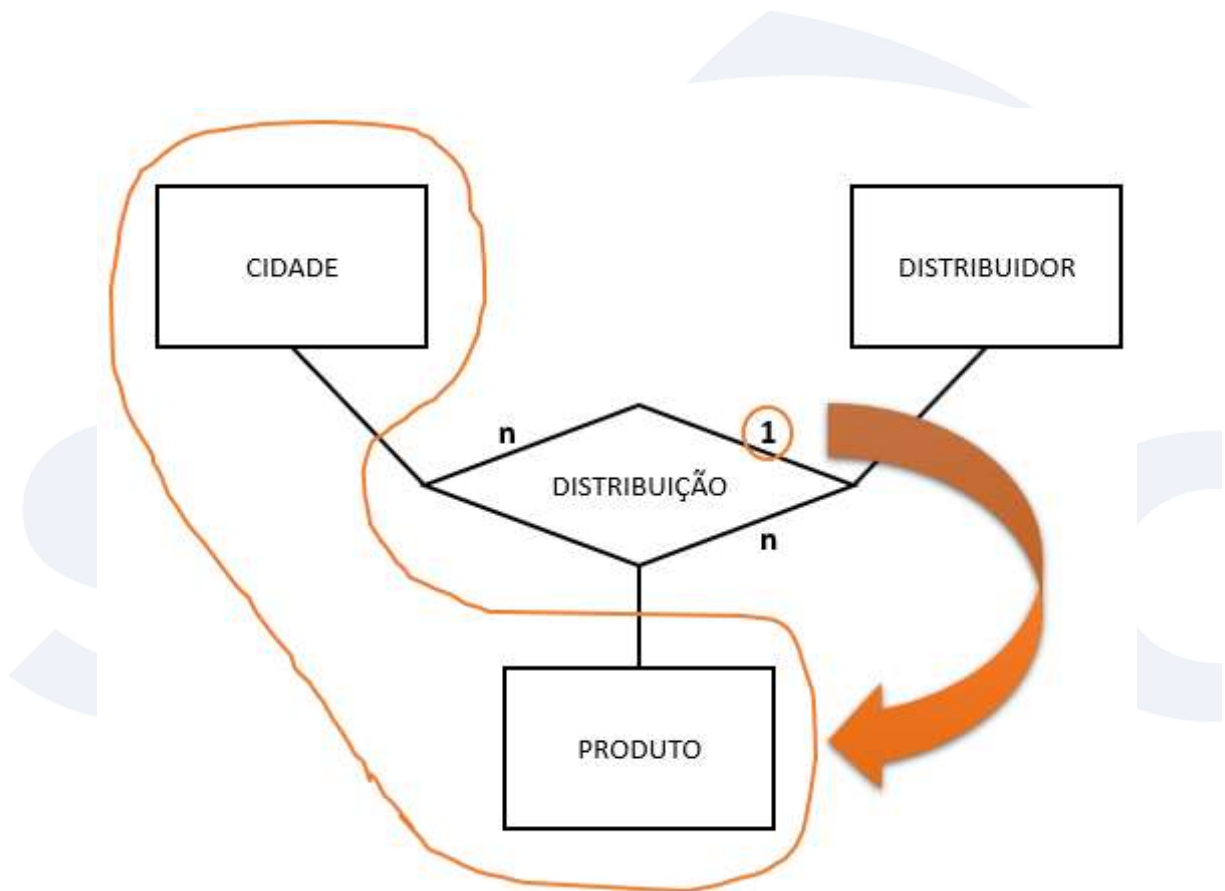
## Diagrama ER no MySQL Workbench



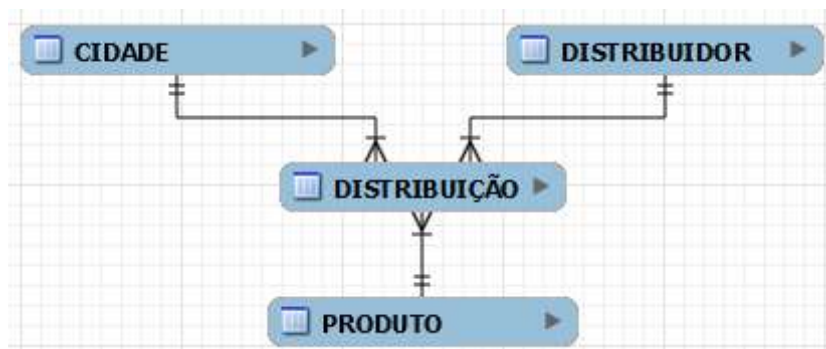
Quadro 8 – Quadro comparativo entre relacionamento ternário na notação de Chen e no diagrama ER no MySQL Workbench

No caso de relacionamentos ternários, a **cardinalidade máxima refere-se a pares de entidades**. Logo, em um relacionamento “R” e entre as três entidades “A”, “B” e “C”, a cardinalidade máxima de “A” e “B” indica quantas ocorrências de “C” podem estar associadas a um par de ocorrências de “A e B”. Veja o quadro a seguir.

