

# Data Modeling

## ► ER Modeling

# Entity Relationship Modeling

- ▶ Allows you to model the real world as **entities** and **relationships**
- ▶ Graphical notation is easy to understand, allowing technical architects to convey complex designs to others
- ▶ Can be used to model databases table structure

# Entity Relationship Modeling

## ► Entities

- The object which is holding the data
- Are often recognizable objects like persons, places, events or concepts
  - Patient
  - Employee
  - Company
  - Project
- Each Entity is analogous to a table in a database

# Entity Relationship Modeling

- ▶ Entities may be independent or dependent
- ▶ Independent entities do not rely on other entities for identification
- ▶ Dependent entities rely on other entities for identification

# Entity Relationship Modeling

## ▶ Relationships

- Relationships may exist between entities
  - Project Managers have projects
  - Car dealerships have cars
  - Facebook Friends have other friends
  - Observations belong to Encounters

# Connectivity

- ▶ **One-to-One (1:1)** One instance of a particular entity is associated to exactly one instance of another entity
  - Example: Each unique instance of a student is given a unique instance of a chair. There is no sharing of chairs

# Connectivity

- ▶ **One-to-many (1:n)** An instance of a particular entity is assigned 0, 1, or many instances of another entity. Each of the other entities only belongs to one entity.
  - One album has many songs in it, however, each song belongs to exactly one album

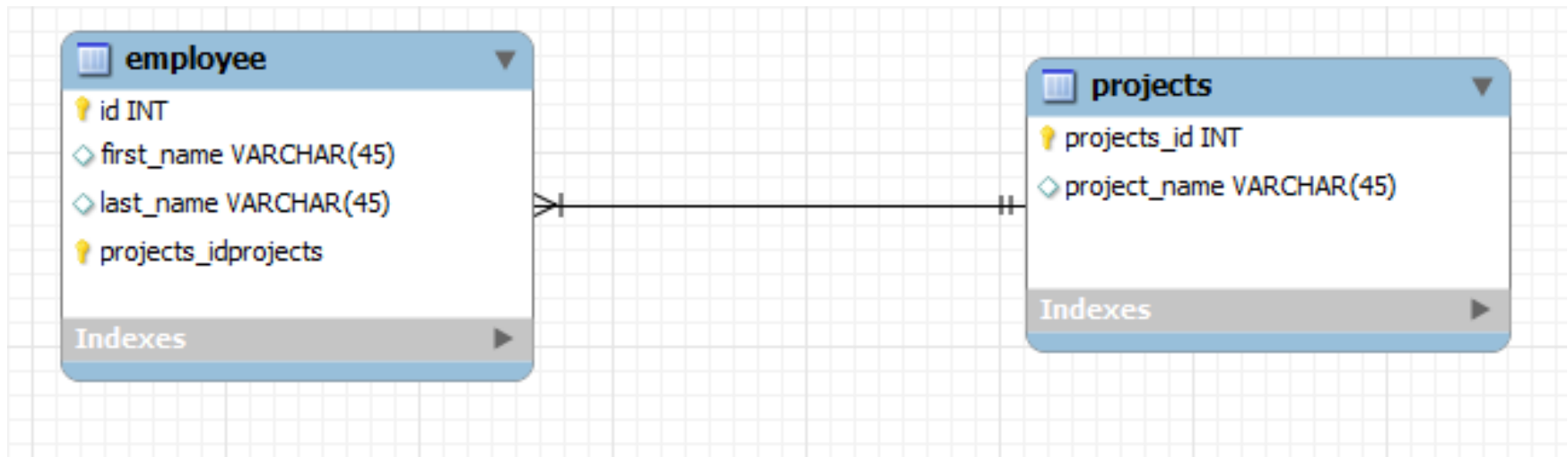


# Connectivity

- ▶ **Many-to-Many (m:n)** An instance of entity A is assigned 0, 1, or many instances of entity B. Entity B is assigned 0,1 or, many instances of entity A.
  - Employees are assigned multiple projects. Each project has many employees

