

Module 06 – Entity Relationships

Module Objectives

Discuss the importance a data models

List purposes of models

Identify the key components of ER modeling

Discuss the importance a data models

Data modeling is the process of capturing the important concepts and rules that shape a business and depicting them visually on a diagram. This process is achieved through modeling the relationship between each entity in the database. When the graphical representation of this model is drawn, it creates an entity relationship diagram, an ERD.

ERDs show the graphical interactions and relations between each entity in your database. ERDs display each entity and all of that entity's attributes. Moreover, the focus at this point is not on the creation of entities, but how to display the relationships in a graphical format between each entity. This is where an Entity Relationships Diagram comes in handy. When you can show a visual illustration of something to someone, it is often easier to understand.

List purposes of models

There are several different purposes for modeling.

Modeling helps to:

- describe exactly the information needs of a business.

- prevent mistakes and misunderstanding.

- forms important ideal system documentation.

- forms a sound basis for a physical database design.

A model is important to a business because it documents the processes of the business. It also takes into account regulations and laws that govern the particular industry the business is working within.

Identify the key components of ER modeling

The three main components of ER modeling are:

Entities – the object itself.

Attributes – the characteristics that describe the object.

Relationships – how one object relates or depends on another object.

An entity represents a life-like object, such as a student, an automobile, or a department. An entity is your actual table. The attributes of your entity are the characteristics that describe that entity. These attributes are your column names from your table. If your entity were student, some of your attributes may be name, gpa, field of study, etc. If your entity were an automobile, some of your attributes may be vehicle type, number of doors, size of engine, etc. If your entity were department, some of your attributes may be department number, department name, location, etc.

Each entity could represent any object that you need to store data about in your database. Each of those entities could have numerous attributes that describe them. There is no limit to the different types of data you could store based upon various entities.

There are three basic types of relationships:

One-to-One

In a one-to-one relationship, there exists one instance of an object in each table.

One-to-Many

In a one-to-many relationship, there exists one instance of an object in one table, but zero or many instances of that object in the other table.

Many-to-Many

In a many-to-many relationship, there exist many instances of an object in one table and many instances of that object in another table.

Of all the relationships, one-to-many is the most common. Many developers often claim that a one-to-one relationship is a waste of database space. Furthermore, most developers claim that a many-to-many relationship is extremely poor design and difficult to manage.

Take for instance this example.

Consider the following tables:

STUDENT

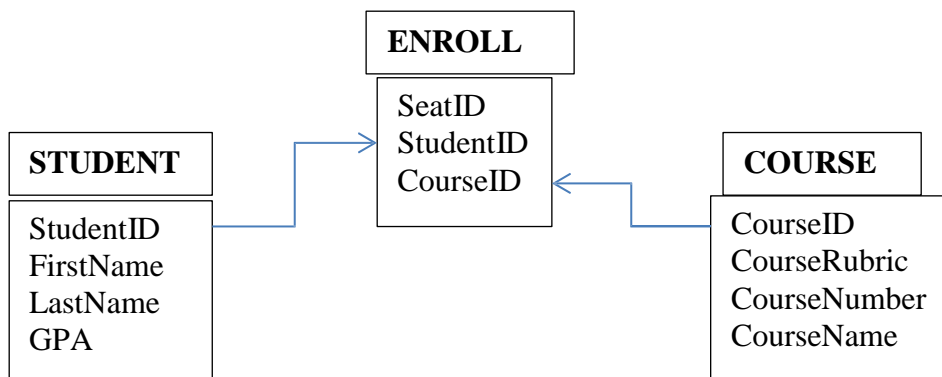
Student ID, First Name, Last Name, GPA

COURSE

Course ID, Course Rubric, Course Number, Course Name

ENROLL

Seat ID, Student ID, Course ID



Using the simple ERD, you can observe the relationships or dependency between these three tables. The STUDENT table has a relationship or dependency on the ENROLL table. The COURSE table has a relationship or a dependency on the ENROLL table.