## Notação de Chen



## Diagrama ER no MySQL Workbench



Quadro 9 – Quadro comparativo entre cardinalidade em relacionamento ternário na notação de Chen e no diagrama ER no MySQL Workbench

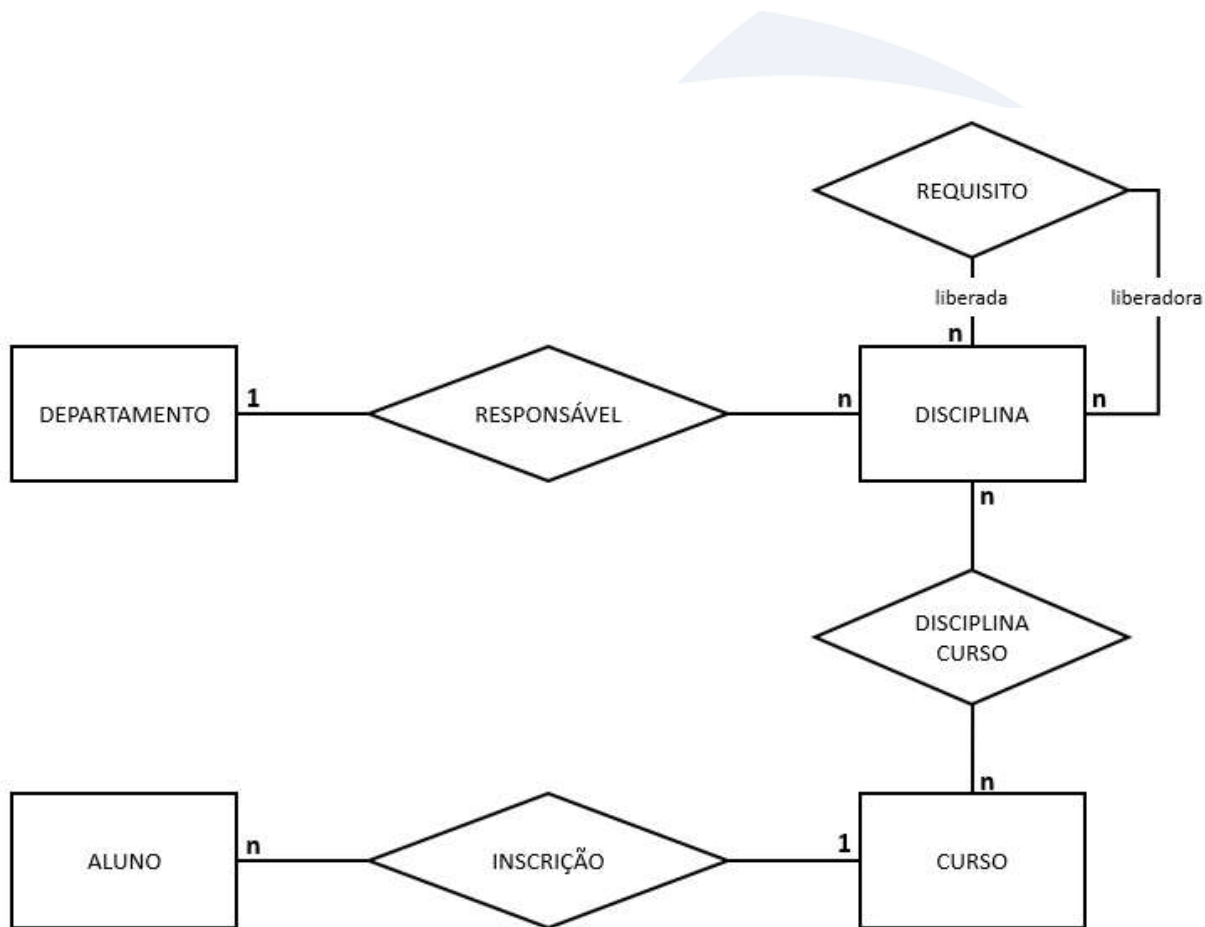
No exemplo do quadro 9, cada par de “cidade e produto” está associado a uma ocorrência de distribuidor, indicando que um distribuidor pode distribuir muitos produtos em muitas cidades.