# Entity Relationship Modeling

- ▶ A diagram modeling a one-to-many relationship between projects and employees



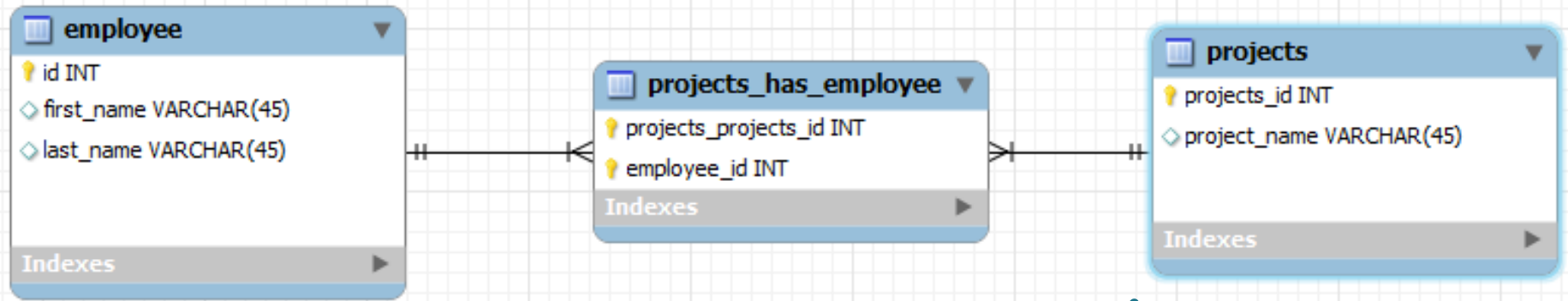
# Entity Relationship Modeling



# Entity Relationship Modeling

- ▶ But wait a second, employees could be on more than one project couldn't they?
- ▶ Yes...let's change our diagram and make it reflect the many-to-many relationship

