

TED (10)-4070

Reg. No.

(REVISION—2010)

Signature.....

FOURTH SEMESTER DIPLOMA EXAMINATION IN ENGINEERING/
TECHNOLOGY—MARCH, 2013

DATABASE MANAGEMENT SYSTEMS

(Common for CT and IF)

[Time : 3 hours

(Maximum marks : 100)

PART—A

I. Answer the following questions in one or two sentences. Each question carries 2 marks.

1. Define Data and Attribute.

Data: Real world entity

Attribute: Represents some property of interest

Further describes an entity

2. List the types of Keys in Database.

Primary key

Super key

Candidate key

Foreign key

3. Define BCNF.

BCNF does not allow dependencies between attributes that belong to candidate keys.

BCNF is a refinement of the third normal form in which it drops the restriction of a non-key attribute from the 3rd normal form.

4. Name any four constraints in Relational Database.

1. Key constraints

2. Entity integrity constraints
3. Referential integrity constraints
4. Domain constraints

5. Write an example of SQL Query using GROUPBY

```
SELECT column_name, aggregate_function(column_name)
FROM table_name
WHERE column_name operator value
GROUP BY column_name;
```

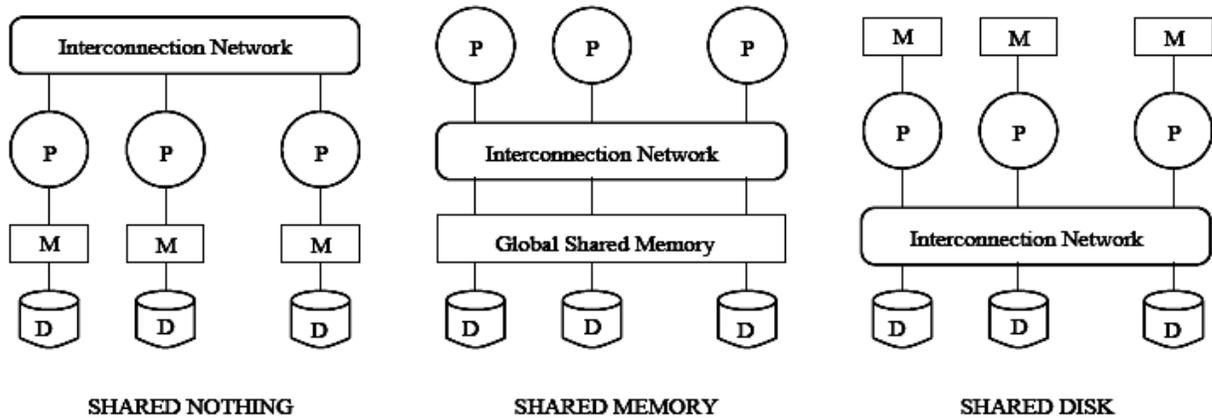
PART--B

II. Answer *any five* of the following. Each question carries 6 marks.

1. Explain the different architecture for building parallel database system.

Three main architectures are proposed for building parallel databases:

1. **Shared - memory system**, where multiple CPUs are attached to an interconnection network and can access a common region of main memory.
2. **Shared - disk system**, where each CPU has a private memory and direct access to all disks through an interconnection network.
3. **Shared - nothing system**, where each CPU has local main memory and disk space, but no two CPUs can access the same storage area; all communication between CPUs is through a network connection.



2. State the advantages of distributed database system.

Advantages of distributed database:

- 1) In a distributed database, data can be stored in different systems like personal computers, servers, mainframes, etc.
- 2) A user doesn't know where the data is located physically. Database presents the data to the user as if it were located locally.
- 3) Database can be accessed over different networks.
- 4) Data can be joined and updated from different tables which are located on different machines.
- 5) Even if a system fails the integrity of the distributed database is maintained.
- 6) A distributed database is secure.