Antes de prosseguir, é importante que você tenha uma visão mais abrangente, considerando que existem diversas entidades e relacionamentos contemplando todos os conhecimentos vistos até o momento. Observe o exemplo a seguir:

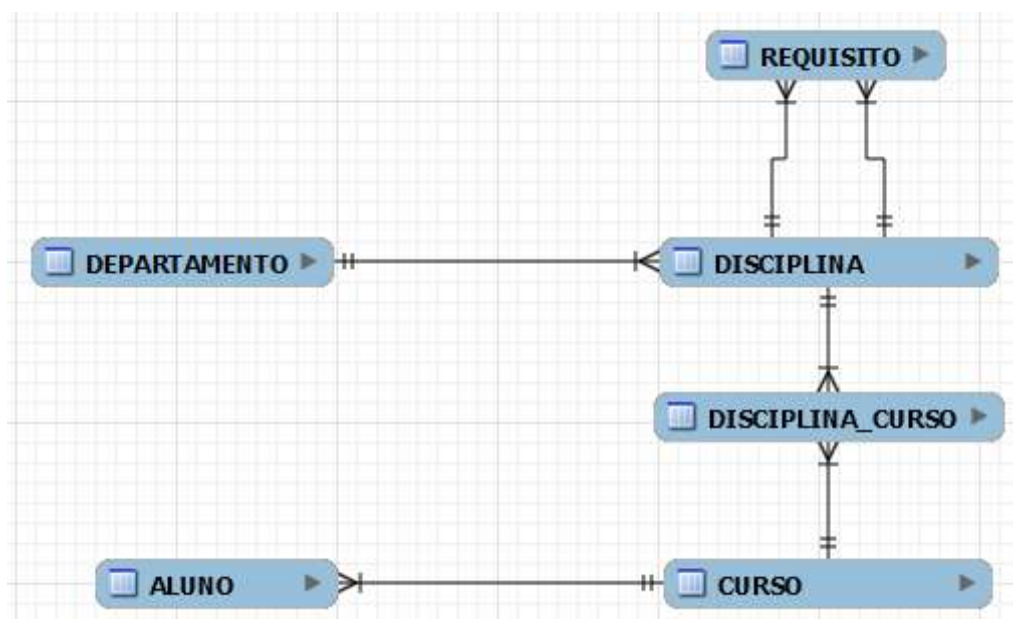




## Notação de Chen



## Diagrama ER no MySQL Workbench



Quadro 10 – Quadro comparativo entre o DER de controle acadêmico na notação de Chen e no diagrama ER no MySQL Workbench

O modelo do quadro 10, segundo Heuser (2001) descreve que:



- Deseja-se manter informações sobre alunos, cursos, disciplinas e departamentos.
- Deseja-se manter informações sobre a associação de alunos a cursos, de disciplinas a cursos, de disciplinas a departamentos e de disciplinas a disciplinas requisito.
- As cardinalidades expressam que:
  - Cada disciplina tem apenas um departamento responsável e um departamento é responsável por muitas disciplinas.
  - Uma disciplina pode ter muitas disciplinas requisitos e uma disciplina requisito pode liberar muitas disciplinas.
  - Uma disciplina pode pertencer a vários cursos e um curso pode ter várias disciplinas.
  - Um aluno está inscrito em apenas um curso e um curso pode ter vários alunos inscritos.

