



Database Management System

Assignment No. 1.

Q1. Differentiate between file system and database system with an example.

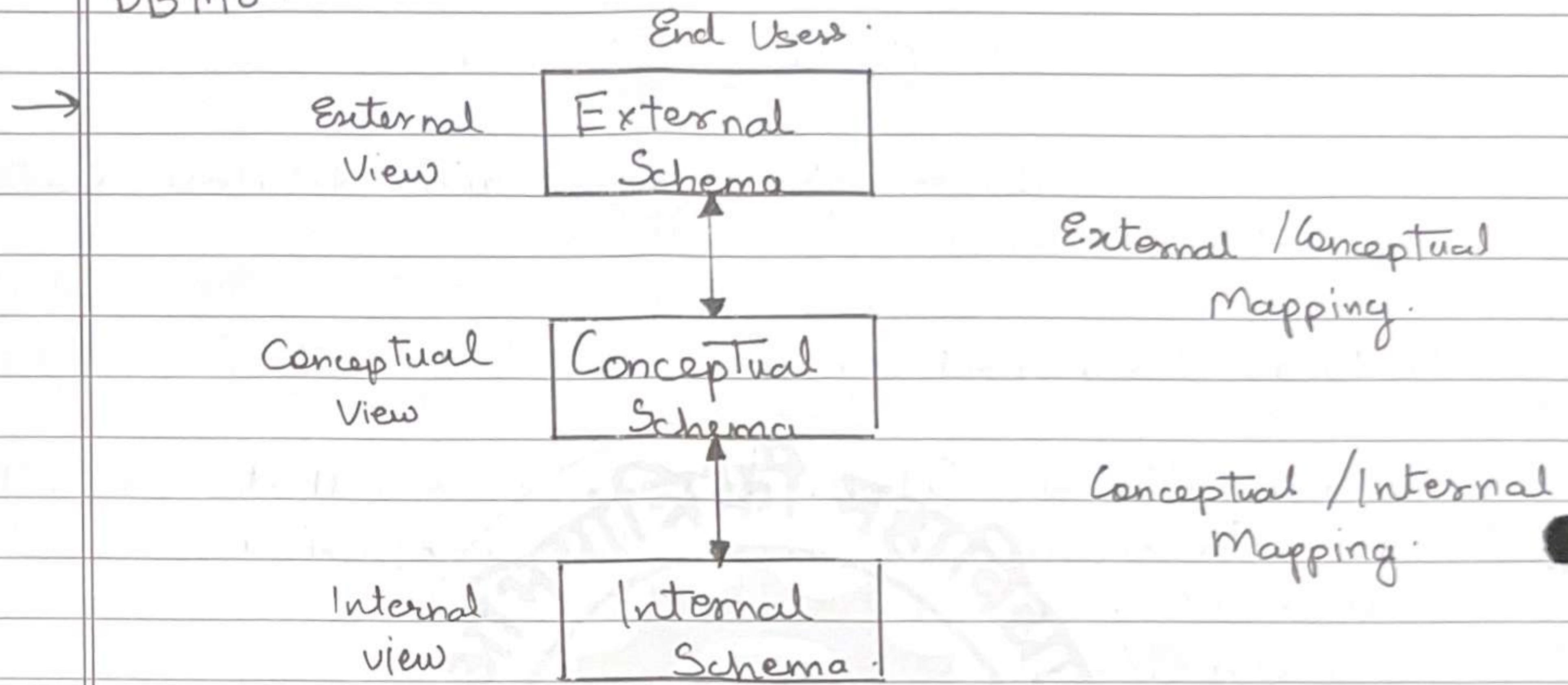
→ Database Management system

File System

- i) It's a collection of software helps in creation and management of databases i.e. hence it's a computerised record keeping system.
- ii) It allows flexible access to data.
- iii) It coordinates both the physical and logical access to data.
- iv) Unauthorized access is restricted.
- v) It has a multiple user interface.
- vi) Redundancy can be controlled.
- vii) For ex. MySQL is a platform where data is stored and accessed using queries in a space efficient and secure way, which can be accessed if the password is correct and provide backup.
- File System
 - It is designed to allow predetermined access of data.
 - It coordinates ~~at~~ only the physical access to data.
 - Data is unauthorized access can't be restricted.
 - Data is isolated enough so it doesn't have multiple interfaces.
 - Redundancy can't be controlled.
 - For ex. C++ can help you store data but accessing, updating and deletion of data is very difficult and not secure and doesn't provide backup.



