



BABU MADHAV INSTITUTE OF INFORMATION TECHNOLOGY, UTU
Integrated M.Sc.(IT)

Semester-I

060010110 | CC2 Database Management Systems |
Question Bank-Unit: 03

Unit-3: Data Modelling

Short Questions [1 Mark]

1.	Which two major difficulties would you come upon while designing a database schema? or Which two major difficulties we must avoid while designing a database schema?
Ans:	➤ We must avoid two major difficulties, while designing a database schema are: 1. Redundancy 2. Incompleteness
2.	What is an Entity? Give example.
Ans:	➤ Entity is a 'thing' or an 'object' existing in the real world that is different from all other objects. ➤ <u>For Example:</u> <ul style="list-style-type: none">▪ Person: Student, Instructor, Patient, Doctor▪ Place: City, Country, State▪ Event: Seminar, Sale, Competition▪ Object: Building, Automobile, Machine, Toy▪ Concept: Course, Account, Loan
3.	What is a Relationship set?
Ans:	➤ Relationship is an association among several entities. ➤ "A relationship set is a set of relationships of the same type". ➤ <u>For Example:</u> <ul style="list-style-type: none">▪ In banking application, each and every account is associated with some particular customer. In other words, each and every customer owns an account in a bank. This kind of association is known as relationship.
4.	What are attributes? Give examples.
Ans:	➤ Attributes are descriptive properties possessed by each member of an entity set. ➤ <u>For Example:</u> An entity set Customer has attributes like customer id, name, address and contact number

	<p>denoted by cid, cname, address, contact_no respectively.</p> <p>➤ Attribute values that describe each entity become a major part of the data stored in a database.</p>
5.	List different types of attributes in E-R Diagram.
<u>Ans:</u>	<p>➤ The different types of attributes in ER Diagram are as follows:</p> <ol style="list-style-type: none"> 1. Simple attribute 2. Composite attribute 3. Single attribute 4. Multi-valued attribute 5. Stored attribute 6. Derived attribute
6.	Define the term Domain.
<u>Ans:</u>	<p>➤ A Domain is the set of the permitted values for each attribute.</p> <p>➤ A domain of an attribute is also known as value set of an attribute.</p> <p>➤ <u>For example:</u></p> <p>A bank is organized in three branches named 'Bardoli', 'Surat' and 'Vyara'.</p> <p>So, a set of permitted values for an attribute branch_name is {'Bardoli', 'Surat', 'Vyara'}.</p>
7.	What are constraints?
<u>Ans:</u>	<p>➤ Constraints are restrictions which the contents of the underlying database must confirm.</p> <p>➤ Each and every data model specifies its own constraints.</p> <p>➤ The E-R model specifies three types of constraints:</p> <ol style="list-style-type: none"> 1. Mapping Cardinalities 2. Key Constraints 3. Participation Constraints
8.	List different types of key constraints.
<u>Ans:</u>	<p>➤ The different types of keys are:</p> <ol style="list-style-type: none"> 1. Super key 2. Candidate key 3. Primary key 4. Alternate key
9.	What is a Primary key?
<u>Ans:</u>	<p>➤ Primary key is an attribute (or set of attributes) through which each tuple (record) in a relation (table) can be uniquely identified.</p>

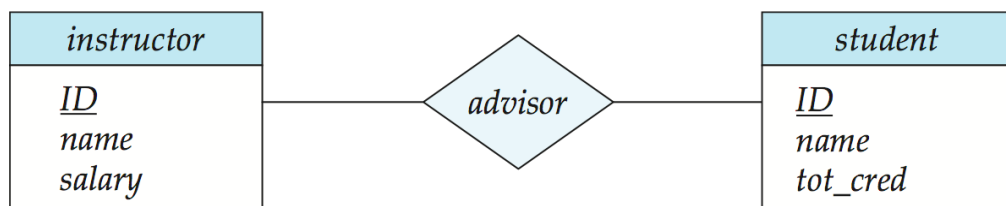
- Primary key is a not null and unique attribute.
- For Example:

STUDENT(ID, NAME, DEPT_NAME, TOT_CRED)

 - Where ID is the primary key of relation student which uniquely identifies each tuple in a student relation.

10. What is a Candidate key?

- Ans:**
- A candidate key is a minimal (in terms of number of attributes) super key.
 - Candidate key uniquely identify a record in a relation. It is a not null and a unique attribute.
 - One of the candidate keys is chosen by database designer as primary key.
 - For Example:



- Where, ID is candidate key of *instructor* and ID is candidate key of *student*.
- Although several candidate keys may exist, one of the candidate keys is selected to be a primary key.

11. What is a Super key?

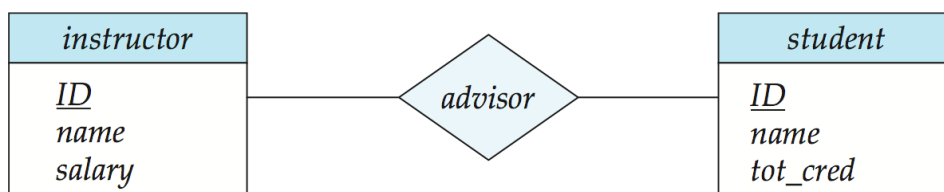
- Ans:**
- A super key is a set of one or more attributes that collectively allows us to identify uniquely an entity in an entity set.
 - For Example:

INSTRUCTOR(ID, NAME, DEPT_NAME, SALARY)

- Where id attribute of the relation *instructor* is sufficient to distinguished one instructor tuple from another. Thus, ID is the super key. The name attribute of instructor, on the other hand, is not a super key, because several instructors might have the same name.

OR

- A super key of an entity set is a set of one or more attributes whose values uniquely determine each entity.
- For example:



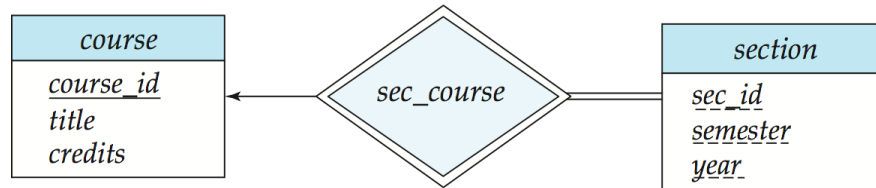
- Here, (s_id, i_id) is the super key of advisor.