- ◆ Deseja-se manter informações sobre alunos, cursos, disciplinas e departamentos.
- ◆ Deseja-se manter informações sobre a associação de alunos a cursos, de disciplinas a cursos, de disciplinas a departamentos e de disciplinas a disciplinas requisito.
- ◆ As cardinalidades expressam que:
  - ◆ Cada disciplina tem apenas um departamento responsável e um departamento é responsável por muitas disciplinas.
  - ◆ Uma disciplina pode ter muitas disciplinas requisitos e uma disciplina requisito pode liberar muitas disciplinas.
  - ◆ Uma disciplina pode pertencer a vários cursos e um curso pode ter várias disciplinas.
  - ◆ Um aluno está inscrito em apenas um curso e um curso pode ter vários alunos inscritos.

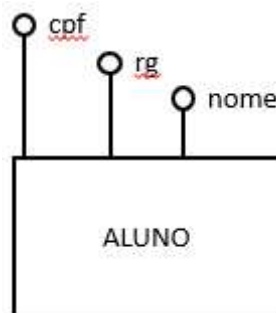
Agora que você concluiu o estudo dos principais conceitos e entendeu o que são relacionamentos, autorrelacionamentos, relacionamentos binários, relacionamentos ternários e as cardinalidades, aprofunde seu conhecimento e entendimento estudando o que são atributos e atributos identificadores. Aos poucos, você focará mais seus estudos nos DERs no MySQL Workbench.

## Atributo

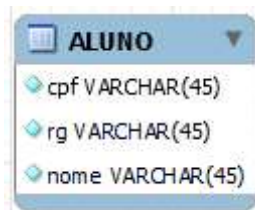
Para Heuser (2001), atributo representa o dado que é associado a cada ocorrência de uma entidade ou de um relacionamento. Portanto, pode-se concluir que atributos são os **dados dos quais se deseja manter registros acerca de uma entidade ou relacionamento**.

São representados graficamente na forma de pequenos círculos com seu identificador ao lado ligados às entidades ou aos relacionamentos por meio de linhas, conforme você pode observar nos quadros 11 e 12.

## Notação de Chen



## Diagrama ER no MySQL Workbench

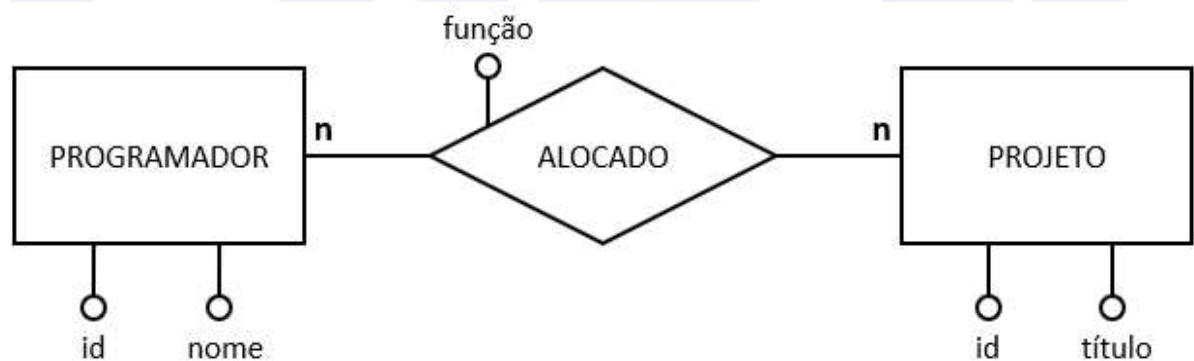


Quadro 11 – Quadro comparativo entre atributos de entidade na notação de Chen e no diagrama ER no MySQL Workbench

