POD (Principles of Database)

2 Marks Questions

1 Define data independence. List its types.

Data independence: The ability to modify or change schema

definition of one level without affecting schema definition in the next Higher level.

Types of data independence: Logical data independence and physical

data independence.

2 Define

i) Tuple

ii) Relation

Tuple: A row or a record is called as tuple in relational database

management system.

2) Relation: A relation is nothing but a table which can store data in

rows and columns form I relational database management system

3 Define primary key. Give its example.

A primary key is a column or a group of columns from a table that

can uniquely identify the rows of data in that table. It accepts unique

and not null values.

Example : Empno is a primary key in table employee, which

identifies each row of employee table.

4 List any two advantages of relational database.

1) Controlled redundancy

2) Sharing of data

3) Improved data security

4) Consistency

5) Higher integrity

5 List any two types of database.

(Any 2 names from following can be considered)

1. Centralized database.

2. Distributed database.

3. Personal database.

4. End-user database.

5. Commercial database.

6. No SQL database.

7. Operational database.

8. Relational database.

9. Cloud database.

10. Object-oriented database.

11. Hierarchical database.

12. Network database.

13. Graph database.

14. Parallel database

6 Explain syntax of Alter table command.

i)To add a new attribute:-

Syntax:

Alter table <Table_name>

Add(<newcolumnname> <datatype(size)>);

Example:

Alter table student

Add(age numeric(2));

This alter command adds a new column to the existing table.

ii) Drop an attribute:-

Syntax : Alter Table table_name

Drop(<columnname>);

Example:

Alter table student

Drop(age);

This alter command removes the existing column from the table.

iii) Adding an constraint:-

Syntax:

Alter table <table_name>

Add constraint <constraint_name>(<columnname>);

Example:

Alter table student

Add constraint unique(Name);

This alter command adds a new constraint to a particular column in the existing table.

iv)Modifying :-

Syntax: Alter table <table_name>

modify (<columnname> <newdatatype(size)>);

Example:

Alter table student

Modify(Rollno numeric(20));

This alter command modify the existing column in the table.

v) Rename:-

Syntax

Alter table table_name

Rename column<old_column_nname> to <new_column_name>;

Example:

Alter table student

Rename column Rollno to Stid;

This alter command renames the existing column in the table.

7 List any two inference rules for functional dependency.

(Any two rules can be considered)

1. Reflexivity: X->X // An attribute(s) determines itself.

2. Augmentation: if X->Y then XZ->YZ.

3. Transitivity: if X->Y & Y->Z then X->Z.

4. Additivity or Union : if X->Y & X->Z then X->YZ.

5. Projectivity or Decomposition: If X->YZ then X-> Y & X->Z.

6. Pseudo-Transitivity: If X->Y, YZ->W then XZ->W

8 Define (i) Data Abstraction, (ii) Data Redundancy.

(i) Data Abstraction:

Data Abstraction is hiding the details of data organization and storage and highlighting the essential features for an improved understanding of data.

(ii) Data Redundancy:

The Data redundancy is the storing of same data multiple times.

This leads to duplication of effort. Second, storage space is wasted.

9 Define the term tuple and domain.

tuple: A row is called a Tuple.

domain: A domain is a set of all possible (or permissible) values in an attribute.

OR

A Domain is defined as a kind of data represented by an attribute.

10 Define primary key and candidate key

Primary key: The PRIMARY KEY uniquely identifies each record in a database table. Primary keys must contain unique values. A primary key column cannot contain NULL values. Each table should have a primary key, and each table can have only one primary key.

Candidate key: A minimal super key is called a candidate key. An entity set may have more than one candidate key.

A candidate key is a column, or set of columns, in a table that can uniquely identify any database record without referring to any other data. Each table may have one or more candidate keys, but one candidate key is special, and it is called the primary key

11 Define constraints, list types.

Constraints are used to limit the type of data that can go into a table. Constraints are used to ensure accuracy and consistency of data in a relational database.

Types of Constraints :

1.NOT NULL Constraint

2.DEFAULT Constraint

3.UNIQUE Constraint

4.CHECK Constraint

5.Primary Key Constraint

6. Foreign Key Constraint

12 Define Data and instance.

Data: Data can be defined as facts or information that can be recorded and have an implicit meaning.

Instance: The collection of information stored in the database at a particular moment is called an instance of the database.

13 Write Syntax for create table.

Syntax of Create table:

CREATE TABLE table_name(

column1 datatype (size),

column2 datatype(size),

column3 datatype(size),

.....

columnNdatatype(size)

);

14 Define Normalization, list its types.

Normalization is a systematic approach of decomposing tables to eliminate data redundancy(repetition) and undesirable characteristics like Insertion, Update and Deletion Anomalies. It is a multi-step process that puts data into tabular form, removing duplicated data from the relation tables.

Types of Normalization are:

1NF,2NF,3NF,4NF,5NF

15 List any four DBMS softwares.

(Note: Any four valid DBMS software can be considered)

List of DBMS software are the followings:

i. Oracle RDBMS

ii. IBM DB2

iii, Microsoft SQL Server

iv. MySQL

v. MS Access

vi. SQLite

vii. PostgreSQL

viii. MongoDB

ix. SQL Developer

x. SAP Sybase SE

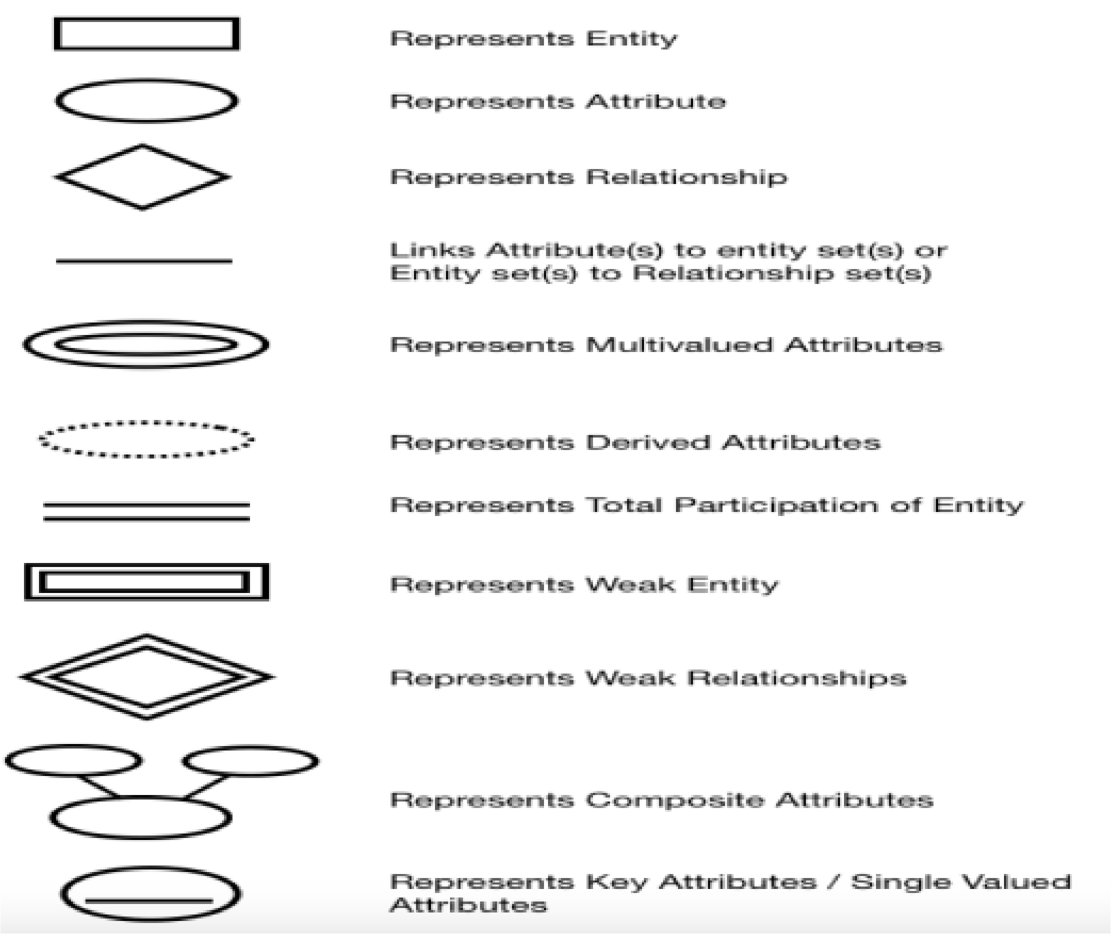
16 Define Domain and Attribute.

A Domain is defined as the set of all unique values permitted for an attribute.

Attributes are the descriptive properties owned by each entity of an entity set.

17 List and draw any four symbols used in ER-Model.

Different symbols used in ER-Model are the following


Represents Entity
Represents Attribute
Represents Relationship
Links Attribute(s) to entity set(s) or Entity set(s) to Relationship set(s)
Represents Multivalued Attributes
Represents Derived Attributes
Represents Total Participation of Entity
Represents Weak Entity
Represents Weak Relationships
Represents Composite Attributes
Represents Key Attributes / Single Valued Attributes

18 Define Constraint.

Constraints are the rules enforced on the data columns of a table. These are used to limit the type of data that can go into a table. This ensures the accuracy and reliability of the data in the database. Constraints could be either on a column level or a table level. The column level constraints are applied only to one column, whereas the table level constraints are applied to the whole table.

19 Define Database. List any two advantages of database system.

A database is an organized collection of data so that it can be easily accessed, managed and updated.

Advantages of database system are the following:

1. Controlling Redundancy of data in a centralized system of DBMS

2. Integrity of data can be enforced in case of database system by enforcing constraints

3. Inconsistency of data can be avoided by reducing duplicacy or redundancy

4. Data can be shared by multiple applications in centralized DBMS

5. Standards can be enforced in DBMS is a central system by enforcing standards easily at Company level, Department level, National level or International level.

6. Restricting unauthorized access among multiple users when sharing of data takes place in a database.

7. Providing Backup and Recovery facilities is provide by DBMS for recovering from hardware or software failures.

20 Define database model.

Definition of database model:

A database model is a type of data model that determines the logical structure of a database. It also fundamentally determines in which manner data can be stored, organized and manipulated.

21 List advantages of Normalization.

List of Advantages of Normalization are the following:

1. More efficient data structure.

2. Avoid redundant fields or columns.

3. More flexible data structure i.e. we should be able to add new rows and data values easily

4. Better understanding of data.

5. Ensures that distinct tables exist when necessary.

6. Easier to maintain data structure i.e. it is easy to perform operations and complex queries can be easily handled.

7. Minimizes data duplication.

8. Close modeling of real world entities, processes and their relationships.

22 Define the term Database Schema

The overall design of the database is called the database schema.

A schema diagram displays only names of record types (entities) and names of data items (attributes) and does not show the relationships among the various files.

23 Enlist components of database.

A database system involves four major components.

1 Data 2 Hardware 3 Software 4 Users

24 State any four characteristics of Database.

The Characteristics of Database are:

1. Persistent Data

2. Meta Data and Self-describing nature of a DB

3. Insulation between programs and data (Data Independence)

4. Support of multiple views of the data

5. Sharing of data and multiuser transaction processing

6. Access flexibility and Security.

7. Controlled Redundancy

25 List 4 types of Database languages.

Four types of database languages are:

1. DDL (Data Definition Language)

2. DML (Data Manipulation Language)

3. DDL (Data Control Language)

4. TCL (Transaction control language)

26 Define the term Data Model.

Underlying structure of the database is called as data model. It is a collection of conceptual tools for describing data, data relationships, data semantics and consistency constraints. Data models define how data is connected to each other and how they are processed and stored inside the system

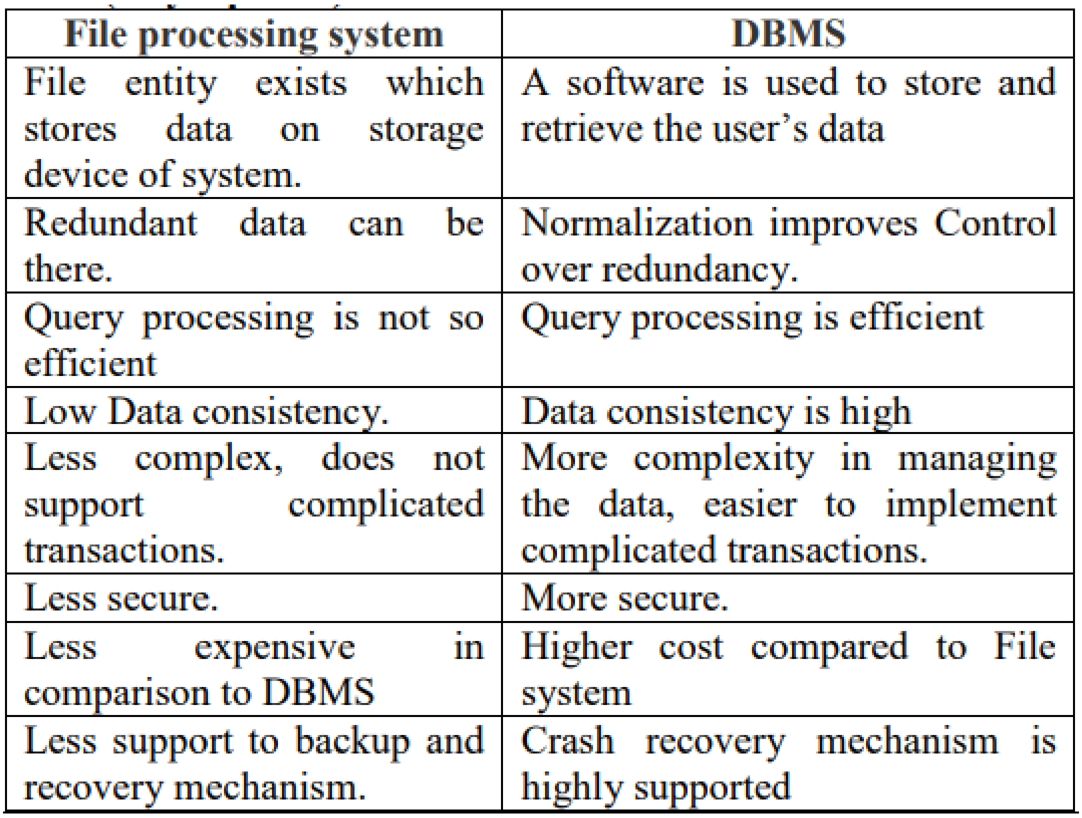
27 Define the term Foreign Key.

A FOREIGN KEY is a key used to link two tables together. A FOREIGN KEY is a field (or collection of fields) in one table that refers to the PRIMARY KEY in another table. It acts as a crossreference between tables because it references the primary key of another table, thereby establishing a link between them

4 Marks Questions

1 Compare file processing system and database management

system (Any 4 points)



File processing system	DBMS
File entity exists which stores data on storage device of system.	A software is used to store and retrieve the user's data
Redundant data can be there.	Normalization improves Control over redundancy.
Query processing is not so efficient	Query processing is efficient
Low Data consistency.	Data consistency is high
Less complex, does not support complicated transactions.	More complexity in managing the data, easier to implement complicated transactions.
Less secure.	More secure.
Less expensive in comparison to DBMS	Higher cost compared to File system
Less support to backup and recovery mechanism.	Crash recovery mechanism is highly supported

2 Describe types of attributes with suitable example.

1. Simple Attributes

Simple attributes are those that cannot be further divided into subattributes.

For example, A student's roll number of a student or the employee identification number.

2. Composite Attributes

Composite attributes are made up of two or more simple attributes.

For example, a person's address may be a composite attribute that is made up of the person's street address, city, state, and zip code.

3. Single Valued Attributes

Single-valued attributes can only have one value. Single-valued attributes are typically used to provide a unique identifier for an entity and are often used in databases.

For example, a person's Social Security Number is a single-valued attribute.

4. Multivalued Attributes

Multivalued attributes can have more than one value.

For example, a person may have multiple email addresses or phone numbers.

5.Key attributes

Key attributes are those attributes which can identify an entity uniquely in an entity set.