12.	What is Disjoint?
	or
	What is Disjointness constraint?
Ans:	Disjoint is an entity that can belong to only one lower-level entity set.
13.	What is Overlapping?
	or
	What is Overlapping constraint of generalization?
Ans:	Overlapping is an entity that can belong to more than one lower-level entity set.
	or
	Overlapping constraint specify that same entity may belong to more than one lower-level entity set within a single generalization.
14.	Enlist the extended features of E-R Model.
Ans:	<p>➤ Following are the extended features of E-R Model:</p> <ol style="list-style-type: none"> 1. Specialization 2. Generalization 3. Attribute inheritance 4. Aggregation
Short Questions [2 Marks]	
1.	What is Mapping cardinalities of a relationship set? Give example.
Ans:	<p>➤ Mapping cardinalities express the number of entities to which another entity can be associated via a relationship set.</p> <p>➤ <u>For Example:</u></p> <ol style="list-style-type: none"> 1) Many-to-One 2) One-to-One 3) One-to-Many 4) Many-to-Many
2.	Give example of following relationships:
	<ol style="list-style-type: none"> 1. Many-to-One 2. One-to-One 3. One-to-Many 4. Many-to-Many
Ans:	<ol style="list-style-type: none"> 1. Many-to-One: Child – Parent 2. One-to-One: Student – Program

	3. One-to-Many: Program – Course, Bank_account – Customer	
	4. Many-to-Many: Student – Teacher	
3.	What is the difference between Entity and Entity set?	
Ans:	<u>Entity</u>	<u>Entity set</u>
	1) An entity is a “thing” or “object” in the real world that is different from all other objects. 2) <u>For example:</u> <ul style="list-style-type: none">▪ each person in a university is an entity,▪ specific person,▪ company,▪ event, plant, etc.	1) An entity set is a set of entities of the same type that share the same properties or attributes. 2) <u>For Example:</u> <ul style="list-style-type: none">▪ set of all persons,▪ companies,▪ trees,▪ holidays, etc.
4.	What is the difference between Simple and Composite attributes?	
Ans:	<u>Simple Attributes</u>	<u>Composite Attributes</u>
	1. Simple attributes cannot be divided into subpart. 2. <u>For Example:</u> STUDENT(ID, NAME, PHONE_NO) - where, ID cannot be further divided.	1. Composite attributes can be divided into subpart. 2. <u>For Example:</u> STUDENT(ID, NAME, PHONE_NO) - where, NAME can be further divided in to FIRST_NAME, MIDDLE_NAME and LAST_NAME. - STUDENT(ID, FIRST_NAME, MIDDLE_NAME, LAST_NAME, PHONE_NO)
5.	Differentiate Single-valued and Multi-valued attributes.	
Ans:	<u>Single-valued Attributes</u>	<u>Multi-valued Attributes</u>
	1. An attribute has single value for a particular entity is called single valued attribute.	1. An attribute has more than one value for a particular entity is called multivalued valued attribute.

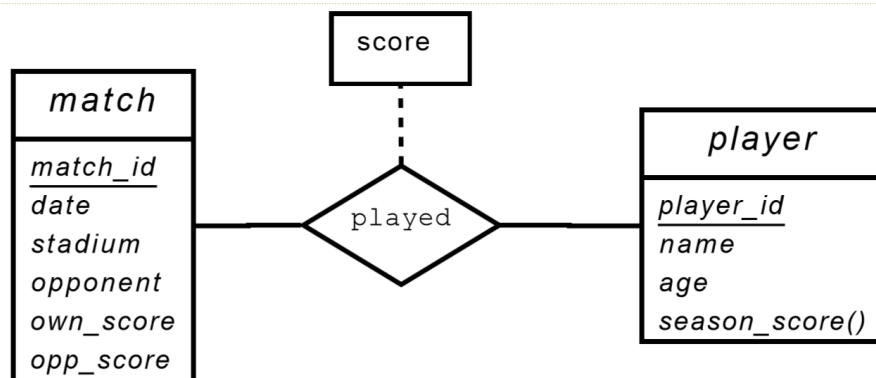
	<p>2. <u>For Example:</u></p> <p>STUDENT(ID,NAME,PHONE_NO)</p> <ul style="list-style-type: none"> - where, ID having only one value for particular student. 	<p>2. <u>For Example:</u></p> <p>STUDENT(ID,NAME,PHONE_NO)</p> <ul style="list-style-type: none"> - where, student having more than one value for PHONE_NO.
6.	Differentiate stored and derived attributes by giving example.	
Ans:	<p><u>Stored Attribute:</u></p> <ul style="list-style-type: none"> ➤ The attributes stored in a database are called stored attributes. ➤ <u>For Example:</u> <p style="padding-left: 40px;">ID, NAME, etc.</p> <p><u>Derived Attribute:</u></p> <ul style="list-style-type: none"> ➤ The value for derived attribute can be derived from the value of other related attribute or entities. The derived attribute is not stored but is computed when required. ➤ <u>For Example:</u> <p style="padding-left: 40px;">STUDENT(ID, NAME, BIRTH_DATE, AGE, PHONE_NO)</p> <p style="padding-left: 40px;">Where, AGE is a derived attribute because its value can be calculated from BIRTH_DATE.</p>	
7.	Explain E-R diagram briefly.	
Ans:	<ul style="list-style-type: none"> ➤ The entity-relationship diagram uses collection basic objects, called entities, and relationships among these objects. ➤ An entity is a “thing” or “object” in the real world that is distinguishable from the other objects. ➤ The overall logical structure (schema) of a database can be expressed graphically by an entity-relationship (E-R) diagram. 	
8.	Explain Strong and Weak Entity Set with proper example.	
Ans:	<ul style="list-style-type: none"> ➤ An entity set that has primary key or super key is termed as <i>strong entity set</i>. ➤ <u>For Example:</u> <div style="text-align: center;"> <pre> graph LR instructor[instructor] <--> advisor student[student] subgraph instructor_entity [instructor] ID1[<u>ID</u>] name1[name] salary[salary] end subgraph student_entity [student] ID2[<u>ID</u>] name2[name] tot_cred[tot_cred] end </pre> <p>The diagram illustrates a strong entity set relationship. On the left is the 'instructor' entity with attributes <u>ID</u>, name, and salary. On the right is the 'student' entity with attributes <u>ID</u>, name, and tot_cred. A diamond-shaped relationship labeled 'advisor' connects the two entities with double-headed arrows, indicating a bidirectional relationship.</p> </div>	

- An entity set that does not have a primary key is referred to as a weak entity set.
- For Example:



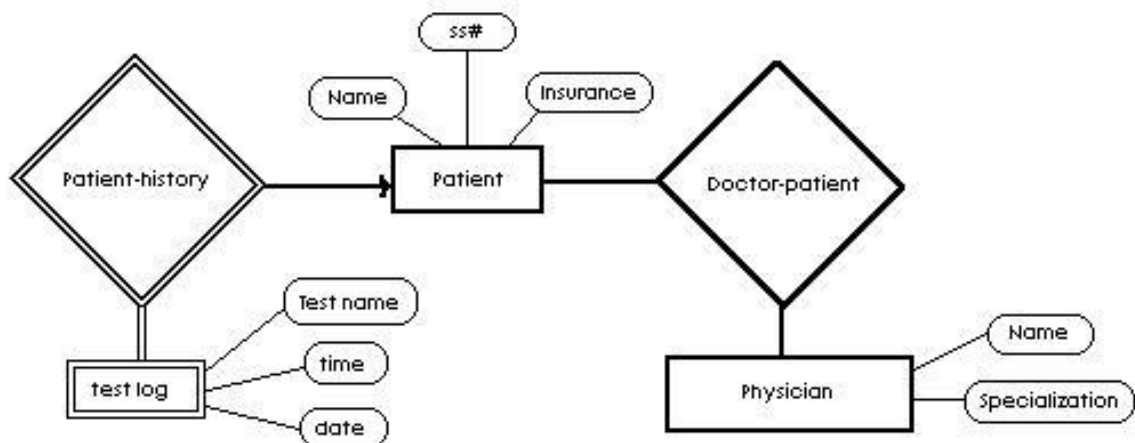
9. Design an E-R diagram for keeping the track of the exploits of your favorite sports team. You should store them at chess played, the scores in each match, the players in each match, and individual player statistics for each match. Summary statistics should be modeled as derived attributes.

