**ST. XAVIER’S COLLEGE**

**(Affiliated to Tribhuvan University)**

Maitighar, Kathmandu



**Database Management System**

**Assignment #6**

**Submitted by:**

PRANESH DHUNJU SHRESTHA  
013BSCCSIT026

**Submitted to:**

|  |  |
| --- | --- |
| **Er. Sanjay Kumar Yadav**  Lecturer  St. Xavier’s College |  |

**Date of Submission:** August 27, 2015

**Entity Relationship model**

1. What do you mean by Entity- Relationship Diagram? Explain

2. Define entity and give an example.

3. Explain the different between an entity class and an entity instance.

4. Define attribute and its types.

5. What is derived attributes?

6. Define relationship and give an example.

7. Explain the difference between a relationship class and a relationship instance.

8. Define degree of relationship.

9. List and give an example of the three types of binary relationships. Draw an E-R diagram

for each.

10. Define the terms maximum cardinality and minimum cardinality.

11. Explain the distinctions among the terms primary key, candidate key and super key.

12. What are the main building modules of the entity relationship model? Discuss each one.

13. What is composite attributes, when it is used?

14. Explain the difference between single-value attributes and simple attributes.

15. Discuss the difference between a composite key and a composite attribute. How would

each indicated in an E-R diagram?

16. What two courses of action are available to a designer when a multivalued attribute is

encountered ?

17. Explain the various terms of an E-R model and how are they represented in an E-R

model?

18. Explain the concept of dependent entities? Give example.

19. What is the difference total and partial participation? Explain.

20. What do you mean by mapping cardinalities ? explain various type of cardinalities.

21. What is the difference between single-value and multivalued attributes? Explain

22. Explain the concept of participation constraints.

23. Difference the binary relationship with ternary relationship with example.

24. Explain the difference between weak and strong entity set.

25. Define the components of extended E-R features.

26. Define the concept of aggregation. Give two examples of where this concept is useful.

27. Explain the distinction between disjoint and overlapping constraints.

28. Explain the distinction between total and partial constraints.

29. Write short notes on:

· Specialization

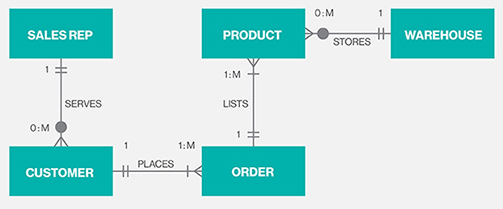
· Generalization

· Aggregation

1.

An entity-relationship diagram (ERD) is a graphical representation of an information system that shows the relationship between people, objects, places, concepts or events within that system. An ERD is a [data modeling](http://searchdatamanagement.techtarget.com/definition/data-modeling) technique that can help define business processes and can be used as the foundation for a [relational database](http://searchsqlserver.techtarget.com/definition/relational-database). While useful for organizing [data](http://searchdatamanagement.techtarget.com/definition/data) that can be represented by a relational structure, an entity-relationship diagram can't sufficiently represent semi-structured or [unstructured data](http://searchbusinessanalytics.techtarget.com/definition/unstructured-data), and an ERD is unlikely to be helpful on its own in integrating data into a pre-existing information system.

Three main components of an ERD are the [entities](http://whatis.techtarget.com/definition/entity), which are objects or concepts that can have data stored about them, the relationship between those entities, and the [cardinality](http://whatis.techtarget.com/definition/cardinality), which defines that relationship in terms of numbers.



2.

An entity can be a real-world object, either animate or inanimate, that can be easily identifiable. Or An [entity](http://cs.hbg.psu.edu/courses/comp419.taw.s97/er.html#rep%20of%20entities) may be an object with a physical existence - a particular person, car, house, or employee - or it may be an object with a conceptual existence - a company, a job, or a university course.

For example, in a school database, students, teachers, classes, and courses offered can be considered as entities. All these entities have some attributes or properties that give them their identity.

3.