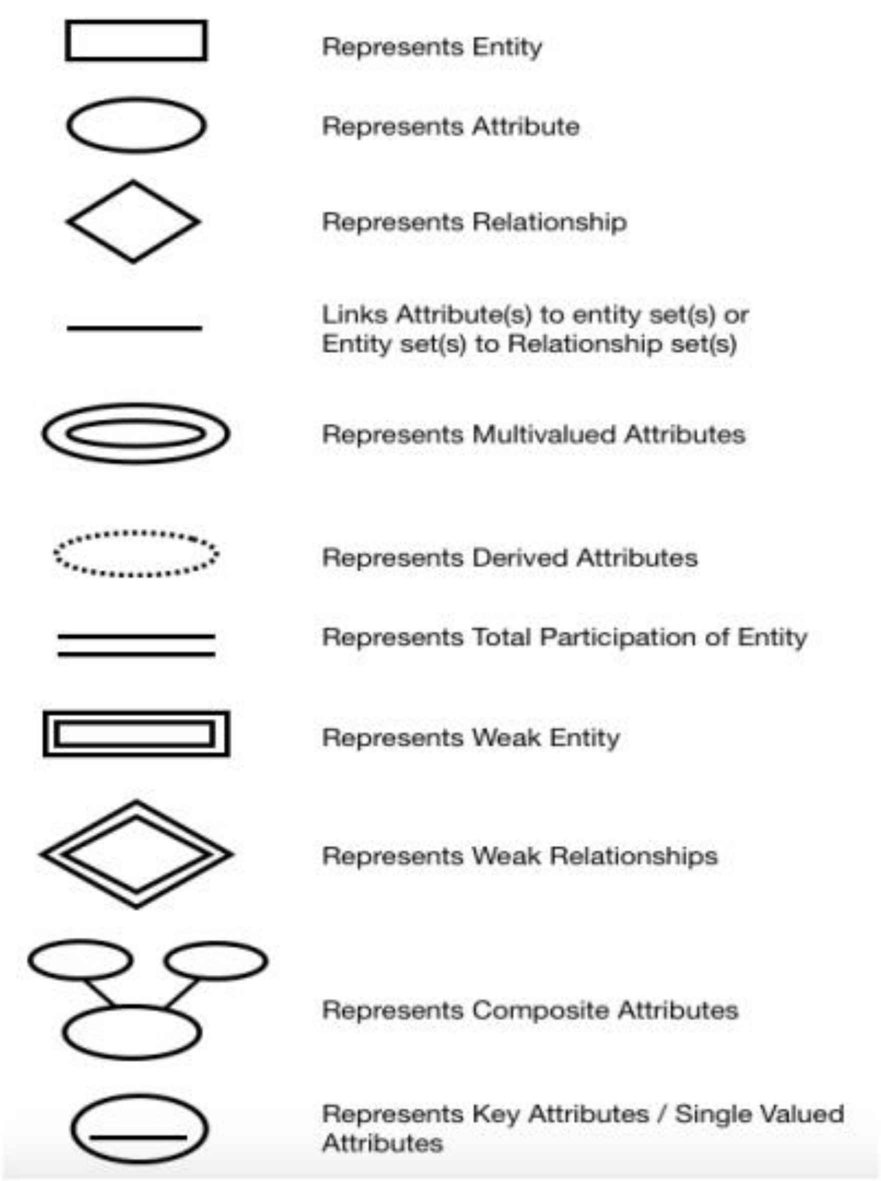
Example: Roll_no in a student table is the key attribute.

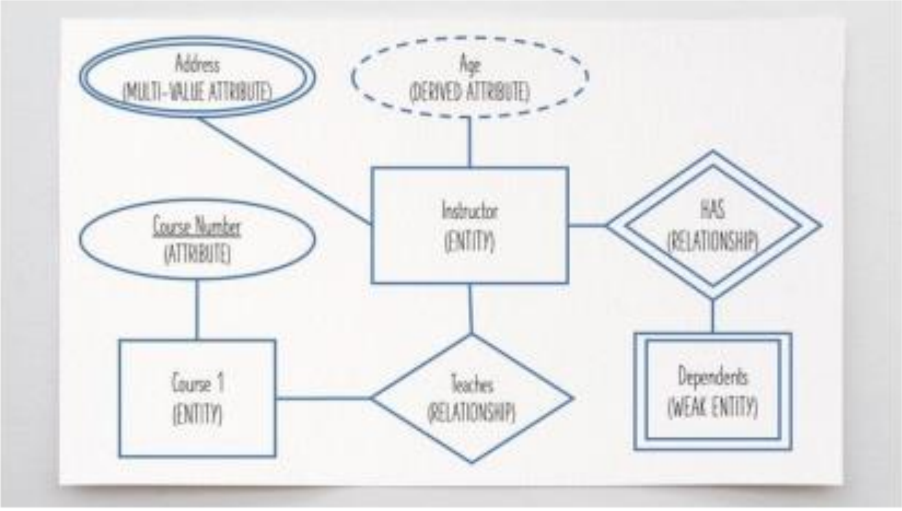
6. Derived Attributes

Derived attributes are based on other attributes and are not stored directly in the database.

For example: Consider a database of employees. Each employee has a date of birth, and we can calculate their age which can be called as derived attribute.

3 List and draw any 4 symbols used in E-R model. Give example of Each


Represents Entity
Represents Attribute
Represents Relationship
Links Attribute(s) to entity set(s) or Entity set(s) to Relationship set(s)
Represents Multivalued Attributes
Represents Derived Attributes
Represents Total Participation of Entity
Represents Weak Entity
Represents Weak Relationships
Represents Composite Attributes
Represents Key Attributes / Single Valued Attributes


Address
(MULTI-VALUE ATTRIBUTE)
Age
(DERIVED ATTRIBUTE)
Course Number
(ATTRIBUTE)
Instructor
(ENTITY)
HAS
(RELATIONSHIP)
Course 1
(ENTITY)
Teaches
(RELATIONSHIP)
Dependents
(WEAK ENTITY)

4 Explain components of database

Components of a Database:

The five major components of a database are:

1. Hardware

Hardware refers to the physical, electronic devices such as computers and hard disks that offer the interface between computers and realworld systems.

2. Software

Software is a set of programs used to manage and control the database and includes the database software, operating system, network software used to share the data with other users, and the applications used to access the data.

3. Data

Data are raw facts and information that need to be organized and processed to make it more meaningful. Database dictionaries are used to centralize, document, control, and coordinate the use of data within an organization. A database is a repository of information about a database (also called metadata).

4. Procedures

Procedures refer to the instructions used in a database management system and encompass everything from instructions to setup and install, login and logout, manage the day-to-day operations, take backups of data, and generate reports.

5. Database Access Language

Database Access Language is a language used to write commands to access, update, and delete data stored in a database. Users can write commands using Database Access Language before submitting them to the database for execution. Through utilizing the language, users can create new databases, tables, insert data, and delete data.

5 Explain domain integrity constraint with example.

Domain integrity constraint contains a certain set of rules or conditions to restrict the kind of attributes or values a column can hold in the database table.

Domain constraints are used to maintain value according to user specification.

There are two types of Domain constraint

 Not Null Constraint

 Check Constraint

Not Null Constraint:

It is applied on a column to avoid null values. When a Not Null Constraint is enforced on column or set of columns it will not allow null values.

For Example:

Apply not null constraint on Ename column.

```
create table employee
(
Empid number (3),
Ename varchar (10)constraint nn not null,
Salary number (7,2),
Phone number (10)
);
```

Check Constraint:

It defines a condition that each row must satisfy. A single column can have multiple check condition.

For Example:

Apply check constraint on Salary column

```
create table employee
(
Empid number(3) ,
Ename varchar(10),
Salary number(7,2)constraint ck check(salary >=5000),
Phone number(10)
);
```

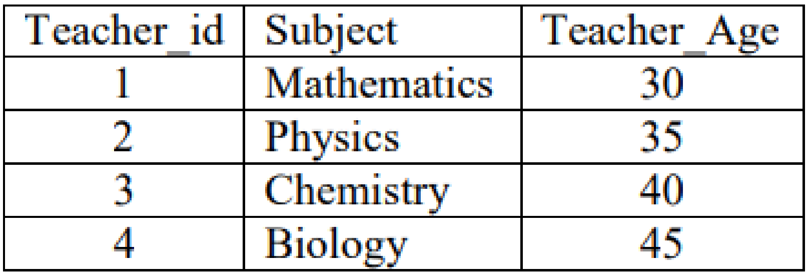
(* key word constraint nn/constraint ck are optional. Without them also query is correct.)

6 State and explain 1 NF and 2 NF with example.

First Normal Form (1NF)

- The table is in 1NF which contains all atomic values. There should be no repeating in any one of the attributes.
- All the attributes are functionally dependent on the primary key.
- 1NF is achieved when all repeating groups are removed and a separate table is created with atomic values.

For Example: Teacher_details (Teacher_id, Subject,Teacher_Age)



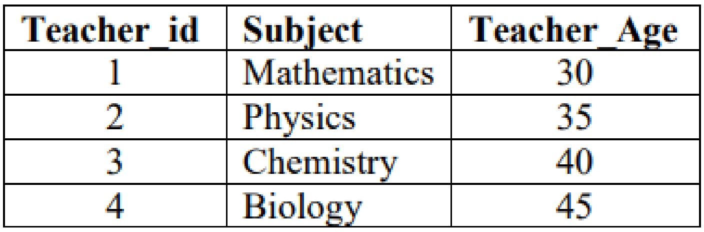
Teacher_id	Subject	Teacher_Age
1	Mathematics	30
2	Physics	35
3	Chemistry	40
4	Biology	45

The above table is in 1NF because every attribute has single (atomic) value.

Second Normal Form (2NF)

- Fully functional Dependency: If a & b are the attributes of the relation, b is fully functionally dependent on a, if b is functionally dependent on a and a proper subset of a.
- So 2NF removes partial dependencies i.e. functionally dependent attributes are removed from the relation by placing them in a new relation along with their copy of determinants.
- 2NF is achieved when relation is in 1NF and each record is fully dependent on primary key of the relation for identification.

For Example If we consider following Teacher_details table.



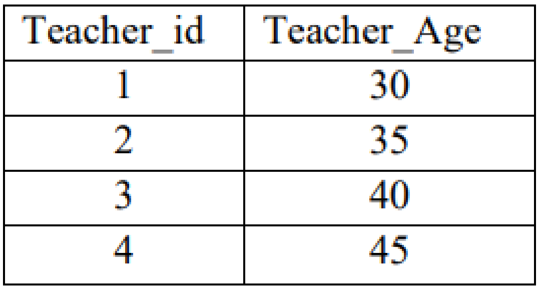
Teacher_id	Subject	Teacher_Age
1	Mathematics	30
2	Physics	35
3	Chemistry	40
4	Biology	45

Functional dependencies are as follows:

Teacher_id->Subject

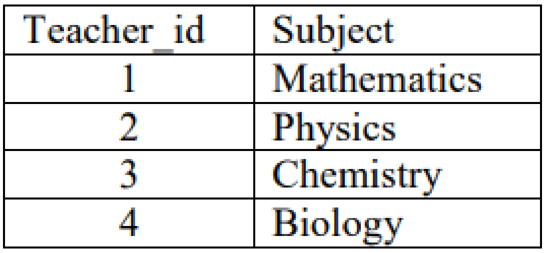
Teacher_id->Teacher_Age

To convert the given table into 2NF, we decompose it into two tables considering above functional dependencies:



Teacher_id	Teacher_Age
1	30
2	35
3	40
4	45

Table 1: Teacher Table



Teacher_id	Subject
1	Mathematics
2	Physics
3	Chemistry
4	Biology

Table 2: Teacher_allocation Table

7 Describe enhanced E-R model with suitable example.

 Enhanced ER is a high-level data model that incorporates the extensions to the original ER model. It is created to design more accurate database schemas.

 EER reflects data properties and constraints more precisely.

 It also includes more complex requirements than traditional application.

 Enhanced ER model includes all concepts of ER model. Additionally, it includes concept of Super Class, Subclass, Generalization, Specialization, Union and Aggregation.

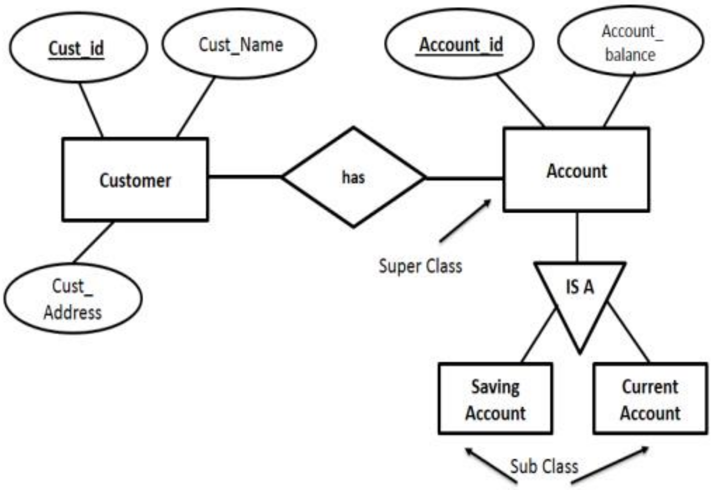
 Generalization is union of two or more entity set to produce higher level entity set. It is bottom up approach.

 Specialization is a process of deriving lower level entities from higher level entity. It is top down approach.

 In aggregation, relation between two entities is treated as a single entity.

 Higher level entities are called Super Class

 Lower level entities are called Sub class



8 Describe parallel database system. Give its two examples.

Parallel DBMS is a Database Management System that runs through multiple processors and disks. They combine two or more processors also disk storage that helps make operations and executions easier and faster.

Advantages

 Execution speed is fast.

 Taking backup is easy because all PC at one site only.

Disadvantages

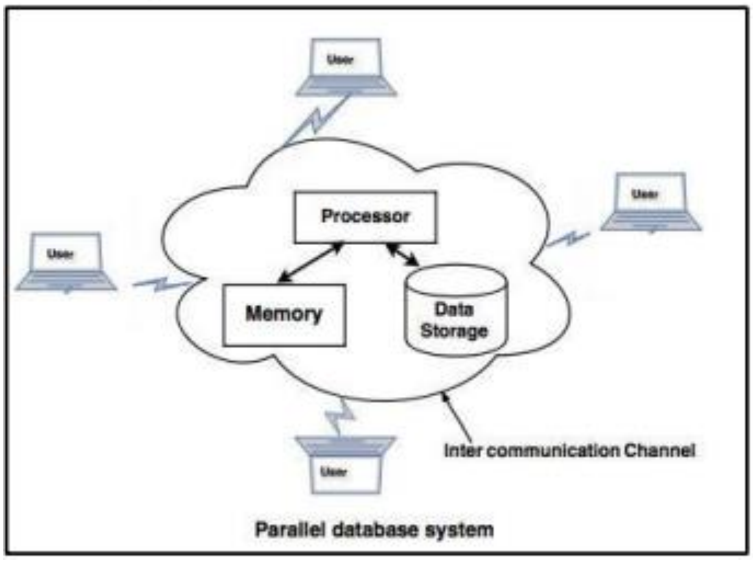
 It is difficult to expand. Not scalable after certain point.

 Startup cost is high

For Example:

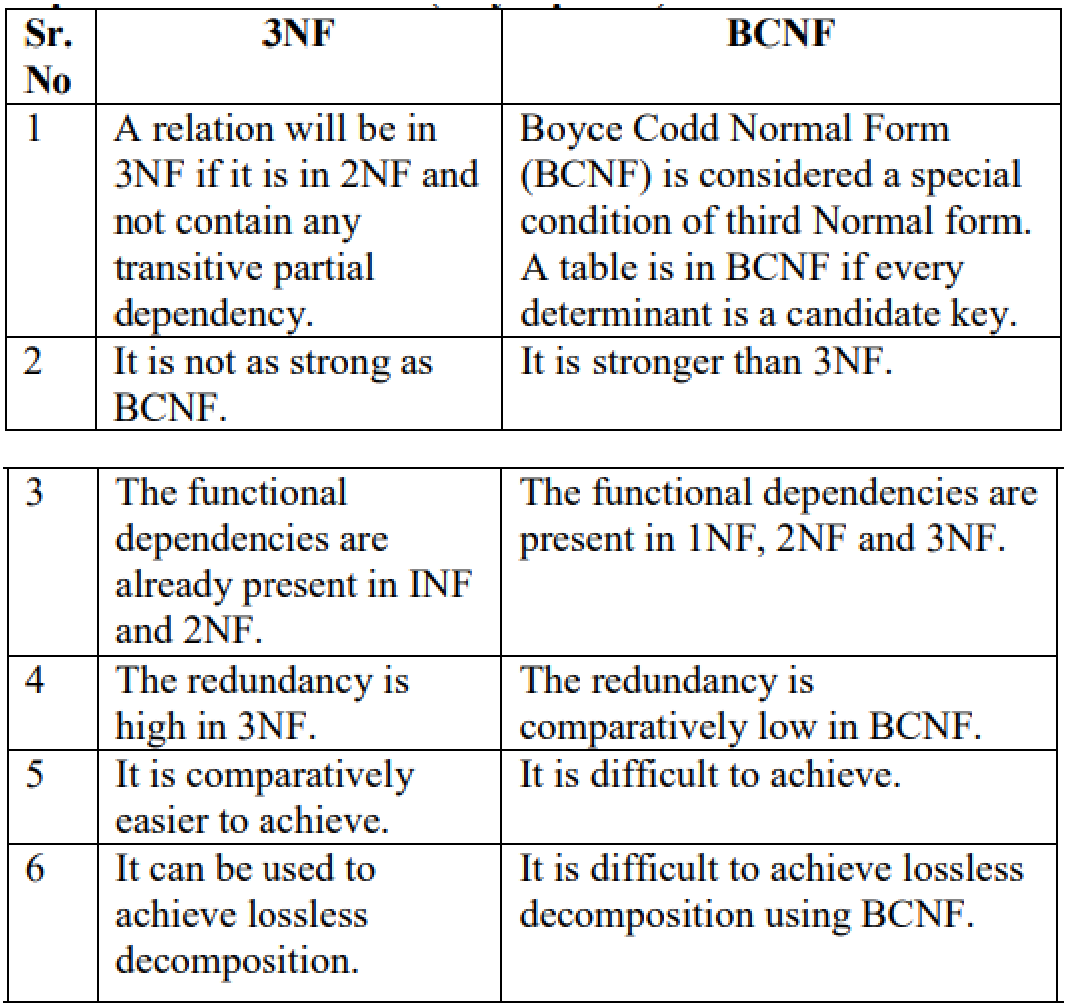
1) Parallel database systems are used in e-commerce