Ans:



10. Construct an E-R diagram for a hospital with a set of patients and a set of medical doctors. Associate with each patient, a log of the various tests and examinations conducted.

Ans:

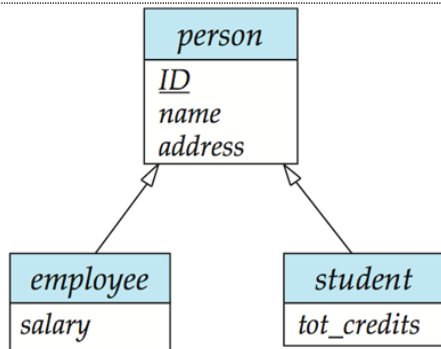


11. Differentiate Specialization and Generalization.+

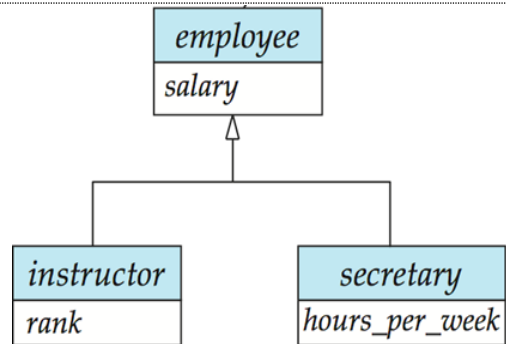
Ans:

Specialization	Generalization
1) It is a top-down approach	1) It is a bottom-up approach
2) Here, one higher level entity can be broken down into two lower level entity	2) Here, two lower level entities combine to form higher level entity

3) Some higher level entities may not have lower level entity sets at all



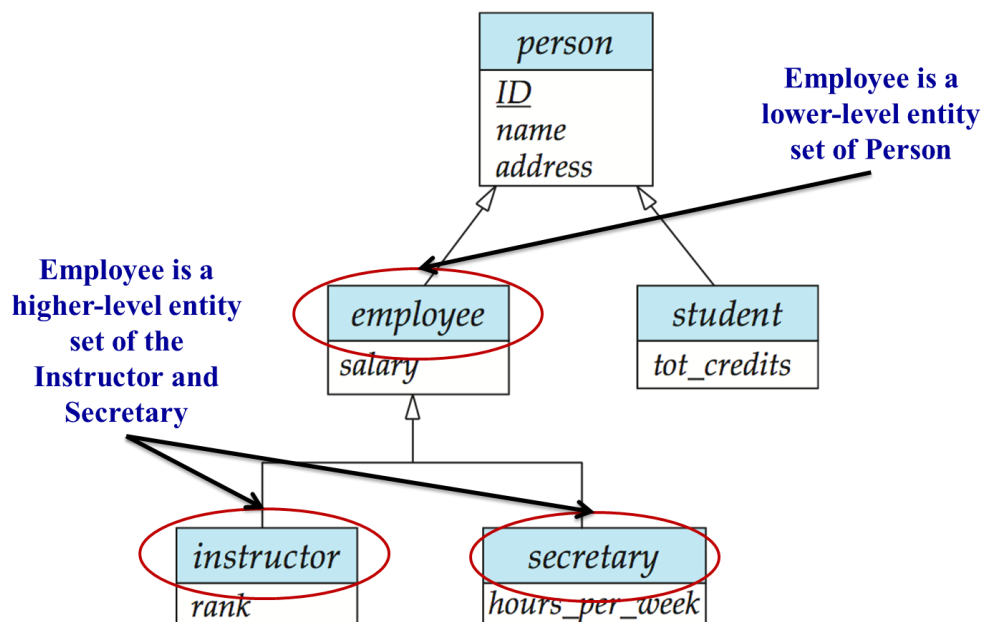
3) Higher level entity can also combine with other lower level entity to make further higher level entity



12. Explain Attribute Inheritance.

Ans: Attribute Inheritance:

- Attribute inheritance is a lower-level entity set which inherits all the attributes and relationship participation of the higher-level entity set to which it is linked.
- The attributes of the higher-level entity sets are said to be inherited by the lower-level entity sets.
- An inheritance of the attribute is key property of the higher and lower level entities which is created by specialization and generalization.



- In a hierarchy, a given entity set may be involved as a lower level entity set in only one ISA relationship; that is, entity sets in this diagram have only single inheritance.

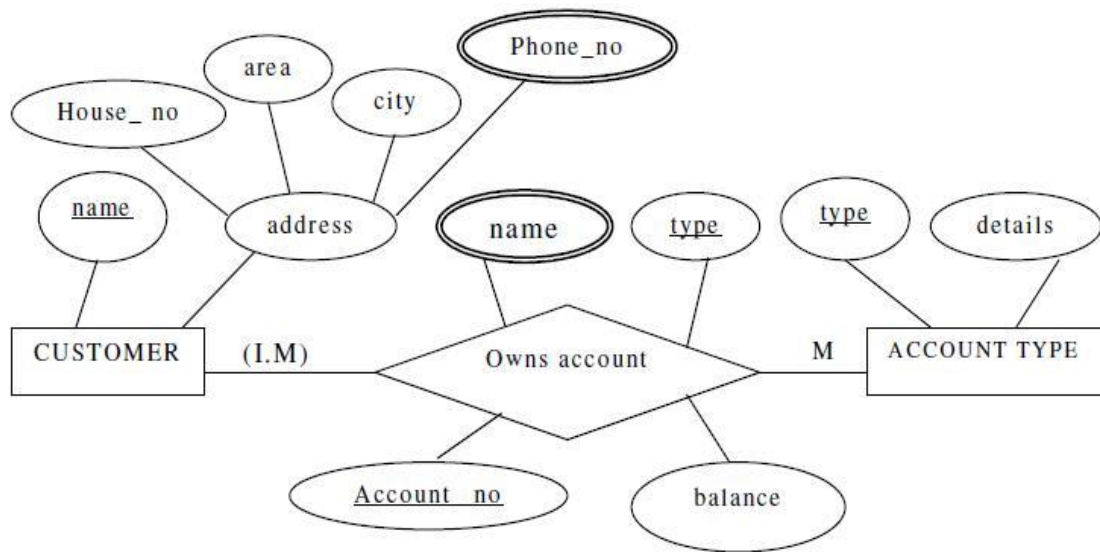
- If an entity set is a lower-level entity set in more than one ISA relationship, then the entity set has multiple inheritances.