# Entity Relationship Modeling

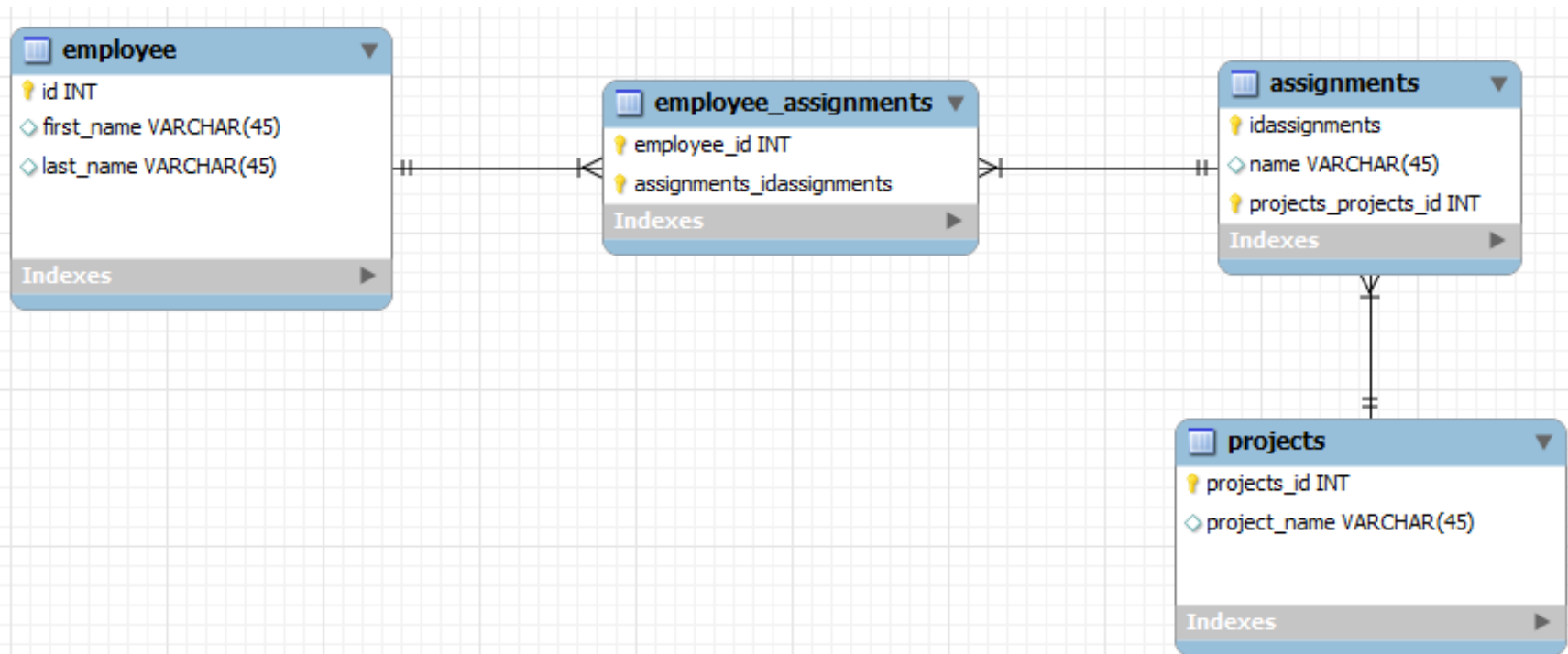


Employees are on many projects

Projects have many employees

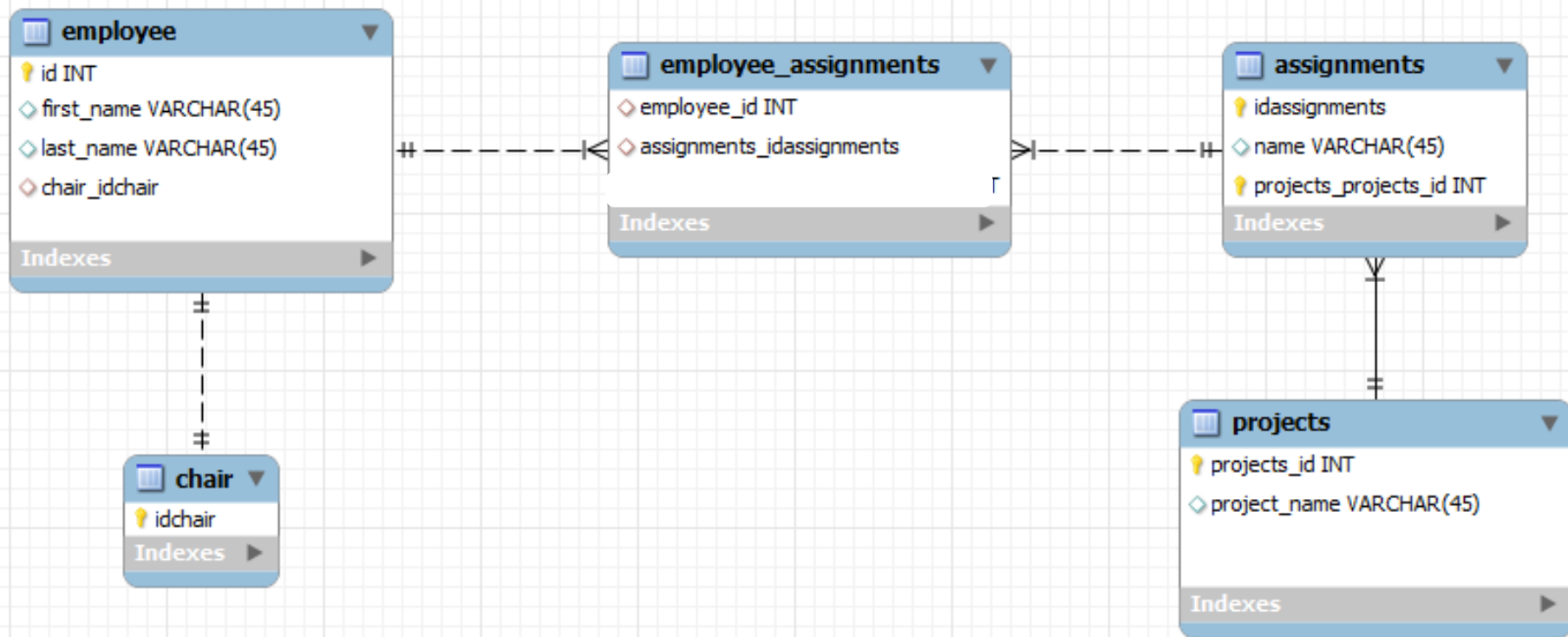
# Entity Relationship Modeling

- ▶ Let's give employees assignments to work on. Projects consist of many assignments



