

DBMS

Page No.:

Date:

youuu

Important Questions from Previous Year

Subject → Database management System (302)

15 Questions & Answers

Saurabh Kumar

BCA - 3rd Semester

Rollno - 61

L.S. College, Muzaffarpur

INDEX

S.No.	Question	Page No.
1.	ER model & example	01-03
2.	What is DBMS? Advantages and Disadvantages	03-06
3.	Database users, role & responsibilities	07-09
4.	Relational data model & its Structure	10-13
5.	Normalisation, Normal Forms	13-17
6.	Generalization, Specialization, Aggregation	18-21
7.	Data Independence	22-23

ER ETD

ER Diagram - Schools, Books, Shop

QUESTION

9. Database designing Process Q6-28

10. Short notes

- Foreign Key, data &

Information, Database - Q8-37
Auditing, Data EncryptionInstance & Schema,
Constraints11. Differences :- File vs DBMS
different types of key Q7-4012. different types of
relation b/w tables 41-44

13. Sol common questions 44-46

14. Three levels of data
abstraction 46-49

15. Log file 49-52

generation
human

Q. What is ER model? Describe in detail the basic concept of ER model with example.

(i) Dimensions & associated (ii)

Ans:- The ER model stands for Entity relationship model. It is defined by Dr. A Peter Chen in 1976. The ER model is high level conceptual data model which is used to define the data elements with their attribute and relationship for a special system. Basically, the Entity relationship model represents the whole database by entity, their attribute and relationship. It is the visual representation of data that describe how the data is related to each other. The ER model is very easy to design and understand.

* There are mainly three component of ER Model

i) Entity :- An entity is a thing or object in the real world which is uniquely identifiable. (Ex. - Teacher, Student, College)

ii) Attribute :- The attribute are the units that describe the

Characteristics or properties of an entity (ex - age, name, gender)

(iii) Relationship → Relationship is among associates. Or it links two or more entities.

There are 4 types of relationship

(a) 1 to 1 (one to one) → 1 to 1

(b) 1 to many → 1 to many

(c) many to one → many to 1

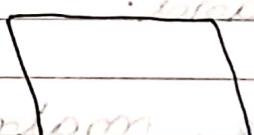
(d) many to many → many to many

Relationships are classified into two types

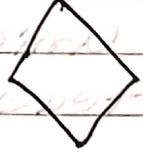
1. Representation → Representation

In ER modelling there are different components

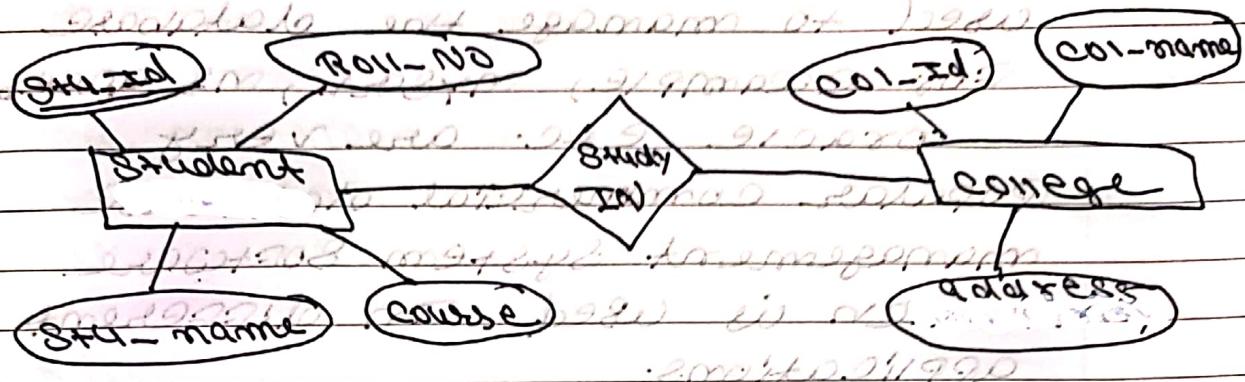
Entity is represented by different shape

Entity → 

Attribute → 

Relationship → 

Ques. Example:- (a) Draw ER diagram
ER model that describe the relationship between student and college.



Here Student and college is entity
and stu-id, Roll-no, collname etc.
are the attributes of these entity.