Scenario based Questions [5 Marks]

1. Model the following scenario into E-R Model.

Information about a bank is about customers and their account. Customer has a name, address which consists of house number, area and city, and one or more phone numbers. Account has number, type and balance. We need to record customers who own an account. Account can be held individually or jointly. An account cannot exist without a customer.

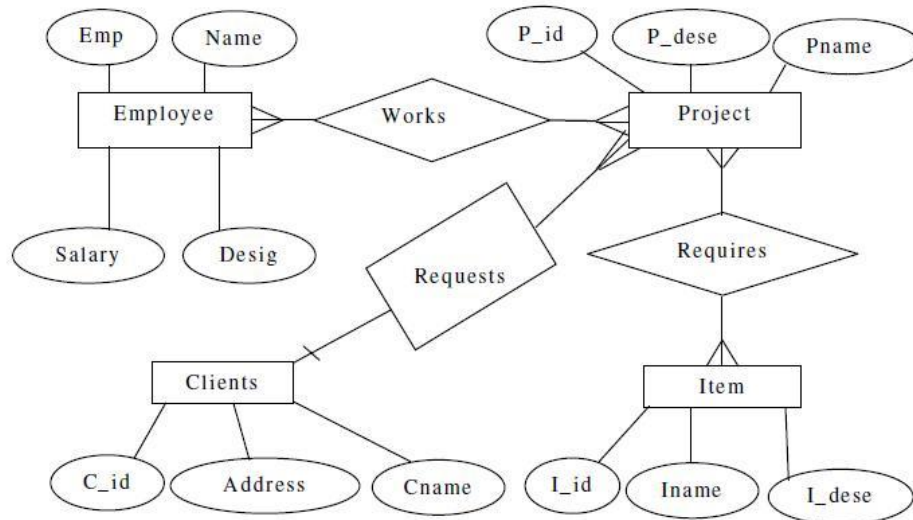
Ans:



2. Model the following scenario into E-R Model.

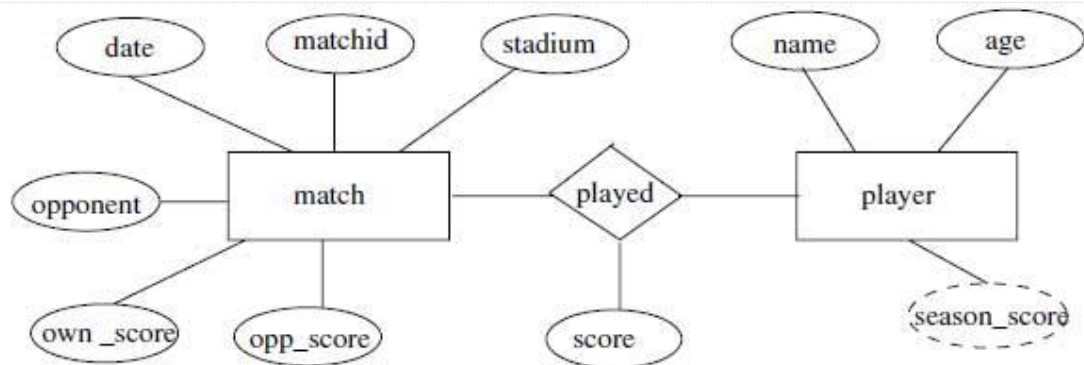
In an organization several projects are undertaken. Each project can employ one or more employees. Each employee can work on one or more projects. Each project is undertaken on the required of client. A client can request for several projects. Each project has only one client. A project can use a number of items and an item may be used by several projects.

Ans:



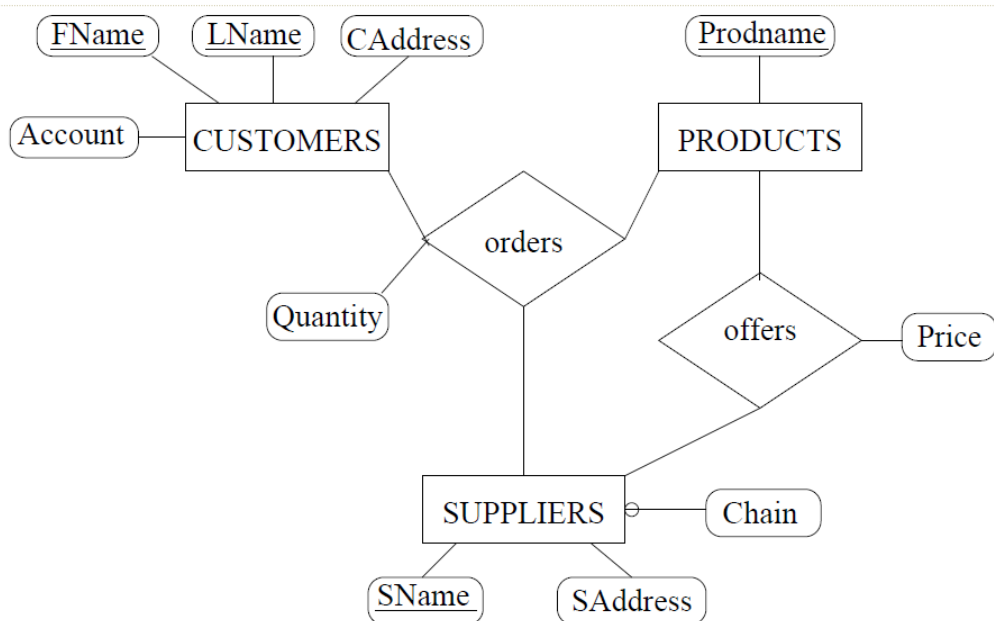
3. Design an E-R diagram for keeping track of the exploits of your favorite sports team. You should store the matches played, the scores in each match, the players in each match, and individual player statistics for each match.

Ans:



4. Create E-R diagram for Customers-Suppliers-Products.

Ans:



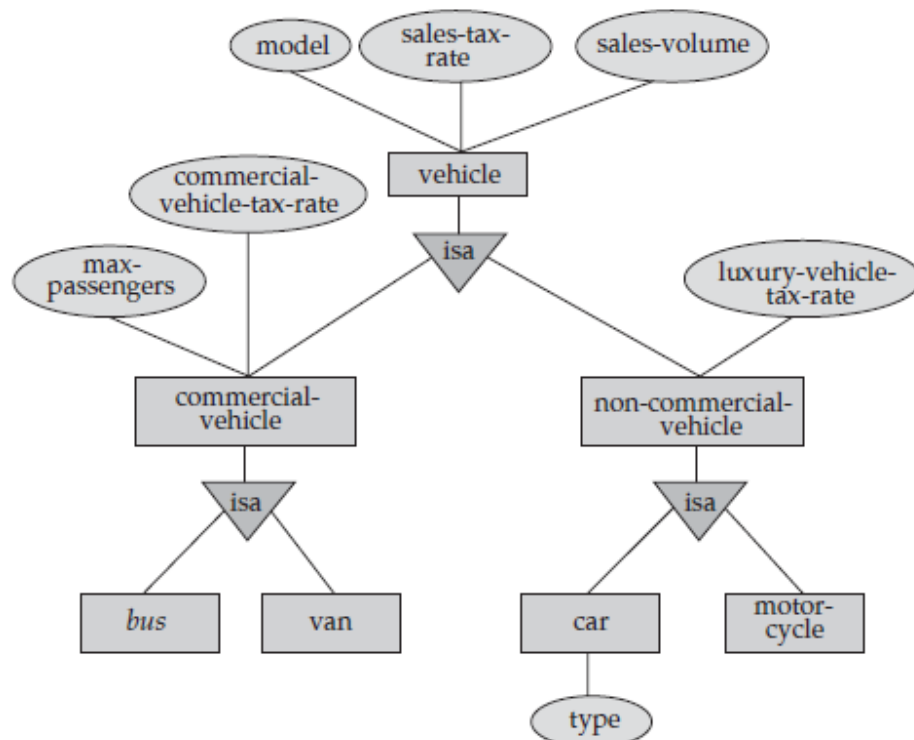
Create the following E-R diagrams.

