Database Management System

Data

- Data are characteristics or attributes, often numerical, collected through observation and can be qualitative(descriptive) or quantitative(numerical). Any facts and figures about an entity is called as Data.
- Data serves a crucial role in various sectors by facilitating analysis, supporting decision-making processes, and being fundamental to research activities.

Information

- Data becomes information when analyzed and placed in context, providing a basis for understanding, decision-making, and further analysis. Processed Data is called information.

Database

- A database is a structured collection of data, facilitating easy access, management, and updates., generally stored and accessed electronically from a computer system.

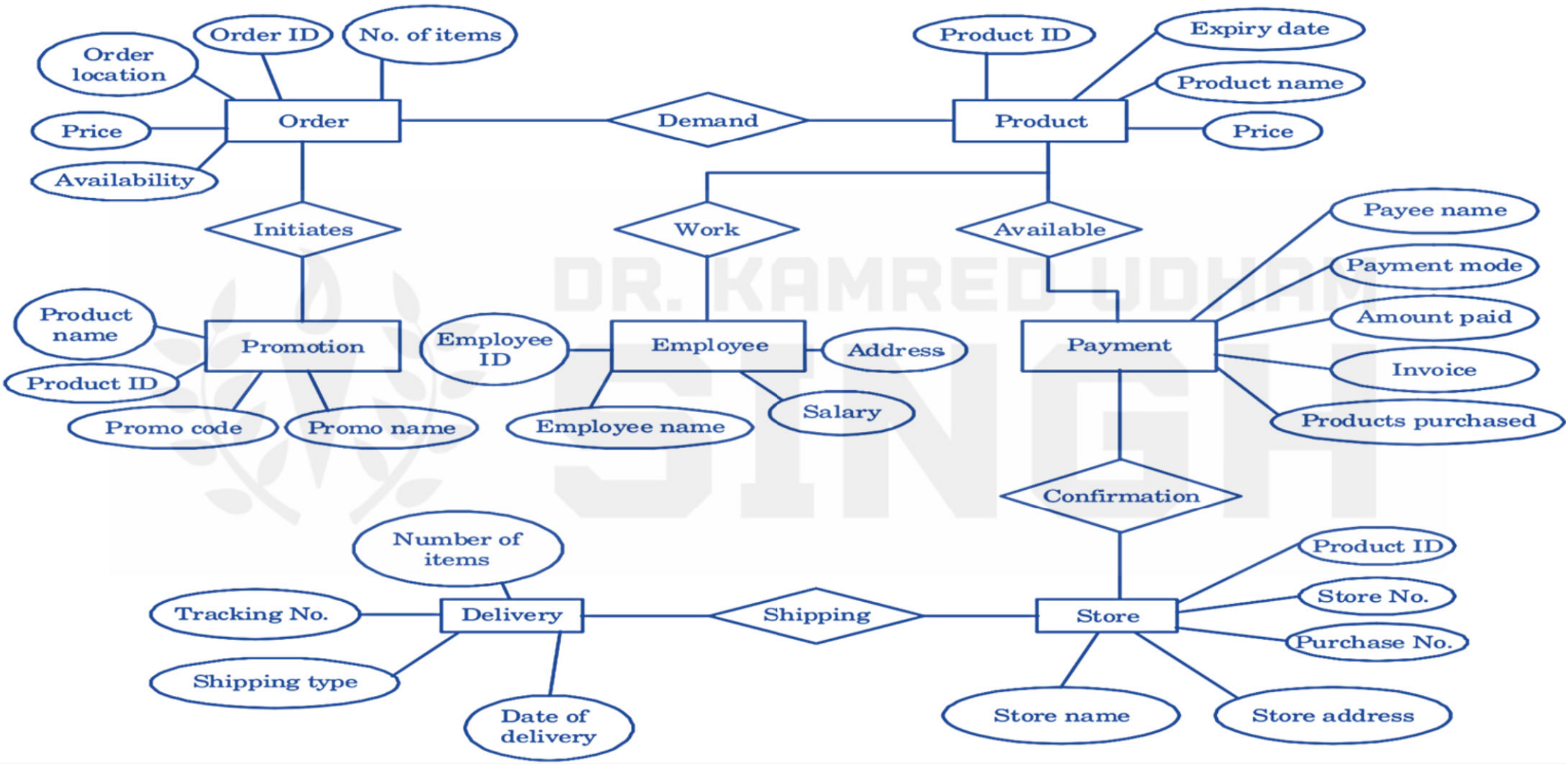
Database Management System

- A DBMS is a software facilitating efficient data storage, retrieval, and management in databases.
- Ensures data safety and integrity while offering accessibility and concurrency control.
- Supports functions like data querying, reporting, and analytics for informed decision-making.

Aspect	File System	Database Management System
Data Access	Slower data retrieval due to unstructured querying capabilities.	Structured querying capabilities allow for quicker data access.
Data Isolation	Challenges in correlating data across separate files leading to data isolation.	Facilitates data integration, reducing data isolation issues.
Data Integrity	Risk of inadvertent data alterations or deletions creating integrity problems.	Features to prevent unauthorized data alterations, maintaining integrity.
Atomicity Problem	Potential for data inconsistency due to incomplete operations, leading to atomicity problems.	Supports transaction properties like atomicity, ensuring operations are either completed fully or not at all.
Concurrent Access Anomalies	Conflicts and inconsistencies from simultaneous data access/modifications, causing concurrent access anomalies.	Advanced concurrency controls to manage multiple users accessing the database simultaneously, reducing anomalies.

ER Diagram

- Developed by Dr. Peter Chen in 1976, this conceptual-level method, grounded in real-world perceptions, facilitates diagrammatic data representation, simplifying comprehension for non-technical users. http://bit.csc.lsu.edu/~chen/chen.html
- The E-R data model, central to database design, encapsulates entities and their attributes within an enterprise schema, serving as a clear, standardized tool for translating real-world enterprise interactions into a conceptual schema.
- The E-R data model was developed to facilitate database design by allowing the specification of an enterprise schema that represents ***the overall logical structure of a database***.


Order ID
No. of items
Order location
Order
Price
Availability
Demand
Product ID
Expiry date
Product name
Price
Product
Initiates
Work
Available
Payee name
Payment mode
Amount paid
Invoice
Products purchased
Payment
Product name
Product ID
Promo code
Promo name
Promotion
Employee ID
Employee
Address
Employee name
Salary
Confirmation
Number of items
Tracking No.
Delivery
Shipping type
Date of delivery
Shipping
Product ID
Store No.
Purchase No.
Store
Store name
Store address

Conclusion

- It consists of collections of basic objects, called entities and of relationships among these entities and attributes which defines their properties.
- E-R model is very useful in mapping the meanings and interactions of real-world enterprises onto a conceptual schema.
- It is free from ambiguities and provides a standard and logical way of visualizing the data.
- As basically it is a diagrammatical representation easy to understand even by a non-technical user.

ENTITY

- An entity is a thing or an object in the real world that is distinguishable from other object based on the values of the attributes it possesses.
- 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.

Name	FName	City	Age	Salary
Smith	John	3	35	$280
Doe	Jane	1	28	$325
Brown	Scott	3	41	$265
Howard	Shemp	4	48	$359
Taylor	Tom	2	22	$250

Types of Entity

- **Tangible** - *Entities which physically exist in real world. E.g. - Car, Pen, locker*
- **Intangible** - *Entities which exist logically. E.g. – Account, video*

- In ER diagram we cannot represent an entity, as an entity is an instant not schema, and ER diagram is designed to understand the schema
- In a relational model entity is represented by a row or a tuple or a record in a table.

ENTIY SET- Collection of same type of entities that share the same properties or attributes.

- In an ER diagram an entity set is represented by a rectangle
- In a relational model it is represented by a separate table

Name	FName	City	Age	Salary
Smith	John	3	35	$280
Doe	Jane	1	28	$325
Brown	Scott	3	41	$265
Howard	Shemp	4	48	$359
Taylor	Tom	2	22	$250

Entity Set

Components of ER diagram-

An ER diagram is mainly composed of the following three components-

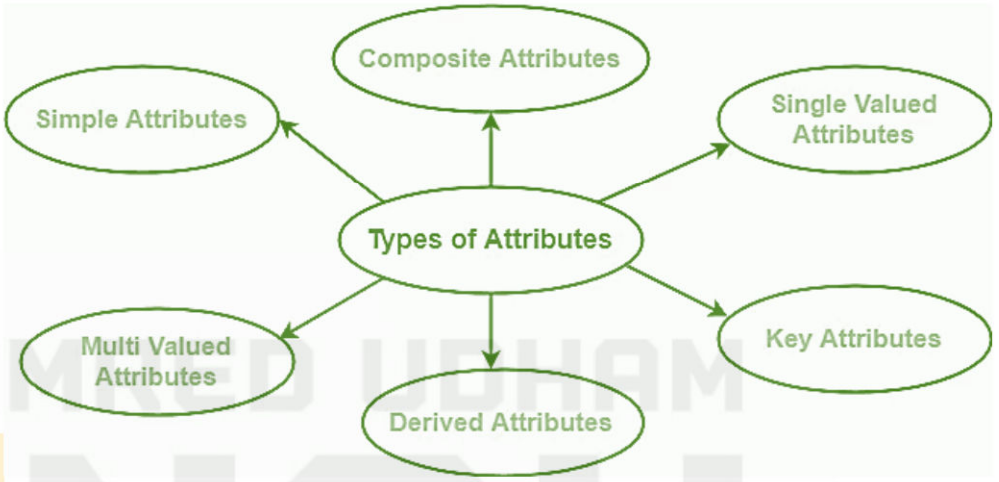
1.Attributes

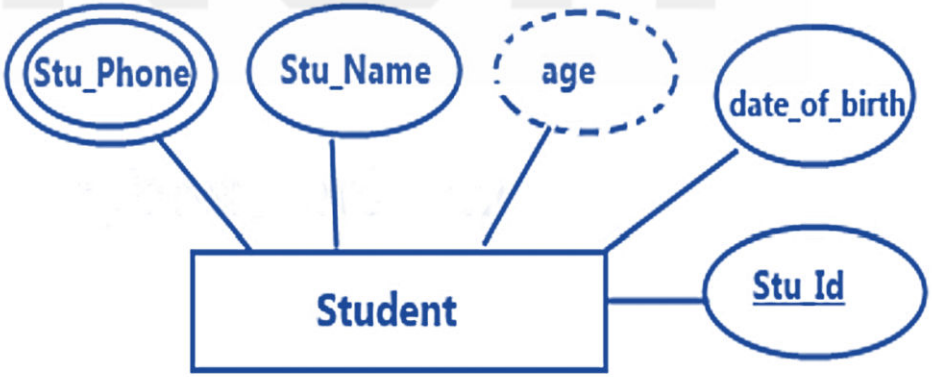
2.Entity Sets

3.Relationship Set

ATTRIBUTES

- Attributes are the units that define and describe properties and characteristics of entities.
- Attributes are the descriptive properties possessed by each member of an entity set. for each attribute, there is a set of permitted values called the domain.
- In an ER diagram, attributes are represented by an ellipse or oval connected to a rectangle.
- While in a relational model they are represented by independent column. e.g. Instructor (ID, name, salary, dept_name)





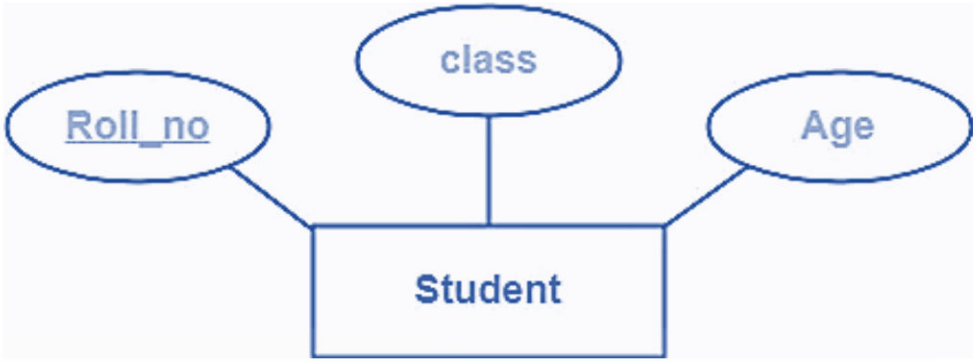
Types of Attributes

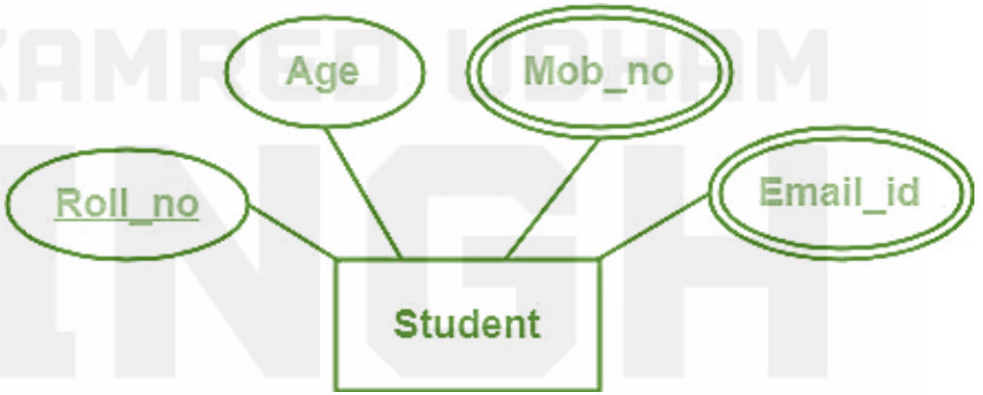
Simple - Attributes that cannot be divided further into sub-parts. E.g. Age

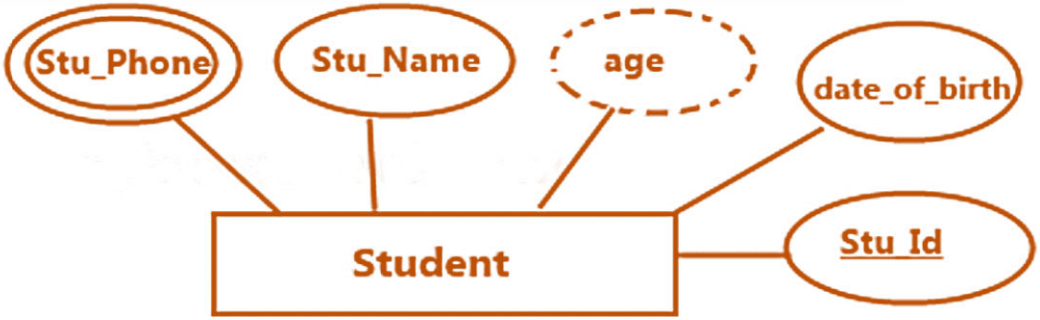
Single valued- Attributes having single value at any instance of time for an entity. E.g. – Aadhar no, dob.