Q. What is DBMS? Q. What are the advantages & disadvantages of DBMS.

⇒ The DBMS stand for Database management System. The Software which allows to setup the database, Insert records into database, delete records from database, update the records into database, extract information or desire records and also protect the data. and many more operation over the data is called database management System.

In simple words we can say that the database management system is a software that is used to manage the database. For example, MySQL, MS Access, Oracle etc. are very popular commercial database management system software which is used in different applications.

Advantages:

i) Redundant data

The same DBMS are used by the multiple user so, the same records are not stored multiple time. It save storage and access time.

Data Consistency

⇒ If any data is updated in DBMS, the data is updated simultaneously for all the user at same time.

⇒ So, there is no inconsistency.

Data Security

⇒ The DBMS provide high security feature at different levels and different records.

⇒ It is very hard to damage data.

Easy to do operation on data

⇒ In DBMS all the data is stored at single place. So, it

is easy to do operation over the data in less time.

- v) **Support multiple view**
 - The DBMS supports multiple view. So, the multiple users can access the same data at same time from different computers.
- vi) **Easy to take backup**
 - The DBMS provides various tools to take backup of the data, and it is very easy because all the data are stored at one place. DBMS also provides auto backup tool.
- vii) **Easy recovery**
 - The DBMS keeps the backup of data so, it is very easy to recover all the data in case of any failure.
- viii) **Easy access**
 - The DBMS works processing software and all the data are stored at some places. So, it is very easy to access any data in less time.
- ix) **Simple view**
 - The DBMS gives simple, logical and graphical or various types of view according to the user requirement.

Disadvantages

(i) High initial Cost

⇒ To Implement the DBMS In an organization, high initial costs required to purchase specific software and hardware and establish network.

(ii) Well trained manpower needed

⇒ DBMS required a well trained manpower to maintain, manage and do operation over data. In albums, it is +20000. It is hard to find such skilled manpower.

(iii) Not Suitable for Small organisation

⇒ The DBMS is costlier and required a well trained manpower which is costly to hire. So, the small organisations have not afford that much costs.

(iv) Security threat

⇒ The DBMS store all the data at one place so, it is quite easy to hack all the data at same time. The hackers and crackers can easily crack the security and steal the data of the organisation.

Ques. Q. 19. If you want to implement DBMS in your organization, what are the problems?

3. Define different types of database user and their role and responsibility.

⇒ Q. answer all your parts

Ans:- There are many types of database users. The most common database users are as follows :-

(i) Database Administrator :- The DBA is a team of single person who define Schema and controls 3 levels of database.

The DBA is incharge of implementing the database system within the organisation. The main role of a DBA is to monitor the other users and database. It provide security to the database, control the database, add new user and remove old user.

It is also responsible for repair of the DBMS. In short we can say that it is the main person who is responsible for overall design and performance of the DBMS.

(ii) System Analyst ⇒ The System analyst is a user who finally tells the requirements of the end users that their requirement are satisfied or not. If requirement are not satisfied their responsibility is to improve the current system which can satisfy the requirements of the end users. If required, he creates a model.

(iii) Database designers :- The user who is responsible for designing the structure of database such as table, index, views, procedure, constraints etc. It controls how the data are stored in the database in a system. It is also responsible to train the new staff of website of the database and help general to understand it fast. It is application programmers who are in possession of proper knowledge of programming and DBMS. They write codes for the application. It helps developer to write application programs or user interface. The application programmer is also responsible for finding and debug the error of the software during use.

(iv) End users :- The end users are the basically those peoples who performs various operations like querying, updating and generating reports. They use the developed database application and they don't have knowledge about design and working of database.

Q1 There are mainly 4 types of end users

(i) Naive End users: The naive end users have the knowledge of DBMS and who do not have large knowledge about the DBMS but they frequently uses the database in their daily life to obtain desired output except queries in any language.

(ii) Casual End users: The end users who access or use the database occasionally and for short period of time. They require different information each time.

(iii) Sophisticated End users: The end users including engineers, scientists, business analyst etc. who have proper knowledge of DBMS and familiar to the database. These users try to learn most of DBMS facilities in order to achieve their complex requirement.

(iv) Standalone End users: The end users whose job is basically to maintain personal database by using a ready-made program package that provide easy to use menu base or graphical interface.