5. Library Management System
6. Institute Management System
7. Hospital Management System
8. Event Management System

[From the above any one can be asked for 5 Marks]

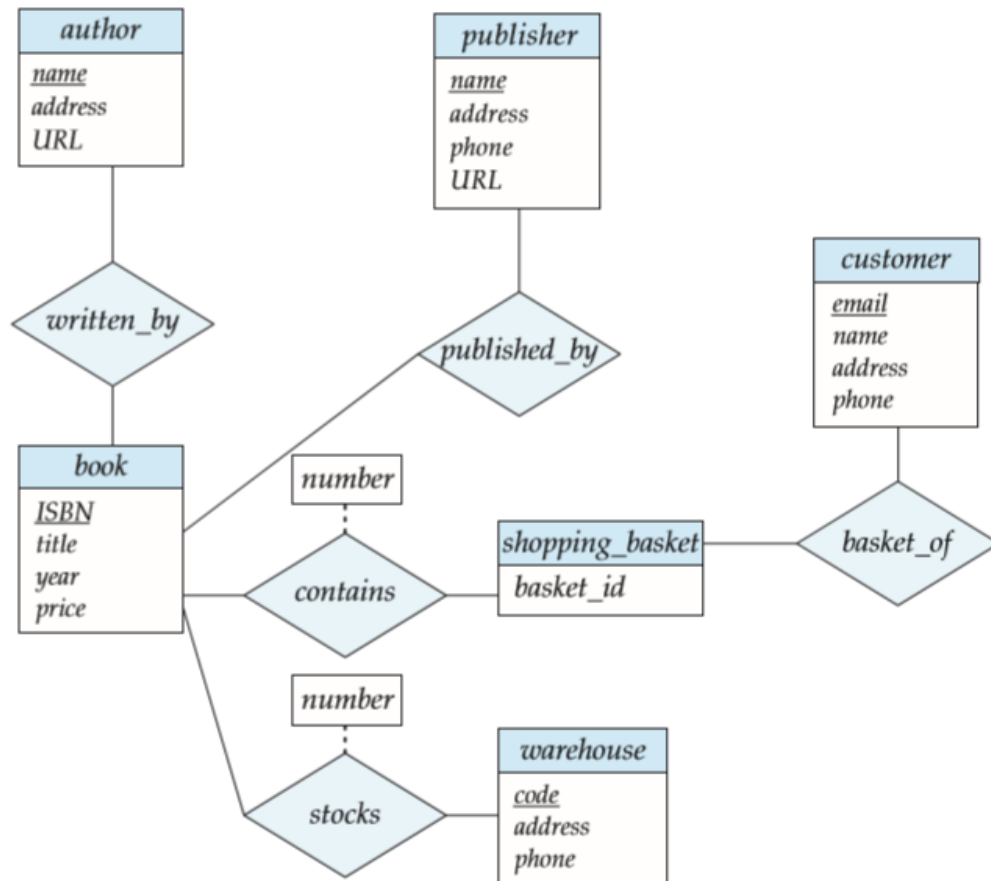
9. Design a generalization–specialization hierarchy for a motor vehicle sales company. The company sells motorcycles, passenger cars, vans and buses.

Ans:



10. Design E-R diagram for Online Bookstore.

Ans:



Long Questions [5 Marks]

1. Explain E-R Model in detail with suitable example.

Ans:

- The E-R data model considers the real world consisting of a set of basic objects, called entities, and relationships among these objects.
- The E-R data model employs three basic notions:
 1. Entity sets
 2. Relationship sets
 3. Attributes

1. Entity Sets:

- An entity is “thing” or “object” in the real world that is distinguishable from all other objects.
- For Example: Each person is an entity.
- An entity has a set of properties, and the values for some set of properties may uniquely identify an entity.
- An entity may be concrete, such as person or a book, or it may be abstract, such as a loan, or a holiday.

- An entity set is a set of entities of the same type that share the same properties, or attributes.

2. Relationships and Relationships sets:

- Relationship is an association among several entities.
- Relationship set is a set of relationships of the same type.
- The association between entity set is referred to as participation. That is, the entity sets E1, E2, . . ., En participate in relationship set R.

3. Attributes:

- The property that describes an entity is called attributes.
- The attributes of customer entity set are customer_id, customer_name and city.
- Each attribute has a set of permitted values called the domain or value set.
- Each entity will have value for its attributes.
- For Example:

Customer Id – 321

Customer Name – John

2. List and explain different types of attributes giving proper examples.

Ans:

- The different types of Attributes are:

1. Simple and composite attributes.
2. Single-valued and multivalued attributes.
3. Derived attributes.

1. Simple and composite attributes:

- A **Simple attribute** is an attribute that is not divided into subparts.
- A **Composite attribute** can be divided into subparts.
- Suppose address to the student entity-set. 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.

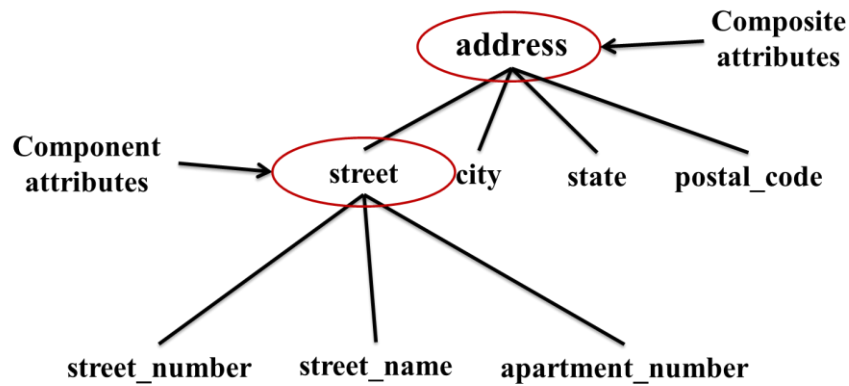


Fig.: Composite attributes instructor *name* and *address*.

2. Single-valued and multivalued attributes:

- 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**.

98988	Tanaka
12345	Shankar
00128	Zhang
76543	Brown
76653	Aoi
23121	Chavez
44553	Peltier

student

- Suppose we add a phone number attribute, to the instructor entity set.
- An instructor may have zero, one or several phone numbers and different instructors may have different numbers of phones. This type of attribute is said to be **multi-valued**.
- To denote a multi-valued attribute, we enclose it in braces.
- For Example:

{phone number} or {dependent name}.