Multivalued - Attributes which can have more than one value for an entity at same time. E.g. - Phone no, email, address.

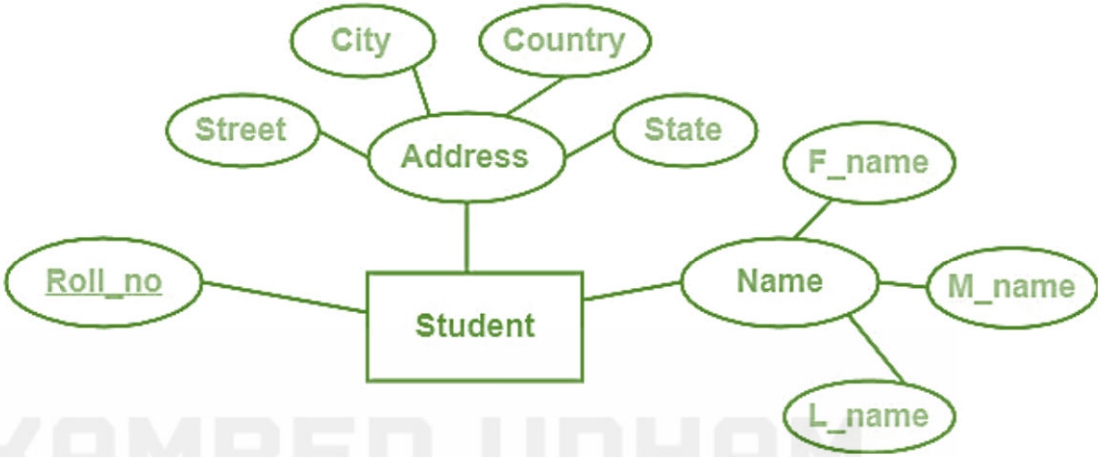
- A multivalued attribute is represented by a double ellipse in an ER diagram and by an independent table in a relational model.
- Separate table for each multivalued attribute, by taking mva and pk of main table as fk in new table



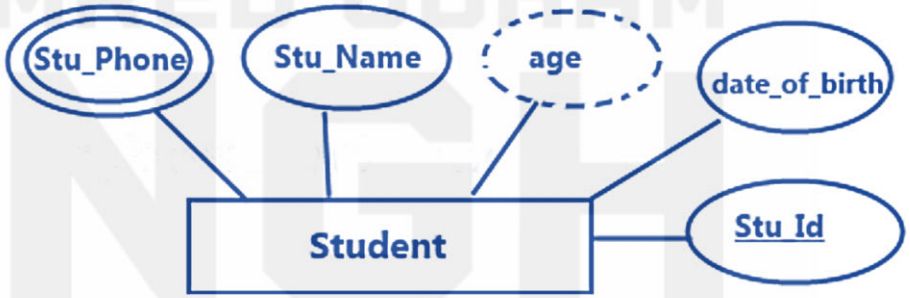




Composite - Attributes which can be further divided into sub parts, as simple attributes. A composite attribute is represented by an ellipse connected to an ellipse and in a relational model by a separate column.

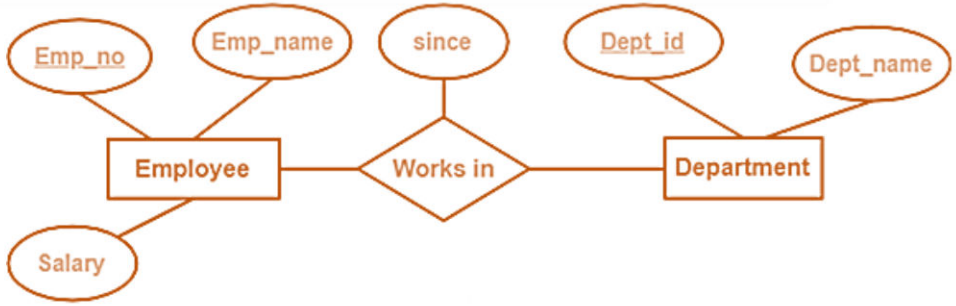


Stored - Main attributes whose value is permanently stored in database. E.g. date_of_birth



Derived- The value of these types of attributes can be derived from values of other Attributes. E.g. - Age attribute can be derived from date_of_birth and Date attribute.

Descriptive attribute - Attribute of relationship is called descriptive attribute.

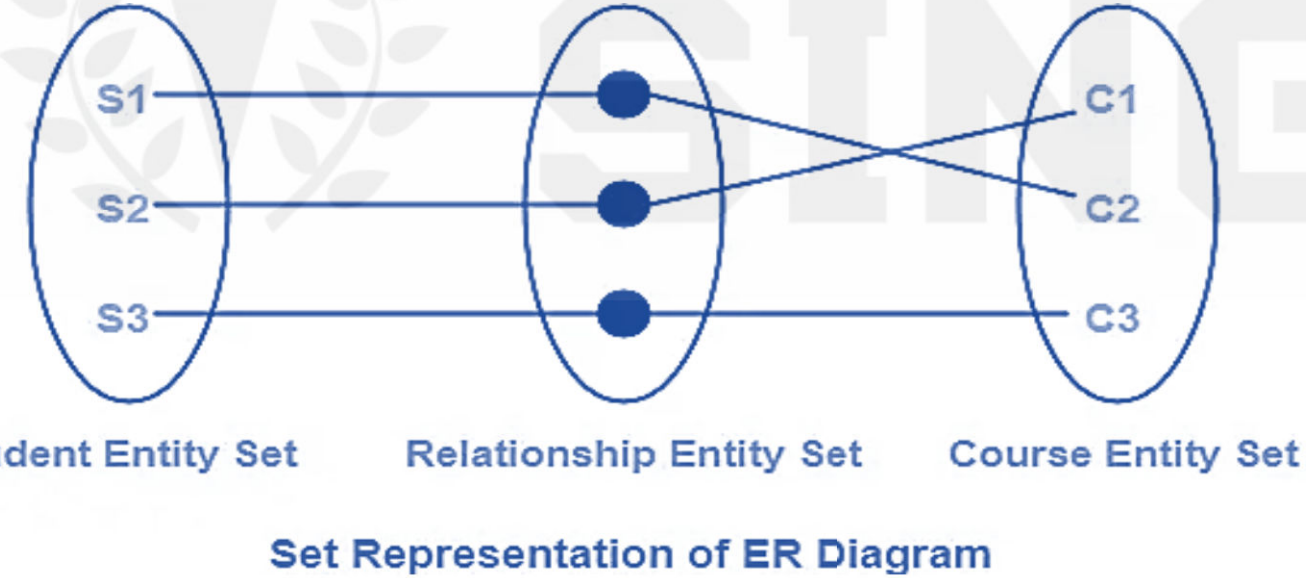


Relationship / Association

- Is an association between two or more entities of same or different entity set.
- In ER diagram we cannot represent individual relationship as it is an instance or data.

Note: - normally, people use the word relationship for relationship type, so don't get confused.

Enrolled in' is a relationship that exists between entities **Student** and **Course**.



Set Representation of ER Diagram

In an ER diagram it is represented by a diamond, while in relational model sometimes through foreign key and other time by a separate table.

Roll no	name	Age	Br_code	Br_name	Br_hod_name
1	A	19	101	Cs	Abc
2	B	18	101	Cs	Abc
3	C	20	101	Cs	Abc
4	D	20	102	Ec	Pqr

Roll no	name	Age	Br_code
1	A	19	101
2	B	18	101
3	C	20	101
4	D	20	102

Br_code	Br_name	Br_hod_name
101	Cs	Abc
102	ec	Pqr

Every relationship type has three components.

- Name
- Degree
- Structural constraints (cardinalities ratios, participation)

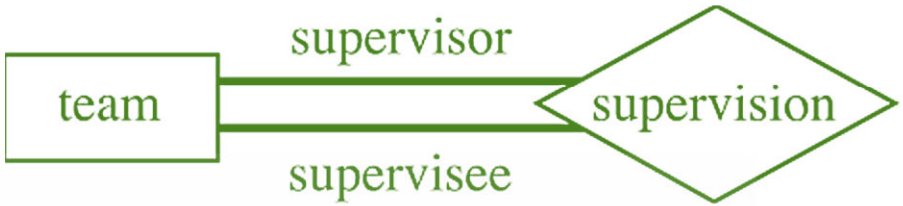
NAME- every relation must have a unique name.

Degree of a relationship/relationship set

- Means number of entities set(relations/tables) associated(participate) in the relationship set.
- Most of the relationship sets in a database system are binary.
- Occasionally, however, relationship sets involve more than two entity sets.
- Logically, we can associate any number of entity set in a relationship called N-ary Relationship.

Unary Relationship

- One single entity set participates in a Relationship, which means two entities of the same entity set are related to each other.
- These are also called self-referential Relationship sets.
- E.g.- A member in a team maybe supervisor of another member in team.

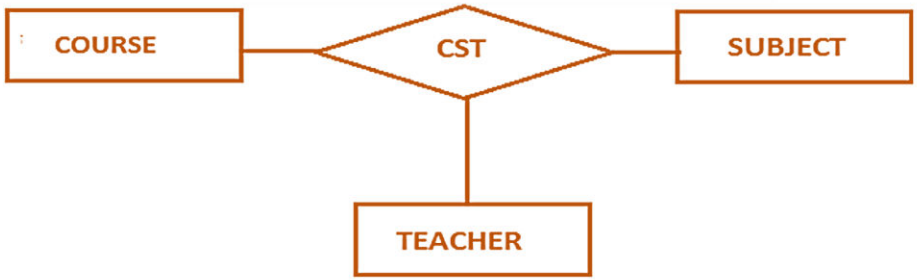


Binary Relationship

Two entity sets participate in a Relationship. It is most common Relationship.

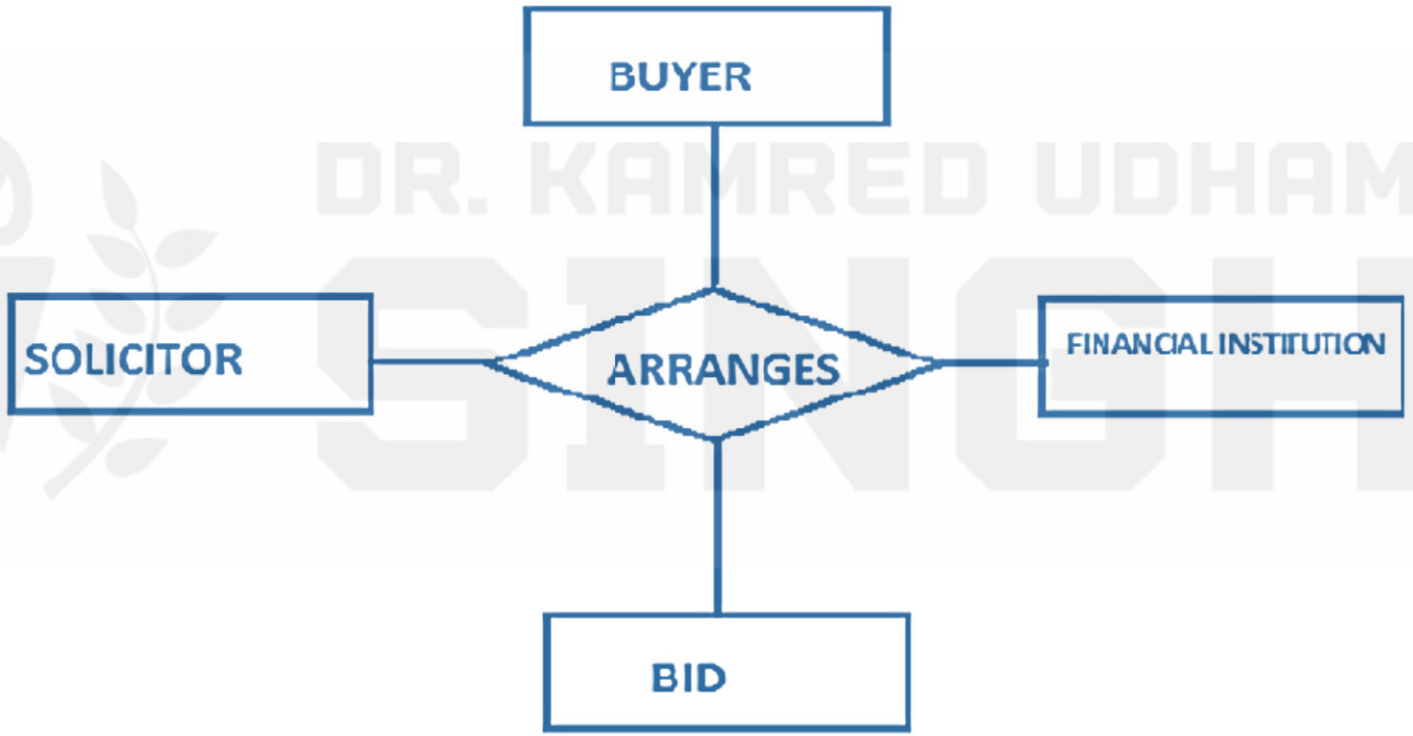
Ternary Relationship

When three entities participate in a Relationship. E.g. The University might need to record which teachers taught which subjects in which courses.



Quaternary Relationship

When four entities participate in a Relationship.



- But the most common relationships in ER models are ***Binary***.