Q2. Explain three level schema structure architecture of DBMS.



I. Internal level (Physical Level)

The internal level is very close to physical storage of data. This level describes the physical storage structure of the data in database. The internal schema at its ground level, it is stored in the form of bits with the physical addresses on the secondary storage devices. At the highest level, it can be viewed in the form of files and simple data structure.

The Internal schema defines the various stored data types and specifies what type of indexes exist, how stored fields are represented and so on. It uses a physical data model.

Ex. Create database Company;
Use Company;
(Create table Employee (Emp-no int , name text),



II Conceptual level:-

This describes the structure of the whole database for a group of users. The conceptual is also known as a data model. The conceptual schema hides the internal details of physical storage and targets on describing entities, data types, relationships and constraints. The conceptual schema contains all the information to build relevant external records, as the conceptual model is derived from physical model.

The conceptual schema is a representation of the entire content of database. It includes the definitions of various conceptual datatypes.

Ex :- Show databases;
Show tables;

III External Schema.

External view is a representation of the entire content of the database. The con-

External level is related with the way data is viewed by individual end users. The external level includes a number of user views. Each external schema describes the segment of the database that is required for a particular user group and hides the rest of the database from that user group.

External view are the proper interface between the user and the database, as an individual user can hardly be expected to be interested in the entire database. It's derived from conceptual schema. It consists of definitions of each of the various external data types in that external view.



Mapping :- The process of transforming requests between various levels of architecture called mappings. These could be time consuming. There are External \leftrightarrow Conceptual mapping and Conceptual \hookrightarrow Internal mapping.

Q3. Discuss the role of Database Administrator.

→ The database administrator is responsible for over-all planning of the company's data resources, design of data etc.

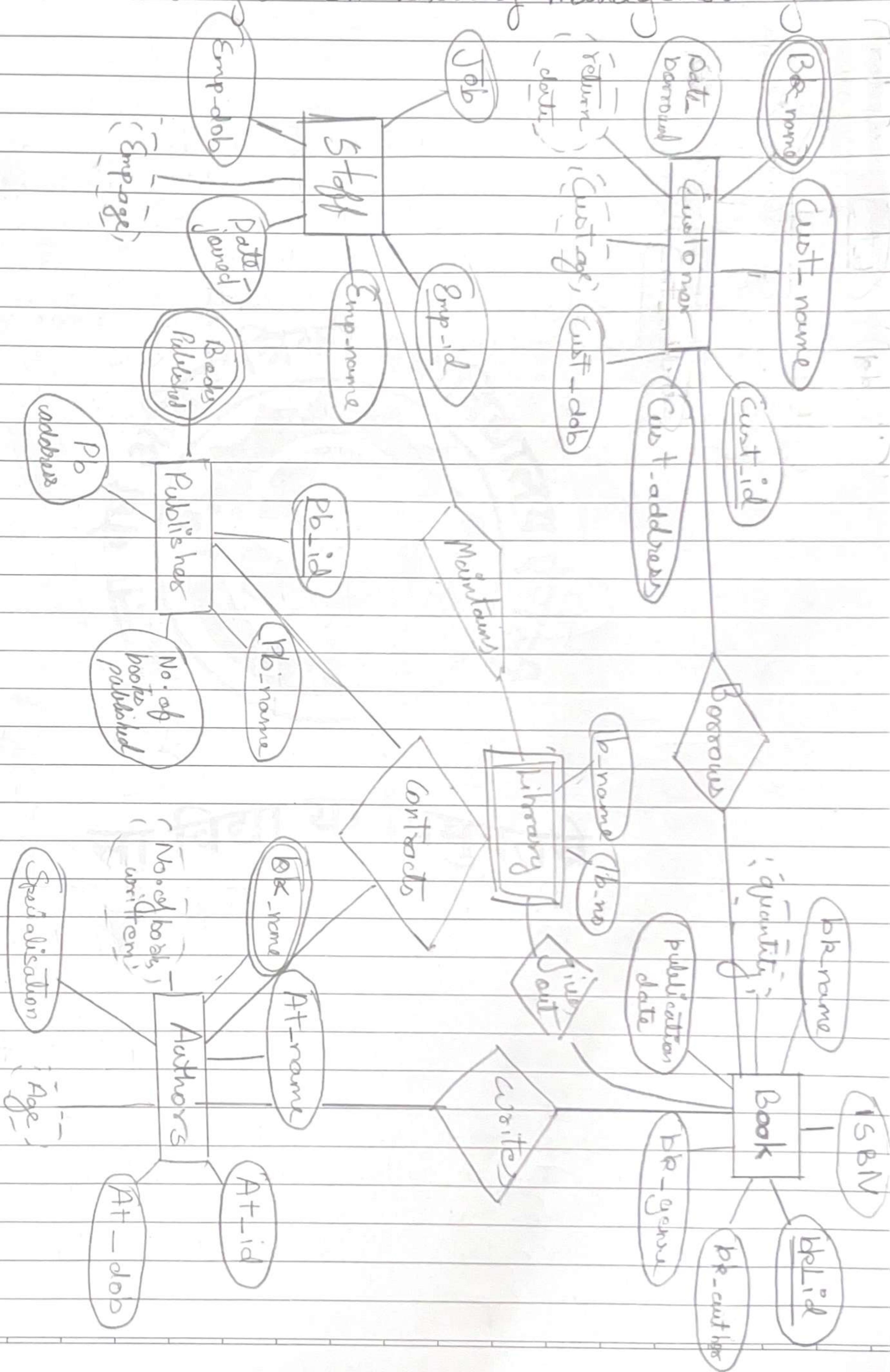
The role of Database Administrator is as following:

- * They need to perform many roles to keep the database up and running.
- * They have to administer or design the system.
- * They need to manage Database Management software as well as the server.
- * They are responsible for deciding on the storage and access methods.
- * They performs all data field updates or adding new fields into database.
- * They also have to be database developer or programme.
- * They writes the programs to design databases and to design the means of reorganising databases periodically.
- * They also determines and implements databases searching strategies.
- * They have to be system analyst.
- * They needs to analyse the system performance and fine tune the DBMS activities.
- * BBA needs to take care of system crashes by planning proper backup procedures.
- * He will also specify techniques for monitoring database performance.



Q4.

Draw an ER diagram on library management System.



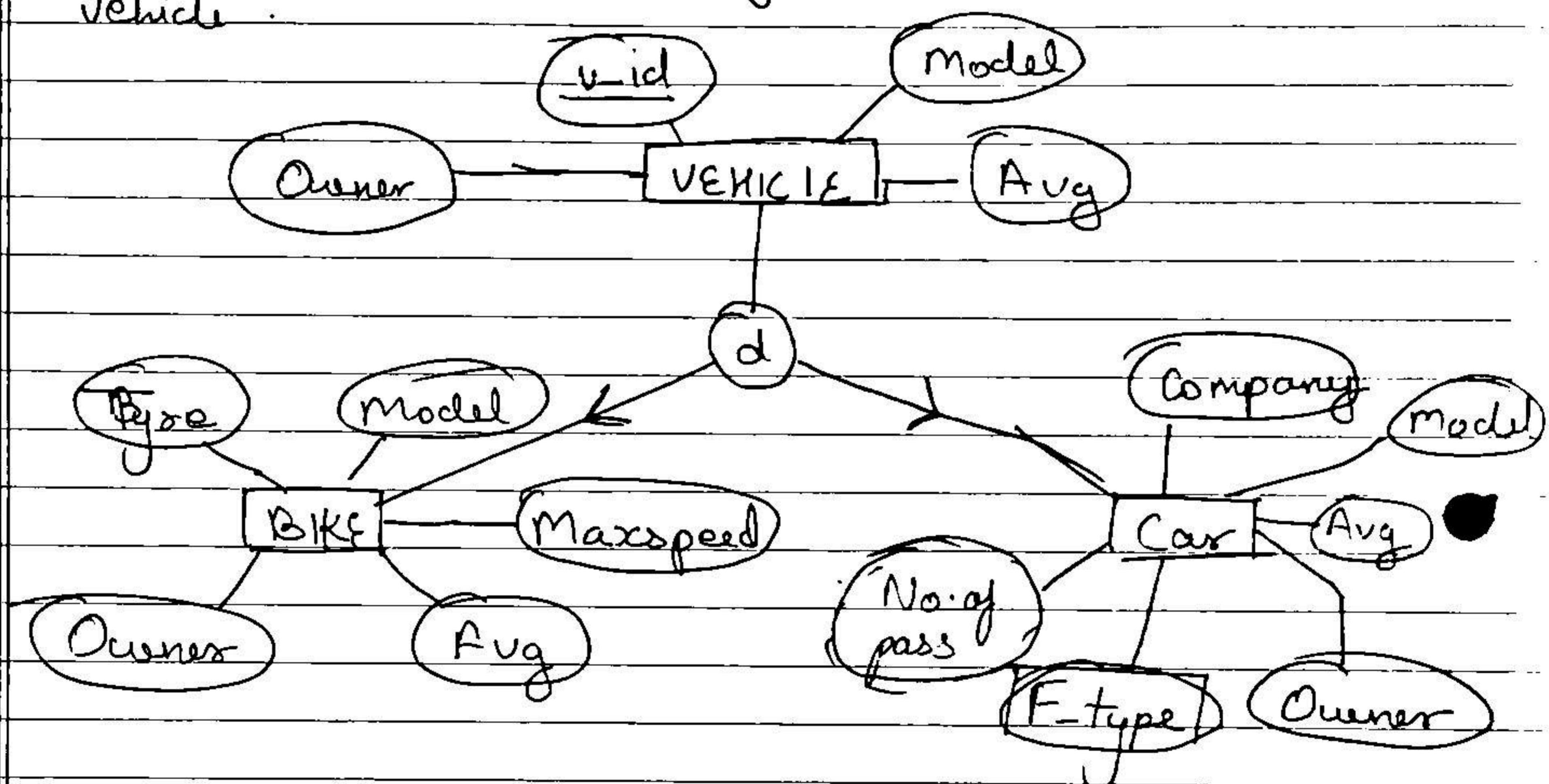


Q. Explain Generalization and Specialization with an example.

→ Generalization is a process in which we differentiate among several entity types identifying their common features and generalizing them to a single super class of which original entity type are special subclass.

This is a bottom up representation of superclass - subclass relationship.

For ex: Car and Bike all have several common attributes that can be generalized to a super class vehicle.