2) Parallel databases are used in data warehousing and data mining



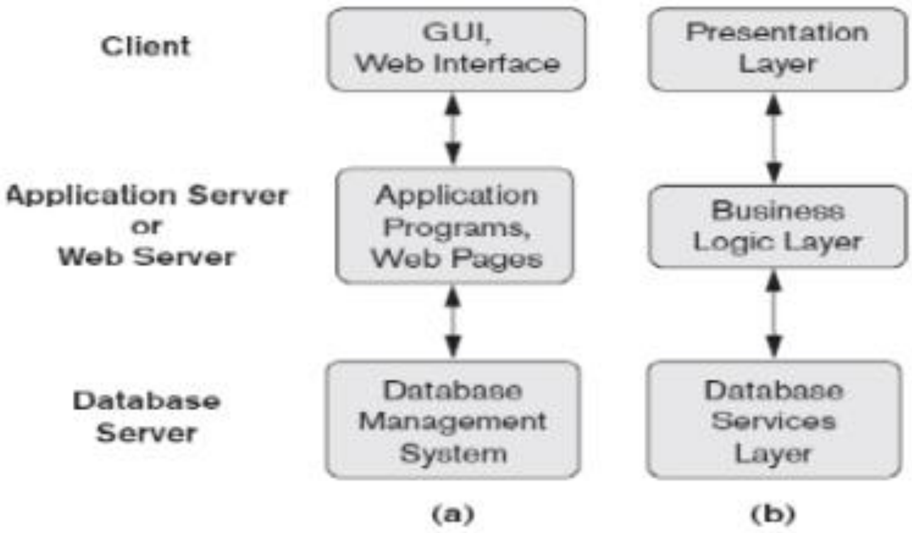
Parallel database system

9 Compare 3 NF and BCNF (Any 4 points)



Sr. No	3NF	BCNF
1	A relation will be in 3NF if it is in 2NF and not contain any transitive partial dependency.	Boyce Codd Normal Form (BCNF) is considered a special condition of third Normal form. A table is in BCNF if every determinant is a candidate key.
2	It is not as strong as BCNF.	It is stronger than 3NF.
3	The functional dependencies are already present in INF and 2NF.	The functional dependencies are present in 1NF, 2NF and 3NF.
4	The redundancy is high in 3NF.	The redundancy is comparatively low in BCNF.
5	It is comparatively easier to achieve.	It is difficult to achieve.
6	It can be used to achieve lossless decomposition.	It is difficult to achieve lossless decomposition using BCNF.

10 Describe 3 tier architecture with its advantages and disadvantages.



In 3 tier

architecture communication take place from client to application server and Application server to Database. Clients contain GUI interfaces and some additional application specific business rules. Application server is called “Middle Layer”. It processes application code.

Accepts requests from clients. Database server process database queries. It is used in W.W.W(World Wide Web)

Advantage:

 Improve data integrity.

 Improve security

Disadvantage:

 It is more complex than the 2-tier architecture system

 Cost is higher than 2- tier architecture system

11 Describe how to express M : N relation with suitable example.

Many to Many Relationship (M:N)

When many instances of an entity A are associated with many instances of entity B.

OR

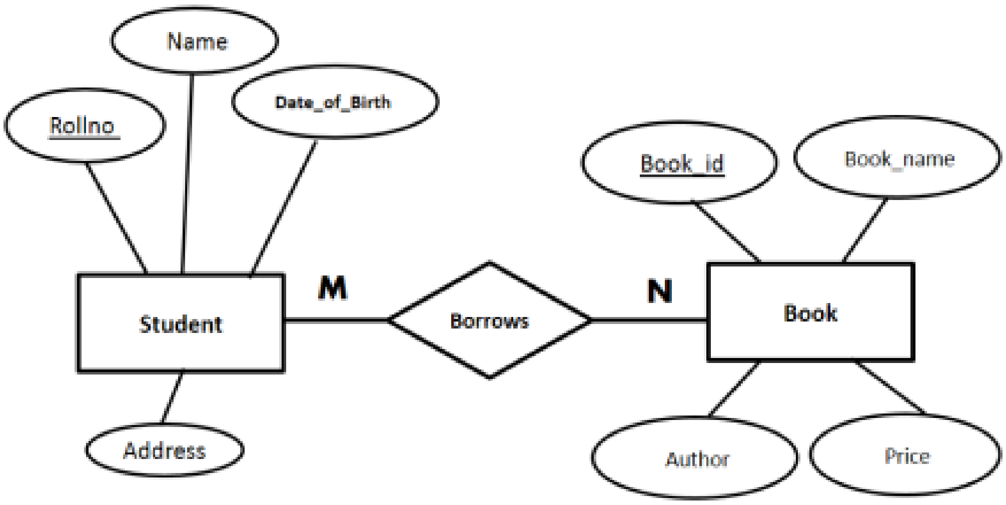
When many instances of an entity are associated with many instances of other entity.

Many to many cardinality is represented by (M: N)

For Example:

Many Students can borrow many Books.

The above relationship indicates M: N(many-to-many) relation type because many students can borrow many books from library.



12 List and explain any four Codd's rules of RDBMS

Rule 1: The information rule

All information in relational database is represented by values in a table.

Rule 2: Guaranteed Access Rule

Whole data should be available or accessible to the user without any ambiguity. The ambiguity can be avoided only through the perfect combination of the table name, primary key, and column name.

Rule 3: Systematic treatment of null values

The null values i.e. absence of the values in the table should be treated properly. RDBMS Distinguish between ZERO (0) and Null Values.

Rule 4: Active on-line catalog based on the relational model

There are certain system tables that stores the database definition should be present. The data accessing tools should be used to access the database structure information. Description of the table and Contents of the table can be queried by DML.

Rule 5: The comprehensive data sub language rule: RDBMS supports many languages but at least one of them should allow user to

Define table, view, Query and Constraints.

Rule 6: The view updating rule: All views those can be updated theoretically, must be updated by the system.

Rule 7: High-level insert, update, and delete: A database must support high-level insertion, update and deletion. This must not be limited to a single row, that is, it must also support union, inter section and minus operations to yield sets of data records

Rule 8: Physical data independence: Changes to the physical level(how the data is stored, whether in arrays or linked lists etc.) must not require a change to an application based on the structure.

Rule 9: Logical data independence: Changes to the logical level(tables, columns, rows, and so on) must not require a change to an application based on the structure.

Rule 10: Integrity independence: Integrity constraints must be specified separately from application programs and stored in the catalog. It must be possible to change such constraints as and when appropriate without unnecessarily affecting existing applications.

Rule 11: Distribution independence: The distribution of portions of the database to various locations should be invisible to users of the database. Existing applications should continue to operate successfully: when a distributed version of the DBMS is first introduced; and when existing distributed data are redistributed around the system.

Rule 12: The non-subversion rule: If the system provides a lowlevel(record-at-a-time) interface, then that interface cannot be used to subvert the system, for example, bypassing a relational security or integrity constraint.

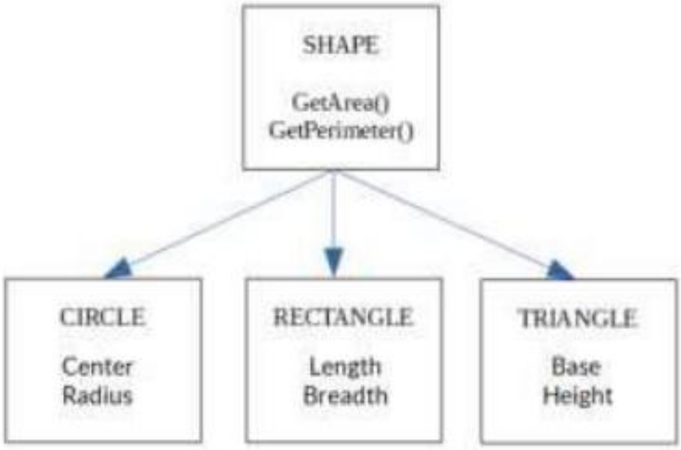
13 Describe object oriented database model with example. Give two advantages.

- Object oriented models were introduced to overcome the short comings of conventional models like Relational, Hierarchical and network model.
- An object oriented database is collection of objects whose behavior, state and relationship are defined in accordance with object oriented with object oriented concepts(objects, class, class hierarchy)
- Object Oriented Database Model is product of object oriented programming and Relational Model.
- This model supports, Object oriented concepts like data encapsulation, polymorphism, inheritance and Relational Database concepts like integrity, query, concurrency etc.

Example:

An Example of the Object Oriented data model is –

- Shape, Circle, Rectangle and Triangle are all objects in this model.
- Circle has the attributes Center and Radius.
- Rectangle has the attributes Length and Breath
- Triangle has the attributes Base and Height.
- The objects Circle, Rectangle and Triangle inherit from the object Shape



An Example of the Object Oriented data model is –

 Shape, Circle, Rectangle and Triangle are all objects in this model.

 Circle has the attributes Center and Radius.

 Rectangle has the attributes Length and Breath

 Triangle has the attributes Base and Height.

 The objects Circle, Rectangle and Triangle inherit from the object Shape.

Advantages:(consider any 2 relevant points)

 Object oriented data model allows the real world to be modeled closely. The object encapsulates both state and behavior. The object can also store the relations with other objects.

 Object Oriented features provide clear modular Structure which is good for defining abstract datatype where internal implementation is hidden. It allows new data types to be built from existing types.

 Redundancy can be reduced as common factors of several classes

can be grouped into a super class and can be shared by the subclasses.

 It can be used to store a variety of data

14 Describe client server system with example.

Client server system consists of two logical components. One is “Client” and the other one is “Server”. Clients are those who send the request to perform a specific task to the server. Servers normally receive the command sent by the clients, perform the task and sendthe appropriate result back to the client.

Example of client is PC where as the server is a large work station. The Client machine runs own copy of an operating system. It runs one or more applications through client’s CPU and memory. But server runs a database management system which manages the whole database

15 Explain Generalization with example.

Generalization uses bottom-up approach where two or more lower

level entities combine together to form a higher level new entity if

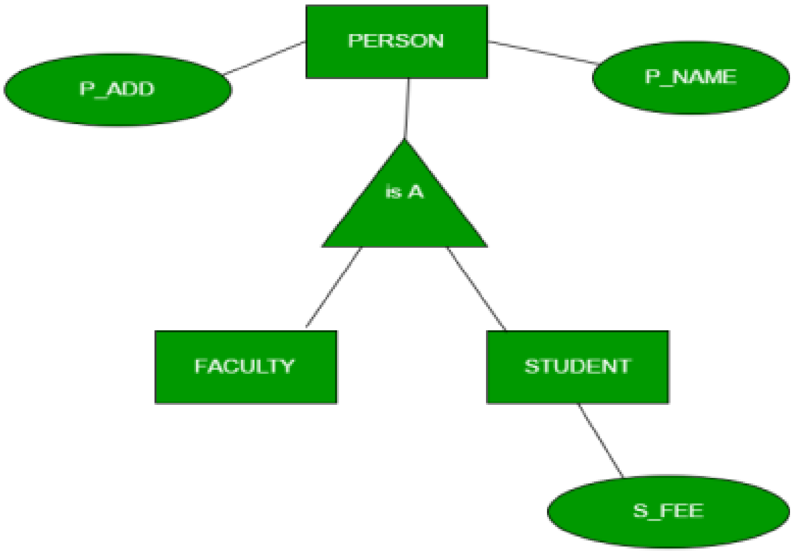
they have common attributes in common.

The new generalized entity can further combine together with lower

level entity to create a further higher level generalized entity.

For Example, STUDENT and FACULTY can be generalized to a

higher level entity called PERSON



15 Explain components of database in detail.

Components of a DBMS:

(i) Query processor: The query processor transforms user queries into a series of low level instructions. It is used to interpret the online user's query and convert it into an efficient series of operations in a form capable of being sent to the run time data manager for execution.

(ii) Run time database manager: Run time database manager is the central software component of the DBMS, which interfaces with user-submitted application programs and queries. It handles database access at run time. It converts operations in user's queries coming. It accepts queries and examines the external and conceptual schemas to determine what conceptual records are required to satisfy the user's request. It enforces constraints to maintain the consistency and integrity of the data, as well as its security. It also performs backing and recovery operations.

(iii) Authorization control: The authorization control module checks the authorization of users in terms of various privileges to users.

(iv) Command processor: The command processor processes the queries passed by authorization control module.

(v) Integrity checker: It .checks the integrity constraints so that only valid data can be entered into the database.

(vi) Query optimizer: The query optimizers determine an optimal strategy for the query execution.

(vii) Transaction manager: The transaction manager ensures that the transaction properties should be maintained by the system.

(viii) Scheduler: It provides an environment in which multiple users can work on same piece of data at the same time in other words it supports concurrency.

(ix) Data Manager: The data manager is responsible for the actual handling of data in the database. It provides recovery to the system which that system should be able to recover the data after some failure. It includes Recovery manager and Buffer manager. The buffer manager is responsible for the transfer of data between the main memory and secondary storage (such as disk or tape). It is also referred as the cache manger.

16 Explain Domain constraints with Syntax and example.

Domain constraints are used to maintain value according to user specification

Domain constraints are:

1. Not null-such constraints are applied to an attribute when we have to specify that the attribute cannot accept null value. Null is in the domain of all attributes unless not null is applied.

Example:

Consider the schema student.Student{rollno, name,sscper}. The name of the student should not be null. So we can apply the not null constraint to the name attribute.

General syntax (While creating table)

Create table tablename(attr1 datatype(size), attr2 datatype(size) not null,attr3 datatype(size));

After creating the table

Alter table tablename modify attr not null;

Example:

Create table student(rollno number(5),name varchar(30) not null,sscper number(3));

Alter table student modify name not null;

2. Check – allows enforcing domain integrity by limiting the values

accepted by an attribute.

Eg: consider an attribute age of the entity employee. If age should be limited to 60, check constraint can be used

General syntax:

Create table tablename(attr1 datatype(size),attr2 datatype(size) constraint nameofconstraint check(attr<value));

or

Alter table tablename add constraint nameofconstraint check(attr<value)

Eg:

Create table emp(empno number(4),name varchar(30),age number(3) constraint chk_emp check(age>60));

or

Alter table emp add constraint chk_emo check(age>60)

17 Describe benefits and drawbook of denormalization.

Benefits of denormalization:

- Reduce number of relations: It reduce the number of relations because it combines two relations into one new relation.
- Reduce number of foreign keys: It reduce number of foreign keys because number of relations is reduced.
- Minimizes need for joins: It minimizes need for joins because it combines many relations into one.
- Increase Performance: It increase performance of database by

adding redundant data or by grouping data.

Drawbacks of demoralization:

 Slow Data Updates: It may speed up the retrieval but can slow down database updates

 Increase size of relations: It can increase size of the relations due to combining multiple relations into one single relation.

 Complex implementation: It may simplify implementation in some cases but may make it more complex in other.

 Application Specific: It is always application-specific and needs to be re-evaluated if the application changes.

