**ST. XAVIER’S COLLEGE**

**(Affiliated to Tribhuvan University)**

Maitighar, Kathmandu



**Data Base Management System Theory Assignment #5**

**Submitted by:**

Yub Raj Basnet  
013BSCCSIT048 (4th Semester)

**Submitted to:**

|  |  |
| --- | --- |
| Er. Sanjaya Kumar Yadav  Lecturer, St. Xavier’s College |  |

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

ER-modeling is a data modeling technique used in software engineering to produce a conceptual data model of a information system. Diagrams created using this ER-modeling technique are called Entity-Relationship Diagrams, or ER diagrams or ERDs. So you can say that Entity Relationship Diagrams illustrate the logical structure of databases.

There are three basic elements in ER-Diagrams:

1. Entities are the "things" for which we want to store information. An entity is a person, place, thing or event.
2. Attributes are the data we want to collect for an entity.
3. Relationships describe the relations between the entities.
4. Define entity and give an example.

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

4. Define attribute and its types.

In simple, attribute is descriptive property of entity set. Set of attributes describes entity set.

For example

customer = (customer-id, customer-name, customer-city)

account=( account\_number, balance)

loan = (loan\_number, amount)

Types of attributes

Simple and Composite attribute

Attribute which can not be divide into subparts (i.e. into other attributes) called simple attribute

Single-valued and Multivalued attributes

Attribute that can take only one value in every entry called singled-valued attribute.

Stored and Derived attribute

Attribute whose values can be derived from the values of other related attributes or entities called derived attribute.

5.What is derived attributes?

Attribute whose values can be derived from the values of other related attributes or entities called derived attribute. For example, in customer entity set, attribute age is derived attribute if customer entity set has attribute date\_of\_birth. We can derive age of customer from date\_of\_birth and current\_date. Here the attribute date\_of\_birth is stored attribute and the attribute age is derived attribute. The value of derived attribute is not stored, it is computed when required.

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

Specialization follows top down design approach. Entity sets are subgroups in distinct entity sets. For example entity set person with attributes name, street and city can further subgroup into two entities sets customer and employee. Each of these person types can describes by set of attributes that includes all the attributes of entity set person plus all possible attributes of itself. For example, customer entity set can further described by set of attributes: customer\_id, enroll\_date etc. Similarly entity attributes can further describes by set of attributes: emplouee\_id, salary etc. The process of sub groupings within an entity set is called specialization. We can apply specialization repeatedly to refine a design schema. For instance bank employees may be further classified into officer, teller or secretary.

In E-R diagram, specialization can be represented by a triangle component labeled ISA. The label ISA stands for “is a “. For example customer is a person, officer is an employee etc. The ISA relationship also called super class-subclass relationship.

· Generalization

Generalization follows bottom-up approach in which multiple entity sets are synthesized into higher-level entity set on the basis of common features. For example, the database designer may have first identified a customer entity set with the attributes: name,street, city and customer\_id and employee entity set with the attributes name, street, city, employee\_id and salary. In both entities some attributes are common. These similarities between these two entities can be express by generalization.

During the course of database design or E-R schema for enterprise database designer may use both specialization and generalization process. Specialization and generalization in E-R diagram represent by a same way. The terms specialization and generalization are used interchangeably.



Figure: Specialization and generalization.

· Aggregation

E-R model can not express relationship among relationship. To illustrate this, let us consider quaternary relationship manages among employee, branch, job and manager. Its main job is to record managers who manages particular job/task perform by particular employee at particular branch.



E-R diagram with redundant relationships.

This quaternary relationship is required since binary relationship between manager and employee can not represent required information. This E-R diagram is able to represent the required information but information are redundant since every employee, branch and job exist both relationship set “work-on” and “manages”. Here aggregation is better to represent such information.

Aggregation is in fact an abstraction it treats relationships as higher level entities. In our example, it treats relationship set work-on (including entity set employee, branch and job) as entity set. So now we can create binary relationship set “manages” between work-on and manager. This removes redundant information.



E-R Diagram with aggregation