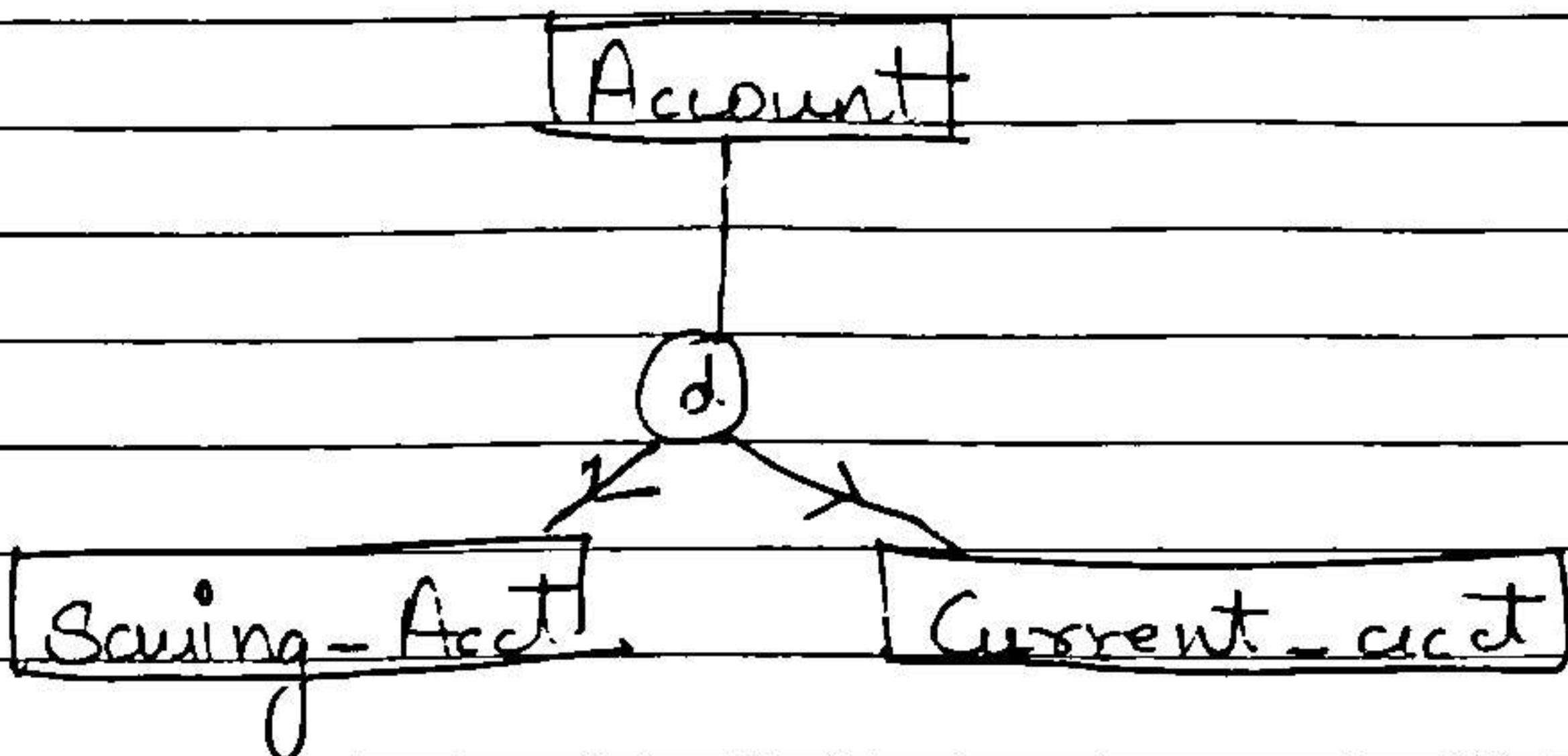
Generalized Vehicle Entity

Specialization is a process of defining set of subclass of entity type. This is called super class of specialization. The set of subclass transforms a specialization as defined on the basis of some distinguishing characteristics of entity in super class.



It's a top down approach of (super-sub) class i.e.

For ex. Set of sub-class (Saving - acc, Current acc) are specializations of Account Entity super class.



Specialization of account entity.

Q6. Explain any five relational algebra operators with an example.

→

Student

Exam

R-no	Name	Branch	P-no.	S-rol	Marks	P-grade
1	Aman	CS	3	M1	32	m
2	Baman	IT	2	P2	57	p
3	Chaman	Etey	1	C3	68	c
4	Daman	Mech	5	d.h4	79	ll
5	Eurman	Mech	4	N5	25	n

I Selection Operator

This operator is used to select rows from table which satisfy particular condition. It selects a set of tuples and the output is exactly same as input schema.

Syntax $(\text{attribute name} \rightarrow \text{comparison op} \rightarrow \text{constant}) \text{ (Table-name)}$



Ex. Query: Select all Student with the roll no. less than equal to 3

→ $\sigma_{\text{rno} \leq 3}(\text{Student})$

R.no	Name	Branch
1	Aman	CS
2	Baman	IT
3	Chaman	Elex

II Projection Operator (π)

This operator is used for selecting some of many columns in table to display in result set.

Syntax: $\pi_{<\text{column-name}>}(\text{Table-name})$

Ex. Query: Select Roll no and Marks.

→ $\pi_{\text{R.no}, \text{Mks}}(\text{Exam})$

R.no	Mks
3	32
2	57
1	68
5	79
4	25

III Rename operator. (ρ)

This gives alternative name to any attribute or any table of query expression.

Syntax $\rho_{<\text{new_name}>}(<\text{Old_name}>)$



Ex - Query : Find Name and Branch for Student :

$S \rightarrow \pi_{\text{st_name}, \text{st_branch}} (\sigma_{\text{st}} (\text{Student}))$

IV Cross Product (\times)

performs

A cross join, relational product of two entities specified.
In this every row in first table will be joined with
every row in second ~~condition~~ table. So finally the no. of
rows is equal to product of number of rows in 1 times.

b Syntax : (Expression 1) \times (Expression 2)

Query - Combination of First 3 Roll no. of student and
~~Highest Marks~~ marks above 50.