Structural constraints (Cardinalities Ratios, Participation)

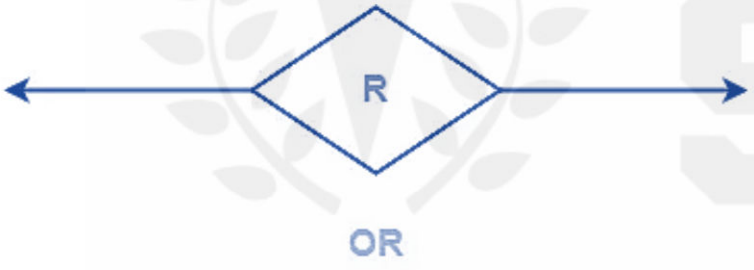
- An E-R enterprise schema may define certain constraints to which the contents of a database must conform.
- Express the number of entities to which another entity can be associated via a relationship set. Four possible categories are-

  - One to One (1:1) Relationship.
  - One to Many (1: M) Relationship.
  - Many to One (M: 1) Relationship.
  - Many to Many (M: N) Relationship.

One to One (1:1) Relationship –

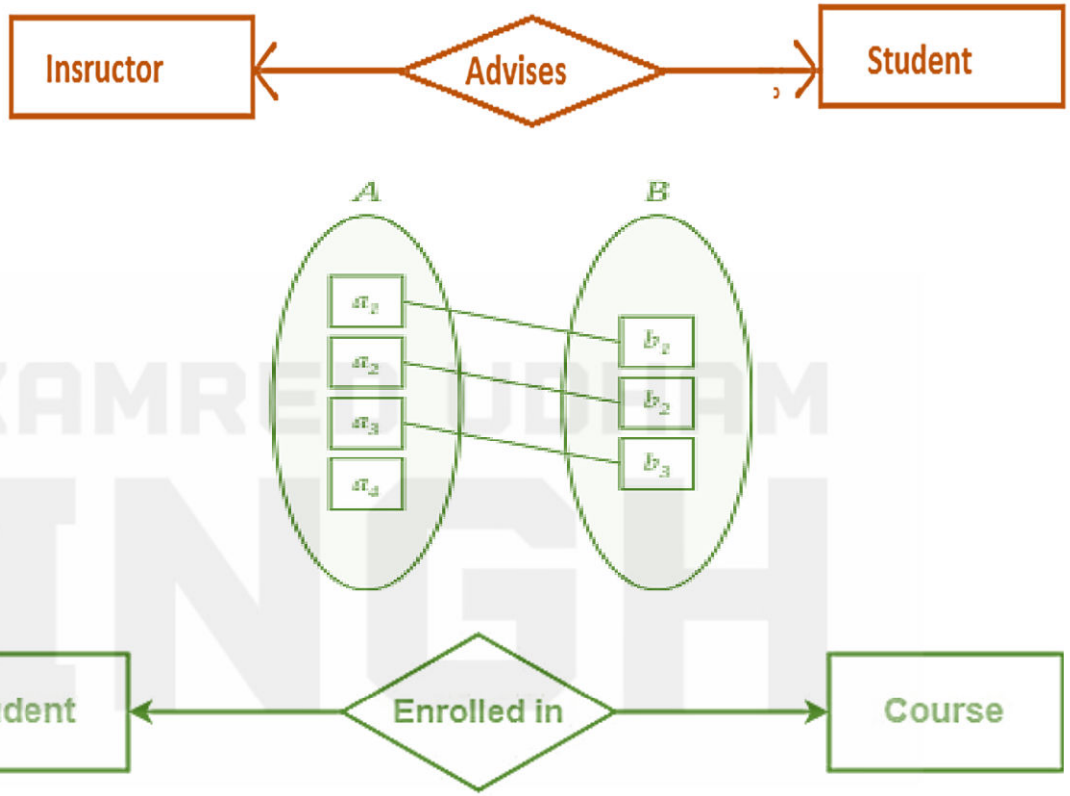
An entity in A is associated with at most one entity in B, and an entity in B is associated with at most one entity in A.

E.g.- The directed line from relationship set advisor to both entities set indicates that 'an instructor may advise at most one student, and a student may have at most one advisor'.



OR

Cardinality Ratio = 1 : 1



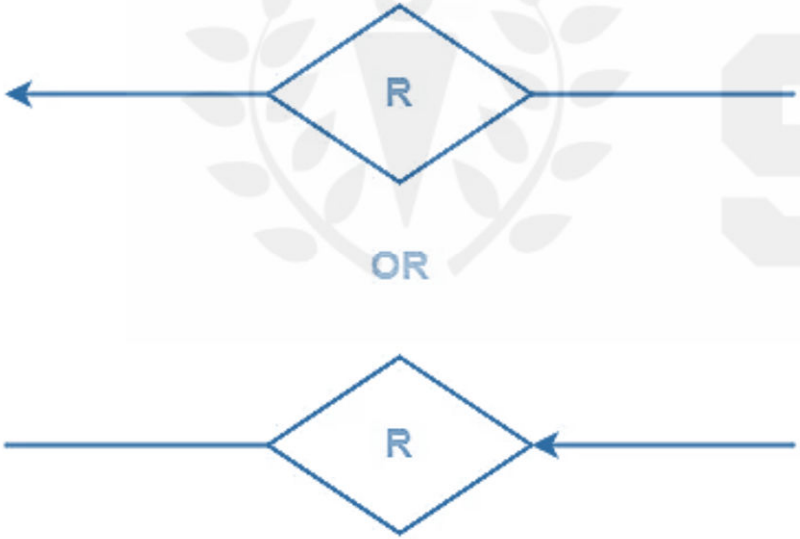
One to One Relationship

- One student can enroll in at most one course.
- One course can be enrolled by at most one student.

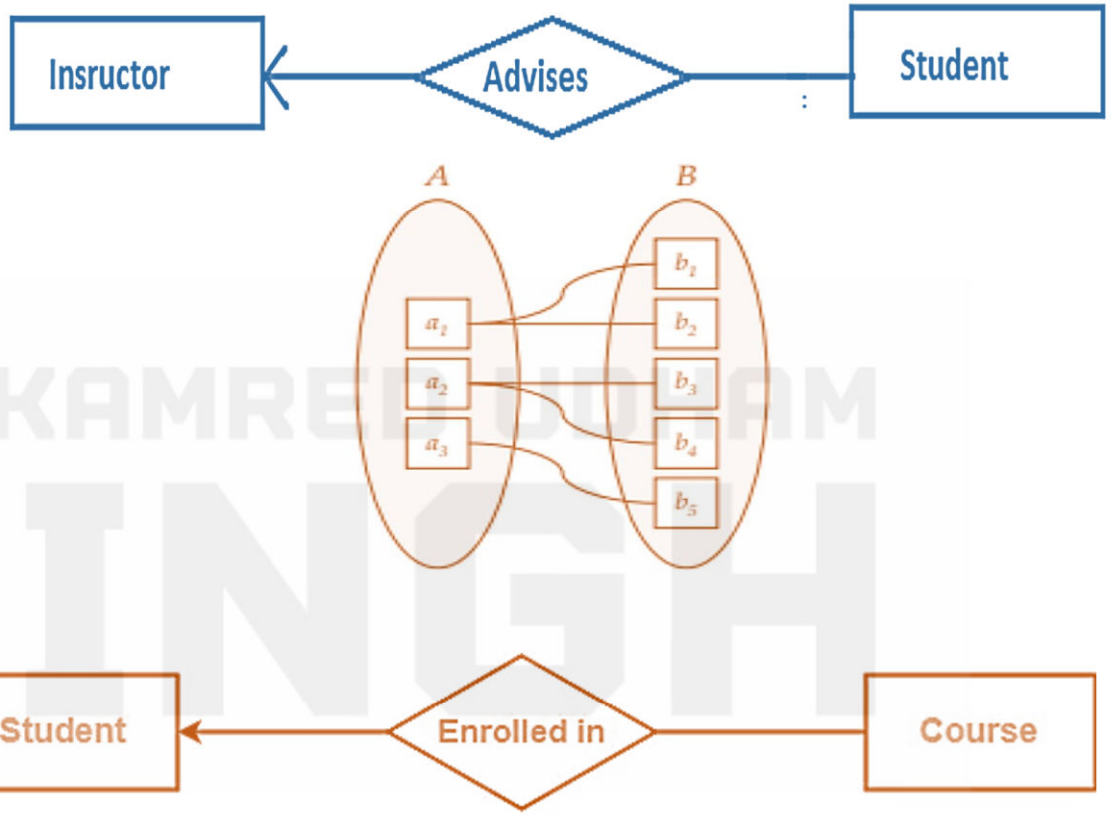
One to Many (1: M) Relationship

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.

E.g.- This indicates that an instructor may advise many students, but a student may have at most one advisor.



Cardinality Ratio = 1 : n



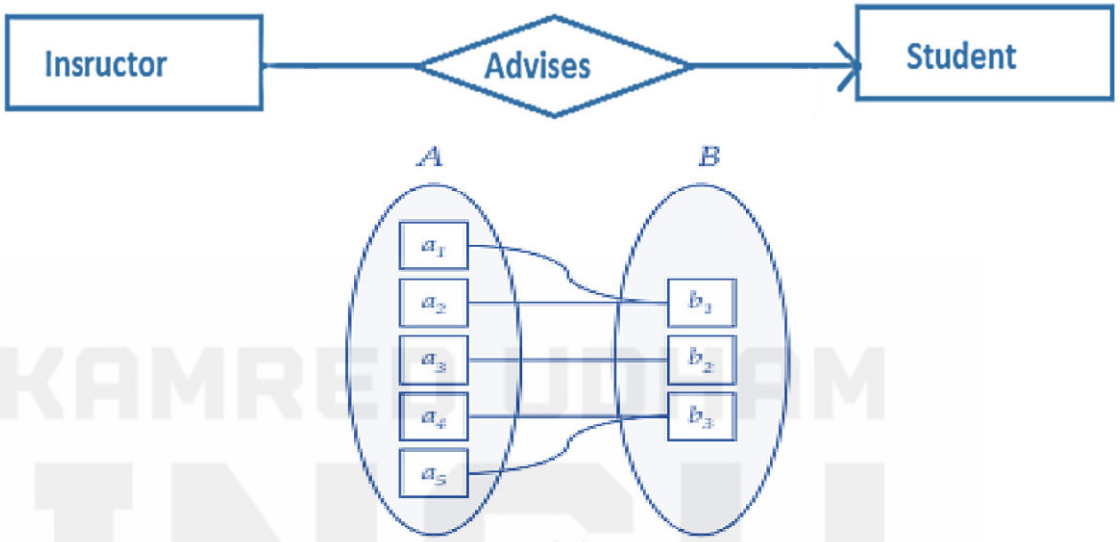
One to Many Relationship

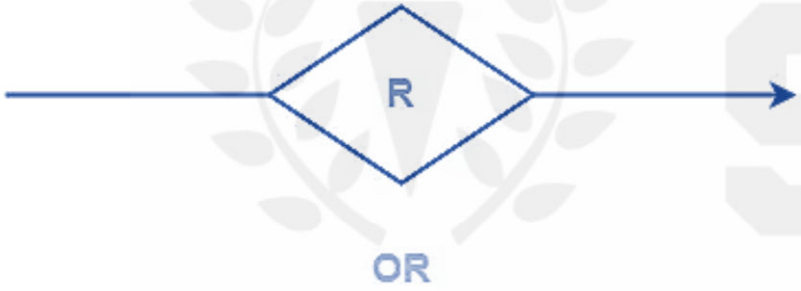
- One student can enroll in any number (zero or more) of courses.
- One course can be enrolled by at most one student.

Many to One (M: 1) Relationship –

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.

E.g.- This indicates that student may have many instructors but an instructor can advise at most one student.





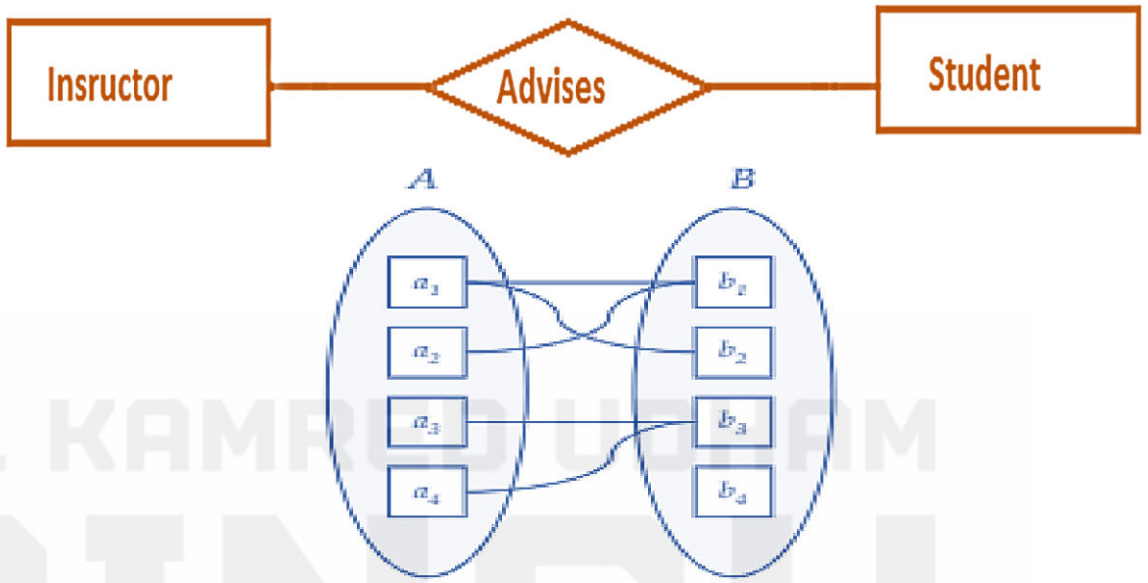
Many to One Relationship

Cardinality Ratio = m : 1

- One student can enroll in at most one course.
- One course can be enrolled by any number (zero or more) of students.

Many to Many(M:N) Relationship –
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.

E.g.- This indicates a student may have many advisors and an instructor may advise many students.



Cardinality Ratio = m : n

Many to Many Relationship

- One student can enroll in any number (zero or more) of courses.
- One course can be enrolled by any number (zero or more) of students.