3. Derived attributes:

- Derived attributes can be added from other attributes.
- For Example:
 - age,
 - date_of_birth, etc.

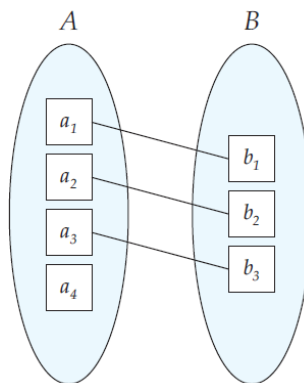
- Suppose that the instructor entity set has an attribute age, that indicates the instructor's age.
- If the instructor entity set also has an attribute date of birth, we can calculate age from date of birth and the current date.
- Thus, age is a derived attribute.

3. Explain Mapping Cardinalities with examples.

- Ans:**
- Mapping cardinalities, or cardinality ratios, express the number of entities to which another entity can be associated via a relationship set.
 - Mapping cardinalities are most useful in describing binary relationship sets, although they can contribute to the description of relationship sets that involve more than two entity sets.
 - For a binary relationship set R between entity sets A and B, the mapping cardinality must be one of the following:
 1. One-to-one
 2. One-to-many
 3. Many-to-one
 4. Many-to-many

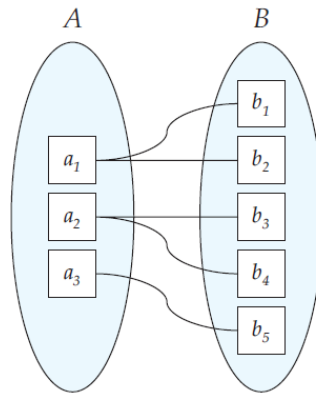
1. One-to-one:

- An entity in A is associated with at most one entity in B, and an entity in B is associated with at most one entity in A.



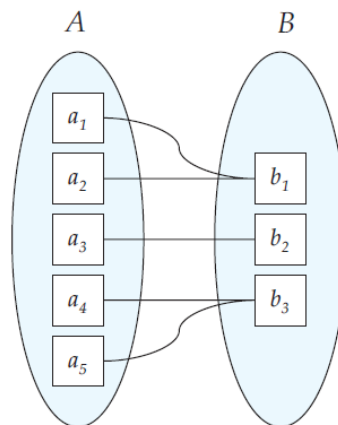
2. One-to-many:

- An entity in A is associated with any number (zero or more) of entities in B. An entity in B, however, can be associated with at most one entity in A.



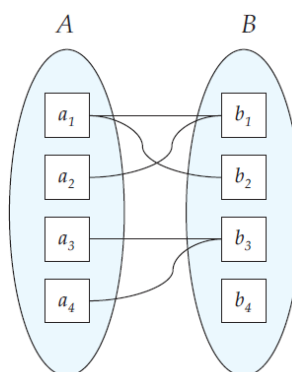
3. Many-to-one:

- An entity in A is associated with at most one entity in B. An entity in B, however, can be associated with any number (zero or more) of entities in A.



4. Many-to-many:

- An entity in A is associated with any number (zero or more) of entities in B, and an entity in B is associated with any number (zero or more) of entities in A.



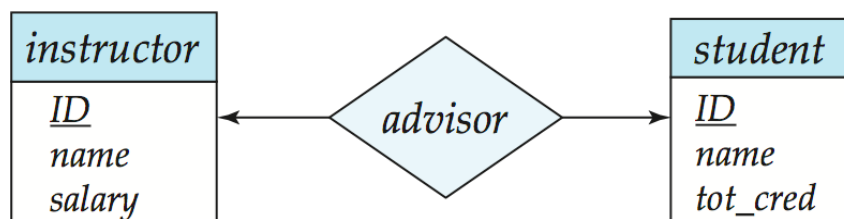
4. Explain Mapping Cardinalities with E-R Diagram notations.

- Ans:**
- Mapping cardinalities, or cardinality ratios, express the number of entities to which another entity can be associated via a relationship set.
 - Mapping cardinalities are most useful in describing binary relationship sets, although they can contribute to the description of relationship sets that involve more than two entity sets.

- The mapping cardinality must be one of the following:
 1. One-to-one
 2. One-to-many
 3. Many-to-one
 4. Many-to-many
- We express mapping cardinalities by drawing either a directed line (\rightarrow), signifying “one,” or an undirected line ($—$), signifying “many,” between the relationship set and the entity set.
- For Example:
 - For a binary relationship set *advisor* between entity sets *student* and *instructor*.

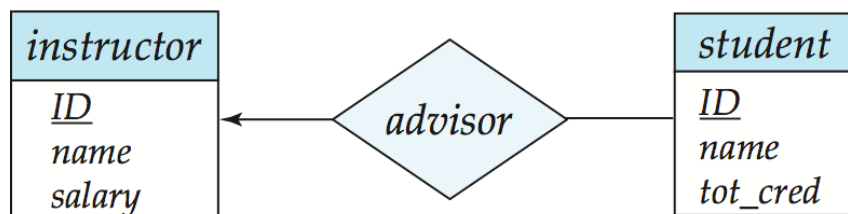
1. One-to-One relationship:

- For Example:
 - One-to-One relationship between an *instructor* and a *student*.
 - ✓ an *instructor* is associated with at most one *student* via *advisor* and
 - ✓ a *student* is associated with at most one *instructor* via *advisor*.



2. One-to-Many relationship:

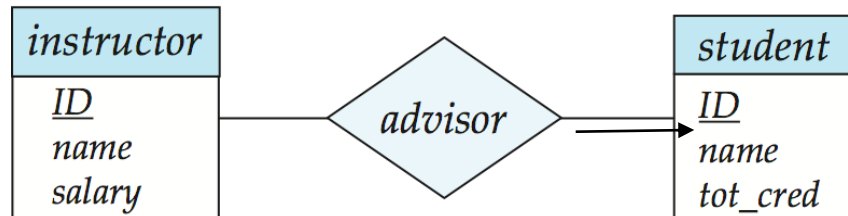
- For Example:
 - One-to-Many relationship between an *instructor* and a *student*.
 - ✓ an *instructor* is associated with several (including 0) *students* via *advisor* and
 - ✓ a *student* is associated with at most one *instructor* via *advisor*.



3. Many-to-One relationship:

➤ For Example:

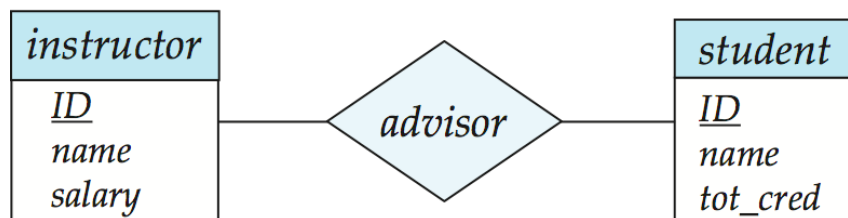
- In a Many-to-One relationship between an *instructor* and a *student*,
 - ✓ an *instructor* is associated with at most one *student* via *advisor* and
 - ✓ a *student* is associated with several (including 0) *instructors* via *advisor*.