Q. Explain what is Relational Data model?

Illustrate the structure of Relational data model

Ans:- The relational data model is a

type of data model which

concerns organization

of information or data in

row form of rows or columns

in one or more tables. In

Relational data model each

table is connected to each

other on common field name

basis and there can be a

number of tables in the

database. It was proposed by

E.F. Codd for IBM in

1970 with 12 rules also

known as Codd's 12 Rules.

In short we can say that the

Relational database model is

the theoretical basis of

Relational database. It represents

how the data is stored in

Relational database. The DBMS

which is based on the

Relational data model is

known as Relational database

management system. It is

now part of most of the

Q. Draw a data table and define its various components in detail.

Page No.:	Date:
-----------	-------

Answer:

Ex- Example → Student (22 x 7)

1000 200 50 20 100 200

ID	Name	Course	Year	City	Mobile no.
201	Sonam	B.C.A	3	Pune	7312456170
202	Selmi	B.Com	2	Ratna	920485611
203	Aditya	B.B.T	2	Ranbir	840141486
204	Mukesh	B.E	1	Banglore	99325412
205	Rakesh	B.A	1	Ratna	899683210
206	Pratik	B.B.A	3	Muzaffarpur	871238698
207	Aakash	B.Sc	2	Bhopal	728698012
208	Aditi	B.C.A	1	Gizmoring	9832561090

100 200 200 50 20 =

→ Structure of Relational data - model.

100 200 200 50 20 =

→ In Relational data model there are several term which represent the different things of relational data model. These are:- i) AT STUDENT

100 200 200 50 20 =

ii) Table :- A Table is the combination of rows and column which represent the data and their relation.

100 200 200 50 20 =

iii) TUPLES:- The set of information of each entity with their attribute is called TUPLE. Each row in a table is tuple. 100 200 200 50 20 =

100 200 200 50 20 =

iv) Attribute:- Each column in table represent the attribute or properties. Each attribute column contain similar type of data (e.g - Rollno)

(iv) Cardinality: It is represented by the total no. of records, i.e., number of rows in the table.

Primary key: It is the key which the attribute contains which is different from all other attributes. Using this key we can easily identify the each row.

Super key: It is a primary key plus some other attributes.

Super key is needed for defining the degree of the table.

— The degree of any table represent the total no. of column or attribute present in the table.

(v) Domain: The domain is set of allowable values for the attribute. It is pre-defined value scope.

Normalised data is stored. \therefore NOT

(vi) Relation Schema \rightarrow The Relation Schema represent the name of the table with their attributes.

$\text{Relation Schema} = \text{tablename}(\text{attribute}, \text{attribute})$

(vii) Tuple variable \rightarrow It is the data store for a particular record / Section of the table.

(viii) Table name \rightarrow It is the name of the table by which each table can be uniquely identifiable.

Example: 1988 student merit 2021

SQL: 20 queries about student merit

DB schema of Student → table name
(2.14.2020) ~~student~~ ~~student attribute~~

Key	Roll	Name	Course	degree = 3
1	1	Ranu	B.C.A.	swami
Cardinality	2	babu	B.B.A.	→ tuple
⇒ 6.2	3	mukund	B.com	all
	4	Santosh	B.A	
	5	bittu	B.Sc	soft
	6	I sawabh	B.C.A	all

Tuple Variable:

any Table

Domain for Roll → five (0 to 100) (int)

Relation Schema = Student(Roll, Name, Course)

5. What is process of normalization? Discuss any three normal forms with example.

example: you want to add DR

Normalized out over 3 tables

→ Normalization is the process of organizing the data in the database to minimize redundancy

From a relation or set of relations.

The normalization is used to eliminate the undesirable characteristics like insertion, update and deletion anomalies.

Basically, it divides the larger table into smaller tables and

link them using relationship.

The main purpose of the normalization is to remove or eliminate redundant (repetitive) data in the database and ensure data is stored logically. The normalization process is also introduced by E.F. Codd.

E.F. Codd is +

There are mainly 7 types of normal forms.

- (i) 1NF
- (ii) 2NF
- (iii) 3NF
- (iv) BCNF
- (v) 4NF
- (vi) 5NF
- (vii) 6NF

Three normal form with example.

(i) 1NF (First Normal Form)

\Rightarrow A relation will be 1 NF if it contain only atomic value. The attribute of a table cannot hold multiple values. It must hold only single valued attribute. The 1NF disallows the multi-valued attribute, composite attribute and their combinations.