PARTICIPATION CONSTRAINTS

It defines the participation of entities of an entity type in a relationship.

- **Partial participation**
- **Total Participation**

Total Participation-

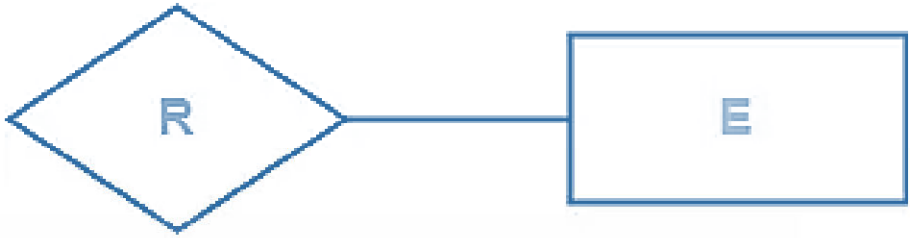
- It specifies that each entity in the entity set must compulsorily participate in at least one relationship instance in that relationship set.
- That is why, it is also called as **mandatory participation.**
- Total participation is represented using a double line between the entity set and relationship set.

Total Participation

- Double line between the entity set "Student" and relationship set "Enrolled in" signifies total participation.
- It specifies that each student must be enrolled in at least one course.

Partial Participation-

- It specifies that each entity in the entity set may or may not participate in the relationship instance in that relationship set.
- That is why, it is also called as **optional participation.**
- Partial participation is represented using a single line between the entity set and relationship set.

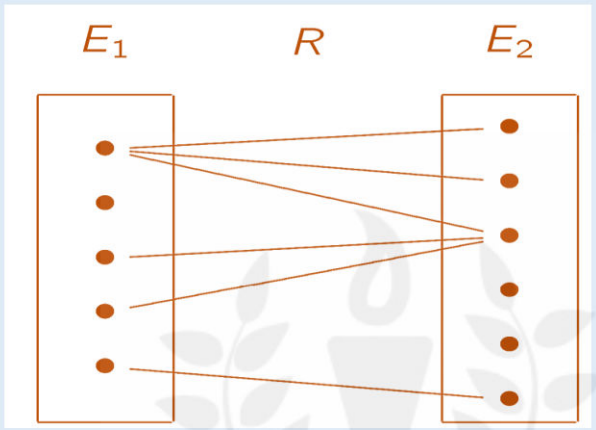


Partial Participation

- Single line between the entity set "Course" and relationship set "Enrolled in" signifies partial participation.
- It specifies that there might exist some courses for which no enrollments are made.

Relationship between Cardinality and Participation Constraints-

General ER relationship



- In general, there is no restriction on how often an entity participates in an relationship R.
- An entity can be connected to one entity of the other type, to more than one, or it can have no R-partner at all.
- However, specific application semantics dictate to how many E2 entities an E1 entity can be related:

Relationship between Cardinality and Participation Constraints……..Cont..

- The ER model introduces the (min, max) notation to specify an interval of possible participations in an relationship: E1 (m1, n1) R (m2, n2) E2
- An entity of type E1 may be related to at least m1 and at most n1 entities of type E2.
- Likewise, m2 is the minimum number and n2 is the maximum number of E1 entities to which an E2 entity is related

- "∗" may be used as maximum if there is no limit.
- (0, ∗) means no restriction at all (general relationship)

- ***Max cardinality*** – it defines the maximum no of times an entity set occurrence participating in a relationship.
- ***Min cardinality*** **-** it defines the minimum no of times an entity set occurrence participating in a relationship.

Minimum cardinality tells whether the participation is partial or total.

- ❑ If minimum cardinality = 0, then it signifies partial participation.
- ❑ If minimum cardinality = 1, then it signifies total participation.

Maximum cardinality tells the maximum number of entities that participates in a relationship set.

Relationship between Cardinality and Participation Constraints……..Cont..

Marriage

"A man can be married to at most one woman and vice versa."

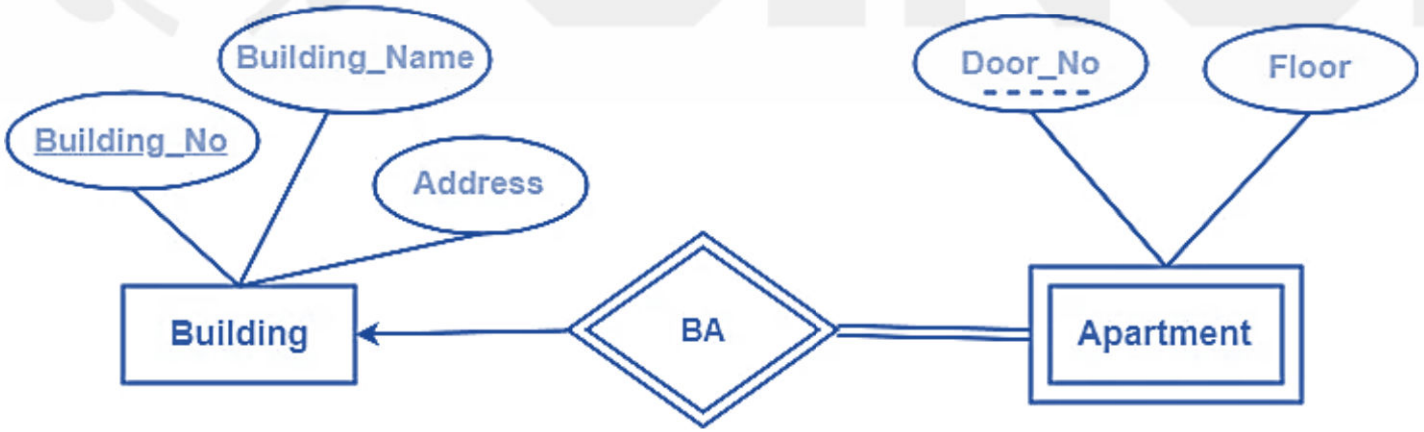
Airport Locations

"An airport lies in exactly one country. A country may have arbitrarily many airports (and maybe none at all)."

- Normally, the minimum cardinality will be 0 or 1, and the maximum cardinality will be 1 or $*$.
- Thus, only the (0, 1), (1, 1), (0, $*$), (1, $*$) cardinalities are common in practice.
- To understand a relationship, one must know the cardinality specifications on both sides.
- The maximum cardinalities on each side are used to distinguish between many-to-many, one-to-many / many-to-one, and one-to-one relationships.

STRONG AND WEAK ENTITY SET

- An entity set is called strong entity set, if it has a primary key, all the tuples in the set are distinguishable by that key.
- An entity set that does not process sufficient attributes to form a primary key is called a weak entity set.
- It contains discriminator attributes (partial key) which contain partial information about the entity set, but it is not sufficient enough to identify each tuple uniquely. Represented by double rectangle.
- For a weak entity set to be meaningful and converted into strong entity set, it must be associated with another strong entity set called the identifying or owner entity set i.e. weak entity set is said to be existence dependent on the identity set.
- The identifying entity set is said to own weak entity set that it identifies.
- A weak entity set may participate as owner in an identifying relationship with another weak entity set.
- The relationship associating the weak entity set with the identifying entity set is called the **identifying relationship (double diamonds).**
- The identifying relationship is many to one from the weak entity set to identifying entity set, and the participation of the weak entity set in relationship is always total.
- The primary key of weak entity set will be the union of primary key and discriminator attributes.

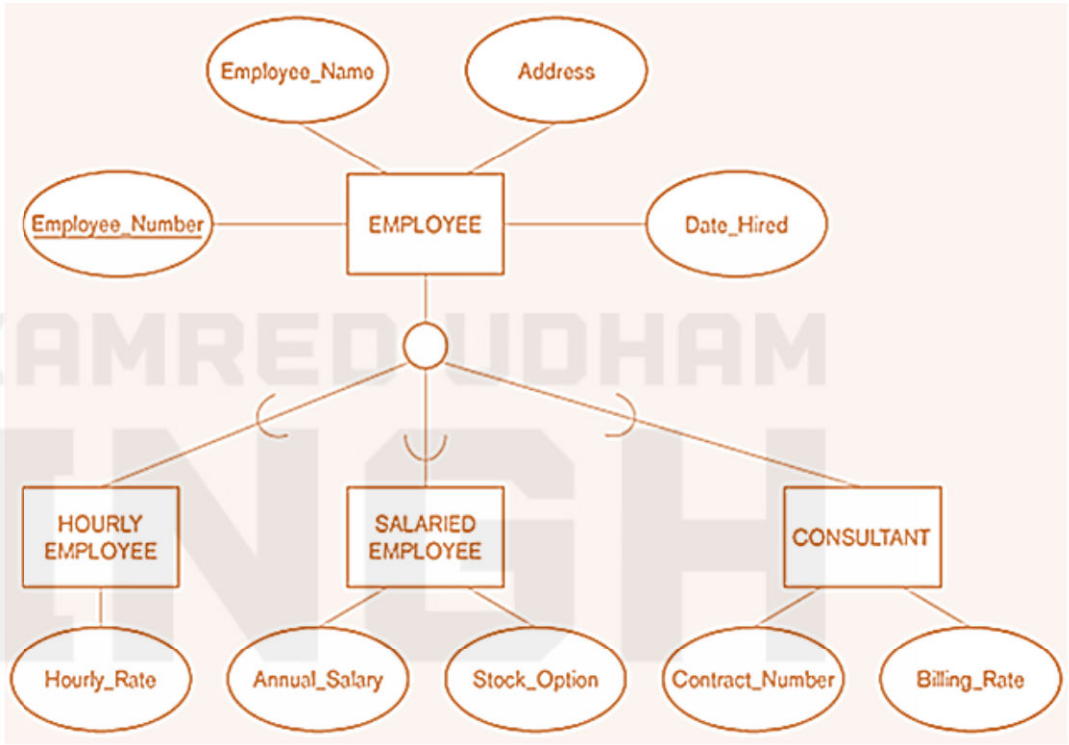


Differences between Strong entity set and Weak entity set-

Strong entity set	Weak entity set
A single rectangle is used for the representation of a strong entity set.	A double rectangle is used for the representation of a weak entity set.
It contains sufficient attributes to form its primary key.	It does not contain sufficient attributes to form its primary key.
A diamond symbol is used for the representation of the relationship that exists between the two strong entity sets.	A double diamond symbol is used for the representation of the identifying relationship that exists between the strong and weak entity set.
A single line is used for the representation of the connection between the strong entity set and the relationship.	A double line is used for the representation of the connection between the weak entity set and the relationship set.
Total participation may or may not exist in the relationship.	Total participation always exists in the identifying relationship.

Extended ER model:

The EER model includes all the modeling concepts of the ER model. In addition, it includes the concepts of **subclass and superclass, attribute and relationship inheritance** with the following additional concepts. The enhanced ER (EER) model concept we take up is that of a **subtype** or **subclass** of an entity type. The name of an entity type is used to represent both a *type of entity* and the *entity set* or *collection of entities of that type* that exist in the database.