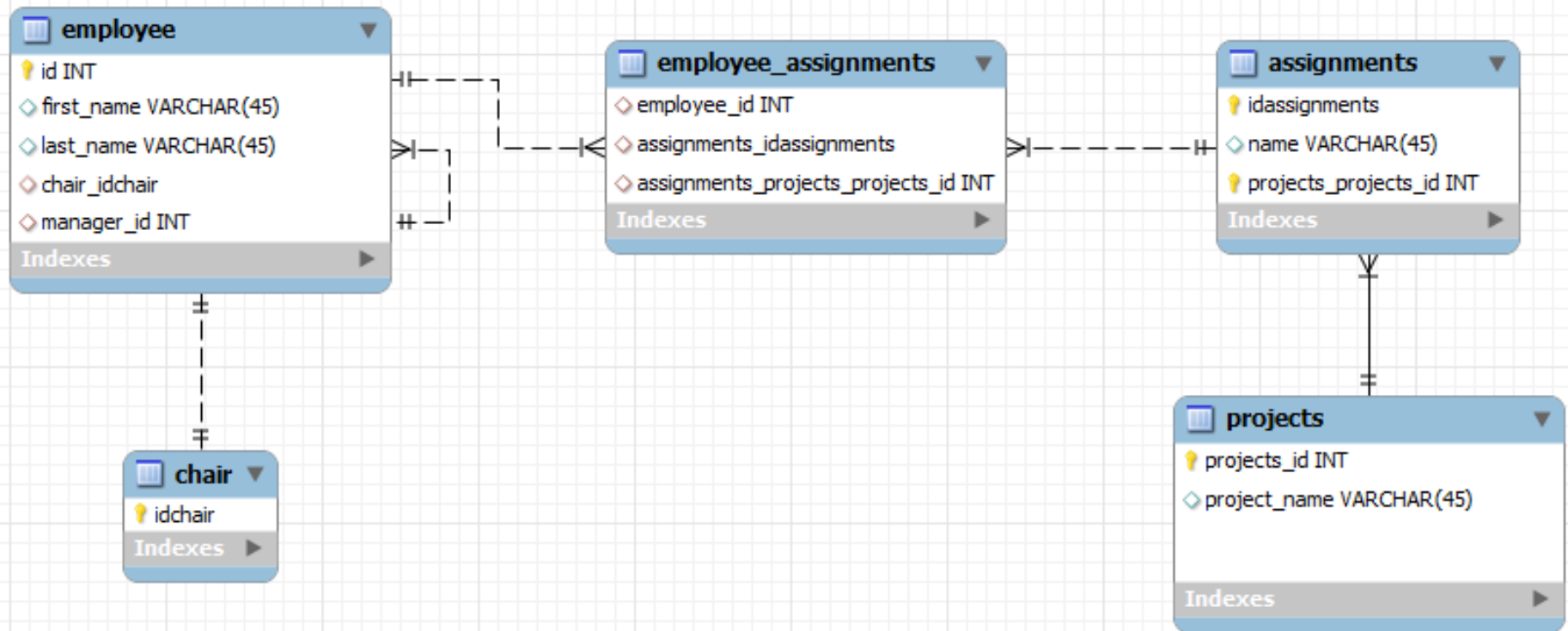
# Entity Relationship Modeling

- ▶ Let's give each employee a chair



# Entity Relationship Modeling

- ▶ Let's add a manager. Each manager has multiple employees. This is a **recursive** relationship





# Get the Design Tool

▶ [\\intare\Public\Software\Database](#)