4. Many-to-Many relationship:

➤ For Example:

- In a Many-to-Many relationship between an *instructor* and a *student*,
 - ✓ an *instructor* is associated with several (possibly 0) *students* via *advisor* and
 - ✓ a *student* is associated with several (possibly 0) *instructors* via *advisor*.

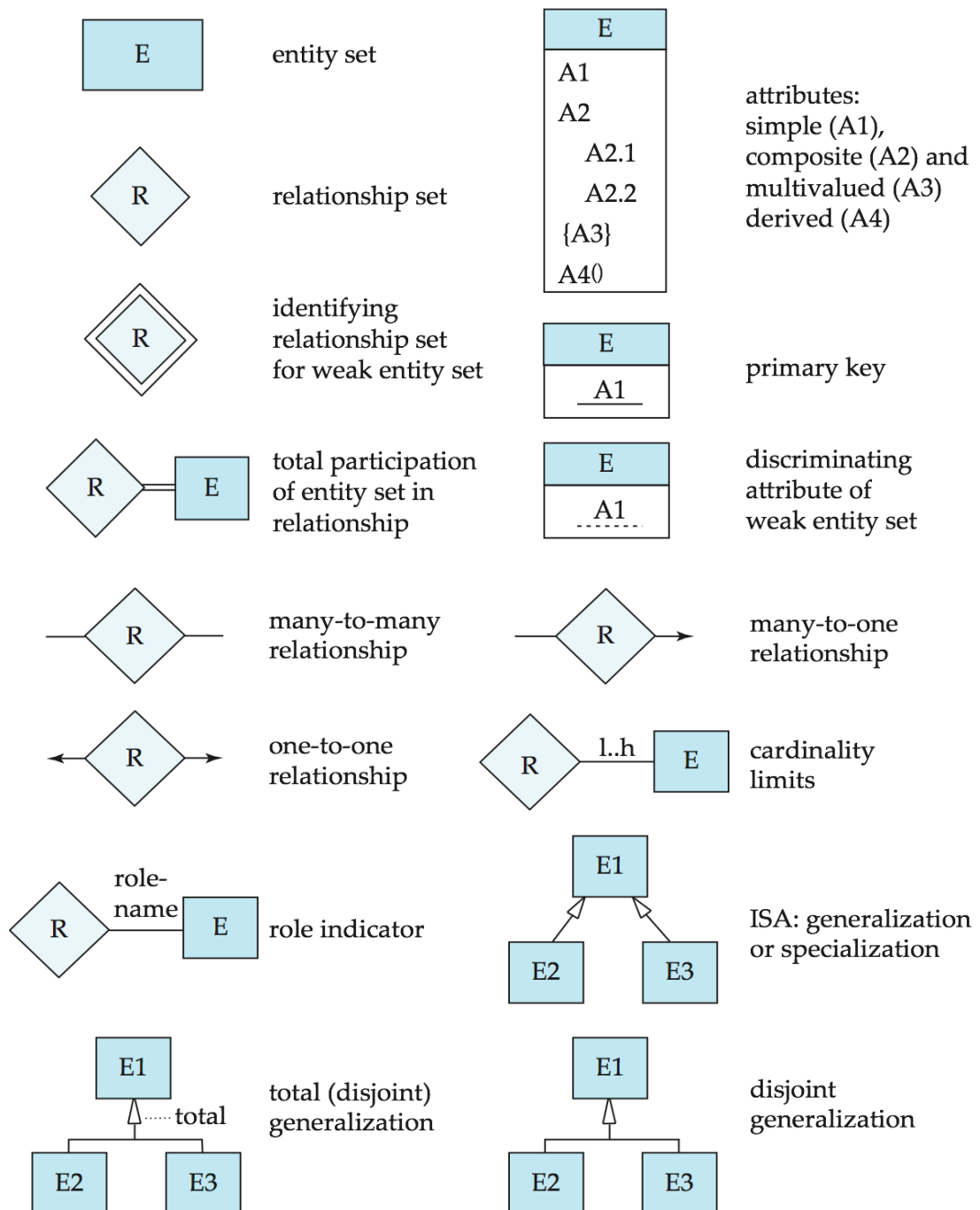


5. Discuss different notations used in E-R Diagrams.

Ans:

- E-R diagram can express the overall logical structure of a database graphically.
- E-R diagrams are simple and clear—qualities that may well account in large part for the widespread use of the E-R model.
- The E-R data model employs three basic concepts:
 1. Entity Sets
 2. Relationship Sets
 3. Attributes
- An E-R diagram consists of the following major components:
 - **Rectangles divided into two parts** represent entity sets.
 - ✓ The first part contains the name of the entity set.
 - ✓ The second part contains the names of all the attributes of the entity set.

- **Diamonds** represent relationship sets.
- **Undivided rectangles** represent the attributes of a relationship set. Attributes that are part of the primary key are underlined.
- **Lines** link entity sets to relationship sets.
- **Dashed lines** link attributes of a relationship set to the relationship set.
- **Double lines** indicate total participation of an entity in a relationship set.
- **Double diamonds** represent identifying relationship sets linked to weak entity sets.

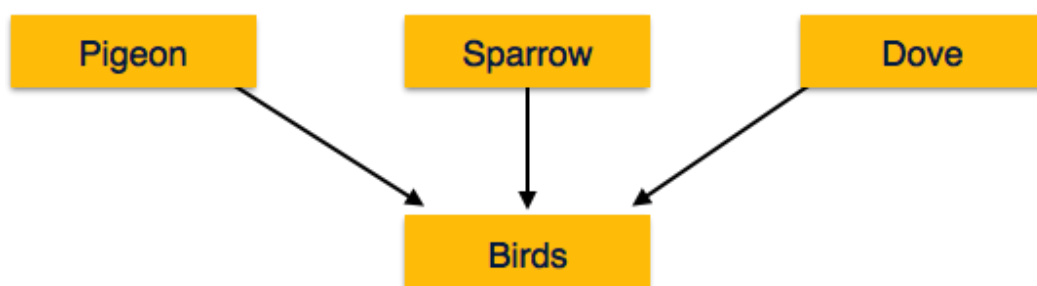


E-R Diagram Software Use Different Notations

6. Explain the concept of generalization and specialization with proper example.

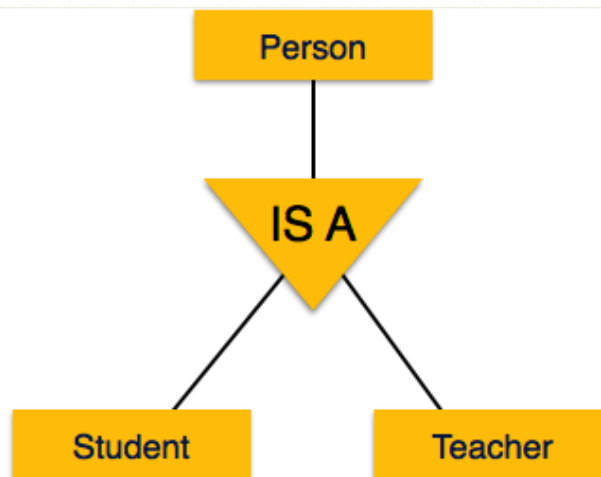
Ans: ❖ **Generalization:**

- Generalization is a bottom-up design process, which combines number of entity sets together, to represent more generalized view.
- For Example:
A particular student named “Raj” can be generalized along with all the students.
- In Generalization, shared characteristics can be attributes, associations, or methods.
- For example:
Pigeon, house sparrow, crow and dove can all be generalized as birds.