In 1NF each column store value must be of same type.

Example in 3NF

Student

Student

Roll	Name	Subject
1	Arun	C, C++
2	Vikash	Java
3	Aditi	C, Java

1NF

Roll	Name	Subject
1	Arun	C
2	Arun	C++
3	Vikash	Java
3	Aditi	C
3	Aditi	Java

(Candidate key) student

student

Here Roll no. 1 and 3 have 2 subjects
 So, there Subject attribute is
 multivalued So, we convert it into
 atomic value by storing them as
 different tuples.

(ii) 2NF (Second Normal Form)

→ Any relation is said to be in 2NF
 If the relation is in 1NF and
 It should not have any partial
 dependency i.e. no non-prime
 attribute (attribute which are not
 part of candidate key) is dependent
 on any proper subset of any candidate
 key of the table. or [All non-key
 attribute are fully functional
 dependent on Primary Key.]

Ex Student_detail

STUDID	COURSENOS	COURSEFEE
101	C1	1000
304	C2	1500
809	C3	1800
009	C4	1200

Here course no.
 is dependent on
 Student id and
 course fee is
 dependent on course no.

So, the course fee is dependent on course no. which is proper subset of candidate key.

To convert the relation into 2NF we need to split the table into two tables

TABLE 1. - Student (SId, courses)

TABLE 2 = Course (courseNo, courseFee)

Student		Course	
SId	courses	courseNo	courseFee
1	C1	C1	1000
2	C2	C2	1500
3	C3	C3	1800
4	C4	C4	1200

3NF (Third Normal Form)

⇒ A Relation is in third normal form if it is in second normal form (2NF) and there is no transitive dependency for non-prime attribute. We change a table into 3NF by removing transitive function dependency of non-prime attribute on super key.

Example.

Employee.

EID	Ename	E2ZIP	ESTATE	ECITY
Q22	Ranul	843328	Bihar	Muz
Q23	bittu	843326	Bihar	Sitamukhi
304	Santosh	842000	Bihar	Patna
984	lively	820010	U.P.	Vermesi
678	Rakesh	600123	M.P.	Bhopal

Here ESTATE and CITY are dependent on E2ZIP and E2ZIP is dependent on EID. The non-prime attribute ESTATE, ECITY is dependent on EID. It violates the Rule of 3NF.

So, we convert it into two table

TABLE 1 - EmployeeDetail (EID, Ename, E2ZIP)

TABLE 2 - EmployeeZIP (E2ZIP, ESTATE, ECITY)

EmployeeDetail and Employee ZIP

EID	Ename	E2ZIP	E2ZIP	ESTATE	ECITY
Q22	Ranul	843328	843328	Bihar	Muz
Q23	bittu	843326	843326	Bihar	Sitam
304	Santosh	842000	842000	Bihar	Patna
984	lively	820010	820010	U.P.	Vermesi
678	Rakesh	600123	600123	M.P.	Bhopal

Q. How Generalization & Specialization
is reverse to each other? Also
describe aggregation.

Ans. Describing all three topic:-

i) Generalization:

⇒ The Generalization is the process
of combining several lower
level entities set to produce a
one higher level entity set.