3. A company stores the database :

(a) Employee details which include employee number, name, gender, date of birth, date of joining, address, phone number.

```
SQL>Create table Employee details(employee number int(4), name char(5),gender char(4), DOB char(10),DOJ char(10),address char(10),phone num int(10));
```

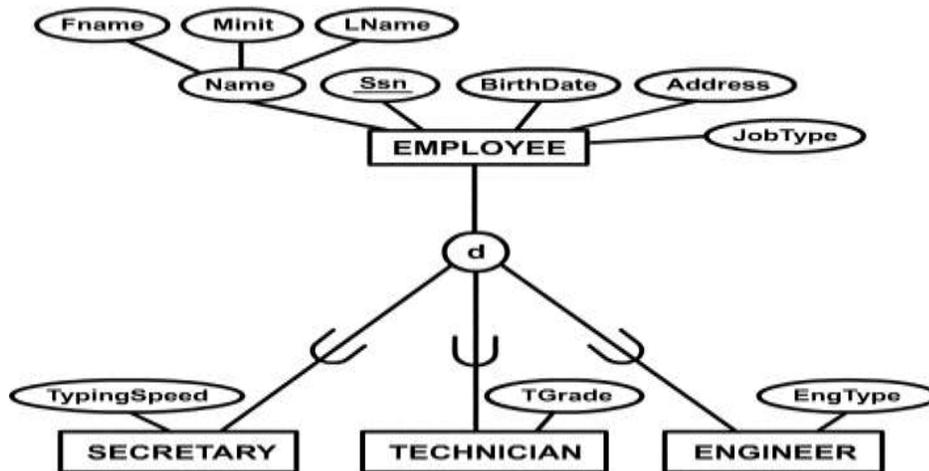
(b) Department details which contains department number, department name, department manager.

```
SQL>Create table Department details(dept_num int(4),dept_name char(5),dept_managr char(5));
```

(c) Two departments in the company are inventory control and customer service.

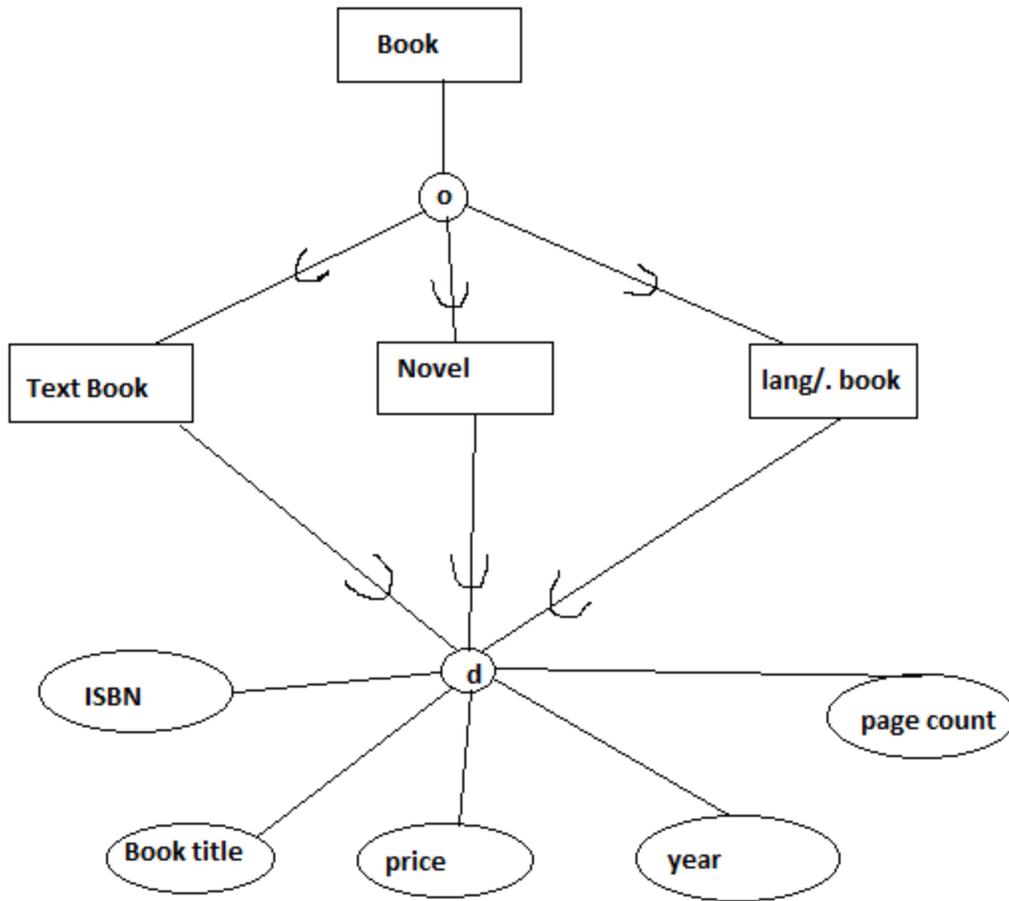
```
SQL>Select dept_name from department having dept_managr="inventory control and customer service";
```

Draw the EER diagram for the company database by showing the specialization of employee entity type.



