Entity Relationship Data Model or, a Diagrammatic ER Diagram (ERD):

- The entity-relationship (E-R) data model was developed to facilitate **database design** by allowing **specification of an overall logical structure** of a database.
- The E-R model is very useful in **mapping the meanings and interactions** of real-world enterprises onto a conceptual schema.
- The E-R data model employs three basic concepts: entity sets, relationship sets, and attributes.

Entity Sets

1. An entity is a **"thing" or "object" in the real world.**
2. An entity **has a set of properties**, and the values for some set of properties may uniquely identify an entity.
3. An **entity may be concrete**, such as a person or a book, **or it may be abstract**, such as a course, a course offering, or a flight reservation.
4. An entity set is a **set of entities of the same type** that share the same properties. E.g. An entity set is a set of entities of the same type that share the same properties.
5. Entity sets **do not need to be disjoint.**

Relationship Set

1. A relationship is an **association among several entities**.
2. A relationship **set is a set of relationships** of the same type.
3. Formally**, it is a mathematical relation** on $n \geq 2$ (possibly nondistinct) entity sets.

   If E1, E2,..., En are entity sets, then a relationship set R is a subset of :

   $$\{(e1,e2,...,en) \mid e1 \in E1, e2 \in E2, ..., en \in En\}$$

   where (e1,e2,...,en) is a relationship.

4. Consider the two entity sets instructor and student. We define the relationship set advisor to denote the association between instructors and students.

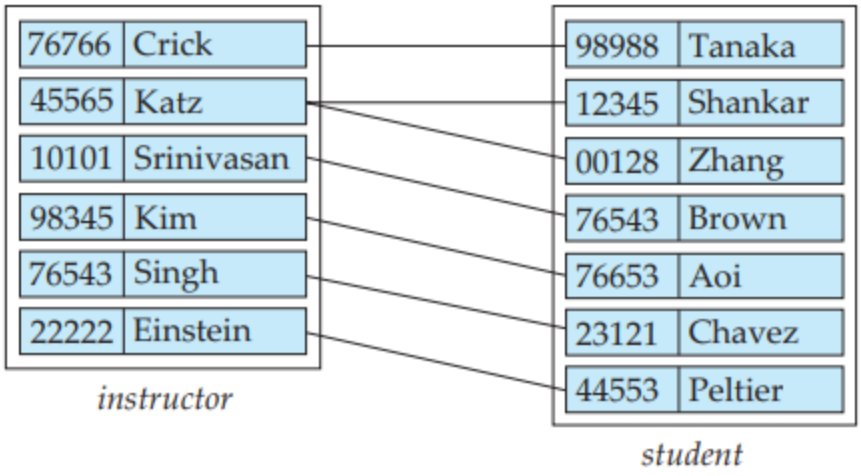

Figure 7.2 Relationship set *advisor*.

5. The association between entity sets is **referred to as participation**; that is, the entity sets E1, E2,..., En participate in relationship set R.
6. The function that an entity plays in a relationship is called that **entity's role**.

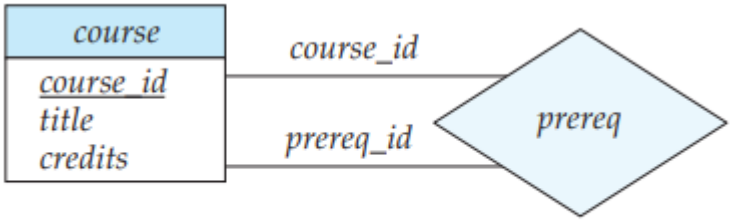

Figure 7.12 E-R diagram with role indicators.

7. A relationship may also have attributes called **descriptive attributes.**

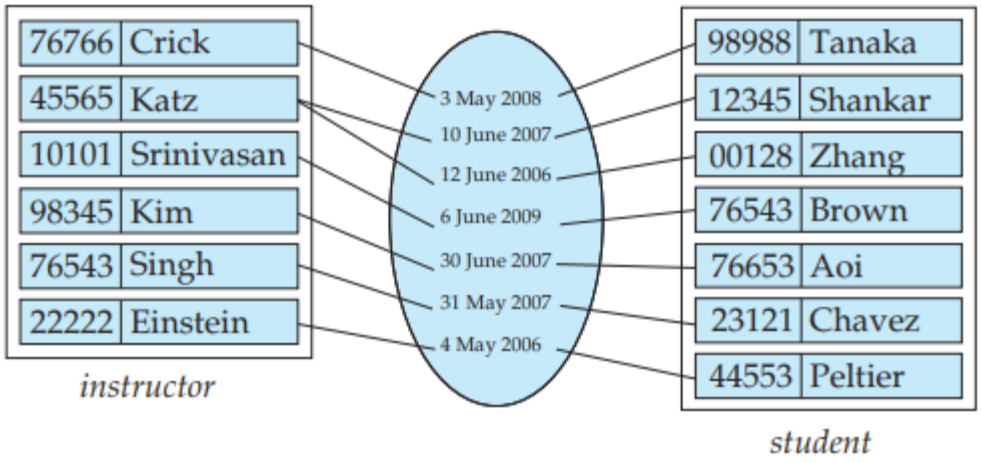

Figure 7.3 *date* as attribute of the *advisor* relationship set.