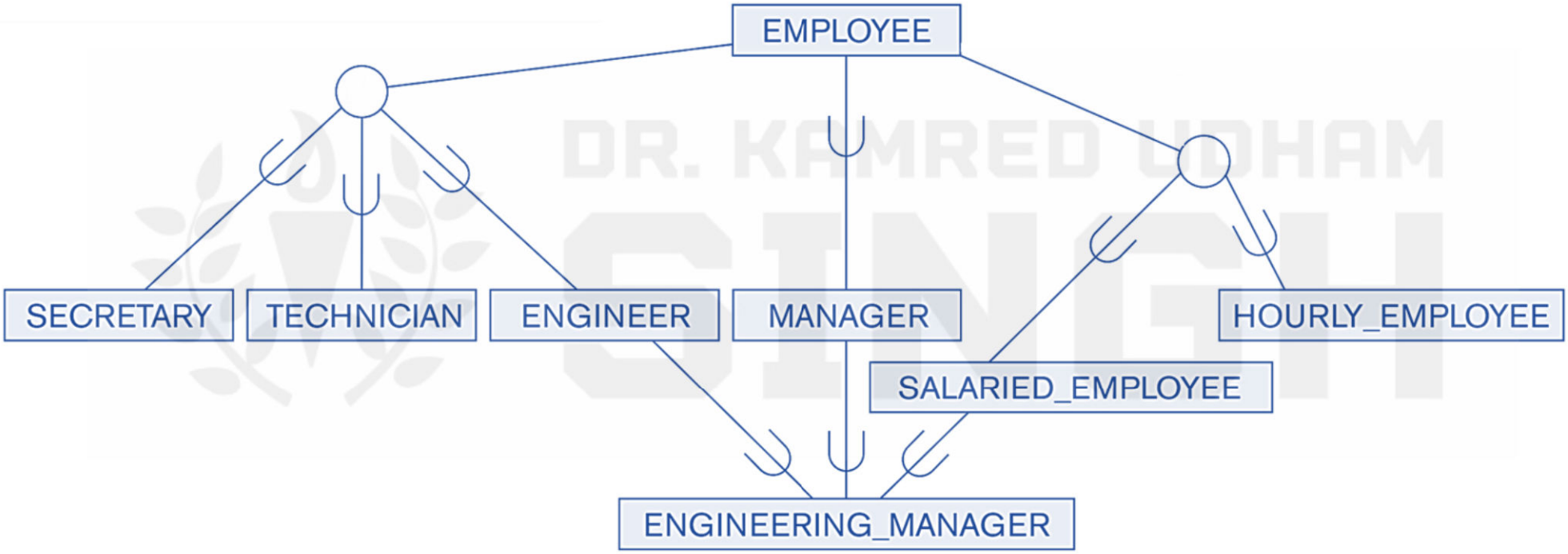


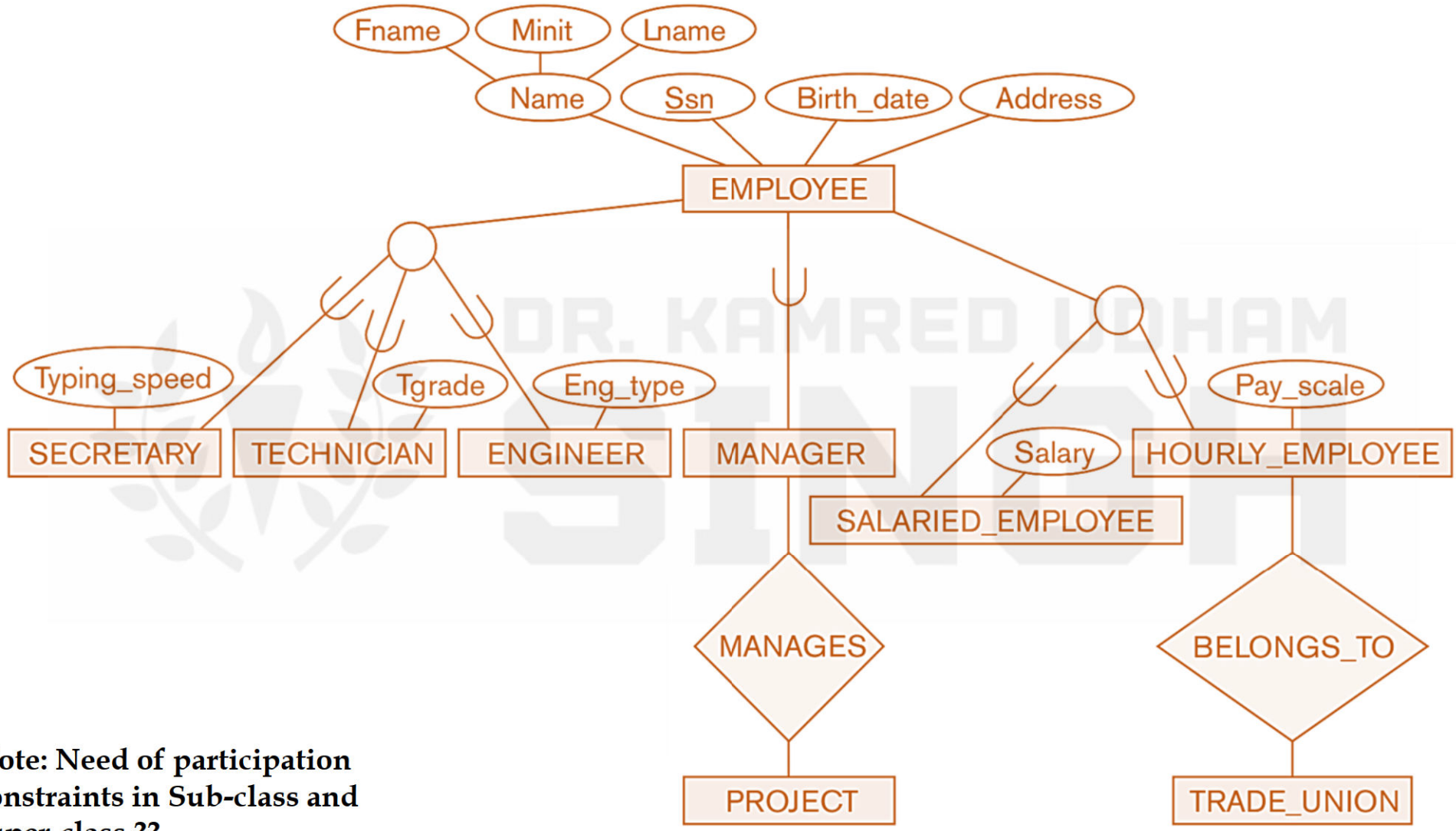
- ❑ Specialization
- ❑ Generalization
- ❑ Categorization or union-type

Subclass or subtype and superclass or supertype :

This feature allows us to model a general entity and subdivide it into several specialized entity sets. Thus, a superclass is an entity set that includes distinct super-classes required to be represented in the data model. A sub-class entity set that has a distinct role and is also a member of a superclass.

Subclass and superclass and multiple inheritance cont.

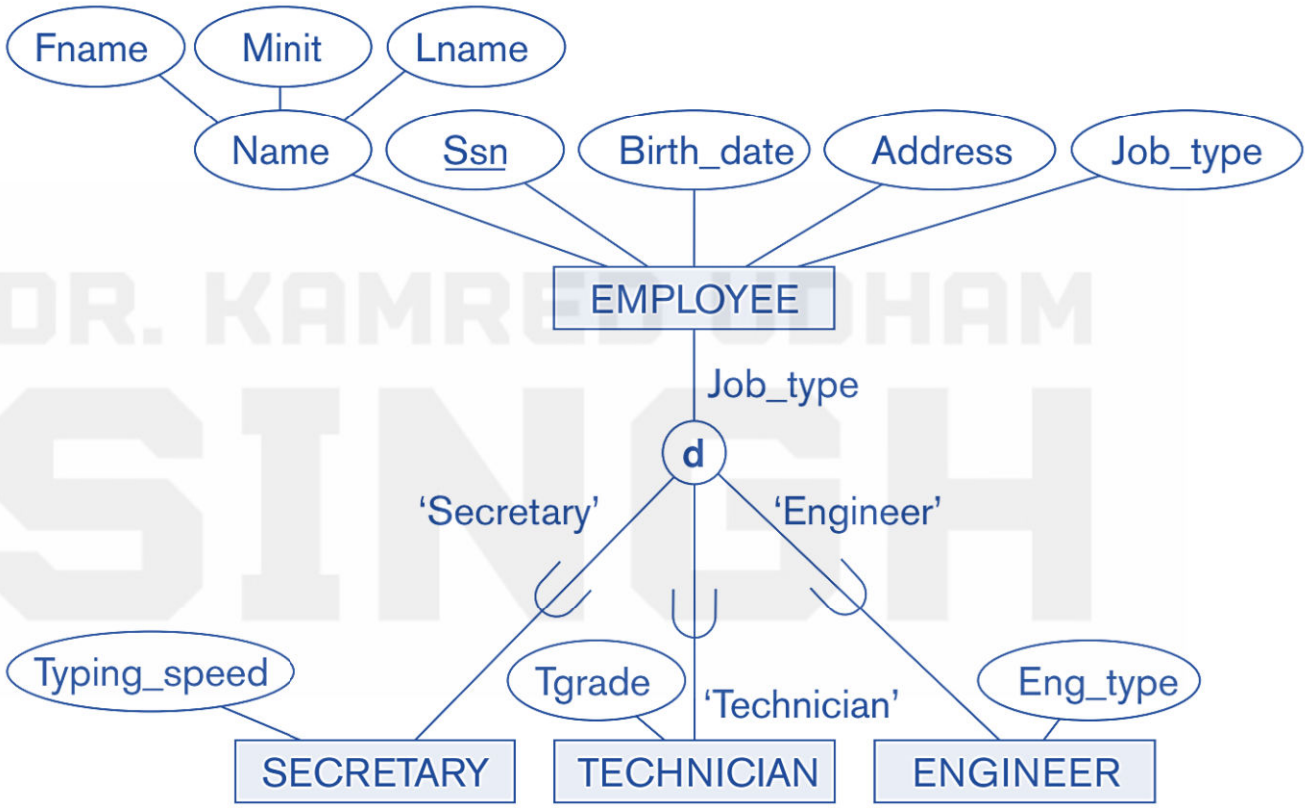




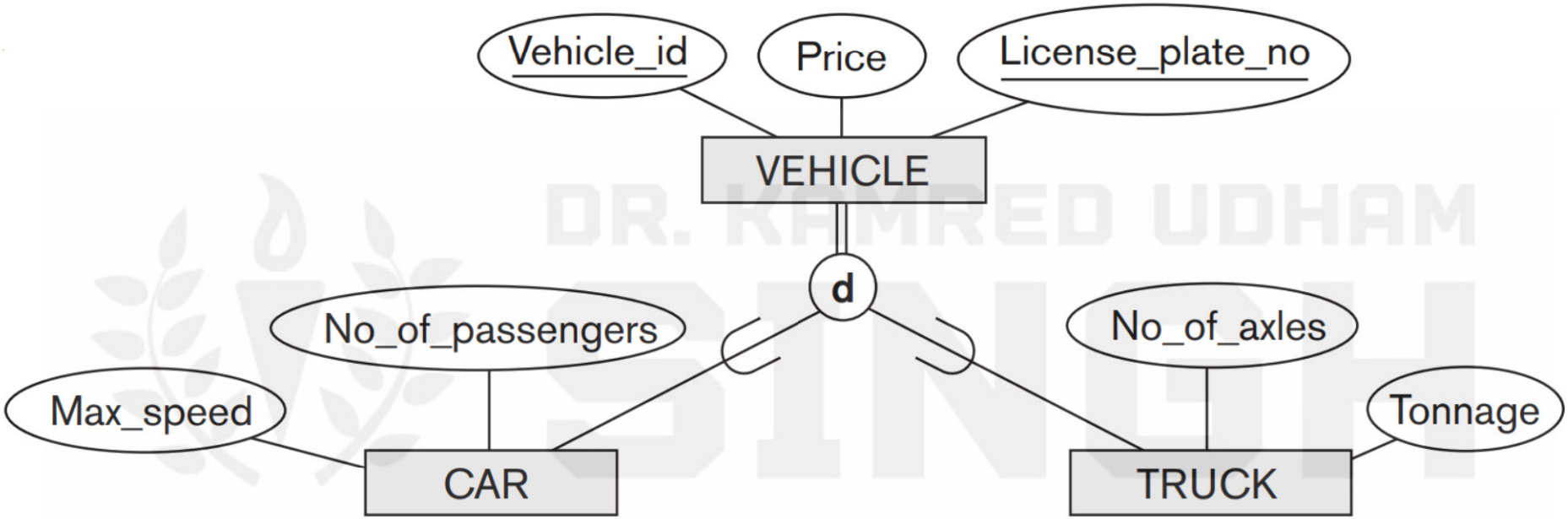
Note: Need of participation constraints in Sub-class and super-class ??

CONSTRAINTS ON SUBCLASSES

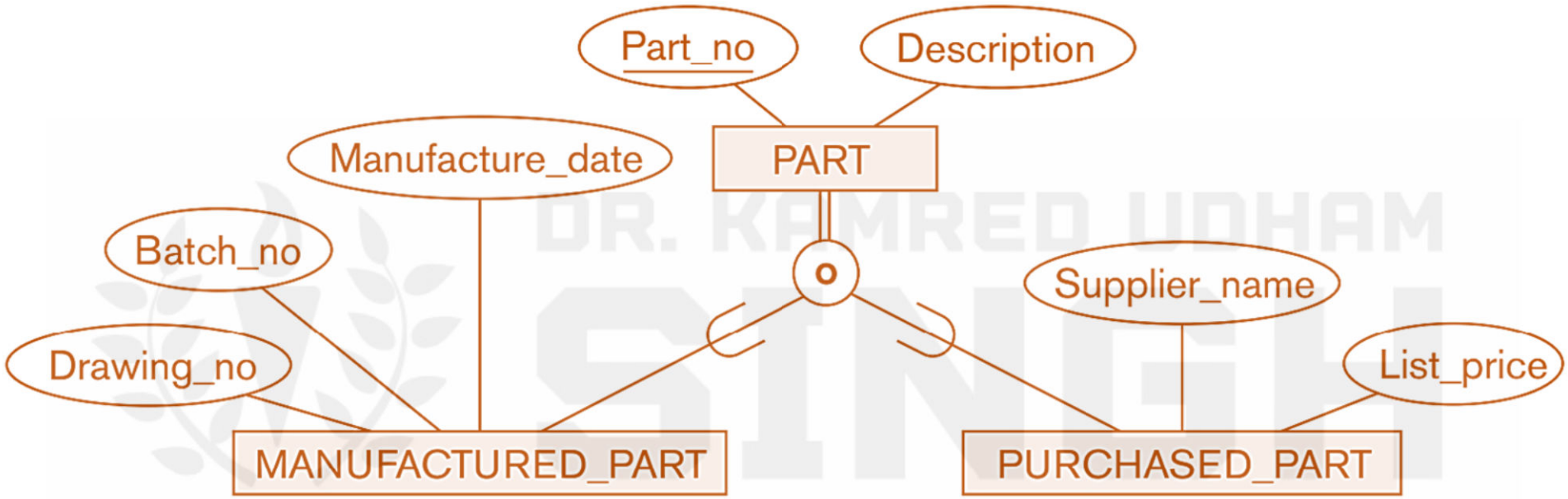
- Disjoint, partial
- Disjoint, total
- Overlapping, total
- Overlapping, partial