4. Consider the book relation given below. Convert this relation into an ER diagram with generalization.

<i>ISBN</i>	<i>Book Title</i>	<i>Category</i>	<i>Price</i>	<i>Year</i>	<i>Page_Count</i>
001-1324	Understanding C	Text book	200	2000	200
001-987	Wings of Fire	Novel	325	2009	300
002-678	My experience with truth	Novel	400	1980	250
003-456	Introduction to foreign languages	Language book	375	1990	300
004-765	Delphi	Text book	290	2005	375



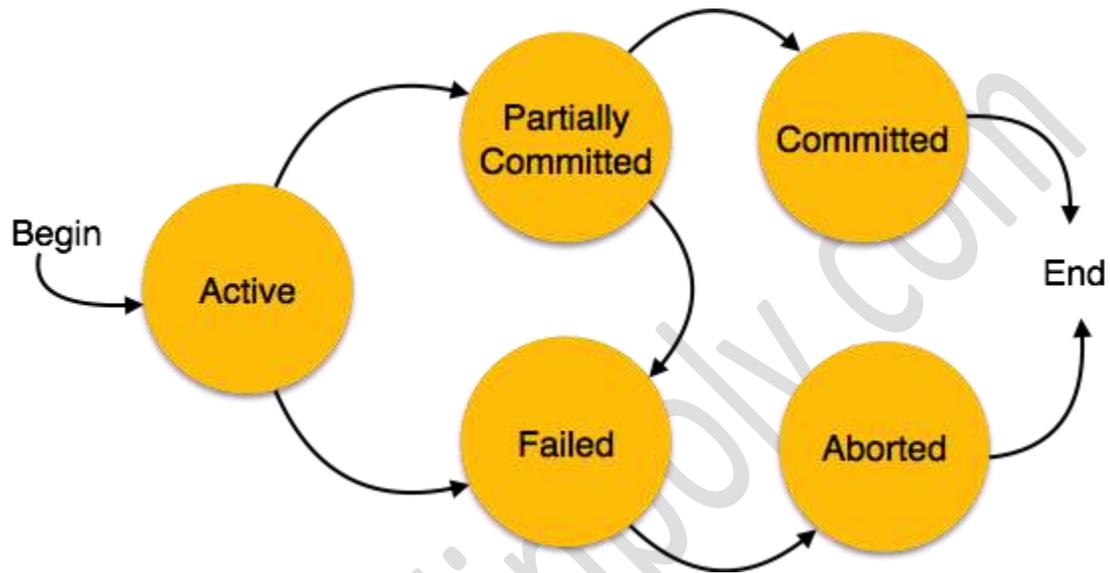
5. Consider the table given below :

Ledger_ID	Branch	Name	Address	Att_percent age
CST-001	Computer	Fiza	Sun city,Kochi	70
EST-002	Electronics	Saventh	New colony, Alappuzha	89
PST-003	Polymer	Lal	Southern plaza,thrissur	94
MST-004	Mechanical	Amit	Dev nagar, tvn	68

Is this table in 1NF or 2NF ? Justify your answer.

Table is in first normal form. A table is considered to be in 1NF if all the fields contain only scalar values (as opposed to list of values).

5. Arrange the states of transaction during the execution of DBMS.



transaction in a database can be in one of the following state:

- **Active:** In this state the transaction is being executed. This is the initial state of every transaction.
- **Partially Committed:** When a transaction executes its final operation, it is said to be in this state. After execution of all operations, the database system performs some checks e.g. the consistency state of database after applying output of transaction onto the database.
- **Failed:** If any checks made by database recovery system fails, the transaction is said to be in failed state, from where it can no longer proceed further.
- **Aborted:** If any of checks fails and transaction reached in Failed state, the recovery manager rolls back all its write operation on the database to make database in the state where it was prior to start of execution of transaction. Transactions in this state are called aborted. Database recovery module can select one of the two operations after a transaction aborts:
 - Re-start the transaction
 - Kill the transaction
- **Committed:** If transaction executes all its operations successfully it is said to be committed. All its effects are now permanently made on database system.