18 Explain different types of attribute with example and their symbols used in ER diagram.

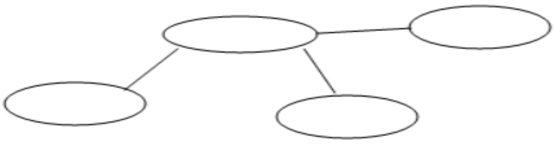
Different types of attributes are:

1. Simple attribute: A simple attributes are those which cannot be subdivided.

Eg:Rollno– symbol

2. Composite attribute: a composite attribute is that which can be subdivided

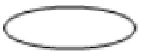
Eg: name – can be divided into first_name, middle_name and last_name symbol



3. Single valued attribute- an attribute which can have only one value for an entity.

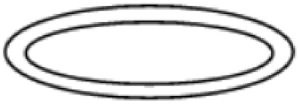
Eg:ssc_per

Symbol :



4. Multivalued attribute - an attribute that can take more than one value for an entity.

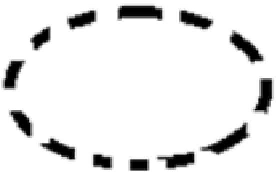
Eg:phoneno symbol :



5. Derived attribute - an attribute for which the value can be calculated or determined from another attribute

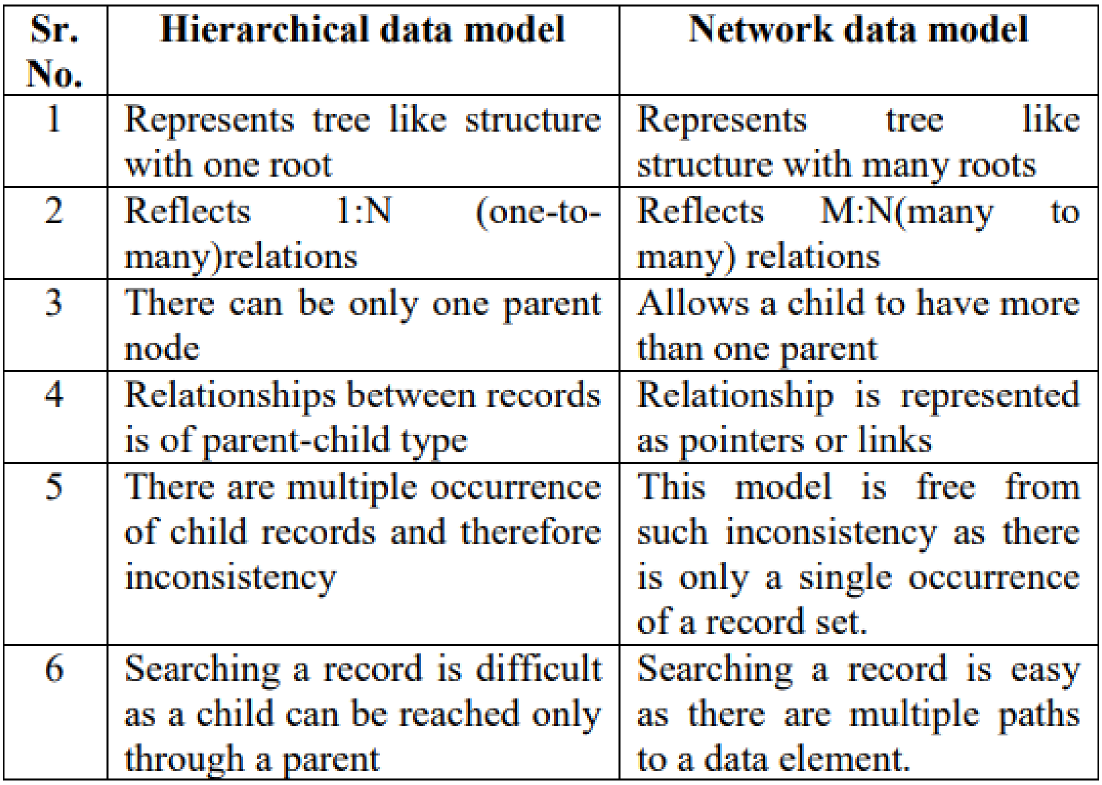
Eg: age from dateofbirth

Symbol



19 Differentiate between Hierarchical Database model and network

database model.



Sr. No.	Hierarchical data model	Network data model
1	Represents tree like structure with one root	Represents tree like structure with many roots
2	Reflects 1:N (one-to-many)relations	Reflects M:N(many to many) relations
3	There can be only one parent node	Allows a child to have more than one parent
4	Relationships between records is of parent-child type	Relationship is represented as pointers or links
5	There are multiple occurrence of child records and therefore inconsistency	This model is free from such inconsistency as there is only a single occurrence of a record set.
6	Searching a record is difficult as a child can be reached only through a parent	Searching a record is easy as there are multiple paths to a data element.

20 Explain functional dependency with example.

A functional dependency occurs when one attribute in a relation uniquely determines another attribute.

OR

Consider a relation say R(X,Y), where X and Y are one or more than one attribute, attribute X is functionally dependent on attribute Y if every value in X in the relation R has exactly one value of Y in the given relation.

The functional dependency is represented as XY, which specifies Yis functionally dependent on X or X attribute functionally determine the attribute Y.

Consider the schema, student(rollno, name, sscper).

rollnoname, rollnosscper are the functional dependencies. rollno uniquely identifies name and sscper. That is, given rollno of a student, the name and sscper can be determined or searched.

21 Explain merits and demerits of Object Oriented Database model.

Object oriented models were introduced to overcome the shortcomings of conventional models like Relational, Hierarchical and network model. An object-oriented database is collection of objects whose behavior, state, and relationships are defined in accordance with object-oriented concepts (such as objects, class, class hierarchy etc).

Merits:

 Object oriented data model allows the real world to be modeled closely. The object encapsulates both state and behavior. The object can also store the relations with other objects.

 It allows new data types to be built from existing types.

Redundancy can be reduced as common factors of several classes can be grouped into a super class and can be shared by the sub classes.

 It can be used to store a variety of data.

 Data evolution is easier.

Demerits:

 There is a lack of universal data model.

 Use of this type of modeling is still limited.

 It lacks standards since there is no universal data model.

 Increased functionality provided by this modeling makes it complex.

 There is no view mechanism

 There is no adequate security mechanism.

22 Explain any 4 Codd's rules.

Codd rules:

Rule 1: The information rule a has to be presented to the user should be in the form of table.

Rule 2: Guaranteed Access Rule

Whole data should be available or accessible to the user without any ambiguity. The ambiguity can be avoided only through the perfect combination of the table name, primary key, and column name.

Rule 3: Systematic treatment of null values

The null values i.e. absence of the values in the table should be treated properly. The table should allow a field to remain empty. This is not applicable to primary keys. Key columns cannot have null values.

Rule 4: Active on-line catalog based on the relational model

Fourth rule specifies need of dynamic on-line catalog based on the relational model. There are certain system tables that stores the database definition should be present. The data accessing tools should be used to access the database structure information.

Rule 5: The comprehensive data sub language rule: The system must support at least one relational language that Has a linear syntax Can be used both interactively and within application programs, Supports data definition operations (including view definitions), data manipulation operations (update as well as retrieval), security and integrity constraints, and transaction management operations (begin, commit, and rollback).

Rule 6: The view updating rule: All views those can be updated theoretically, must be updated by the system.

Rule 7: High-level insert, update, and delete: A database must support high-level insertion, updation, and deletion. This must not be limited to a single row, that is, it must also support union, intersection and minus operations to yield sets of data records

Rule 8: Physical data independence: Changes to the physical level (how the data is stored, whether in arrays or linked lists etc.) must not require a change to an application based on the structure.

Rule 9: Logical data independence: Changes to the logical level (tables, columns, rows, and so on) must not require a change to an application based on the structure.

Rule 10: Integrity independence: Integrity constraints must be specified separately from application programs and stored in the catalog. It must be possible to change such constraints as and when appropriate without unnecessarily affecting existing applications.

Rule 11: Distribution independence: The distribution of portions of the database to various locations should be invisible to users of the database. Existing applications should continue to operate successfully: when a distributed version of the DBMS is first introduced; and when existing distributed data are redistributed around the system.

Rule 12: The non subversion rule: If the system provides a low level (record-at-a-time) interface, then that interface cannot be used to subvert the system, for example, bypassing a relational security or integrity constraint

23 Explain distributed database system with example.

A distributed database is a database that consists of two or more files located in different sites either on the same network or on entirely different networks.

Portions of the database are stored in multiple physical locations and processing is distributed among multiple database nodes.

With distributed databases, data is physically stored across multiple sites and independently managed.

The processors on each site are connected by a network, and they don't have any multiprocessing configuration.

Distributed databases can be homogenous or heterogeneous.

In a homogenous distributed database system, all the physical locations have the same underlying hardware and run the same operating systems and database applications.

In a heterogeneous distributed database, the hardware, operating systems or database applications may be different at each location.

Advantage of Distributed databases:

Better Response – If data is distributed in an efficient manner, then user requests can be met from local data itself, thus providing faster response

More Reliable - When the data and DBMS software are distributed over several sites one site may fail while other sites continue to operate ,which makes database more reliable

Easier Expansion - : Expansion can be easily achieved by adding processing and storage power to the existing network.

Improved Performance -These systems provide greater efficiency and better performance

Resource Sharing -Since data is distributed, a group of users can easily share and use data of different sites

Though there are many distributed databases to choose from, some examples of distributed databases include Apache Ignite, Apache Cassandra, Apache HBase, Couchbase Server, Amazon SimpleDB, Clusterpoint, and FoundationDB

24 Define data abstraction. Explain the levels of data abstraction with neat diagram.

Data abstraction is defined as

Suppression of details of data organization and storage

Highlighting of the essential features for an improved understanding of data

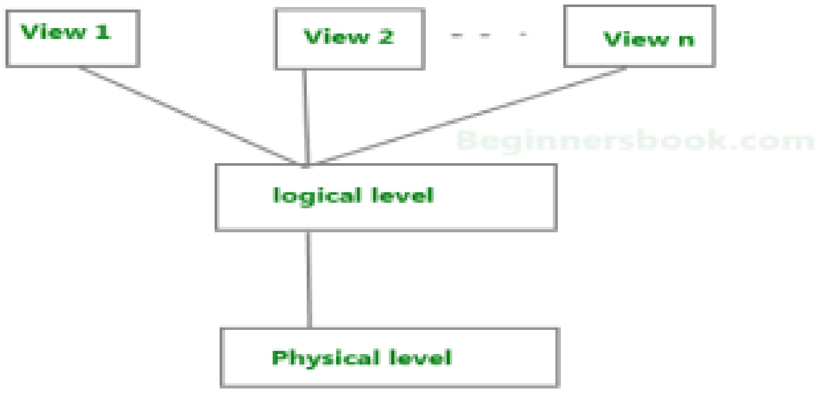
The characteristic that allow program data independence and program operation independence is called data abstraction.

Three levels of abstraction are:

Physical level: This is the lowest level of data abstraction. It describes how data is actually stored in database. The complex data structure details is described at this level.

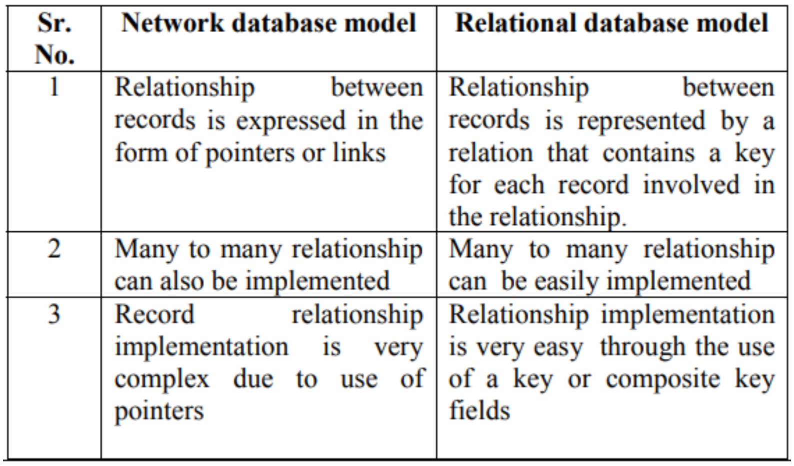
Logical level: This is the middle level of 3-level data abstraction architecture. It describes what data is stored in database and the relationships among the data.

View level: This is highest level of data abstraction. This level describes the user interaction with database system.

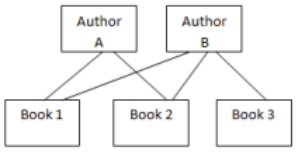


Three Levels of data abstraction

24 Distinguish between network database model and relational database model.



Sr. No.	Network database model	Relational database model
1	Relationship between records is expressed in the form of pointers or links	Relationship between records is represented by a relation that contains a key for each record involved in the relationship.
2	Many to many relationship can also be implemented	Many to many relationship can be easily implemented
3	Record relationship implementation is very complex due to use of pointers	Relationship implementation is very easy through the use of a key or composite key fields

4	Network model is useful for representing such records which have many to many relationships	Relationship model relations are is useful for representing most of the real world objects and relationship among them
5	In Network model also the record relations are physical	Relational model does not maintain physical connection among of records. Data is organized logically in the form of rows and columns.
6	Example: 	Example: Relation :Student Rollno name percentage 101 Abc 89.8

25 Describe enhanced ER model with the help of example.

Enhanced ER is a high-level data model that incorporates the extensions to the original ER model. It is created to design more accurate database schemas.

EER reflects data properties and constraints more precisely.

It also includes more complex requirements than traditional application.

It is a diagrammatic technique for displaying the following concepts

Sub Class and Super Class

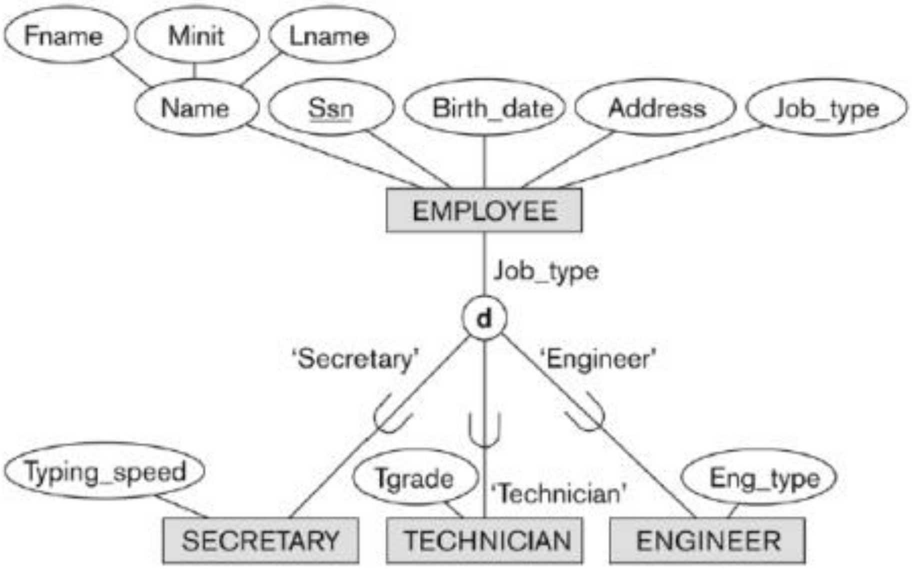
Specialization and Generalization

Union or Category

Aggregation

These concepts are used when they comes in EER schema and the resulting schema diagrams called as EER Diagrams.

For example: Square, Circle, Triangle are the sub class of Shape super class.



26 Describe functional dependency with example.

(Note: Any other example shall be considered)

A functional dependency occurs when one attribute in a relation uniquely determine another attribute.

(OR)

A relation say R attribute X is functionally dependent on attribute Y if every value in X in the relation has exactly one value of Y in the given relation.

The functional dependency is represented as X Y, which specifies Y is functionally dependent on X or X attribute functionally determine the attribute Y.

Example:

Consider table : Employee(Emp_Id, Emp_Name, Emp_Address)

Here Emp_Id attribute can uniquely identify the Emp_Name attribute of employee table because if we know the Emp_Id, we can tell that employee name associated with it.

Functional dependency can be written as:

Emp_Id → Emp_Name

27 Explain different types of attributes.

Types of Attributes:

1) Simple attributes : Attributes that cannot be subdivided (i.e are atomic) into subparts are called as simple attributes.

E.g: Enroll_no, RollNo

2) Composite Attributes: The attributes which can be divided into subparts are called composite attributes.

E.g: attribute name could be structured as a composite attribute consisting of first_name,middle_name and last_name

3) Single Valued Attributes:

The attribute has single value for a particular entity called as single valued attribute.

E.g: Student_id

4) Multivalued Attributes:

The attribute has set of values for a specific entity called as multi valued attribute.

E.g: Phone_no is multivalued attribute because employee may have zero, one or several phone no.

5) Derived Attribute:

The value for this type of attribute can be derived from the values of other related attributes or entities.

E.g: Customer entity has attribute age and date_of_birth. We calculate age from date_of_birth and current_date.

Here age is derived attribute and date_of_birth is base or stored Attribute

6) Stored Attribute:

The stored attributes are such attributes which are already stored in the database and from which the value of another attribute is derived is called stored attribute. For example: date_of_birth is a stored attribute from which age can be derived.

7) Null Attribute:

An attribute takes a null value when an entity does not have a value for it - that is value does not exist

for the entity. E.g apartment_no

28 Explain different operations performed with Data Definition Language.

DDL Operations:

1. Create

2. Alter

3. Drop

4. Rename

5. Truncate

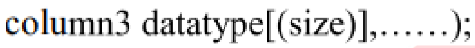
1) Create

in the database. It creates an empty structure of the table.

Syntax:

Create table < table name> (column1 datatype[(size]),

column2 datatype[(size]),



column3 datatype[(size)],……);

Example:

Create table employee (empno number(5), ename varchar2(20),Salary number(8,2));

2) Alter : It is used to add new attributes or to modify the existing attribute in the table structure.

Syntax for add option:

alter table<table name>

add(columnName1 datatype(size)

columnName2 datatyp(size)

...

columnNameNdatatyp(size)

);

Example:

alter table emp add(sal number(8,2));

Syntax for modify option:

alter table<table name> modify (<columnName1><data type>(<size>));

Example:

alter table emp modify sal number(10,2);

3) Rename :This command is used to rename a table, view, sequence or a synonym.

Syntax of Rename command:

rename <oldtable_name> to <newtable_name>;

Example:

rename employee to employee_details;

4) Drop: The DROP command removes a table from the database. All the tables' rows, indexes and privileges will also be removed. No DML triggers will be fired. The operation cannot be rolled back.