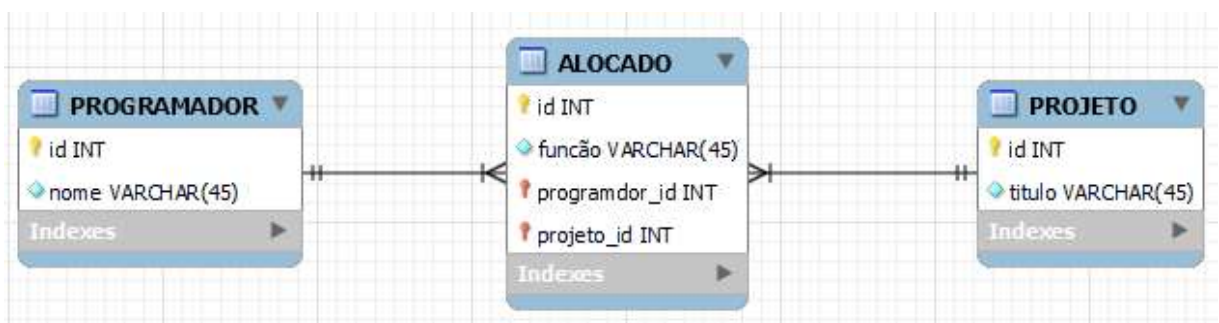
Fonte: Adaptado de Heuser (2001)

No MySQL Workbench, as entidades ou os relacionamentos dos quais se deseja manter registros são conhecidos como tabelas, por isso, os atributos são identificados no seu interior. Perceba que, ao lado dos atributos, existe o tipo de dado que será armazenado naquele atributo. Você pode ter informações sobre tipos de dados nos conteúdos sobre modelagem conceitual de banco de dados e sobre definição de dados desta unidade curricular.

## Notação de Chen



## Diagrama ER no MySQL Workbench



Quadro 12 – Quadro comparativo entre atributos de relacionamento na notação de Chen e no diagrama ER no MySQL Workbench

Fonte: Adaptado de Heuser (2001)

No MySQL Workbench, sempre que se tem um relacionamento do tipo “muitos para muitos” (n:n) entre duas entidades, esse relacionamento, obrigatoriamente, transforma-se em uma entidade, ou seja, transforma-se em tabela. Perceba que, no

relacionamento, guarda-se o “id” do relacionamento, a função do programador dentro do projeto, o “id” do programador alocado no projeto e o “id” do projeto em questão.

No exemplo do quadro 12, foram apresentados alguns atributos introduzindo o conceito de identificadores, que serão abordados a seguir utilizando-se somente o DER do MySQL Workbench.

## Identificadores

### Primary key (chave primária)

Qualquer entidade ou relacionamento dos quais se deseje manter os registros obrigatoriamente devem conter um atributo identificador. Esse identificador deve ser único, ou seja, deve diferenciar uma ocorrência da entidade das demais ocorrências da mesma unidade.

Em bancos de dados relacionais, esses atributos identificadores são conhecidos como “chave primária” e, na maioria dos casos, utiliza-se a nomenclatura de **id**, o seu tipo como **int** (números inteiros), not null (não aceita valores nulos ou vazios); marca-se como *autoincrement* (cria automaticamente o valor do **id** no momento em que é salvo no banco de dados) e também é marcado como *primary key* (chave primária), ou seja, o atributo identificador entre os demais. Observe a figura 1.