$\rightarrow (\pi_{R_{no} >= 3} (\text{Student})) \times (\sigma_{\text{marks} > 50} (\text{Exam}))$

R-no	Name	Branch	R.no	Stock	Mhs	P-Stock
1	Aman	CS	2	P2	57	P
1	Aman	CS	1	C3	68	C
1	Aman	CS	5	LL4	79	LL
2	Baman	IT	2	P2	57	P
2	Baman	IT	1	C3	68	C
2	Baman	IT	5	LL4	79	LL
3	Chaman	Elex	2	P2	57	P
3	Chaman	Elex	1	C3	68	C
3	Chaman	Elex	5	LL4	79	LL

V Difference Operator :

This operator finds all rows that are in 1 but not in 2.

Syntax (Expression 1 - Expression 2)



Ex. Query : Difference between Student and Marks -

→ (Students) - (Marks)

→ returns empty set

Q7. Define Super Key , Candidate key , Primary key and Foreign key .

→ SuperKey :- It is a set of those keys which can identify a tuple uniquely . Its role is to identify the tables of a specified table . All those attributes in a table which are capable of identifying the other attributes of the table in a unique manner are all super keys

Candidate Key :- It is a subset of super key where it doesn't contain any redundant value . They find table row or column uniquely as well as its value can't be ~~be~~ NULL .

Primary Key :- It is a candidate key selected to uniquely identify all other attributes values in any given order .

Foreign Key :- An attribute or combination of attributes in one table whose values must either match the primary key in another table or NULL .



Q8 Explain 4 DDL commands with example.

→ Some Data Definition Language Commands are:-

I Create :-

This is used to create database and tables etc.

Syntax :

CREATE Database <Name>;

CREATE TABLE <Name> (Col-1 datatype,
Col-2 datatype
:

Ex. Create Table Emp (Name text, Age int, Empid text);

II DROP :-

This is used to remove database, Table etc. and all of the data stored in it completely.

Syntax :-

DROP DATABASE <Name>;

DROP TABLE <Name>;

Ex; DROP TABLE Employee;

Show tables;

→ No table found.



III Rename Table:

This changes name of table with or without data at any point of time.

Syntax :-

RENAME TABLE <old-name> to <new-name>;

Ex: Rename table employeesoftheWorld to e;

IV TRUNCATE:

This deletes all the data from a table and is without removing the table itself.

Syntax:-

Truncate Table <Name>

Ex: > Truncate table c;

~~desc e < select * from e;~~
No data:

Q8

Q9. Explain 4 DML commands with an example.

Ex Student

S-id	Name	Class	R-no	Age
211	Aman	7	3	12
212	Baman	6	9	11
213	Chaman	9	5	14
214	Daman	10	20	15



Some Database Management Language Commands are:

~~I~~ Select :-

This displays records from the table, such as

Syntax: Select col-name-1, col-name-2...n from
table-name;
↓
where condition;

Exe. Select Name from Student where S_id = 213;

S	Name Charmain
---	------------------

U Insert :-

This adds the values into a table which should be
written to meet the data types and constraints given earlier.

Syntax : `INSERT INTO <Table-name>(<column-name_1, n>)
VALUES (val1, valn);`

without col name to insert for every column.

Ex. `Insert into Student(Age) values(16);`
→ OK, Query executed

Schett * from Student where age >14



S.id	Name	Class	R-no	Age
214	Daman	40	20	15
NULL	NULL	NULL	NULL	16

III UPDATE

It updates the existing data in table.

Syntax:-

UPDATE <Table-Name> Set [col1 = v1, col2 = v2 ... col-n = v-n]
[col1 = v1, col2 = v2 ... col-n = v-n]

Where Condition;

Ex. UPDATE Student Set [S.id = 215] where Age = 16;
Select * from student where age = 16;

S.id	Name	Class	R-no	Age
215	NULL	NULL	NULL	16

IV DELETE

It deletes the record where the condition has been met.

Syntax: DELETE FROM <table name> where
(condition)

Ex. DELETE FROM Student WHERE Age = 16.

Select * from student where age = 16.

Empty set.



10. Explain DCL commands with an example.

→ Some DCL commands are :- Data - Control

I GRANT :-

A system privilege is the right to perform a particular action or to perform an action on schema objects of a particular types. An authorised user can pass authorisation to others.