7. Write a program segment which uses cursor to update the mark of a student from the relation STUDENT by the value entered by the user. STUDENT relation contain the attributes STIID_ID, CLASS, NAME and TOTAL MARKS.

Sql> insert into STUDENT values (1, 10th, "VIJAY", 190);

1 row created.

Sql> insert into STUDENT values (2, 10th, "ANU", 180);

```
BEGIN
UPDATE student SET status = 'INACTIVE'
WHERE name='vijay';
IF SQL%FOUND THEN
DBMS_OUTPUT.PUT_LINE ('Marks found for 'vijay');
END IF;
IF SQL%NOTFOUND THEN
DBMS_OUTPUT.PUT_LINE ('No marks found for vijay ');
END IF;
IF SQL%ROWCOUNT > 0 THEN
DBMS_OUTPUT.PUT_LINE (SQL%ROWCOUNT ' ||
mark(s) inactivated');
ELSE
DBMS_OUTPUT.PUT_LINE ('No mark(s) inactivated');
END IF;
END;
/
```

PART-C

(Maximum marks: 60)

(Answer one full question from each unit. Each question carries 15 marks.)

UNIT-I

III. (a) Describe areas in which DBMS are applied.

Database Applications:

Banking: all transactions

Airlines: reservations, schedules

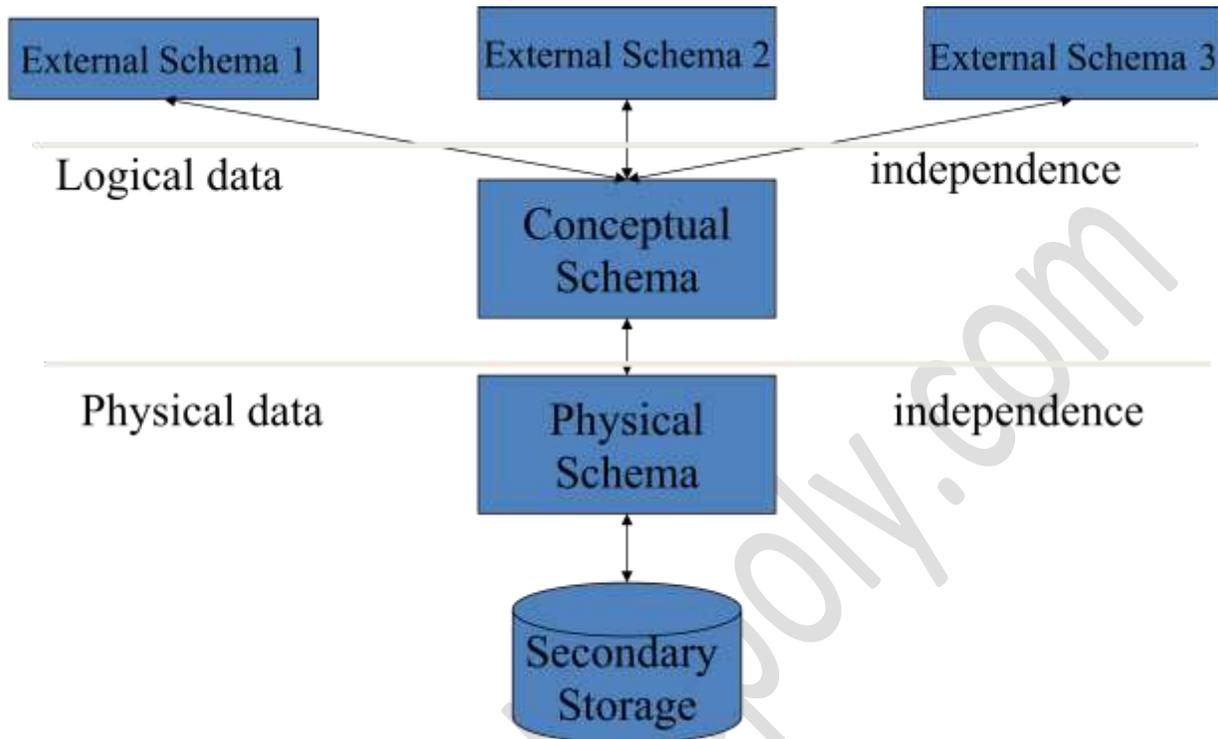
Universities: registration, grades

Sales: customers, products, purchases

Manufacturing: production, inventory, orders, supply chain

Human resources: employee records, salaries, tax deductions

(b) Explain the three schema architecture of DBMS citing an example.



■ Physical Schema

- Storage space allocation for data and indexes
- Record description for storage (sizes for data items)
- Record placement
- Data compression and data encryption techniques
- Conceptual (Logical) Schema

Example:

Book (*bid*: char(2), *title*: char(50), *author*: char(20), *price*: decimal(5,2), *av_q*: integer)

Customer (*cid*: char(2), *l_n*: char(25), *f_n*: char(25), *a_c*: char(30), *phone*: char(10))

Order (*i_n*: char(2), *bid*: char(2), *cid*: char(2), *n_i*: integer, *d_of_sale*: date)

■ External Schema

- Multiple external schemas
- In relational DBMS, External Schema contains views and relations from conceptual schema
- Reduce complexity of DBMS for users

- Support data security
- Support data independence

Or

IV (a) Distinguish between hierarchical, network and relational model of database.

Hierarchical :

Simple structure: The organization of an IMS database was easy to understand. The database hierarchy paralleled that of a company organization chart or a family tree.

- **Parent/child organization:** An IMS database was excellent for representing parent/child relationships, such as "A is a part of B" or "A is owned by B."
- **Performance:** IMS stored parent/child relationships as physical pointers from one data record to another, so that movement through the database was rapid.

The network data model:

extended the hierarchical model by allowing a record to participate in multiple parent/child relationships. For a programmer, accessing a network database was very similar to accessing a hierarchical database. An application program could:

- find a specific parent record by key (such as a customer number),
- move down to the first child in a particular set (the first order placed by this customer),
- move sideways from one child to the next in the set (the next order placed by the same customer), or move up from a child to its parent in another set (the salesperson who took the order).

■ **Relational Data Model**

■ Basic constructs:

- A relation
- A record
- A field (or a column)

■ Integrity constraints

■ Supports all types of relationships between two records which may change over time:

- 1:1
- 1:N
- N:M

(b) Define data dictionary.

data dictionary contains metadata—that is, data about data. The schema of a table is an example of metadata. A database system consults the data dictionary before reading or modifying actual data. **3**

(c) Differentiate the function of query optimizer, query processor and storage manager.

query optimizer	query processor	storage manager.
Query optimization is a function of many relational database management systems in which multiple query plans for satisfying a query are examined and a good query plan is identified. This may or not be the absolute best strategy because there are many ways of doing plans. There is a trade-off between the amount of time spent figuring out the best plan and the amount running the plan. Different qualities of database management systems have different ways of balancing these two. Cost based query optimizers evaluate the resource footprint of various query plans and use this as the basis for plan selection	Query processor takes care of arranging the underlying access routines to satisfy a given query Thus queries can be specified in terms of the required results rather than in terms of how to achieve those results	<ul style="list-style-type: none">• Storage manager is a program module that provides the interface between the low-level data stored in the database and the application programs and queries submitted to the system.• The storage manager is responsible to the following tasks:<ul style="list-style-type: none">▪ interaction with the file manager▪ efficient storing, retrieving and updating of data

6

UNIT-II

V. (a) State the advantages of ER model.

- Straight forward relation representation
- Easy conversion for E-R to other data model

(b) Define information.

Information is data that has been processed in such a way as to be meaningful to the person who receives it. It is anything that is communicated.

(c) Explain the different types of attributes.

- Composite attributes can be divided into smaller subparts, which represent more basic attributes with independent meanings.
- Attributes that are not divisible are called *simple* or *atomic* attributes.
- Attributes which have only one value for an entity are called single valued attributes.
- phone number attribute may have multiple values. Such values are called Multi-valued attributes.