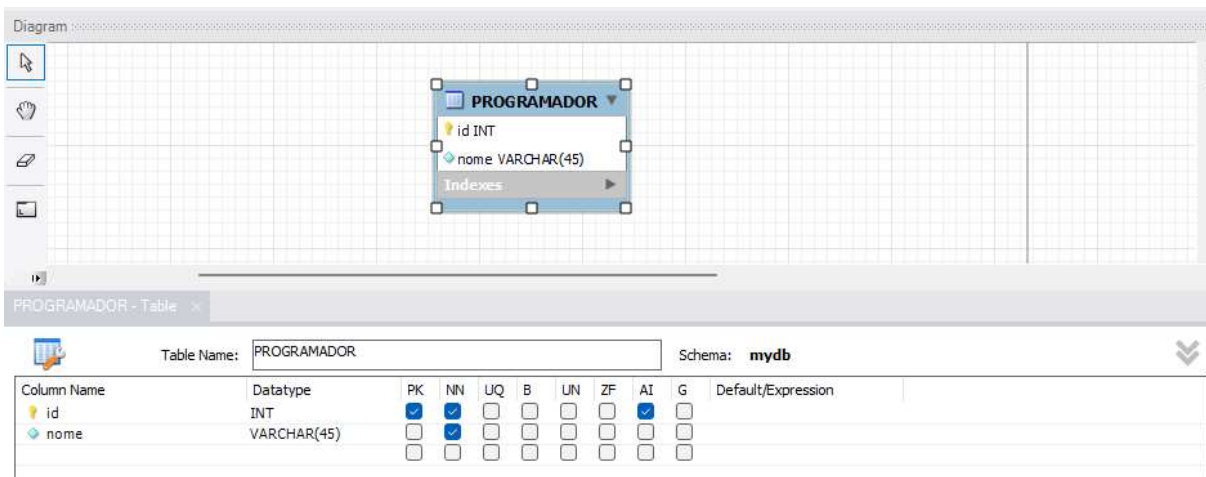


Figura 1 – Identificador simples ou chave primária

No exemplo da figura 1, é possível observar a presença de um único identificador, que é chamado de “identificador simples”. Em algumas situações, por conta dos requisitos ou das regras de negócios, pode haver situações nas quais será necessário utilizar mais de um atributo identificador, que é reconhecido como identificador composto. Observe a figura 2.

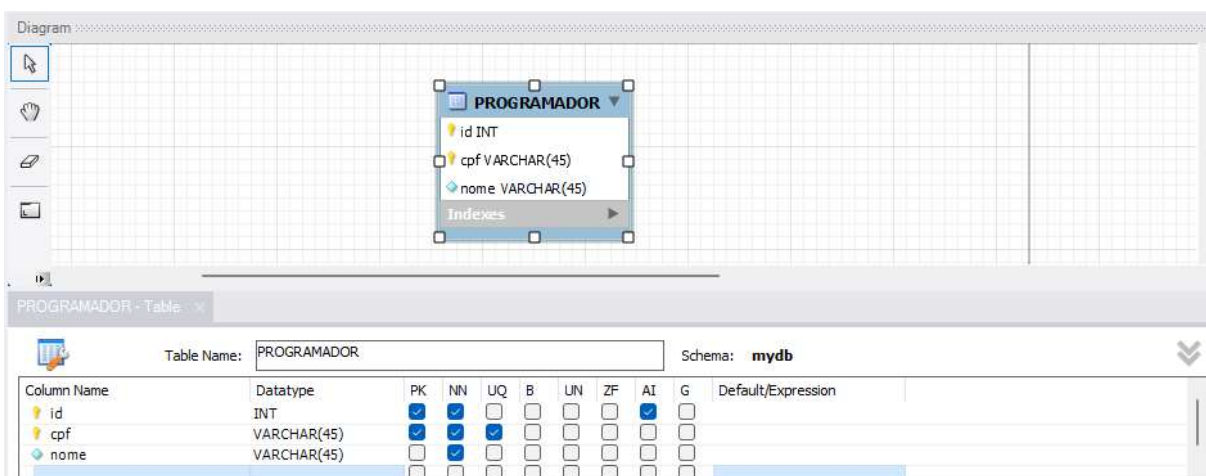


Figura 2 – Identificador composto ou chave primária composta

O atributo **cpf**, apesar de ser numérico, é tipificado como **VARCHAR**, pois não há necessidade de operações aritméticas com ele, portando, ele é guardado em banco na forma de texto. Pelo mesmo motivo, números de telefones são tipificados dessa maneira.