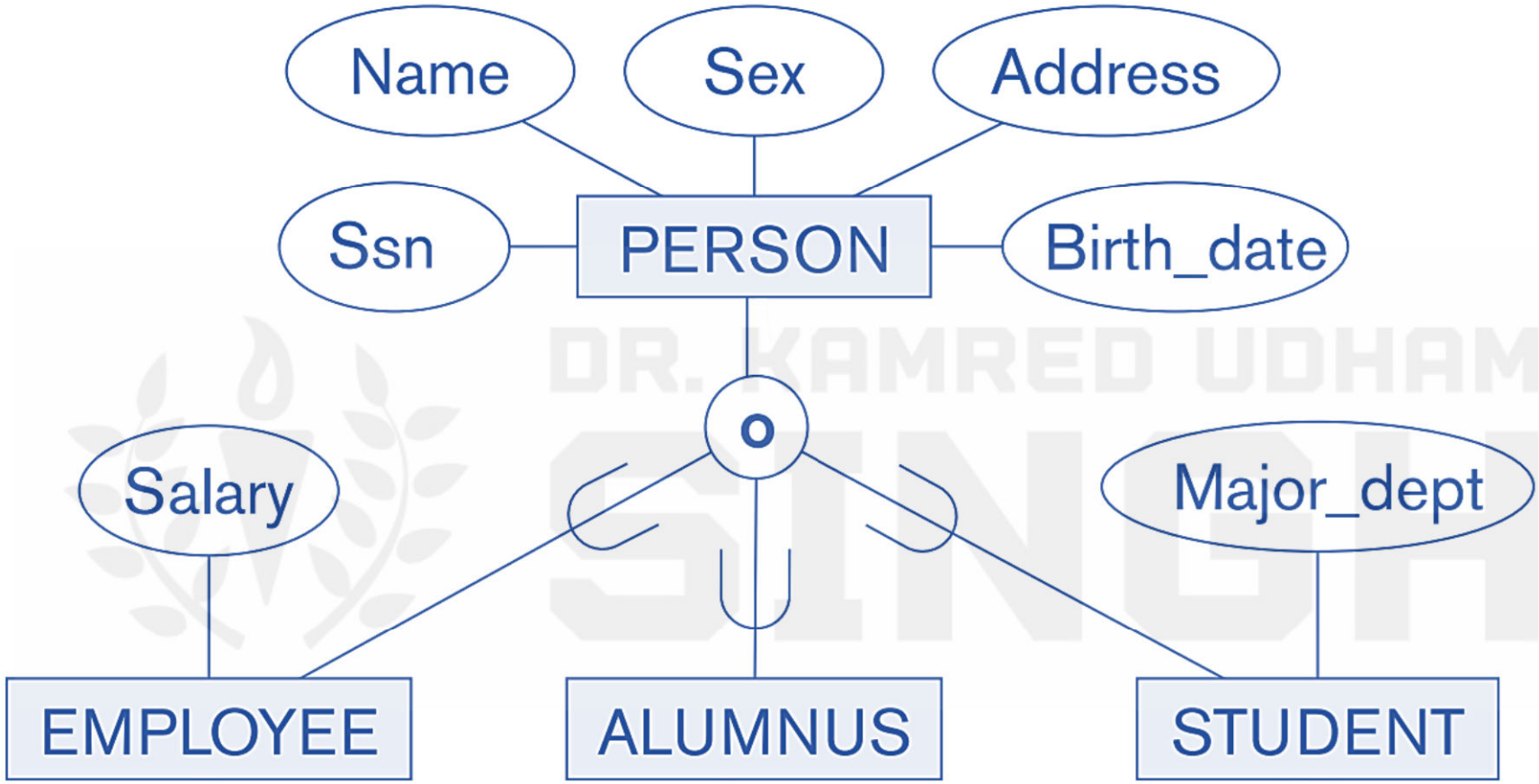
Disjoint, partial



Disjoint, total



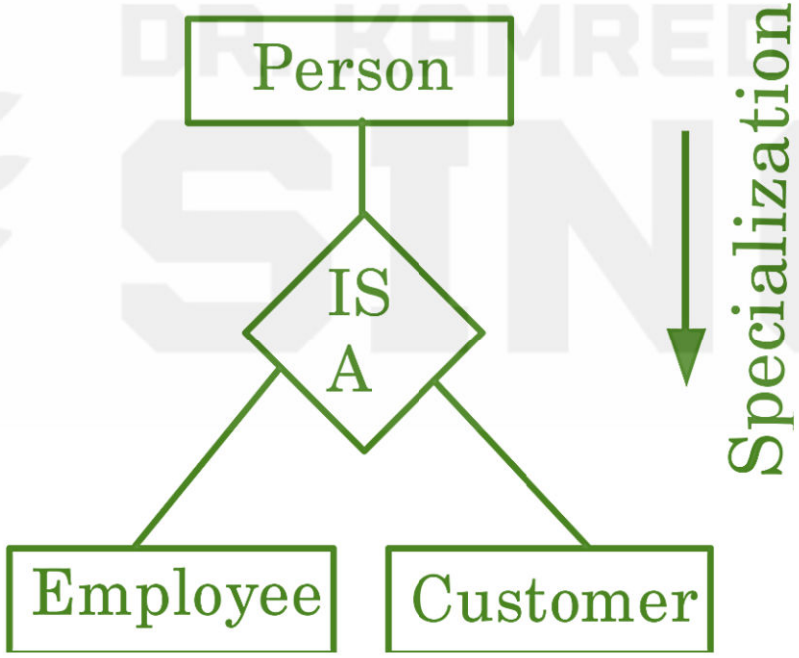
Overlapping, total

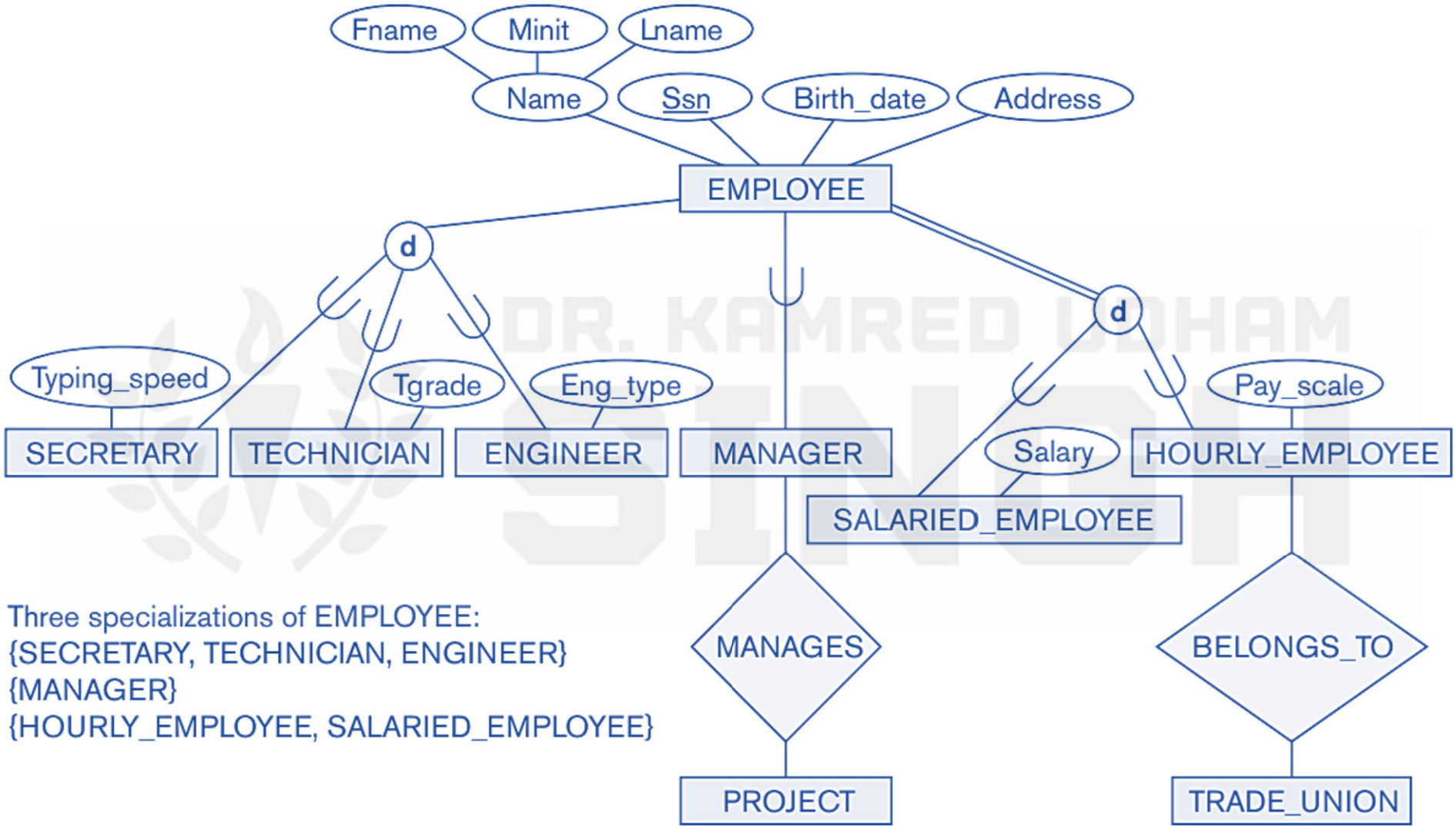


Overlapping, partial

Specialization

Specialization is the process of defining a *set of subclasses* of an entity type; this entity type is called the **superclass** of the specialization. The set of subclasses that forms a specialization is defined on the basis of some distinguishing characteristic of the entities in the superclass. It is top-down approach delineates complexity into simpler components.



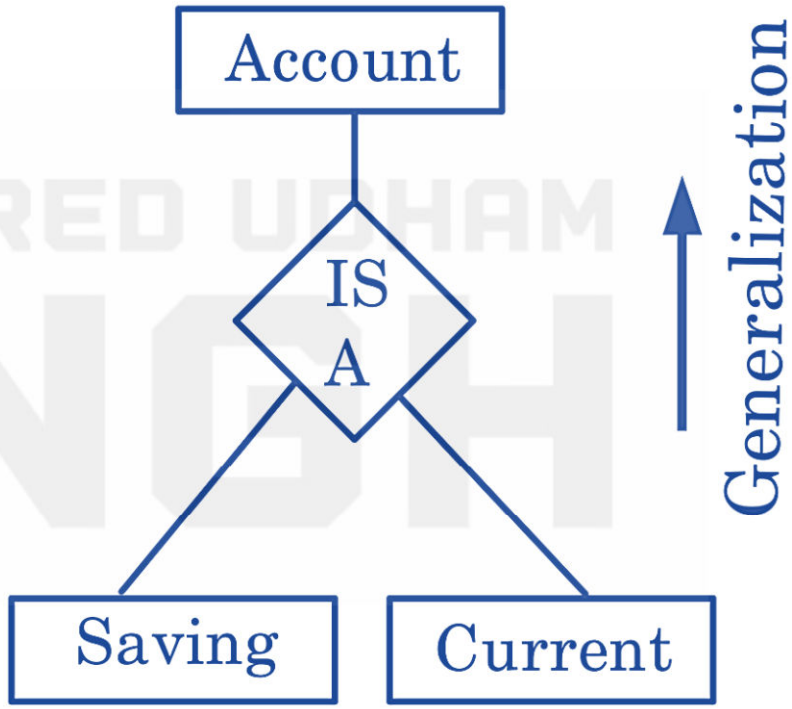


Three specializations of EMPLOYEE:
{SECRETARY, TECHNICIAN, ENGINEER}
{MANAGER}
{HOURLY_EMPLOYEE, SALARIED_EMPLOYEE}