Syntax:

drop table <table name>;

Example: drop table emp;

5) Truncate :Truncate command is used to remove all rows from a table and to release the storage space used by the table keeping the table definition intact.

Syntax:

truncate table <table name>;

Example: truncate table emp;

29 Explain BCNF with example.

(Note: Any other example shall be considered)

BCNF:

Boyce Codd Normal Form (BCNF) is considered a special condition of third Normal form. A table is in BCNF if every determinant is a candidate key. A table can be in 3NF but not in BCNF. This occurs when a non key attribute is a determinant of a key attribute

Example of BCNF:

Let's assume there is a company where employees work in more than one department.

EMPLOYEE(EMP_ID,EMP_COUNTRY,EMP_DEPT, DEPT_TYPE,EMP_DEPT_NO)

In the above table Functional dependencies are as follows:

1.EMP_ID EMP_COUNTRY

2.EMP_DEPT {DEPT_TYPE, EMP_DEPT_NO}

Candidate key: {EMP-ID, EMP-DEPT}

The table is not in BCNF because neither EMP_DEPT nor EMP_ID alone are keys.

To convert the given table into BCNF, we decompose it into three tables:

1.EMP_COUNTRY table: EMP_ID EMP_COUNTRY

2.EMP_DEPT table: EMP_DEPT {DEPT_TYPE,

EMP_DEPT_NO}

3. EMP_DEPT_MAPPING table:EMP_ID,EMP_DEPT

Functional dependencies:

1. EMP_ID EMP_COUNTRY

2. EMP_DEPT {DEPT_TYPE, EMP_DEPT_NO}

Candidate keys:

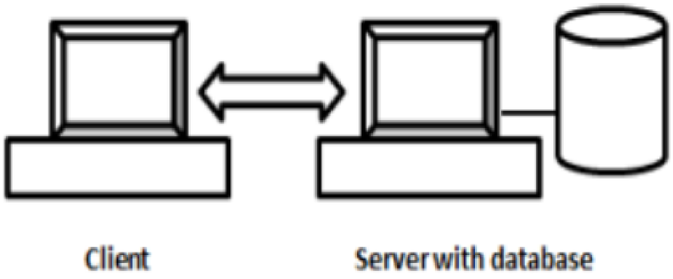
For the first table: EMP_ID

For the second table: EMP_DEPT

For the third table: {EMP_ID, EMP_DEPT}

Now, this is in BCNF because left side part of both the functional dependencies is a key.

30 Explain client/server database system.



Client/Server Database System

1. It has two logical parts client and server.

2. Computer networking allows some task to be executed on a server system and some tasks on client system. This leads to development of client server architecture.

3. Server is the machine which serves to the clients.

4. Server machine provide services to the client machine such as file access, printing, and database access. It is used to manage the database tables optimally among multiple clients who concurrently request the server for the same data.

5. The clients are the machines which requests for the service to the server.

6. There are different types of client/server architecture such as

Two tier architecture

Three tier architecture.

7. In two tier architecture, client systems directly approach database servers whereas in three tier architecture, there exists a middle layer which acts as application server to receive and send requests from client machine to database server and vice versa.

31 Explain terms primary key and candidate key with example.

Primary Key:

A primary key is an attribute in Relation that uniquely identifies the rows in relation. A Primary key does not hold NULL values and duplicate values.

OR

A key which is selected by the designer to uniquely identify the entity is called as Primary key. A primary key cannot contain duplicate values and it can never contain null values inside it.

Example:

In a Student table(Rollno,Name,Percentage), Rollno is the primary key

Candidate key:

In a relation there may be a key or combination of keys which uniquely identify the record. Such a key is called as Candidate key.

Example:

Consider a Student table (Rollno,Name,Percentage), if (Rollno) and(Name)both are unique then both are identified as candidate keys

OR

Consider a Student table (Rollno, Name, Percentage), if (Rollno, Name) is unique , then (Rollno, Name) can be a candidate key if and only if Name and Rollno individually are not unique.

32 Explain entity integrity constraint with example.

Entity integrity constraint:

1) Unique key constraint: It avoids the duplication of values within the rows in table. It allows null values.

Syntax:

Create table <table_name>

(column name1 datatype(size),

column_name2 datatype(size) constraint <constraint_name> unique,

---

column_name n datatype(size)

);

Example:

create table dept

(deptno number(5) constraint dept_deptno_uk unique,

dname varchar2(20),

loc varchar2(20));

2) Primary key constraint: Primary key constraint can be assigned on one or more columns in a table used to uniquely identifies the each row in table. It avoids duplication of rows and do not allow null values.

Syntax:

Create table <table_name>

(column name1 datatype(size),

column_name2 datatype(size) constraint <constraint_name> primary key,

---

column_name n datatype(size)

);

Example:

create table dept

(deptno number(5) constraint dept_deptno_pk primary key,

dname varchar2(20),

loc varchar2(20));

33 Describe centralized database system with example.

(Note: Any other example shall be considered).

Centralized Database System:

1. A centralized database consists of a single data server into which all data are stored and from which all data are retrieved. All the data reside at a single location and all applications must retrieve all data from that location.

2. The centralized database system consists of a single processor together with its associated data storage devices and other peripherals. It is physically confined to a single location.

3. Data can be accessed from the multiple sites with the use of a computer network while the database is maintained at the central site

Following are the advantages of centralised database system:

The data integrity is maximized

The data redundancy is minimal.

Centralized database is much more secure.

Data is easily portable because it is stored at the same place.

The centralized database is cheaper than other types of databases as it requires less power and maintenance.

Example:

Consider a company developing a project. As the project consist of many different types of information like documents, plans, diagrams, can be stored in a database on server which can act as a centralized database from which all the project members will assess the information acting as clients.

34 Describe object-oriented data models.

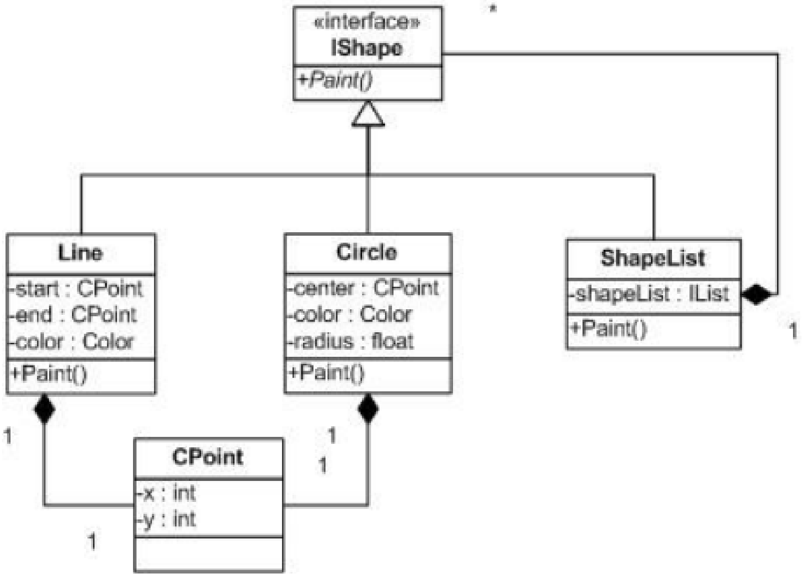
Object Oriented Model

Object oriented models were introduced to overcome the shortcomings of conventional models like Relational, Hierarchical and network model. An object-oriented database is collection of objects whose behavior, state, and relationships are defined in accordance with object-oriented concepts (such as objects, class, class hierarchy etc.)

In this both data and their relationship are organized or contained in a single structure known as object.

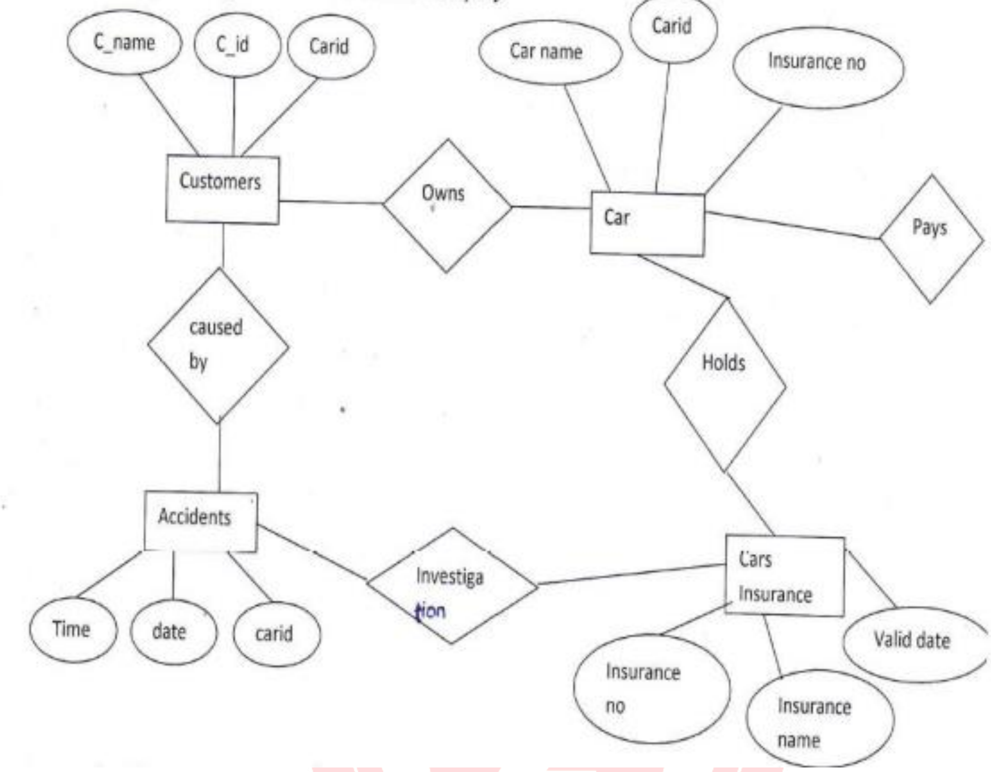
Object includes information about relationship between the facts within the object, as well as information about relationship with other objects.

It is also said to be semantic data model.



36 Construct an E-R diagram for a car insurance company whose customers own one or more cars. Assume suitable attributes.

Ans c) E R diagram for a car insurance company.



37 Explain Integrity constraints with example.

 Not Null: By default, all columns in tables allows null values. When a NOT NULL Constraint is enforced on column or set of columns it will not allow null values.

Example

SQL> CREATE TABLE STUDENT (ROLL_NO NUMBER (5), NAME VARCHAR2 (20) NOT NULL);

 Check Constraint: The constraint defines a condition that each row must satisfy. A single Column can have multiple check condition.

Example

SQL> CREATE TABLE EMP (ID NUMBER (5), NAME VARCHAR2 (10), SAL NUMBER (10) CONSTRINT CHK_SAL CHECK (SAL>15000));

 Primary Key constraint: It is used to avoid redundant/duplicate value entry within the row of specified column in table. It restricts null values too.

Example

SQL> CREATE TABLE EMP (ID NUMBER (5) CONSTRAINT ID_PK PRIMARY KEY, NAME VARCHAR2 (10), SAL NUMBER (10));

 Unique Constraint: The UNIQUE constraint uniquely identifies each record in a database table. The UNIQUE and PRIMARY KEY constraints both provide a guarantee for uniqueness of a column or set of columns. It allows null value.

Example

CREATE TABLE PERSONS (P_ID NUMBER CONSTRAINT P_UK UNIQUE, FIRSTNAME VARCHAR2(20), CITY VARCHAR2(20));

 Referential Integrity Constraint: It is a relational database concept in which multiple tables share a relationship based on the data stored in the tables, and that relationship must remain consistent. A value of foreign key is derived from primary key which is defined in parent table.

Example

CREATE TABLE DEPARTMENT (EMP_ID NUMBER(5)

REFERENCESEMP(EMP_ID), DNO NUMBER(3));

38 Explain benefits and drawbacks of Denormalization.

Benefits of denormalization (consider any 2)

 Reduce number of relations : It reduce the number of relations because it combines two relations into one new relation.

 Reduce number of foreign keys-It reduce number of foreign keys because number of relations are reduced .

 Minimizes need for joins-It minimizes need for joins because it combines many relations into one.

 Increase Performance - It increase performance of database by adding redundant data or by grouping data.

Drawbacks of demoralization.(consider any 2)

 Slow Data Updates-It may speed up the retrieval but can slow down database updates

 Increase size of relations -It can increase size of the relations due to combining multiple relations into one single relation.

 Complex implementation -It may simplify implementation in some cases but may make it more complex in other.

 Application Specific -It is always application-specific and needs to be re-evaluated if the application changes.

39 Explain advantages of centralized and distributed databases.

Advantage of Centralized databases (consider any 2)

 Data integrity is maximized -Data integrity is maximized and data redundancy is minimized because data is stored at a single place.

 Easier Database Administration -It is easy for database administration because Centralized databases are easy to manage, maintain, update, backup etc.

 Cost effectiveness – Cost will be less because, database is located ,stored and maintain at one central location

 Easy Modification, Access and Analysis -Data kept in the same location which makes modification, access and analysis easy.

Advantage of Distributed databases (consider any 2)

 Better Response – If data is distributed in an efficient manner, then user requests can be met from local data itself, thus providing faster response

 More Reliable - When the data and DBMS software are distributed over several sites one site may fail while other sites continue to operate ,which makes database more reliable

 Easier Expansion - : Expansion can be easily achieved by adding processing and storage power to the existing network.

 Improved Performance -These systems provide greater efficiency and better performance

 Resource Sharing -Since data is distributed, a group of users can easily share and use data of different sites

40 Describe the first normal form with its example

First Normal Form (1NF)

 A relation is said to be 1NF if and only if every entry of the relation has at most a single (atomic) value.

OR

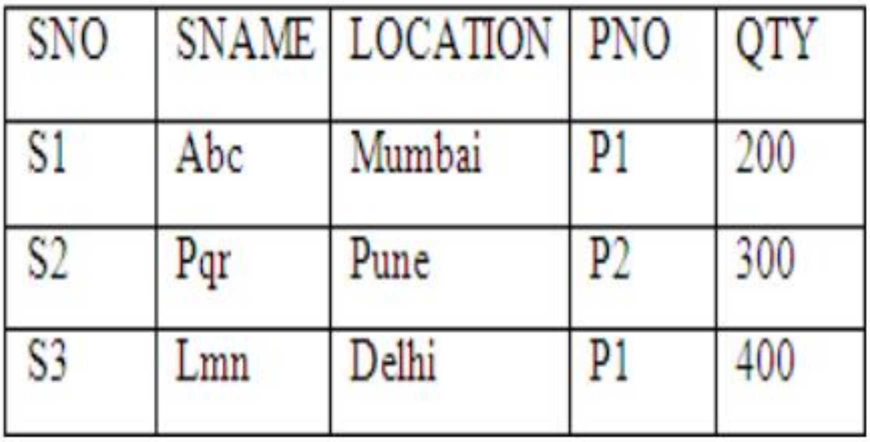
 A relation R is said to be in first normal form (1NF) if the domain of all attributes of R are atomic.

 It does not allow multivalued attributes and composite attributes.

Example

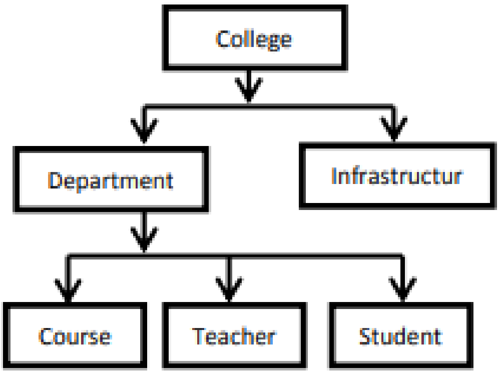
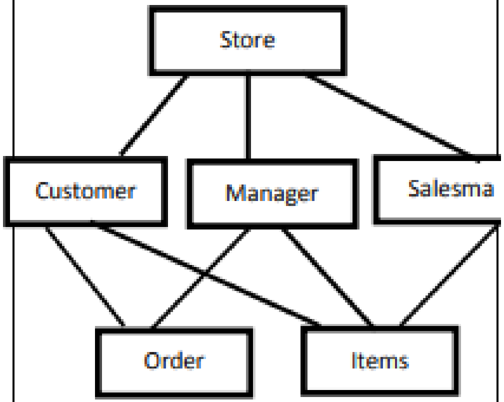
Supplier (sno, sname, location, pno, qty)

The above relation is in 1NF as all the domains are having atomic value. But it is not in 2NF



SNO	SNAME	LOCATION	PNO	QTY
S1	Abc	Mumbai	P1	200
S2	Pqr	Pune	P2	300
S3	Lmn	Delhi	P1	400

41 Compare Hierarchical Database Model with Network Model.

Sr No	Hierarchical Database Model	Network Model
1	Network Model represents tree like structure with one root.	Network Model represents tree like structure with many roots.
2	Reflects 1:N (One to many) relationship	Reflects M:N (Many to many) relationship
3	There can be only one node at the parent level	It allows a record to have more than one parent.
4	Example:  College Department Infrastructur Course Teacher Student	Example : 
5	Relationship between records is of parent child type	Relationship between records is expressed in the form of pointers or links(Graphs).
6	Searching for a record is very difficult since one can retrieve a child only after going through its parent record.	Searching a record is easy since there are multiple access paths to a data element
7	There are multiple occurrences of child records, which lead to problem of inconsistency during the update operations	This model is free from update anomalies because there is only a single occurrence for each record set.
8	Record relationship implementation is simple due to the use of pointers	Record relationship implementation is complex due to the use of pointers

42 Explain three level architecture of Database

There are following three levels or layers of DBMS architecture:

• External Level : Describes part of the database that a particular user group is interested in.