Or

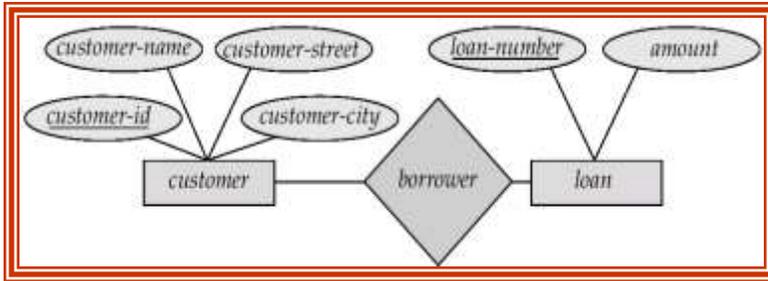
VI. (a) Devise the concept of ER Diagram'

ER diagram is widely used in database design

Represent conceptual level of a database system

Describe things and their relationships in high level

- Basic Concepts
- Entity set – an abstraction of similar things, e.g. cars, students
 - An entity set contains many entities
- Attributes: common properties of the entities in an entity set
- Relationship – specify the relations among entities from two or more entity sets
- **An Example**



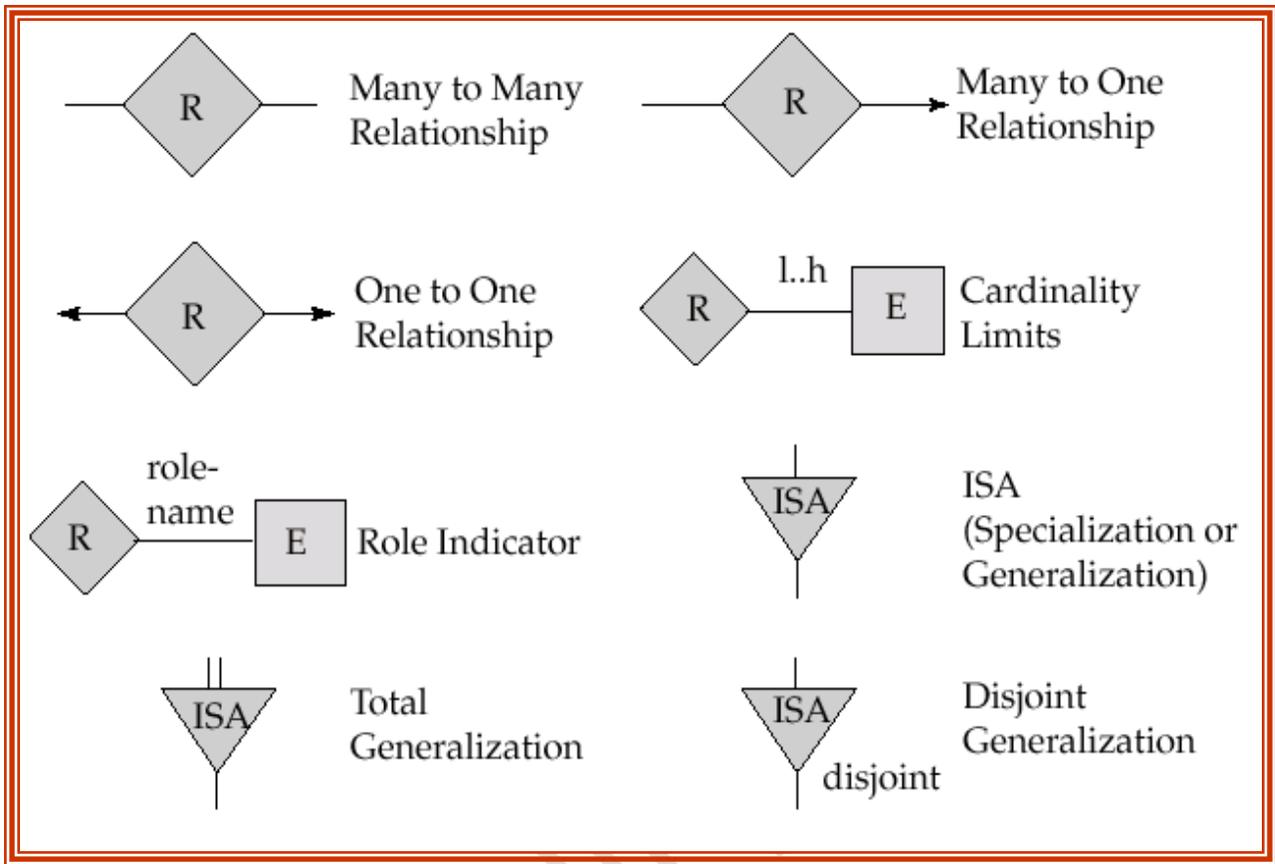
- A relationship may be thought as a set as well