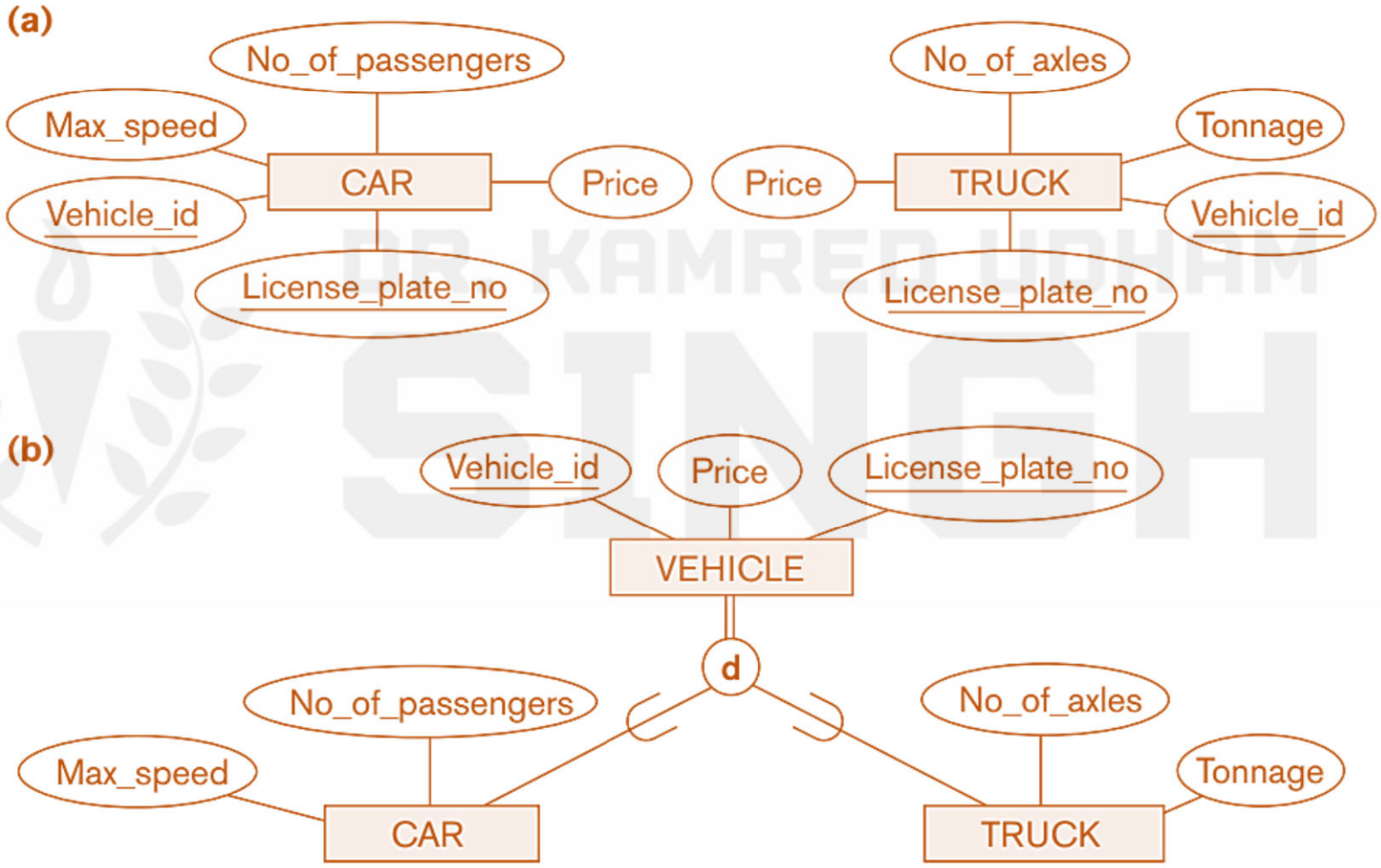
Generalization

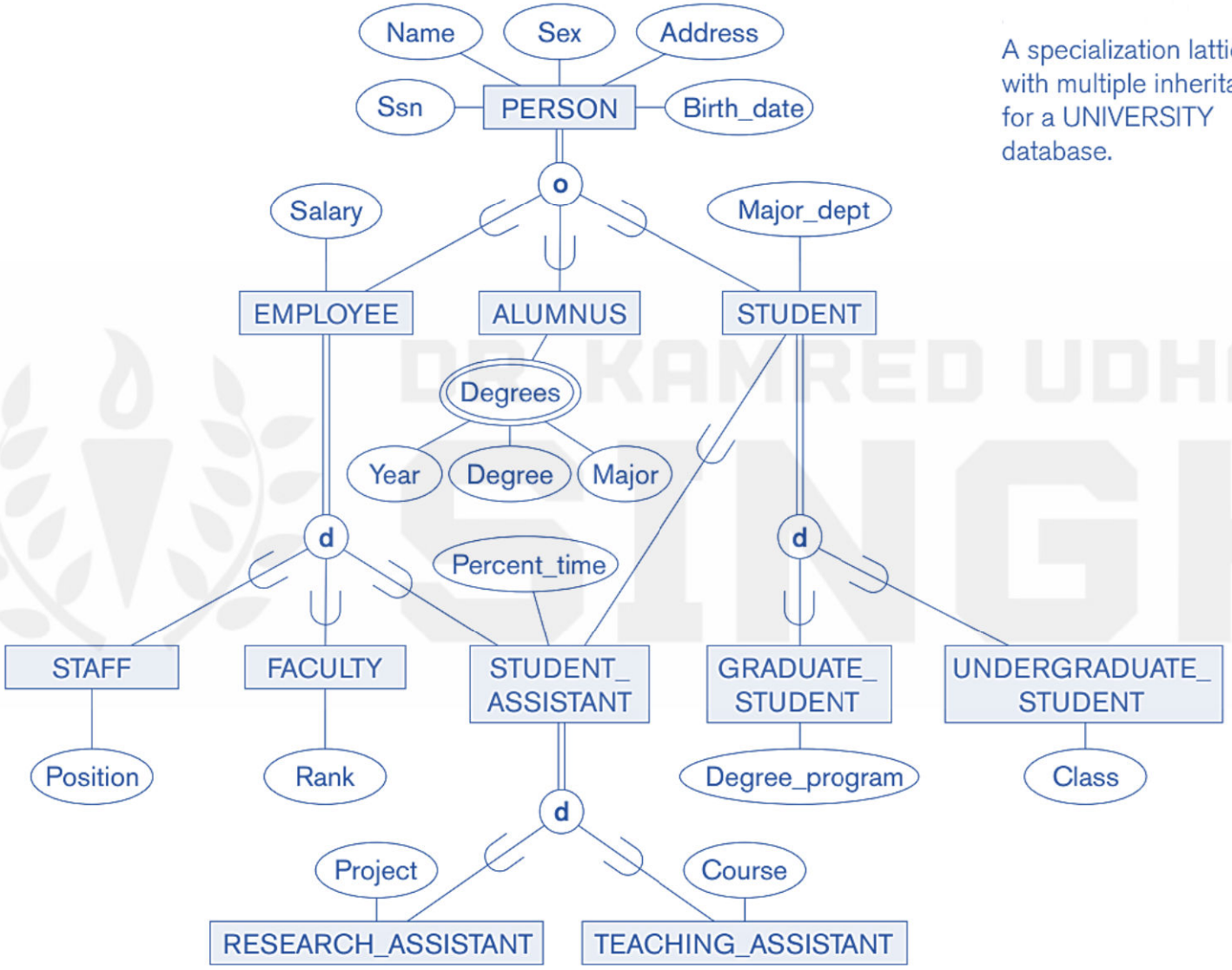
We can think of a *reverse process* of abstraction in which we suppress the differences among several entity types, identify their common features, and **generalize** them into a single **superclass** of which the original entity types are special **subclasses**.

- ❑ Involves merging two lower-level entities to create a higher-level entity.
- ❑ A bottom-up approach that builds complexity from simpler components.
- ❑ Highlights similarities among lower-level entity sets while hiding differences.
- ❑ Leads to a simplified, structured data representation, aiding in database design and querying processes.



Generalization. (a) Two entity types, CAR and TRUCK.
(b) Generalizing CAR and TRUCK into the superclass VEHICLE.

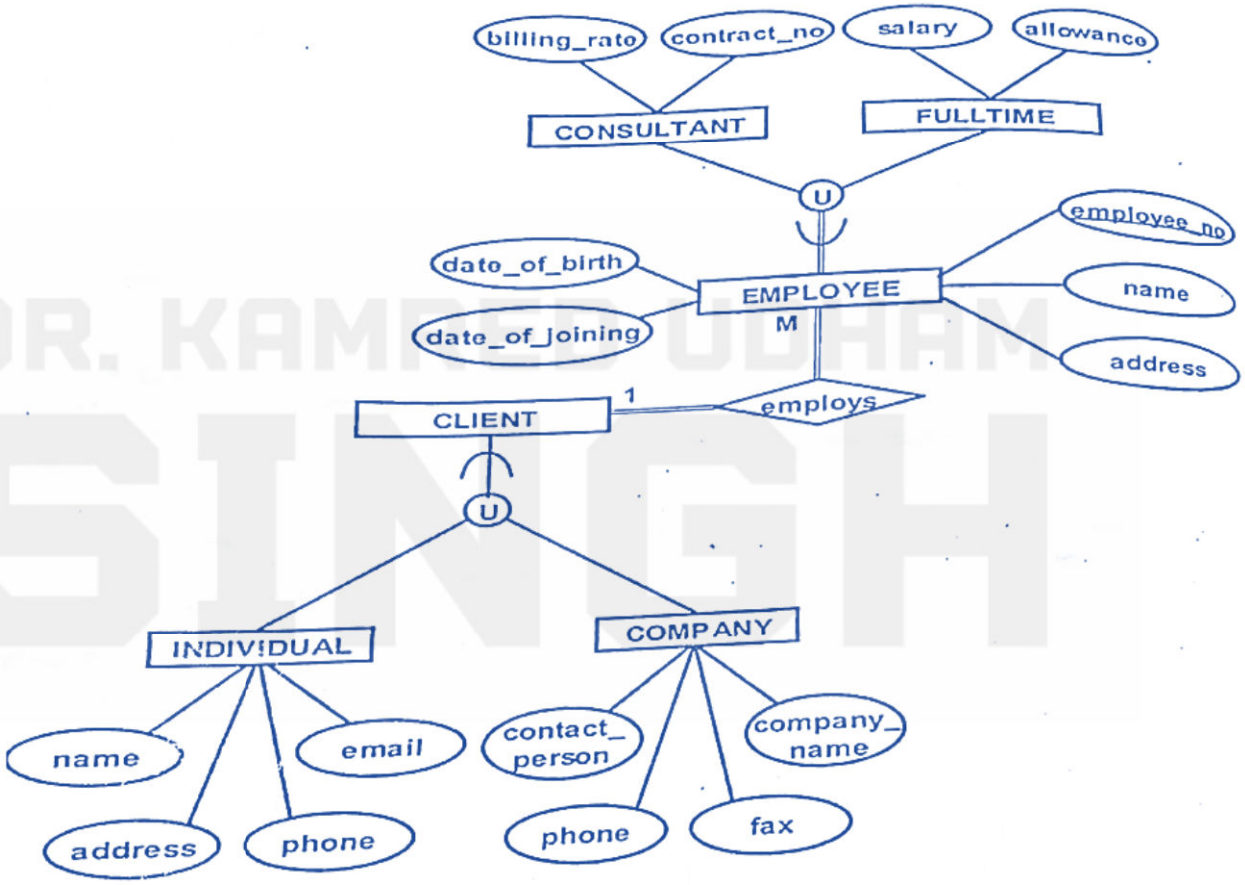


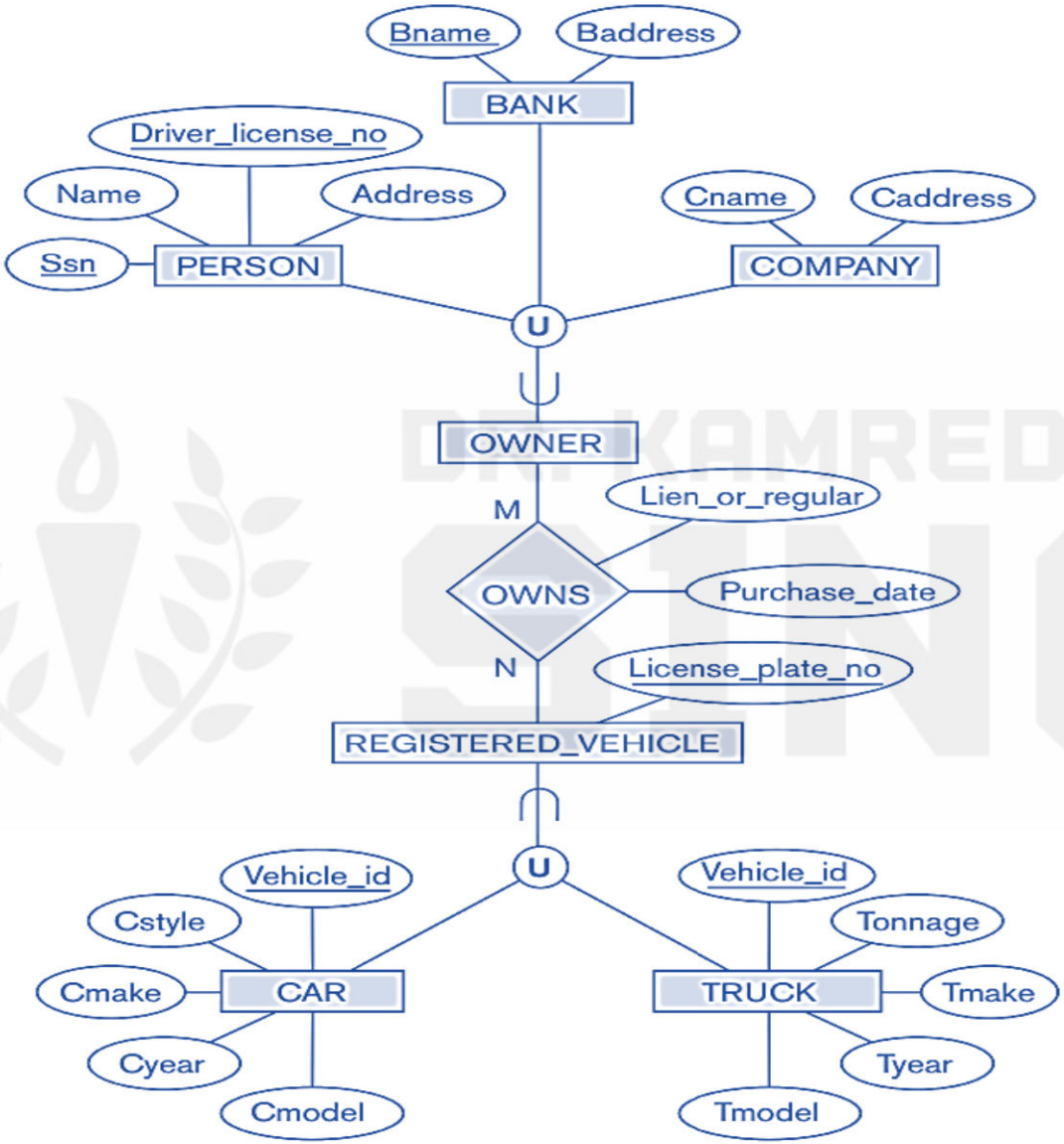


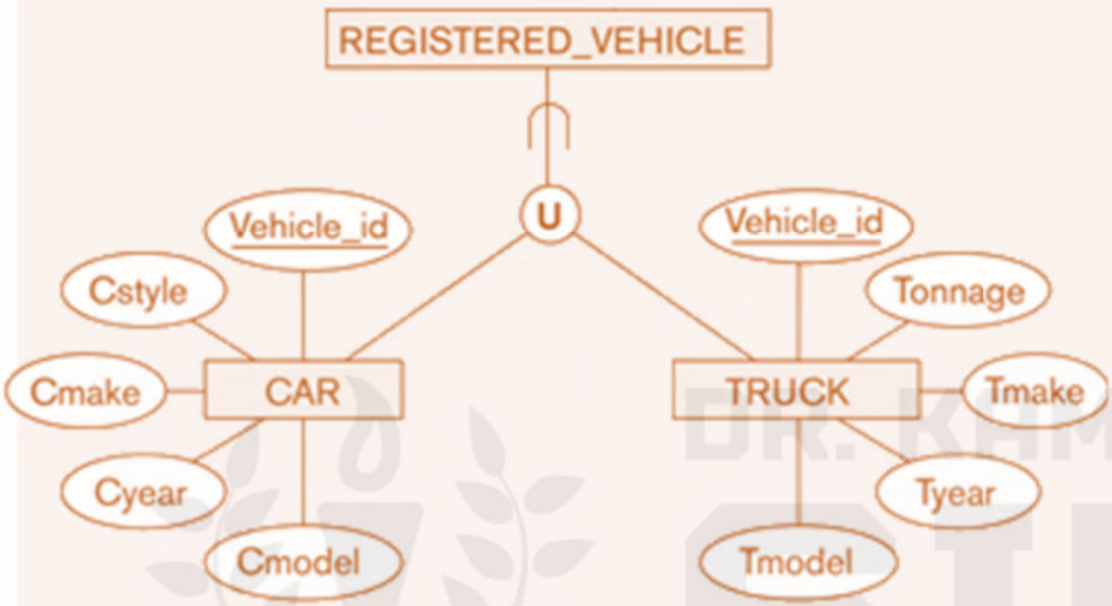
A specialization lattice with multiple inheritance for a UNIVERSITY database.

UNION Types / Categorization

It is sometimes necessary to represent a collection of entities from different entity types. In this case, a subclass will represent a collection of entities that is a subset of the UNION of entities from distinct entity types; we call such a *subclass* a **union type** or a **category**. In this example, client and employee are two categories.