Syntax:-

```
Grant <ALL|privilege list>
ON <relation name>
TO <user|PUBLIC>
```

Ex:-

```
GRANT ALL
ON my_db,*
TO 'me@somehost';
```

II REVOKE:-

This removes privileges given to a particular user.

Syntax:-

```
REVOKE <ALL|privilege list>
ON <relation name>
FROM <user|role list|PUBLIC>
[RESTRICT | CASCADE]
```

```
REVOKE Create
ON *,*
```

```
from 'you@somehost';
```



11. Explain Views & Triggers/ with an example.

→ Views

A view is defined as a database object that allows us to create a virtual table in the database whose contents are defined by a query or taken from one or more table. View is defined to hide complexity of query from user. In view, instead of showing entire table we show only a glimpse of it.

Types of views :-

- Simple view :- The views which are based on only table called as simple view.
- Complex view :- The views which are based on more than one table called as complex view.

Table :-

Student

<u>Ex:-</u>	S_id	S-name	S-cr_no
	1	Aman	21
	2	Baman	25
	3	Chaman	27
	4	Daman	29

Create View h as select * from Student where S_id > 2;

Select * from h;

Sid	S-name	S-cr_no
3	Chaman	27
4	Daman	29



Triggers :-

A trigger is stored procedure in database which automatically invokes whenever a special event in database occurs. It gets triggered when a row is inserted into a specified table or when certain table columns are being updated.

E-C-A model

Event : Event may be insert, update or delete or

Condition : A condition that must be satisfied

Action : Snippet performed after the trigger is triggered.

Types of Triggers

- Row level Trigger :- It's fired each time the table is affected
- Statement level triggers :- A statement level trigger is fired only once on behalf of the triggering statement irrespective of the no. of rows in the table

Ex. Create trigger +

before/after insert .

ON Student

for each row .

Set • S_id=(set S_id+1));

Sid	S-name	R-no
2	Aman	24
3	Baman	25
4	Chaman	27
5	Daman	29
5	Easman	28



12. Explain Stored Procedure and Function with an example.

Stored Procedures

Stored Procedures can be defined as the set of Queries clustered in server. The users can refer from stored stored procedures so they don't have to write again and again. These can be used to perform opⁿ like Insert, Update, Delete. They enhances code reusability but also increases performance by reducing traffic as information sent over network is lessened.

It can return any value i.e. zero, single or multiple. It can return XML. Alter the records in database can be used as well as DML statements.

Types of Stored Procedures

- a) System Stored Procedures:- It is used to do informational tasks or admin level activities in SQL Server.
- b) User defined Stored Procedure:- It can be stored or compiled on SQL server. These are User stored procedure, user defined function and trigger.

Ex : Create PROCEDURE Get Student Name

()

@~~s~~ S-id int,
@ S-name varchar(100)
)

as begin select @S-name = Fn + " Ln from Student
where ~~str~~ s-id = @ S-id
END;



FUNCTIONS

Functions are routine that performs actions like complex calculations accept input parameters and return the result of that action as value. They return only a single value. They can't make any permanent alterations to SQL server.

Types of Functions :-

- a) System Defined Function: There are already defined in the system like round(), ceil(), upper(), lower(), etc and aggregate f^k like sum(), Avg(), Max(), etc
- b) User Defined Function :- Their are also 3 types-
 - i) Scalar - This gives a single value from the step of actions performed by function. It returns any data type value

Ex - Table "Student" has S-id, First-name, Last-Name, Class, etc etc but only Full-name is wanted.

Create Function f Get Fname (
@ F-name varchar(30),
@ L-name varchar(30))
returns varchar(60)

As

Begin return(Select @f-name + "@Last Name");
end.

∴ Calling

=> Select also. f Get Fname(First Name, Last Name)
as Name, Class from student.



(i) **Inline table-valued function :-** It gives the result as a table. It has no Begin/End body. Only a single select statement is used to derive the result.

Create function fSt()

Returns Table

as

return (Select * from Student)

→ Select * from fSt();

(ii) **Multi - Statement Table - valued function :-** It consists of more than one select statements or contains a select statement that can't be updated; then the result returned by that function is not updated.