- For binary relationship, it enumerates the pairs of entities that relate to each other

(b) Draw notations of ER diagram specifying the description.

Notation

	Entity Set		Attribute
	Weak Entity Set		Multivalued Attribute
	Relationship Set		Derived Attribute
	Identifying Relationship Set for Weak Entity Set		Total Participation of Entity Set in Relationship
	Primary Key		Discriminating Attribute of Weak Entity Set



UNIT-III

VII (a) Map the ER model into Relational model.

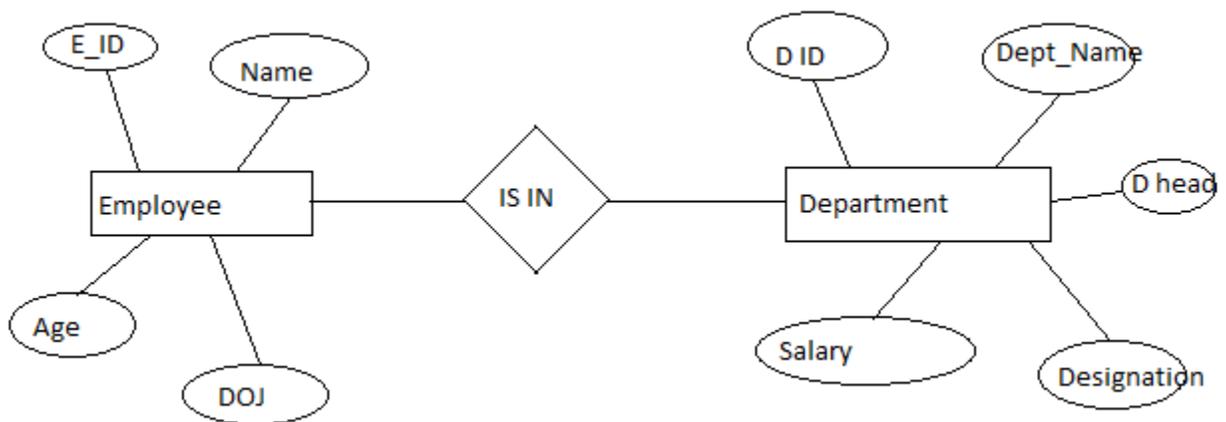


Table 'employee'

E-ID	Name	Age	DOJ
------	------	-----	-----

Table 'Department'

D ID	Dept_Name	D head	Designation	Salary
------	-----------	--------	-------------	--------

(b) Define Project-join normal form. Prove that a relation in the Project-join normal form is in 4NF.

- **Projection**
- Unary operation, selects columns
- Returned schema is different, so returned tuples are not subset of original set, like they are in selection

Eliminates duplicate tuples

Projection Example

Employee

SSN	Name	DepartmentID	Salary
999999999	John	1	30,000
777777777	Tony	1	32,000
888888888	Alice	2	45,000

SSN	Name
999999999	John
777777777	Tony
888888888	Alice

Or

VIII. (a). Discuss the anomalies due to the interleaved execution of transactions. Identify the anomaly in the following schedule:

T1	T2
read(A) A:=A-100	
	read(A) X:=A*0.02 A:=A+X
write(A) read(B)	
	write(A)
B:=B+100 write(B)	

- The following concurrent schedule does not preserve the value of T. Schedules: sequences that indicate the order in which instructions of concurrent transactions are executed.
- A schedule for a set of transactions must consist of all instructions of those transactions
- Must preserve the order in which instructions appear in each individual transaction

(b) Compare 3NF and BCNF with examples.