Entities of a given type are grouped into entity classes. An entity instance is the representation of a particular entity.If an entity is an individual "person, place, event, or thing about which data is collected", then an entity is an instance. Linguistically, entity is just another word for a single. Entities that have the same attributes are grouped in what are best called entity classes .Entity and entity class are data modeling terms. The corresponding object modeling terms are object and class, albeit that a class typically has operations, which are foreign to an entity class.

An example of this would be STUDENT as the *entity* while PRANESH DHUNJU SHRESTHA is an *instance* of that entity.

4.

Entities are represented by means of their properties, called **attributes**. All attributes have values. For example, a student entity may have name, class, and age as attributes.

There exists a domain or range of values that can be assigned to attributes. For example, a student's name cannot be a numeric value. It has to be alphabetic. A student's age cannot be negative, etc.

Types of Attributes

* **Simple attribute** − Simple attributes are atomic values, which cannot be divided further. For example, a student's phone number is an atomic value of 10 digits.
* **Composite attribute** − Composite attributes are made of more than one simple attribute. For example, a student's complete name may have first\_name and last\_name.
* **Derived attribute** − Derived attributes are the attributes that do not exist in the physical database, but their values are derived from other attributes present in the database. For example, average\_salary in a department should not be saved directly in the database, instead it can be derived. For another example, age can be derived from data\_of\_birth.
* **Single-value attribute** − Single-value attributes contain single value. For example − Social\_Security\_Number.
* **Multi-value attribute** − Multi-value attributes may contain more than one values. For example, a person can have more than one phone number, email\_address, etc.

5.

**Derived attribute** − Derived attributes are the attributes that do not exist in the physical database, but their values are derived from other attributes present in the database.

For example, average\_salary in a department should not be saved directly in the database, instead it can be derived. For another example, age can be derived from data\_of\_birth.

6.

The association among entities is called a relationship. Relationship is a situation that exists between two relational database tables when one table has a foreign key that references the primary key of the other table. Relationships allow relational databases to split and store data in different tables, while linking disparate data items.

For example, an employee **works\_at** a department, a student **enrolls** in a course. Here, Works\_at and Enrolls are called relationships.

7.

A relationship class is an association among entity classes; a relationship instance is an association among entity instances.

8.

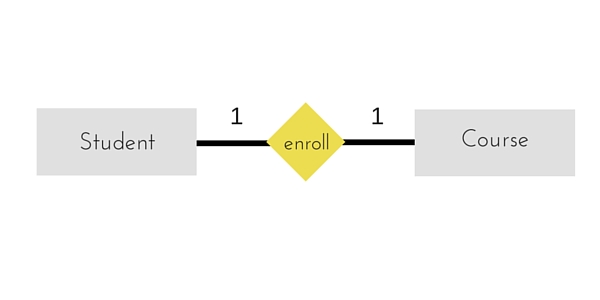
The number of participating entities in a relationship defines the degree of the relationship.

* Binary = degree 2
* Ternary = degree 3
* n-ary = degree n

9.

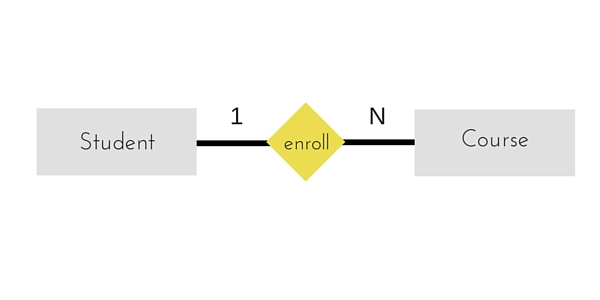
Binary Relationship means relation between two Entities. This is further divided into three types.

1. **One to One :** This type of relationship is rarely seen in real world.



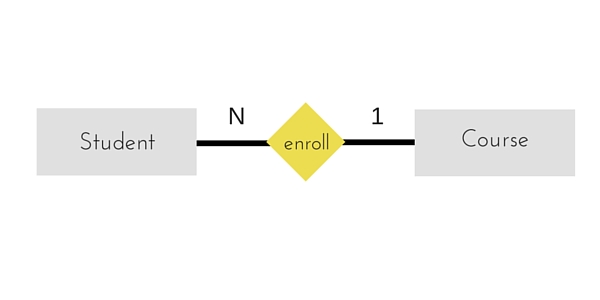
The above example describes that one student can enroll only for one course and a course will also have only one Student. This is not what you will usually see in relationship.