•Conceptual Level: Describes structure of the whole database for a community of users.

• Internal Level : Describes physical storage structure of the database.

External Level or View level

It is the users' view of the database. This level describes that part of the database that is relevant to each user. External level is the one which is closest to the end users. This level deals with the way in which individual users view data. Individual users are given different views according to the user's requirement.

Conceptual Level or Logical level

It is the community view of the database. This level describes what data is stored in the database and the relationships among the data. The middle level in the three level architecture is the conceptual level. This level contains the logical structure of the entire database as seen by the DBA. It is a complete view of the data requirements of the organization that is independent of any storage considerations.

The conceptual level represents all entities, their attributes, and their relationships.

Internal level or physical level

It is the physical representation of the database on the computer. This level describes how the data is stored in the database. The internal level is the one that concerns the way the data are physically stored on the hardware.

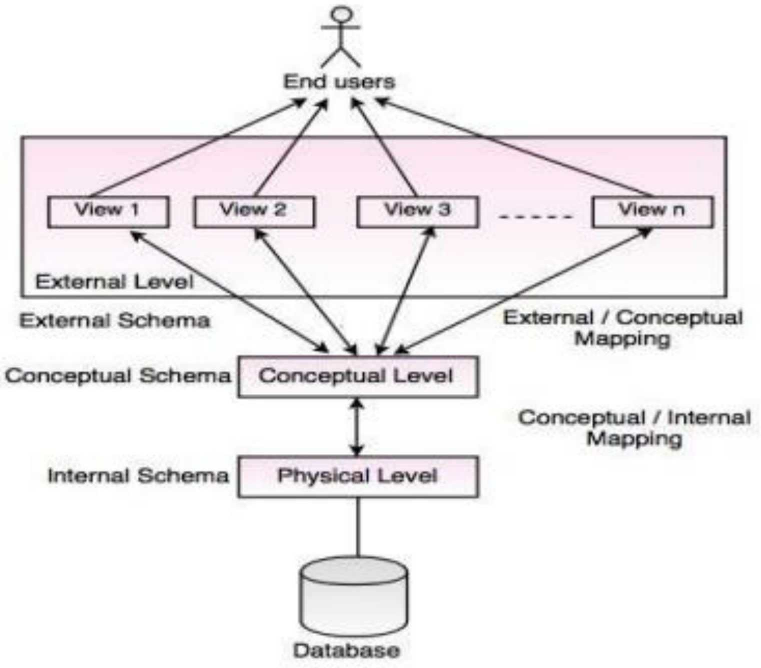


Fig. Three Level Architechture of DBMS

43 Explain various types of Relational constraints.

Relational Constraints

Relational constraints are a set of rules. It is used to maintain the quality of information. Integrity constraints ensure that the data insertion, updating, and other processes have to be performed in such a way that data integrity is not affected. Thus, integrity constraint is used to guard against accidental damage to the database.

Types of Relational integrity Constraints are as follows

1. Domain constraints
2. Entity integrity constraints
3. Referential Integrity Constraints

Domain Constraint - It is used to maintain value according to user specification

For example: Not null, check constraint.

Entity integrity constraints –it provides a way of ensuring that changes made to the database by authorized users do not result in a loss of data consistency.

For example: Primary key, unique constraints

Referential Integrity Constraints – It establishes parent child relationship between two tables.

For example :Foreign key constraints

6 Marks Questions

1 Find 3NF decomposition of given relation schema. Shipping (ship, capacity, date, cargo, value).

Functional dependencies

Ship -> capacity.

Shipdate -> cargo.

Cargo, Capacity -> Value

R= (ship, capacity, date, cargo, value).

Functional dependencies

Ship -> capacity.

Ship, date -> cargo.

Cargo, Capacity -> Value

1)Find all attributes in R that are not involved in any functional dependency. Here no such attribute found.

2)R= (ship, capacity, date, cargo, value)

No functional dependency has all the attributes.

3)For each Functional dependency

i)Ship -> capacity

R1= (ship, capacity)

ii)Ship, date -> cargo.

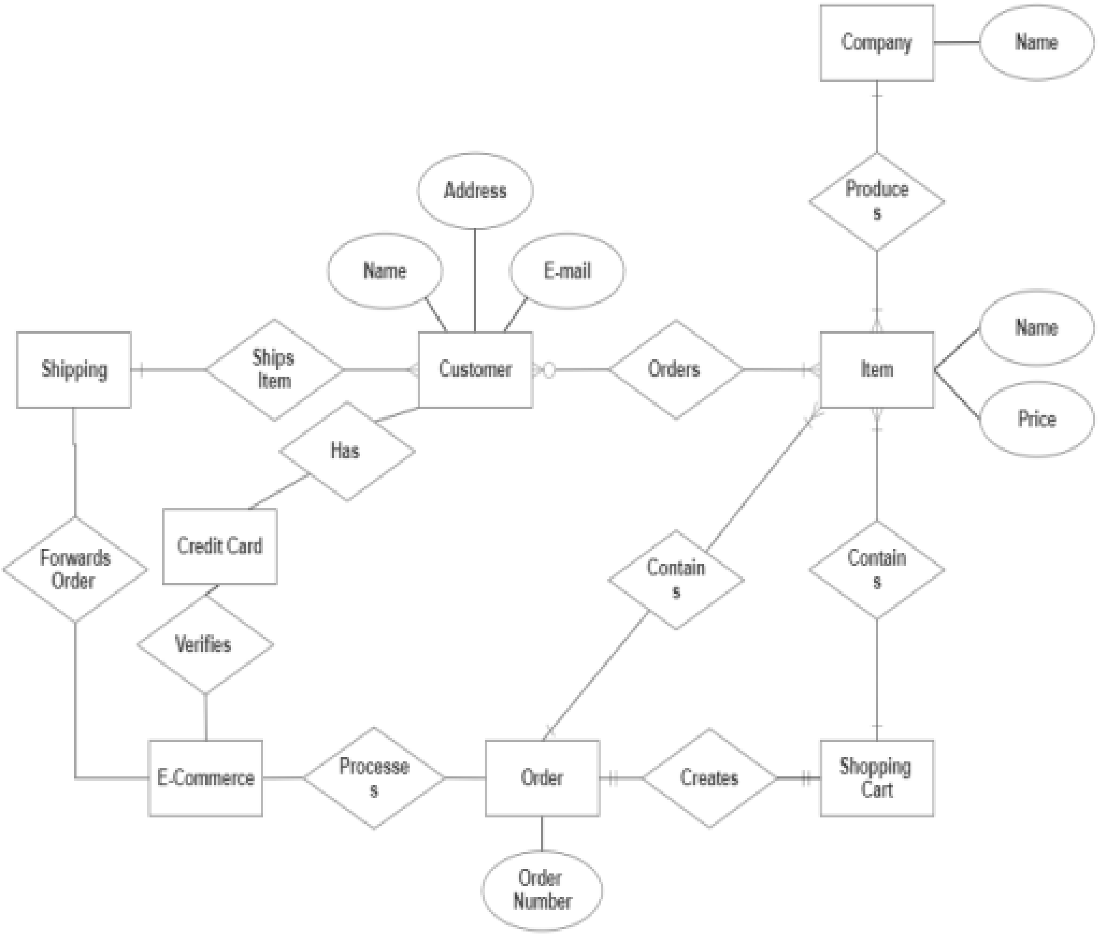
R2= (ship, date, Cargo)

iii) Cargo, Capacity -> Value

R3= (cargo, capacity, value)

Above 3 relations R1, R2 and R3 gives 3NF decomposition which is lossless and dependency preserving

2 Draw an ER diagram for online sales system in which customer can order items online and pay through credit cards



3 Let R = (A, B, C, D) and

F={AB C, C A}

Find BCNF decomposition of R using the algorithm.

Let R1= (A, B, C, D) (Given relation)

Step:1

Find merged minimal cover of Functional Dependencies (FDs) which

contain

AB->C

C->A

Round 1:

Checking whether R1 is in BCNF

The FD [AB->C] violates BCNF as LHS is not superkey , so table is split as below

R2 = (A, B, C)

R3= (A, B, D)

Round 2:

Checking whether R2 is in BCNF

The FD [C->A] violates BCNF as the LHS is not superkey, so table is split as below

R4= (C, A)

R5= (B, C)

Relation R3, R4, and R5 are in BCNF

4 Consider schema student (roll no., name, marks, address, mobile no., birthdate). Write commands for :-

i) create table.

ii) insert values.

iii) alter table.

iv) truncate table.

v) delete row.

vi) drop table

Write proper output of each.

i) SQL>create table student (

rollno number(5),

name varchar2(15),

marks number(5,2),

address varchar2(20),

mobileno number(15),

birthdate date

);

ii)SQL> insert into student values(101,‟Rajesh‟,75, „Thane‟,9889992345 ‟13-JAN-2004‟);

(OR)

SQL>insert into student(rollno,name,marks,address,mobileno,birthdate)values(101,‟Rajesh‟,75,

„Thane‟,9889992345, ‟13-JAN-2004‟);

iii)SQL>Alter table student modify (name varchar2 (20));

(OR)

iii)SQL>Alter table student add (course varchar2 (10));

iv)SQL>truncate table student;

v)SQL>Delete from student where rollno=101;

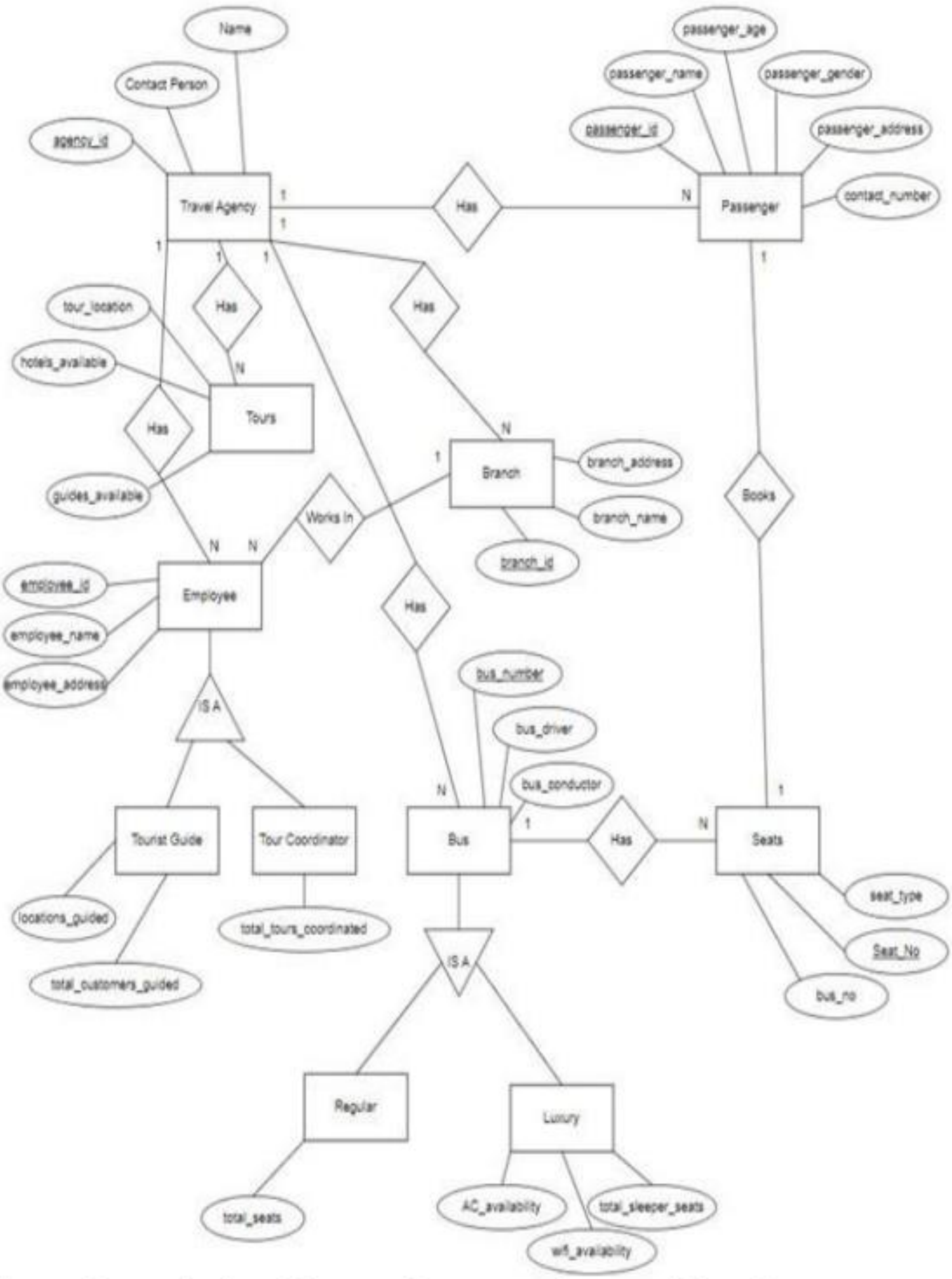
vi)SQL>drop table student;

5 Construct an ER diagram for travel agency consider various

entities such as travel agency, passenger, branch, seat, bus,

employee, tours etc.

Design specialization and generalization EER features.



(Any other relationship, attributes can be considered)

6 Consider following schemas.

i) Course details (course code, course name, fees)

ii) Student details :- (Student-id, name, marks, subjects, course code, dept.)

Identify :- l) Primary key 2) Super key 3) Foreign key

With justification, draw and explain parent child relationship for

above schemas.

Primary key

1)coursecode attribute is a primary key of Coursedetails relation

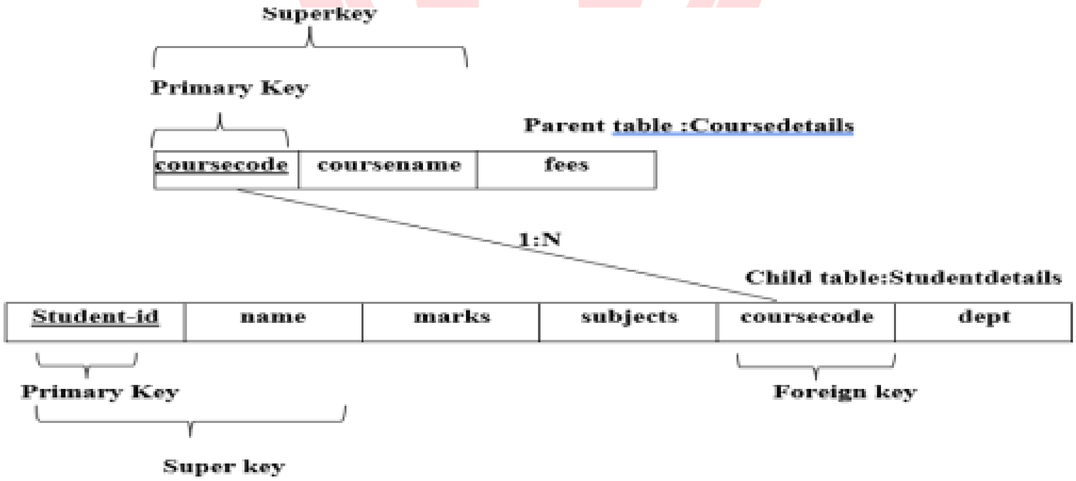
2)Student-id is attribute is a primary key of Studentdetails relation

Super key

1. Coursedetails (coursecode,coursename)

2. Studentdetails (Student-id,name)

Parent child relationship for given schema is:



Foreign key :coursecode is a foreign key of studentdetails relation.

Since there exist a common attribute coursecode in both tables Course details and Student details coursecode attribute uniquely identifies course, is a primary key of Course details relation, coursecode is a foreign key of student details relation. A student canhave a course that exist in Course details table and hence we need to reference coursecode in Student details table from coursecode in Course details table. To ensure this referential integrity coursecode in Student details table becomes the foreign key referenced to coursecode primary key from Course details table

7 Consider following relation

student (Roll_No, name, class, total_marks, percentage, Grade).

Find appropriate dependencies and normalize upto 3NF.

Functional Dependencies:

Roll_no⍰ name

Roll_no⍰ class

total_marks⍰ percentage

percentage ⍰ Grade

1NF: Student(Roll_no,name.class,total_marks,percentage,Grade)

2NF: To convert It into 2NF, We have to decompose the given table into two tables with fully functional dependencies and establishing a referential integrity constraint relationship among the two tables.