## Foreign key (chave estrangeira)



Para Heuser (2001), uma “chave estrangeira” é um identificador ou mais identificadores, cujos valores aparecem, obrigatoriamente, como “chave primária” em outra entidade. Sendo assim, a “chave estrangeira” é o mecanismo que permite a implementação dos relacionamentos nos bancos de dados relacionais.

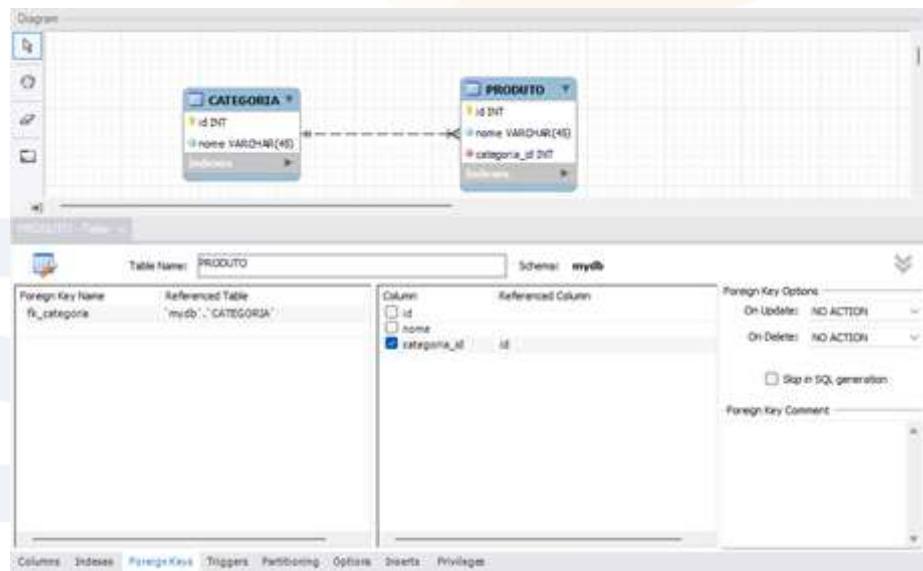


Figura 3 – Chave estrangeira (1:n)

No relacionamento binário (1:n) da figura 3, é possível observar a presença de uma única chave estrangeira na tabela “produto”. Em algumas situações, você terá relacionamentos (n:n) que, obrigatoriamente, transformar-se-ão em uma nova tabela e as chaves primárias serão armazenadas nessa nova tabela. Portanto, em muitas situações, haverá tabelas com duas ou mais chaves estrangeiras. Observe a figura 4.



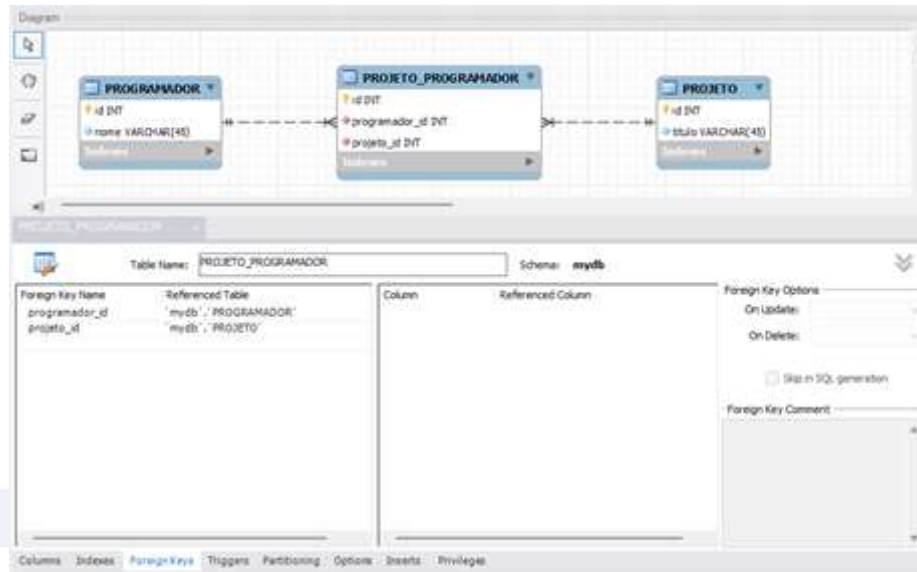


Figura 4 – Chave estrangeira (n:n)

Que tal agora realizar alguns desafios?