❖ **Specialization:**

- Specialization is top-down design process.
- It is the process to split up the entities into further sub-entities on the basis of their functionalities, specialties and features.
- These sub-groupings become lower-level entity sets that have attributes.
- It is represented by a triangle component labeled ISA (E.g., Person “is a” Student or Teacher).
- For example:
A person has name, date of birth, gender, etc. These properties are common in all persons, human beings. But in a company, persons can be identified as employee, employer, customer, or vendor, based on what role they play in the company.



- Specialization and generalization are simple reversals of each other; they are represented in an E-R diagram in the same way.
- It can have multiple specializations of an entity set based on different features.
- The ISA relationship also referred to as **superclass - subclass** relationship.

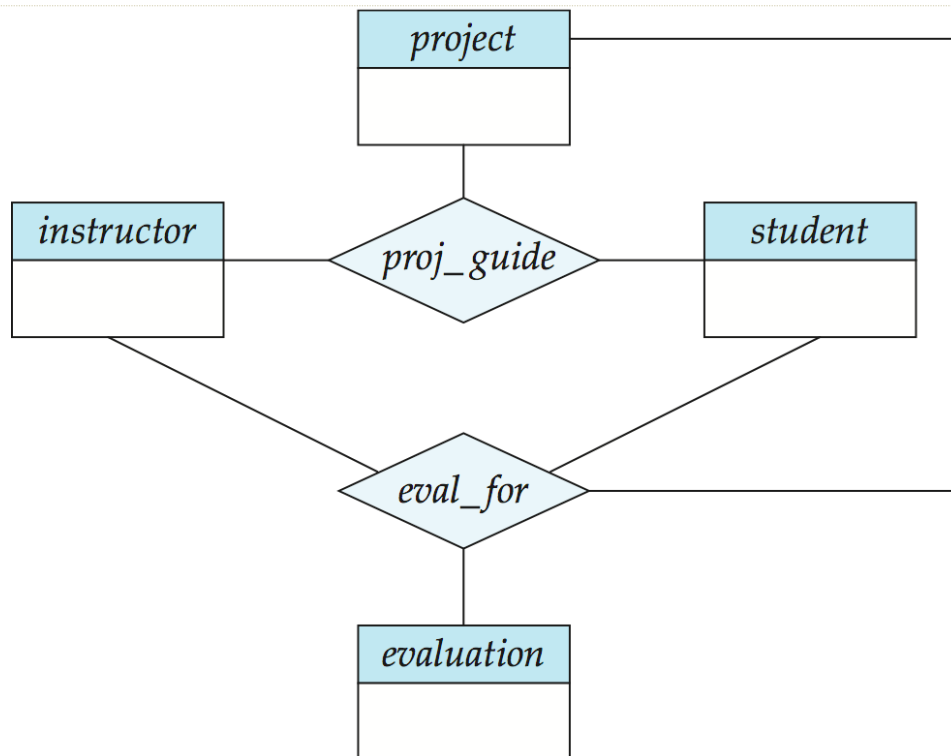
7. Which are the extended features of E-R Model? Explain Aggregation with example.

Ans: The extended features of E-R Model are as follows:

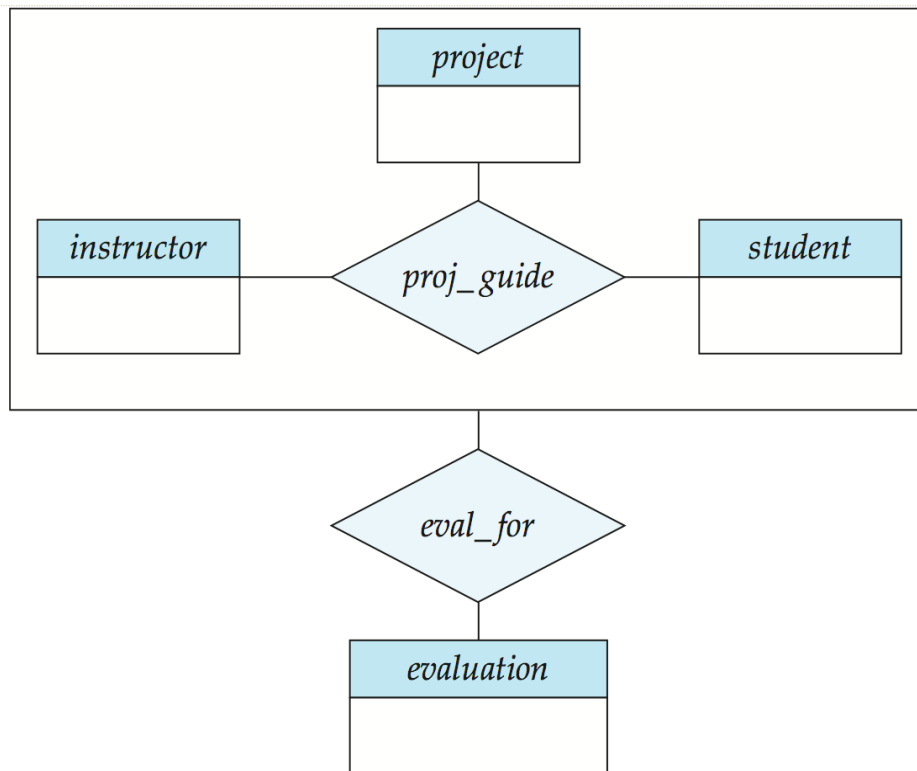
1. Generalization
2. Specialization
3. Attribute Inheritance
4. Aggregation

Aggregation:

- Aggregation is an abstraction through which relationships are treated as higher-level entities.
- Consider the ternary relationship *proj_guide*.
- Suppose we want to record evaluations of a student by a guide on a *project*.



- Relationship sets *eval_for* and *proj_guide* represent overlapping information.
 - Every *eval_for* relationship corresponds to a *proj_guide* relationship.
 - However, some *proj_guide* relationships may not correspond to any *eval_for* relationships.
 - ✓ So, we can't discard the *proj_guide* relationship.
- Eliminate this redundancy via aggregation....
 - Treat relationship as an abstract entity.
 - Allows relationships between relationships.
 - Abstraction of relationship into new entity.
- Without introducing redundancy, the following diagram represents:
 - A student is guided by a particular instructor on a particular project.
 - A student, instructor, project combination may have an associated evaluation.



ER Diagram with Aggregation