1. **One to Many :** It reflects business rule that one entity is associated with many number of same entity. The example for this relation might sound a little weird, but this menas that one student can enroll to many courses, but one course will have one Student.

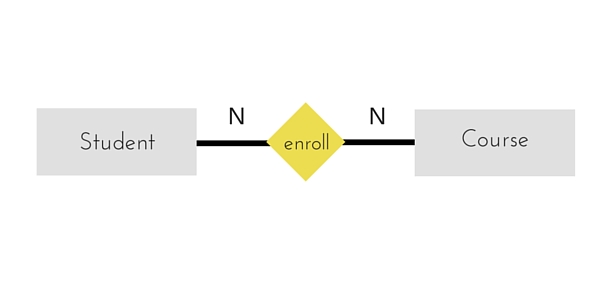


The arrows in the diagram describes that one student can enroll for only one course.

1. **Many to One :** It reflects business rule that many entities can be associated with just one entity. For example, Student enrolls for only one Course but a Course can have many Students.



1. **Many to Many :**



The above diagram represents that many students can enroll for more than one courses.

10.

Maximum cardinality is the maximum number of instances of an entity that can participate in an instance of a relationship. Minimum is the least number of instances of an entity that can participate in an instance of a relationship.

11.

Super Keys : Super key stands for superset of a key.A Super Key is a set of one or more attributes that are taken collectively and can identify all other attributes uniquely.

Candidate Keys: Candidate Keys are super keys for which no proper subset is a super key. In other words candidate keys are minimal super keys.

Primary Key: It is a candidate key that is chosen by the database designer to identify entities with in an entity set. Primary key is the minimal super keys. In the ER diagram primary key is represented by underlining the primary key attribute. Ideally a primary key is composed of only a single attribute. But it is possible to have a primary key composed of more than one attribute.

12.

The Entity-Relationship model is a top-down approach to design database that is based on uniquely identifiable object. If begins by identifying things that are uniquely distinguishable called entities and relationships among these entities. The main building modules of the Entity-Relationship model are:

a. Entities

b. Relationships

c. Attributes

Entities : An Entity is a basic object of ER-model which is an object in real world that can be distinguishable and can exists independently.

Relationships : Relationship defines the association among two entities. Suppose, consider student and a class are the two entities. These entities are associated as “student studies in class”. Hence studies is a relationship between the two entities, student and class.

Attributes : The properties of the entities are called attributes. For example if we consider a mobile phone as an entity then each mobile well have its own color, design, model company. All these are the attributes of the mobile entity.

13

The composite attribute is an attribute that can be subdivided into other single attributes with meanings of their own.

Example: Imagine from the entity Student that instead of having the three attributes: stu\_LastName, stu\_MiddleName, stu\_FirstName it had one attribute called stu\_Name. The attribute stu\_Name would be considered a composite attribute since it can be subdivided into the other three attributes: stu\_LastName, stu\_MiddleName, stu\_FirstName.

14.

A simple attribute

A simple attribute is an attribute that cannot be subdivided. For example, age, sex, and marital status would be classified as simple attributes. To facilitate detailed queries, it is wise to change composite attributes into a series of simple attributes.

A single-valued attribute

A single-valued attribute is an attribute that can have only a single value. For example, a person can have only one Social Security number, and a An attribute broken into component parts manufactured part can have only one serial number. Keep in mind that a single-valued attribute is not necessarily a simple attribute. For instance, a part’s serial number, such as SE-08-02-189935, is single-valued, but it is a composite attribute because it can be subdivided into the region in which the part was produced (SE), the plant within that region (08), the shift within the plant (02), and the part number (189935).

15.

A composite key is one that consists of more than one attribute. A composite attribute is one that can be subdivided to yield attributes for each of its components. If the E-R diagram contains the attribute names for each of its entities, a composite key is indicated in the E‑R diagram by the fact that more than one attribute name is underlined to indicate its participation in the primary key. There is no E-R convention that enables us to indicate that an attribute is a composite attribute.