Construa o diagrama de entidade e relacionamento apresentado no quadro 10 do conteúdo desta disciplina. A seguir, você terá o texto para ajudá-lo a identificar as cardinalidades. Lembre-se de criar os atributos das entidades (o **id** e mais um são suficientes), as chaves primárias e estrangeiras e realizar a criação dos relacionamentos.

- ◆ Deseja-se manter informações sobre alunos, cursos, disciplinas e departamentos.
- ◆ Deseja-se manter informações sobre a associação de alunos a cursos, de disciplinas a cursos, de disciplinas a departamentos e de disciplinas a disciplinas requisito.
- ◆ As cardinalidades expressam que:
  - ◆ Cada disciplina contém apenas um departamento responsável e um departamento é responsável por muitas disciplinas.
  - ◆ Uma disciplina pode ter muitas disciplinas requisitos e uma disciplina requisito pode liberar muitas disciplinas.
  - ◆ Uma disciplina pode pertencer a vários cursos e um curso pode ter várias disciplinas.
  - ◆ Um aluno está inscrito em apenas um curso e um curso pode ter vários alunos inscritos.

Com base no texto a seguir, que representa o escopo de um projeto de software, construa o DER.

A Paróquia de São João deseja informatizar a festa junina que promove todos os anos. Padre Chico, muito responsável, contratou uma equipe de analistas e forneceu as seguintes informações: é necessário cadastrar as barraquinhas de diversões da festa, com informações como: localização, descrição, categoria, pessoas que trabalharão nela e pessoa responsável. Categorias são cadastradas apenas com uma descrição. Em uma barraquinha, trabalham diversas pessoas da comunidade, as quais também são cadastradas para facilitar um novo contato a cada ano. Quando uma barraquinha é cadastrada, automaticamente todas as pessoas que trabalharão nela devem ser selecionadas, informando a hora em que iniciarão o trabalho na barraquinha e a hora de término (uma pessoa pode trabalhar em mais de uma barraquinha na festa, para não se sentir entediada). Cabe salientar que cada barraquinha é de responsabilidade de uma dessas pessoas. Sobre as pessoas, é importante manter: nome, endereço e telefone para contato. Os prêmios também devem ser cadastrados, sendo que um mesmo tipo de prêmio pode ser

distribuído em diferentes barraquinhas de brinquedo. Sobre os prêmios, é importante manter a descrição e a quantidade em estoque. Cada vez que um participante da festa ganhar um prêmio, será necessário cadastrar a premiação, informando o nome do ganhador, a idade dele, o prêmio e em que barraquinha ele ganhou. Isso é importante para saber quem ganhou mais prêmios. No momento em que uma premiação é cadastrada, automaticamente é atualizado o estoque de prêmios disponíveis. Todas essas funcionalidades são realizadas pelos funcionários, em geral, da paróquia.

## Encerramento

Você chegou ao final deste conteúdo, construindo sólidos conhecimentos que servirão de base para as próximas unidades curriculares. Você aprendeu o conceito de entidade, que é a abstração dos objetos que se deseja armazenar em banco de dados, e entendeu que os atributos de uma entidade são os dados relevantes ao *software*, os quais devem ser guardados das entidades mapeadas. Aprendeu sobre a importância das chaves primárias e que as chaves estrangeiras são as responsáveis por estabelecer os relacionamentos entre as entidades. Cabe salientar que os DERs fazem parte da documentação do *software* a ser desenvolvido e todos esses conceitos serão retomados e farão mais sentido no decorrer das próximas unidades curriculares.