Student(Roll_No, name, class)

Marks(Roll_No, total_marks, Percentage, Grade)

3NF: To convert the above tables in 3NF ,We have to decompose them in three tables satisfying the transitive dependencies property

Student(Roll_No, name, class)

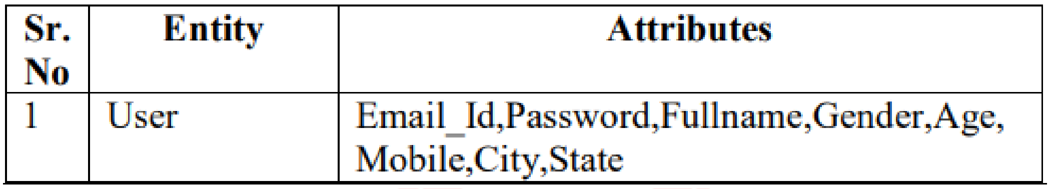
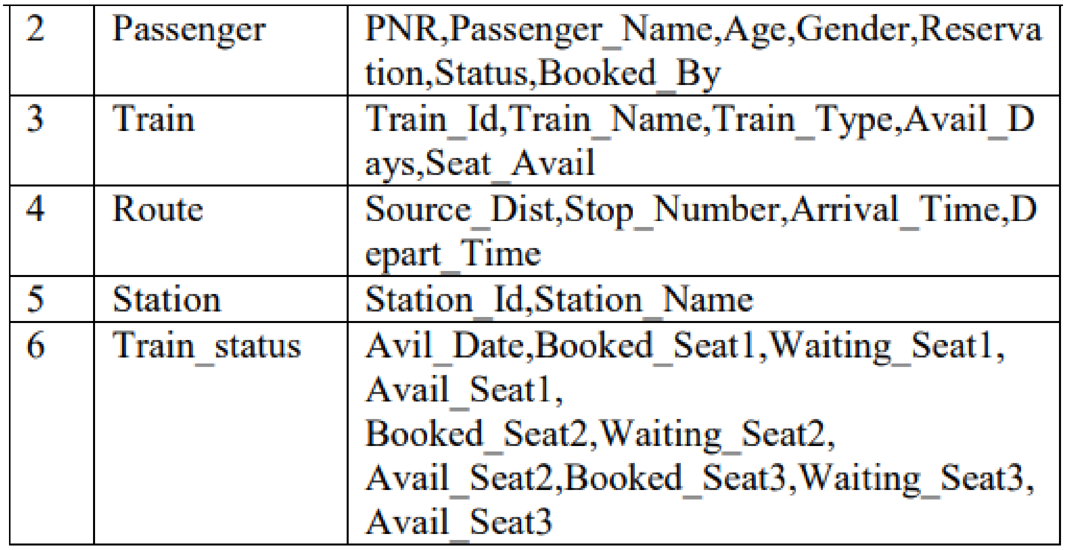
Marks(Roll_No, total_marks, percentage)

Grade (percentage, Grade)

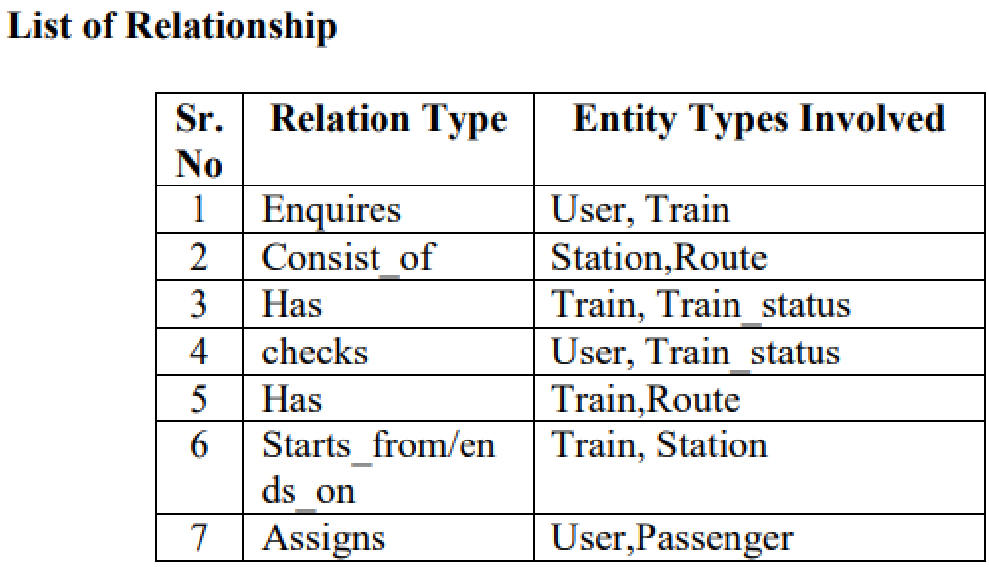
8 Identify entities and their relationship in terms of tables for railway reservation system.

(Note: Any other entity or relationship shall be considered)

List of Entity Types:

Sr. No	Entity	Attributes
1	User	Email_Id,Password,Fullname,Gender,Age, Mobile,City,State
2	Passenger	PNR,Passenger_Name,Age,Gender,Reservation,Status,Booked_By
3	Train	Train_Id,Train_Name,Train_Type,Avail_Days,Seat_Avail
4	Route	Source_Dist,Stop_Number,Arrival_Time,Depart_Time
5	Station	Station_Id,Station_Name
6	Train_status	Avil_Date,Booked_Seat1,Waiting_Seat1, Avail_Seat1, Booked_Seat2,Waiting_Seat2, Avail_Seat2,Booked_Seat3,Waiting_Seat3, Avail_Seat3



List of Relationship

Sr. No	Relation Type	Entity Types Involved
1	Enquires	User, Train
2	Consist_of	Station,Route
3	Has	Train, Train_status
4	checks	User, Train_status
5	Has	Train,Route
6	Starts_from/ends_on	Train, Station
7	Assigns	User,Passenger

9 Consider given relation R = (A, B, C, D, E) with the following functional dependencies {CE D, D B, C A}.

(i) List all key for R.

(ii) Identify the best normal form that R satisfies.

Step1: Find attributes that are neither on the LHS nor on RHS

--None

Step2:Find the attributes that are only on RHS

--A ,B

Step3: Find the attributes that are only on LHS.

--C, E

Step4: Combine the attributes on step 1 and 3- C E

The attributes C and E will belong to candidate key, but to find others we need to calculate closure of CE

Step5: Closure finding :

In our case, because with CE we get D and from D we get B and from C we get A

So we have only one candidate key that is CE

The relation is in 1NF as it does not have any composite as well as multivalued attribute.

But it is not in 2NF as the statement says that

1) It should be in 1NF

2) All non-key attributes are fully functionally dependent on primary key

In our case rule 2) is violated by C⍰A

Thus given relation is best suited for 1NF only.

10 Consider the following schema

student (R_No, Name, DOB, Percentage, D_No).

Write procedure to manipulate given database by adding, modifying and deleting records.

Consider given Schema

Student(R_No,Name,DOB,Percentage,D_No)

For adding records in table:

We use Insert into command for adding/inserting data into Student table.

Syntax for adding the values in the table is as follows:

SQL> Insert into <table name> values (value1, value2, value3…);

Ex:

SQL>insert into Student values(1,’Ram’,’12-Jan-1990’,88,10)

OR

Ex:

SQL>Insert into Student

values(&R_No,’&Name’,’&DOB’,&Percentage,&D_No);

For modifying records in table

We use update command for modifying data of Employee table.

The syntax of update command is:

Update<table name>set

<columnname>=<expression>,<columnname>=<expression>;

Ex:

SQL>update Student set DOB=’22-feb-1995’ where R_No=3;

For deleting records from table:

We use delete command for deleting data of Employee table.

Syntax:-

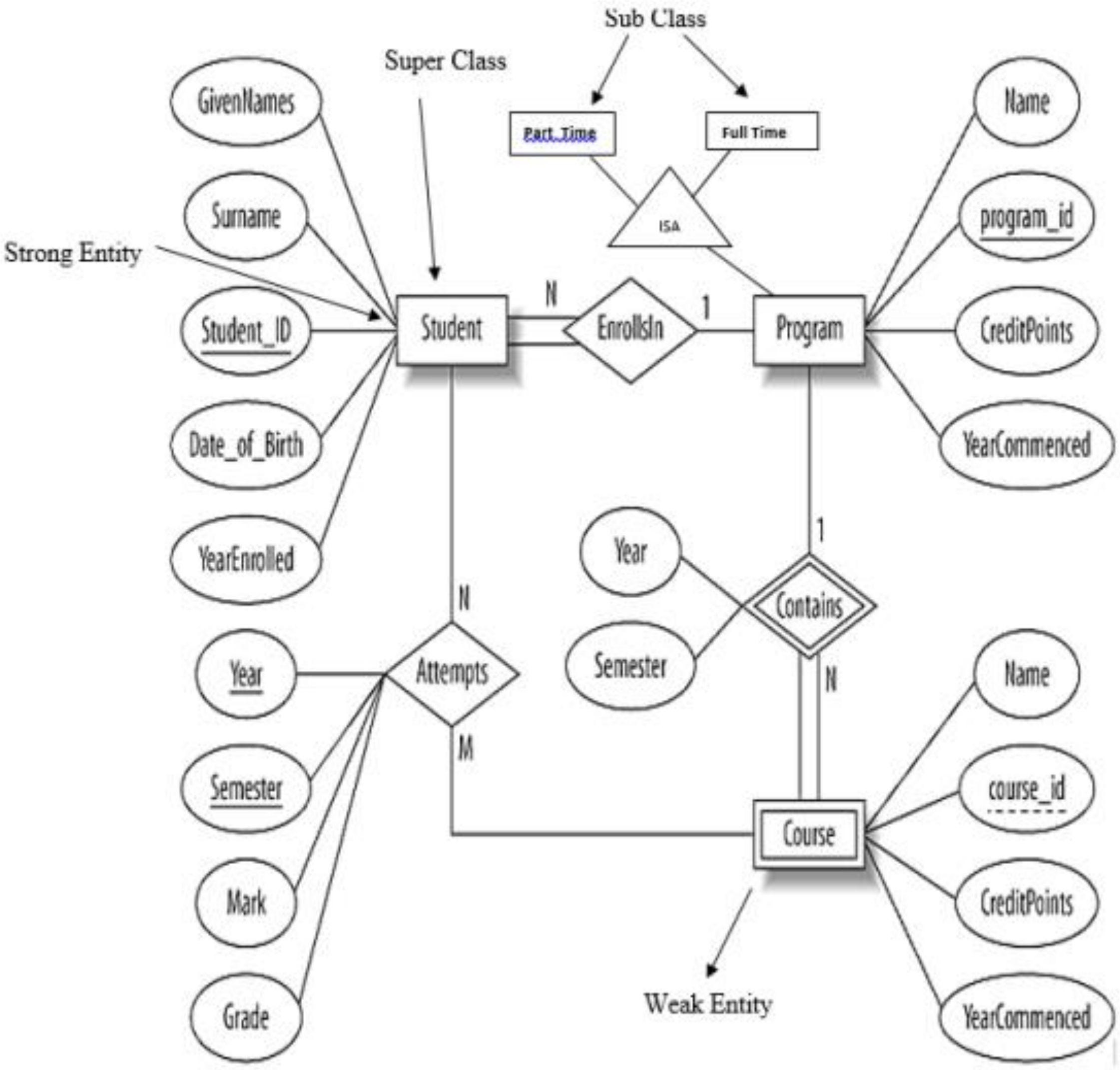
Delete from <table name> where <condition>;

Ex:

SQL>delete from Student where R_No=2;

11 Draw the enhanced E-R diagram for College Management System and show strong entity set, weak entity set, super class and sub class.

(Note: Any relevant diagram shall be considered)



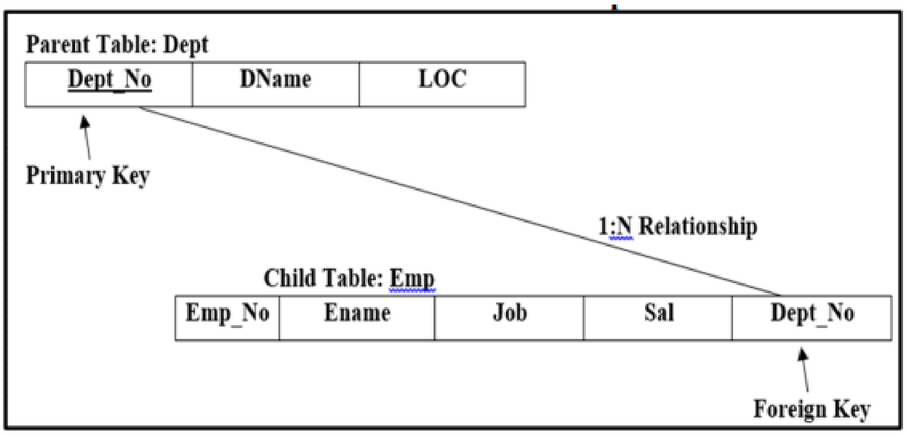
12 Consider the following schemas:

(i) Dept (Dept_No, DName, LOC)

(ii) Emp (Emp_No, Ename, Job, Sal, Dept_No)

Draw and explain parent child relationship for above schemas and apply referential integrity constraint.

Parent child Relationship



Referential integrity constraint:

 It is used to establish the parent child relation between two tables having common column.

 Value of foreign key is derived from primary key.

 We should define the column in the parent table as a primary key and same column in the child table as a foreign key referring to the corresponding parent key

Dept (Dept_No, DName, LOC)

Emp(Emp_No,Ename,Job,Sal,Dept_No)

In table Dept, Dept_No is a primary key containing unique values for deptnos.

To set the relationship between these two tables , we can define Emp.Dept_No as a foreign key as

1. Create table Dept

(

```
Dept_No number(5) constraint Dept_Dept_No_pk primary key,
DName varchar2(20),
LOC char(10)
);
```

2. Create table Emp

```
(
Emp_No number(4),
Ename varchar2(25),
Job char(10),
sal number(10,2)
Dept_No number(5) constraint Emp_Dept_No_fk references
Dept(Dept_No),
);
```

13 Consider a single table consisting following columns. Convert it into 2NF and 3NF Table:

(supplier_no, supplier_name, supplier_city, order_no, order_quantity, order_amount, product_code, product_name)

Given Table Schema - (supplier_no, supplier_name, supplier_city, order_no, order_quantity, order_amount, product_code, product_name)

Second Normal Form (2NF):

To convert it into 2NF, We have to decompose the given table into two tables with fully functional dependencies and establishing a referential integrity constraint relationship among the two tables.

Table 1- Supplier Details

(supplier_no,supplier_name,supplier_city,order_no)

Table 2 - Order Details

(order_no, order_ quantity, order_amount, product_code, product_name)

Now the above two tables are in 2NF.

Third Normal Form (3NF):

To convert the above tables in 3NF ,We have to decompose them in three tables satisfying the transitive dependencies property.

Table 1- Supplier Details

(supplier_no ,supplier_name,supplier_city)

Table 2- Product Details

(product_code,product_name)

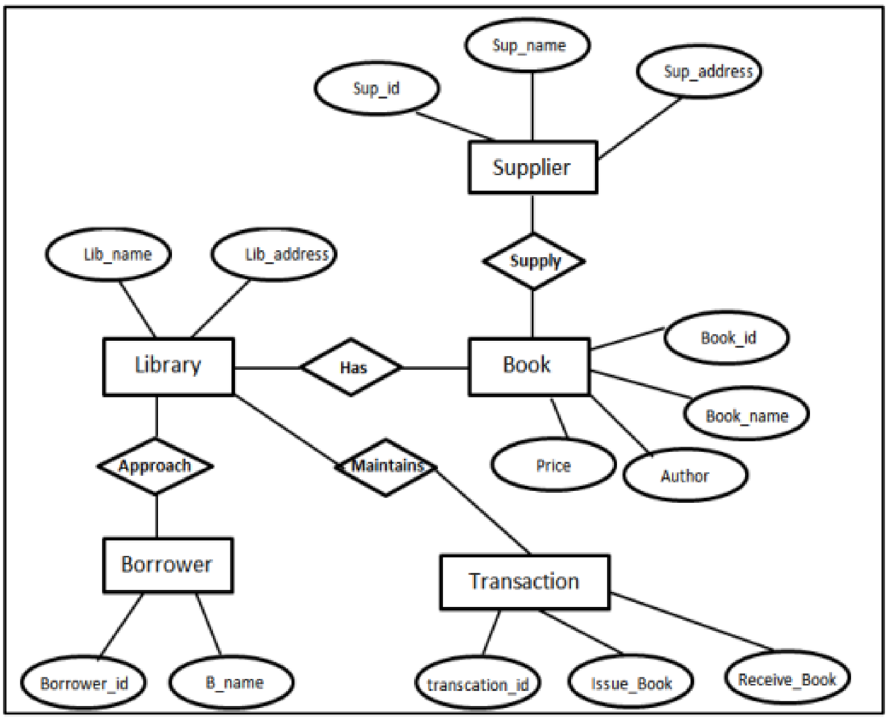
Table 3- Transaction(Order) Details

(order_no, product_code,supplier_no, order_ quantity, order_amount)

Hence the above three tables are satisfying Transitive dependencies

Thus they are in 3NF

14 Draw ER diagram of library management system in which library maintain the data of books, borrowers, issue return details, fine collection, supplier of books etc. Assume suitable data and display the relationship among entities.