- **Third Normal Form (3NF)**

This form dictates that all non-key attributes of a table must be functionally dependent on a candidate key i.e. there can be no interdependencies among non-key attributes.

For a table to be in 3NF, there are two requirements

- The table should be second normal form

No attribute is transitively dependent on the primary key

Scheme \rightarrow {BuildingID, Contractor, Fee}

1. Primary Key \rightarrow {BuildingID}
2. {BuildingID} \rightarrow {Contractor}
3. {Contractor} \rightarrow {Fee}
4. {BuildingID} \rightarrow {Fee}
5. Fee transitively depends on the BuildingID
6. Both Contractor and Fee depend on the entire key hence 2NF

- **Boyce-Codd Normal Form (BCNF)**

- BCNF does not allow dependencies between attributes that belong to candidate keys.
- BCNF is a refinement of the third normal form in which it drops the restriction of a non-key attribute from the 3rd normal form.
- Third normal form and BCNF are not same if the following conditions are true:
 - The table has two or more candidate keys
 - At least two of the candidate keys are composed of more than one attribute
 - The keys are not disjoint i.e. The composite candidate keys share some attributes

Example 1 - Address (Not in BCNF)

Scheme \rightarrow {City, Street, ZipCode }

1. Key1 \rightarrow {City, Street }
2. Key2 \rightarrow {ZipCode, Street}
3. No non-key attribute hence 3NF
4. {City, Street} \rightarrow {ZipCode}
5. {ZipCode} \rightarrow {City}

Dependency between attributes belonging to a key

UNIT-IV

IX. (a). Explain embedded SQL with suitable example. Write the database connection query to connect the embedded SQL to a server named 'myserver' and connection name 'connectone'. The username will be 'user' and password '123456'.

EMBEDDED SQL:

The central idea of embedded SQL is to blend SQL language statements directly into a program written in a "host" programming language, such as C, Pascal, COBOL, FORTRAN, PL/I, or Assembler. Embedded SQL uses the following techniques to embed the SQL statements:

- SQL statements are intermixed with statements of the host language in the source program. This "embedded SQL source program" is submitted to a SQL pre-compiler, which processes the SQL statements.
- Variables of the host programming language can be referenced in the embedded SQL statements, allowing values calculated by the program to be used by the SQL statements.
- Program language variables also are used by the embedded SQL statements to receive the results of SQL queries, allowing the program to use and process the retrieved values.

```
String uri = "jdbc:oracle:www.bookstore.com:3083"
Connection connection;
try {
    Connection connection =
    DriverManager.getConnection(url,userId,password);
}
catch(SQLException excpt) {
    System.out.println(excpt.getMessageO);
return;
}
```

(b). Specify the use of indexes in SQL. Specify the query to create a unique index in the following table :

BOOK (ISBN, Book_Title, Price).

10

Index is an object which can be defined as the ordered list of values of a column or combination of columns used for faster searching and sorting of data. An index speeds up joins and searches by providing a way for a database management system to go directly to a row rather than having to search through all the rows until it finds the one you want. By default, indexes are created for primary keys and unique constraints of a table. However, creating indexes must be avoided on columns that are updated frequently since this slows down insert, update and delete operations. CREATE [UNIQUE] INDEX index_name ON table_name (column_name1 [ASC | DESC] [, column_name2 [ASC | DESC]].....)

CREATE [UNIQUE] INDEX BOOK_ind ON BOOK (ISBN[ASC]);

Or

X. Write the query for the following :

(a) Create a table 'BOOK' appropriate attributes having suitable data type.

SQL>create table BOOK(ISBN int(4), book type char(5), price int(4),category char(4));

(b) Add tuples to the table.

Sql> alter table BOOK add (page count);

(c) Retrieve how many tuples are in the table.

SQL>select * from BOOK;

(d) Retrieve the details of all books which having the price ranges from Rs 250 to Rs 500.

SQL>select book type from BOOK having price between 250 and 500;

(d) Retrieve the average price of books.

SQL>select AVG(price) from BOOK;

(e). Retrieve the number of categories of books.

SQL> select * from BOOK group by category;

(g) Retrieve the category of books where count is greater than 5.SQL> select category from BOOK where page count>5