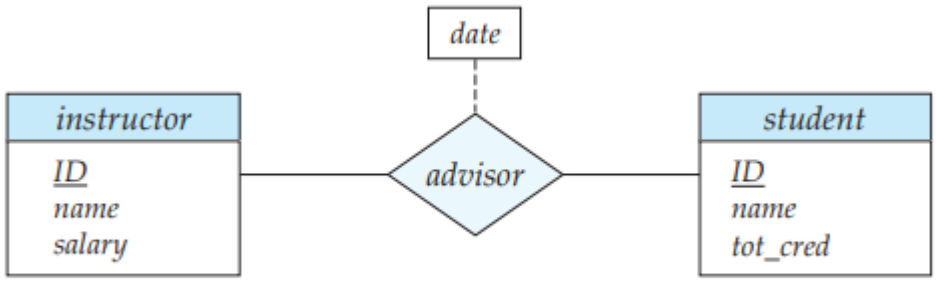

Figure 7.8 E-R diagram with an attribute attached to a relationship set.

8. It is possible to have more than **one relationship set involving the same entity sets**.
9. The relationship sets advisor and dept_advisor provide examples of **a binary relationship set**— that is, one that involves two entity sets.
10. The number of entity sets that participate in a relationship **set is the degree** of the relationship set. *Example: Each project can have multiple associated students and multiple associated instructors.*

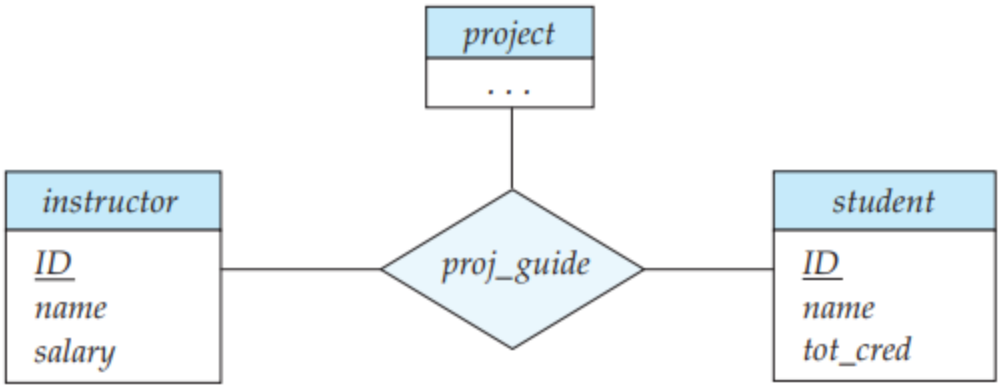

Figure 7.13 E-R diagram with a ternary relationship.

Attributes

1. An entity is represented by **a set of attributes.**
2. The designation of an attribute for an entity set expresses that the **database stores similar information** concerning each entity in the entity set.
3. Each **entity has a value** for each of its attributes.
4. For each attribute, there is a set of permitted values, **called the domain, or value set**, of that attribute.

An attribute, as used in the E-R model, can be characterized by the following attribute types:

Simple and composite attributes

- **The attributes are simple**; that is, they have not been divided into subparts.
- **Composite attributes**, on the other hand, can be divided into subparts (that is, other attributes).
  For example, an attribute *name* could be structured as a composite attribute consisting of first name, middle initial, and last name. Or, the address can be defined as the composite attribute address with the attributes Street, city, state, and zip code.
- Composite attributes help us to group together related attributes, making the modeling cleaner.

Single-valued and multivalued attributes

- The attributes in our examples all **have a single value for a particular entity**. For instance, the student ID attribute for a specific student entity refers to only one student ID. Such attributes are said to be single valued.

- The **attribute would be multivalued**, since any particular instructor may have zero, one, or more dependents. Example, An instructor may have zero, one, or several phone numbers, and different instructors may have different numbers of phones.

Derived attribute or Stored

- ***The value for this type of attribute can be derived*** from the values of other related attributes or entities. *For instance*, let us say that the instructor entity set has an attribute students advised, which represents how many students an instructor advises. We can derive the value for this attribute by counting the number of student entities associated with that instructor.
- *As another example,* suppose that the instructor entity set has an attribute age that indicates the instructor's age. If the instructor entity set also has an attribute date of birth, we can calculate age from date of birth and the current date. Thus, age is a derived attribute. In this case, date of birth may be referred to as a base attribute, or a stored attribute. The value of a derived attribute is not stored but is computed when required.

An attribute takes a null value when an entity does not have a value for it. The null value may indicate "**not applicable**"— that is, that the value does not exist for the entity. For example, one may have no middle name.

An unknown value may be **either missing** (the value does exist, but we do not have that information)

* * *