15 Consider the relation R with five attributes L, M, N, O, P

You are given following dependencies:

L M, MN P, PO L

(i) List all keys for R.

(ii) Is R in 3NF?

Justify your answer.

(i) List all keys for R:

Since Right hand side does not have NO, it can be part of the key.

So, (NO)+ ={NO}

We will try other combinations with NO

(LNO)+ ={LNOMP} it is candidate key.

(MNO)+={MNOPL} it is candidate key.

(PNO)+={PNOLM} it is candidate key .

we get Keys as LNO,MNO,PNO.

(ii) IsRin3NF?:

M, P, L are prime attributes, so R(L,M,N,O,P) is in 3NF

16 Consider the following schemas:

(i) Dept (Dept_no, Dept_name, Dept-loc)

(ii) Staff (Staff_id, Staff_name, Dept_no, Joint_date)

Draw and explain parent-child relationship for above schemas and find out foreign key with justification.

Parent –Child Relationship Diagram for given Schema is as follows:

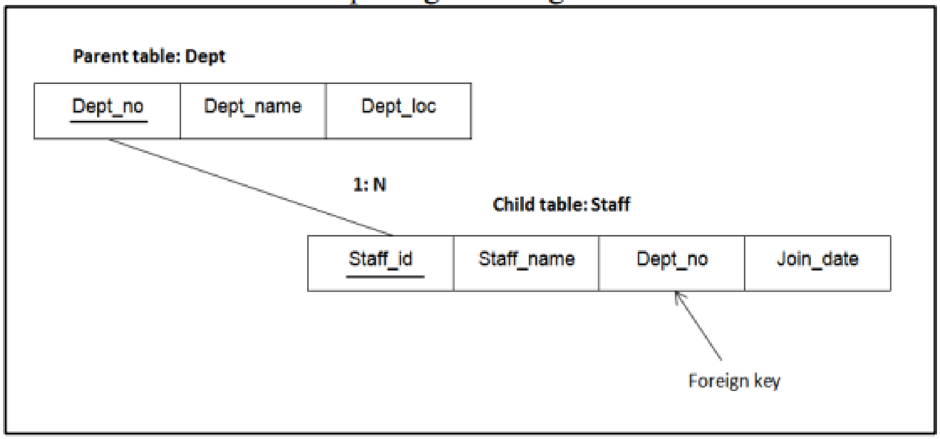


Fig: Parent Child Relationship diagram

Foreign key: Dept_no is Foreign key for table Staff

Justification:

As per above schemas, Dept table is parent table and Staff table is child table.

Dept_no is primary key for Dept table.

There exist Dept_no as a common attribute in both the tables Dept and Staff.

Staff_id is primary key for Staff table.

So, Dept_no is foreign key for table Staff.

17 Draw enhanced ER diagram for loan payment system. Consider the following entities:

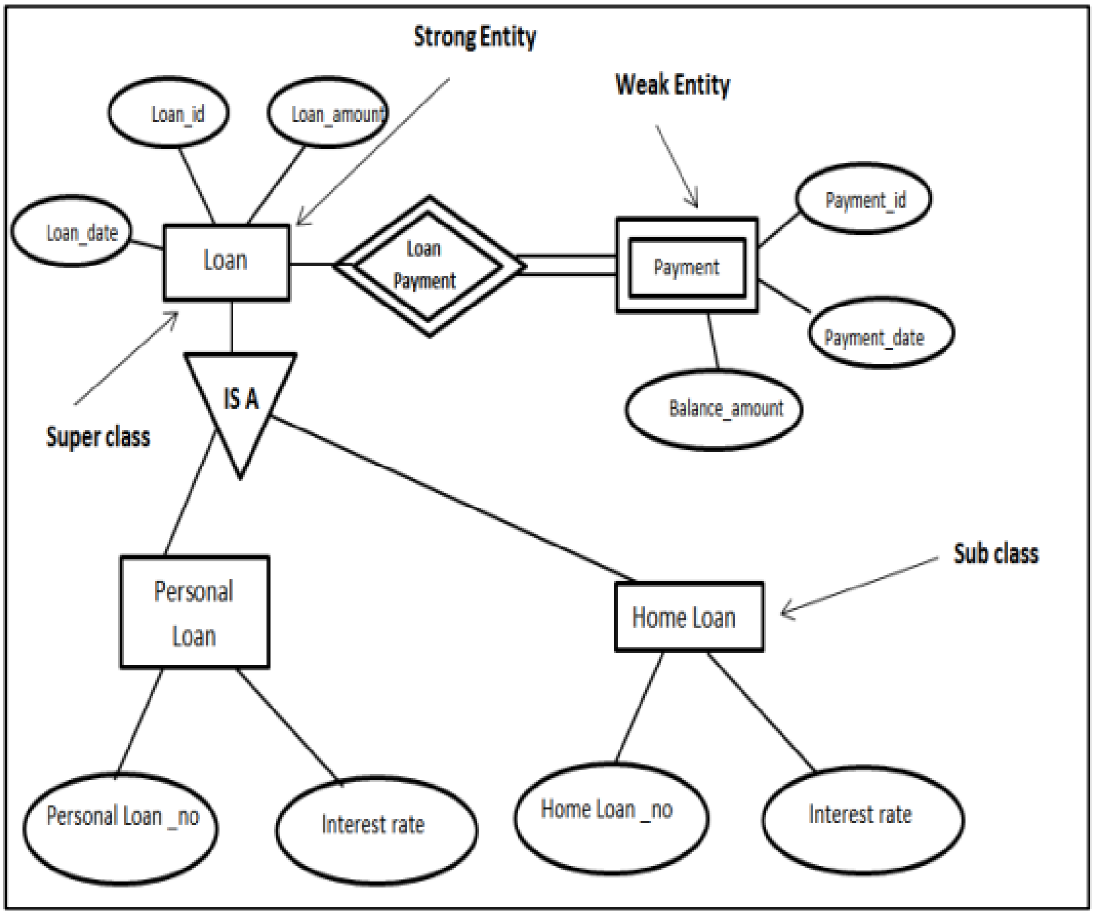
(i) Loan (Loan_id, Loan_amount, Loan_date)

(ii) Payment (payment_id, Payment_date, Balance_amount)

(iii) Personal Loan (Personal Loan_no, Interest rate)

(iv) Home Loan (Home Loan_no, Interest rate)

Show strong entity set, weak entity set, super class and sub class.



18 Consider 'Employee' database with appropriate details. Write a procedure to manipulate given database by adding, modifying and deleting records.

Let us consider a Schema for Employee table

(emp_id,emp_name,emp_addr,emp_salary)

For adding records in table:

We use Insert into command for adding /inserting data into Employee table.

Example:

OR

Example:

SQL> Insert into Employee

values(&emp_id,'&emp_name','&emp_addr,'&emp_salary);

For modifying records in table:

We use update command for modifying data of Employee table.

Example:

SQL> update Employee set salary=30000 where emp_id=3;

For deleting records from table:

We use delete command for deleting data of Employee table.

Example:

SQL>delete from Employee where emp_id=4;

19 Consider relation R with five attributes L, M, N, O, P.

You have been given following dependencies

$L \to M, MN \to P, PO \to L$

(i) List all keys for R.

(ii) In what Normalized form R is? Justify your answer

(i) List all keys for R.

Since Right hand side does not have NO

So (NO)+ =NO

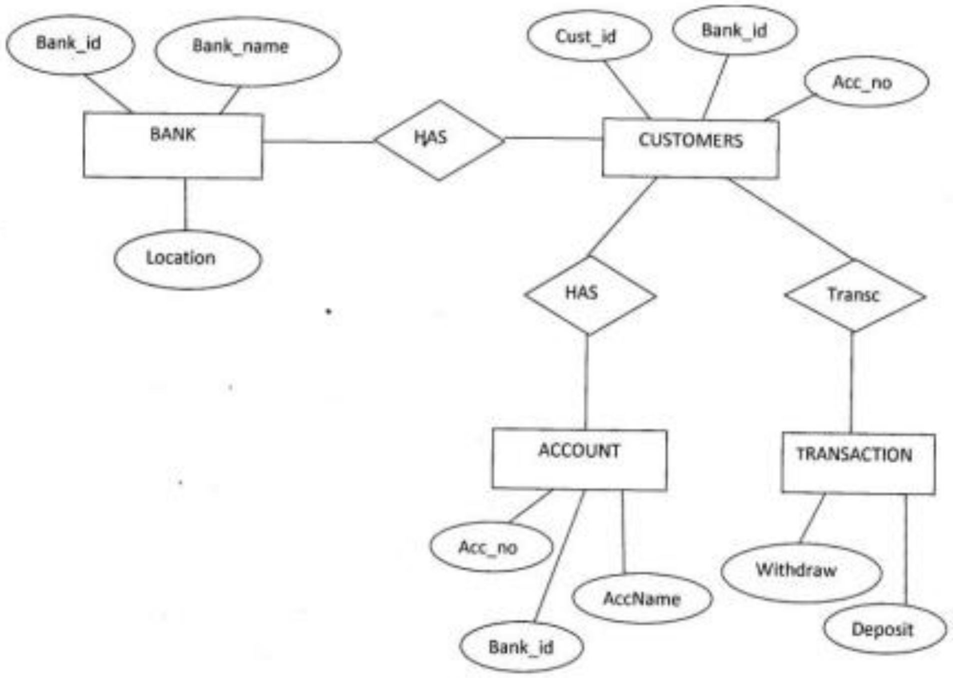
Now Combining NO with L,M,P

we get Keys as LNO,MNO,PNO

(ii) In what Normalized form R is? Justify your answer

M,P,L are prime attributes, so R(L,M,N,O,P) is in 3NF

20 Draw ER diagram for Banking system, to represent a customer has account scenario. Identify entities with their attributes and draw a diagram.



11 Consider a single table consisting following columns. Convert it into 2NF and 3NF. Table (Supplier_no, Supplier_name, Supplier_city,. Order_no, Order_quantity, order_amount, product_name)

Table 1 Schema given:

(Supplier_no,Supplier

Name,Supplier_city,Order_no,Order_quantity,Order_amount,Product_code Product name)

Step 1.To convert It into 2NF, We have to decompose the given table into two tables with fully functional dependencies and establishing a referential integrity constraint relationship among the two tables.

Table2: Supplier Details

(Supplier_no,Supplier_name,Supplier_city,Order_no)

Table 3:Order Details

(Order_no, Order_ quantity, Order_amount, Product_code, product_name)

Now the above two tables are in 2NF

Step 2: To convert the above tables in 3NF ,We have to decompose them in three tables satisfying the transitive dependencies property.

Table 4: Supplier Details

(Supplier_no,Supplier_name,Supplier_city)

Table 5: Order Details

((Order_no, Order_ quantity, Order_amount)

Table 6: Trasaction Details

(Supplier_no, Order_no, Product_code, product_name)

Hence the above three tables are satisfying Transitive dependencies

Thus they are in 3NF.

12 Consider 'student' database with appropriate details. Write a procedure to manipulate given database by adding, modifying

and deleting records.

Let us consider a Schema for student database

(Student_id,Student_name,Student_addr,Student_contact)

1.To add records into the given database , we have to use Insert into command.

Syntax for inserting the values in the table is as follows:

SQL> Insert into <table name> values (value1, value2, value3…);

Example:SQL> Insert into student

values(101,’Rajesh’,Thane,9889923456);

OR

Example:

SQL> Insert into student
values(&Studentid,'&Student_name','&Student_addr,'&Student_contact);

2.To update records in given database, we have to use UPDATE command.

The syntax of update command is:

Update<table name>set

<columnname>=<expression>,<columnname>=<expression>;

Example

SQL> update student set Student_addr= 'Borivili';

3.To delete records from the database, we have to use DELETE command.

Syntax:-

Delete from <table name> where <condition>;

Example:

Delete from student where Student_ addr='Thane';

1 row deleted

13 For each of following relationship indicate type of relationship (1:1, 1:m, m:m)

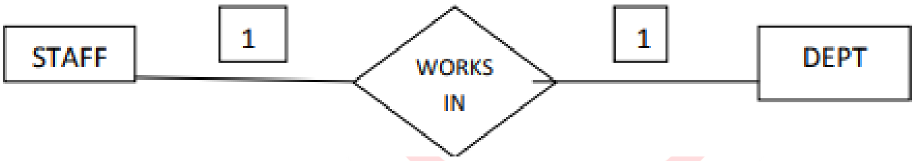
(i) Works in (a relationship between entities dept. and staff)

(ii) Managers (a relationship between entities employee and Manager)

Note: Considering Managers in relationship as Manages

i) Works in(a relationship between entities dept and staff)

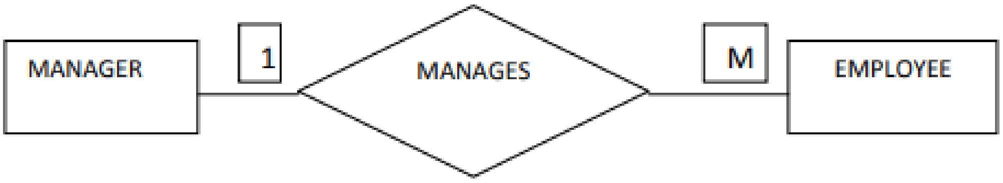
Diagram:



The above relationship indicates 1:1 (one-to-one) relation type because one staff can work in one department only at a given period.

ii) Managers(a relationship between entities employee and manager)

Diagram:



The above relationship indicates 1:m(one-to-many) relation type because one manager can manage many employees in a department.

14 Draw Enhanced ER diagram for loan payment system. Consider following entities:

(i) Loan (Loan_id, Loan_amount, Loan_date)

(ii) Payment (Payment_id, Payment_date, Balance_amount)

(iii) Personal Loan (Personal Loan_no, Interest rate)

(iv) Home Loan (Home loan_no, Interest rate)

Show strong entity set, weak entity set, super class and sub class.

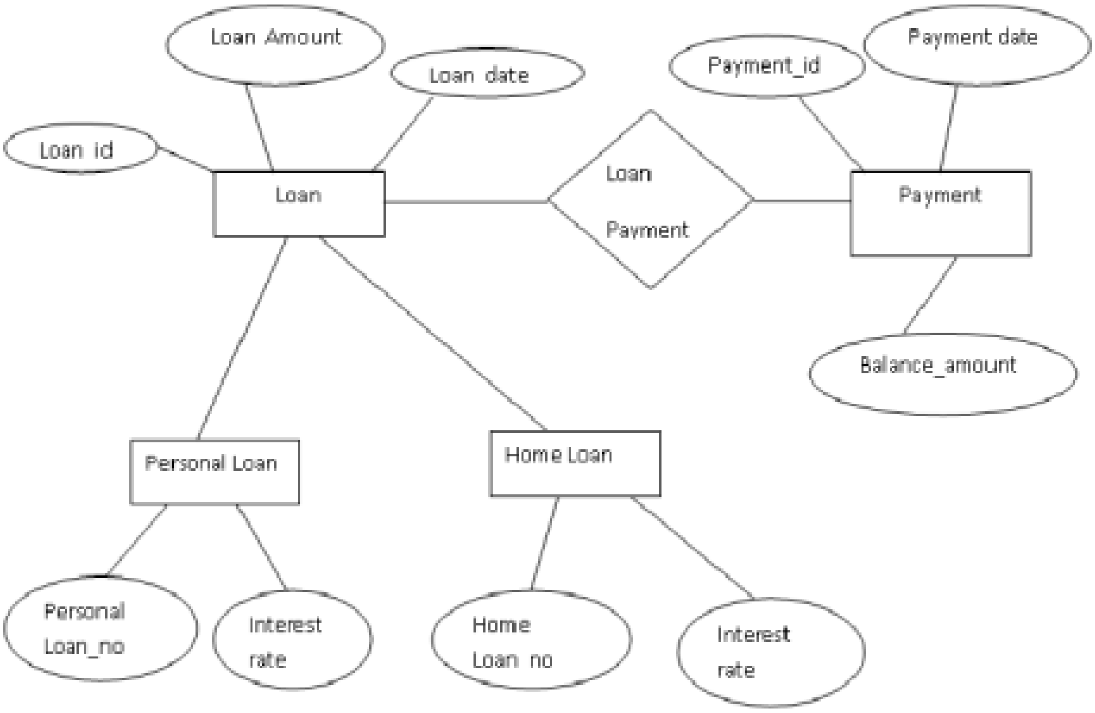
1. All the above given entities contains a primary key attribute. So all the entities are Strong entity sets.

Example: Loan_id is a primary key attribute present in loan entity.

2. There is absence of weak entity sets since all the entities contain a primary key attribute.

3 .Loan is a super class present in the above EER diagram.

4. Personal Loan and Home Loan are the sub classes present above


Loan Amount
Loan date
Loan id
Loan
Loan
Payment
Payment_id
Payment date
Payment
Balance_amount
Personal Loan
Home Loan
Personal
Loan_no
Interest
rate
Home
Loan no
Interest
rate