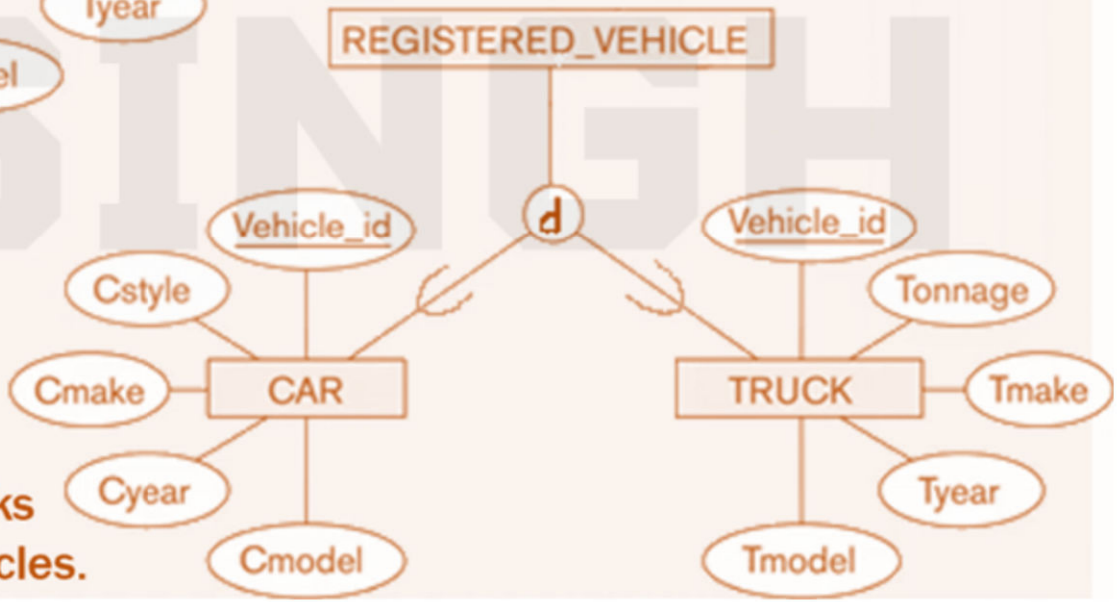



Bname
Baddress
BANK
Driver_license_no
Name
Address
Ssn
PERSON
Cname
Caddress
COMPANY
U
OWNER
M
Lien_or_regular
OWNS
Purchase_date
N
License_plate_no
REGISTERED_VEHICLE
U
Vehicle_id
Cstyle
Cmake
CAR
Cyear
Cmodel
Vehicle_id
Tonnage
TRUCK
Tmake
Tyear
Tmodel

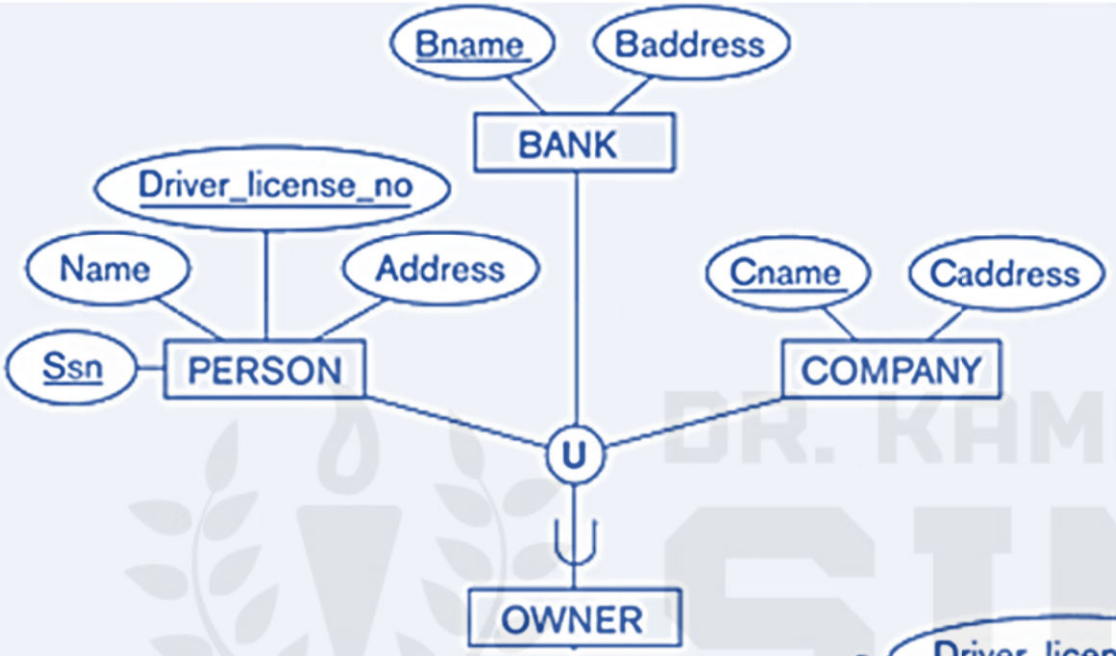


- Subtle difference between generalization and specialization

A registered vehicle is either a car or a truck.

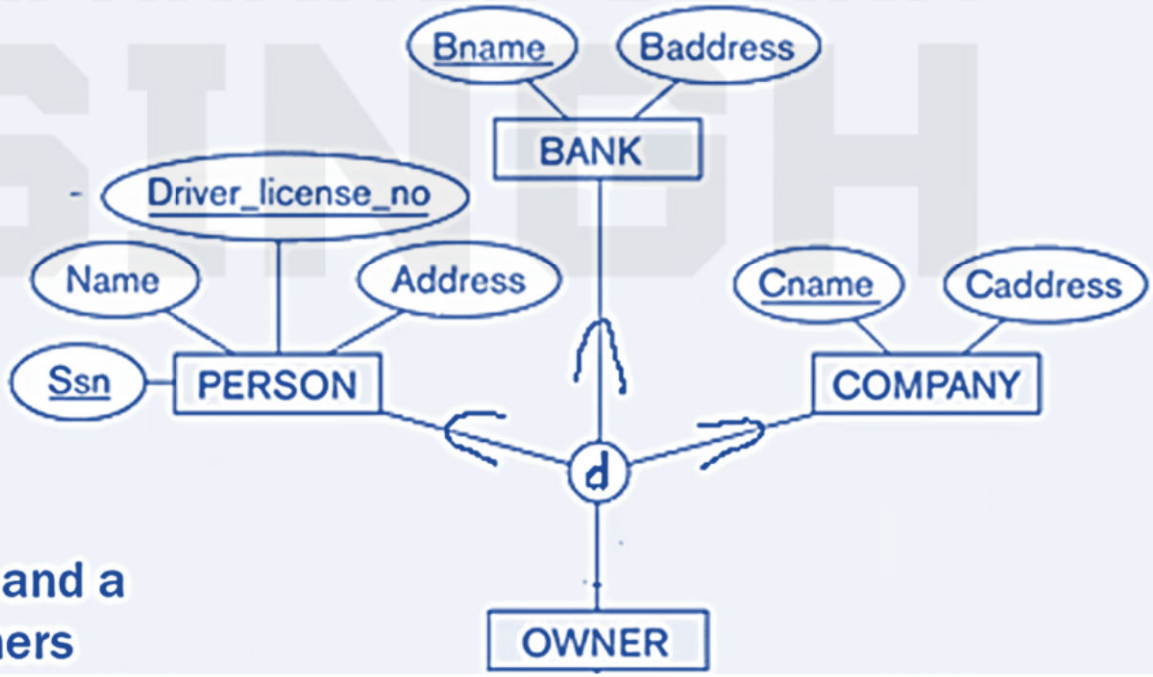


Both cars and trucks are registered vehicles.

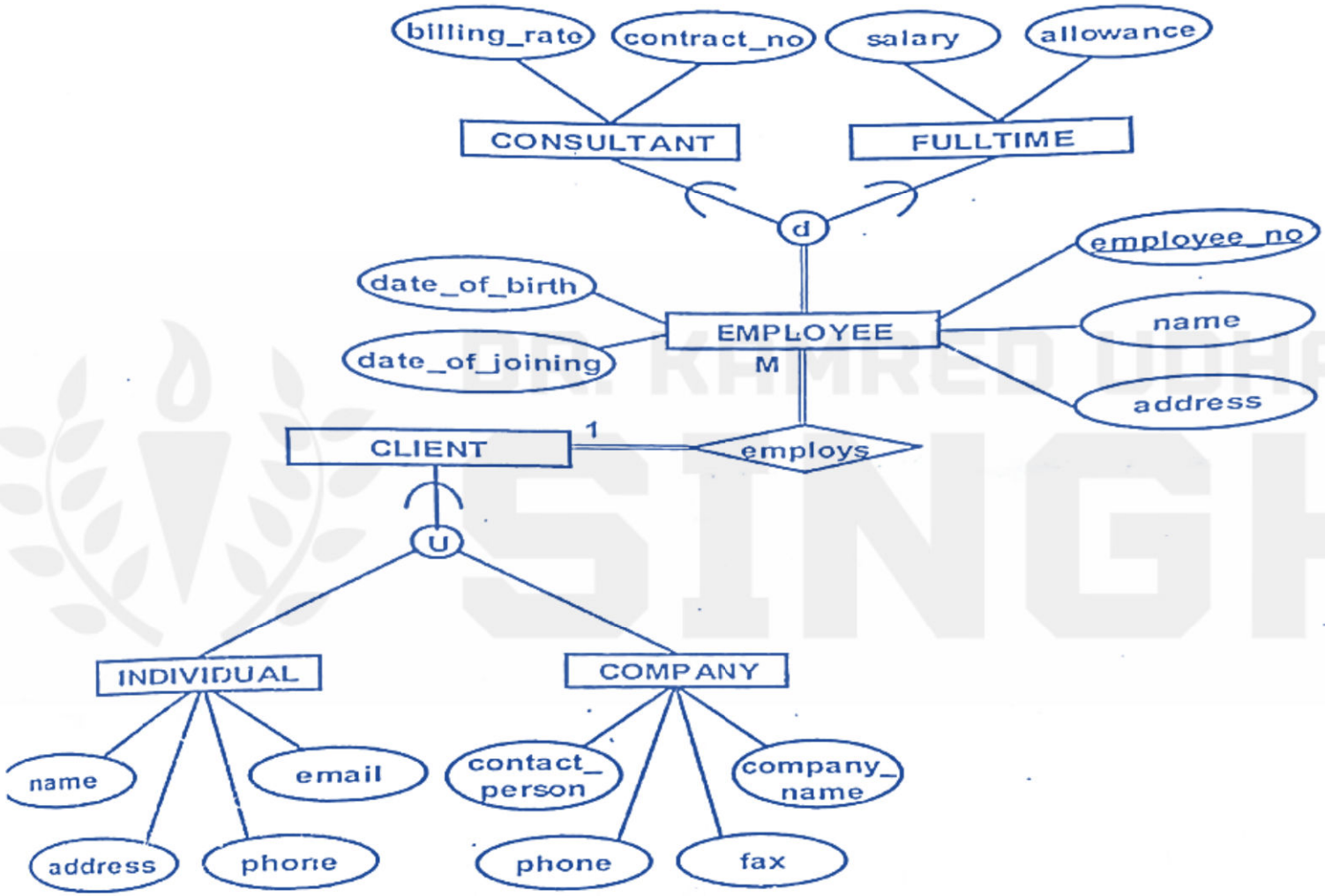


An owner is either a bank, a person, or a company.

- Subtle difference between generalization and specialization



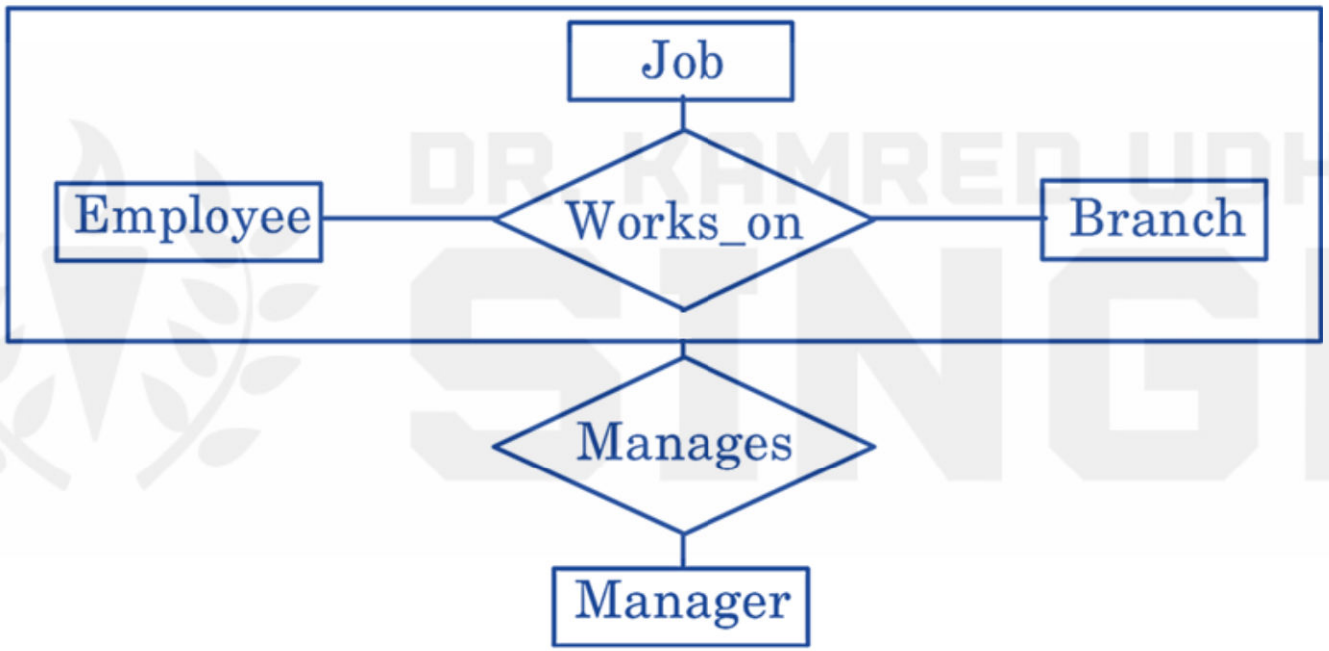
A bank, a person and a company are owners



Categorization and Generalization

Aggregation

A concept wherein relationships are abstracted to form higher-level entities, enabling a more organized representation of complex relationships.



DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH

DR. KAMRED UDHAM

SINGH