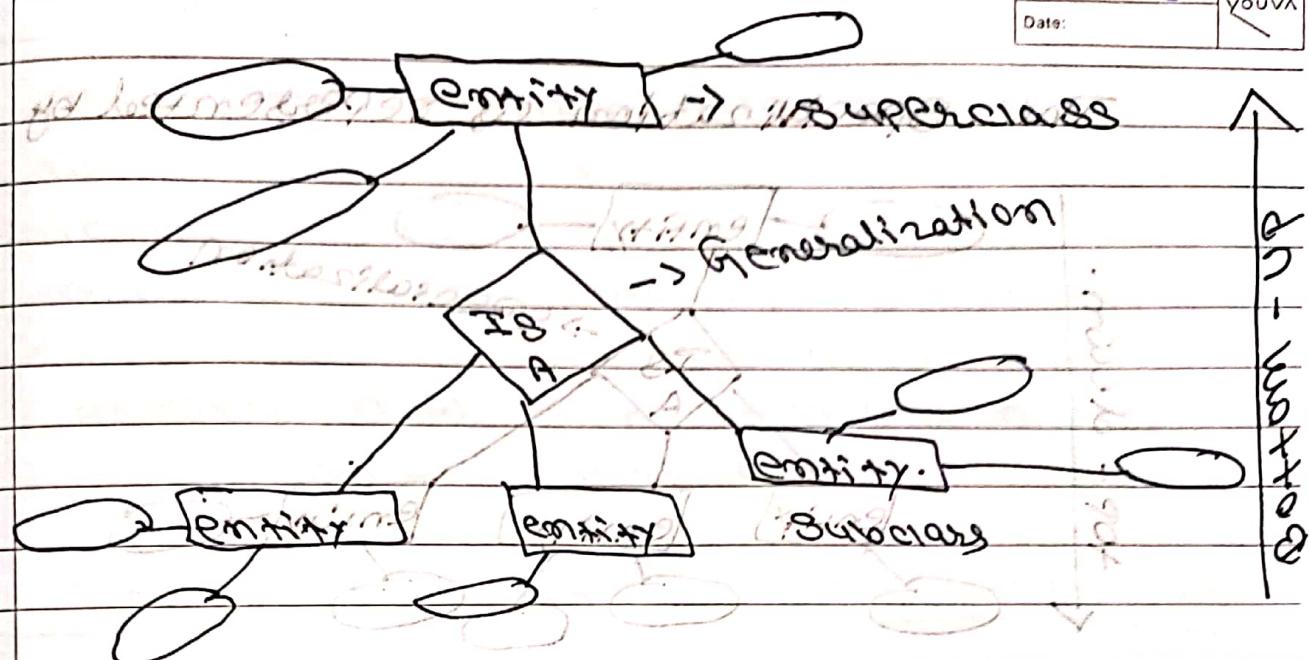
It is a bottom-up approach
in which two or more lower
level entities can be generalized
to a higher level entity, if
they have some common attribute.

The entity which are combined
together is known as subclass
and the entity which made
after the combination of
subclasses is known as superclass.

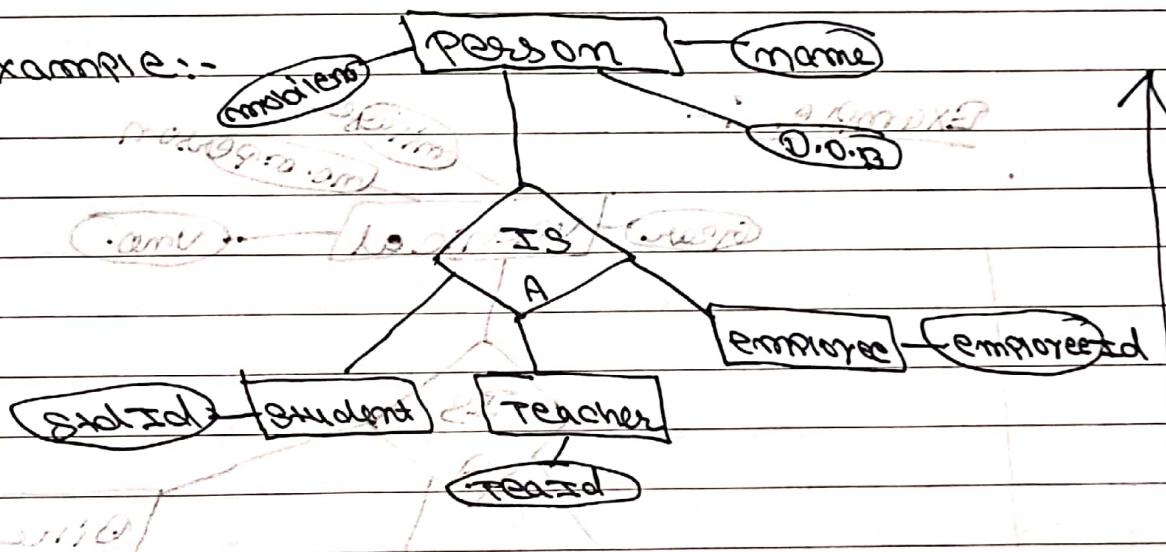
It is used to emphasize or
produce similarity among
the lower level entity and
to hide differences b/w the
lower level entities.

The Generalization is represented
by :-



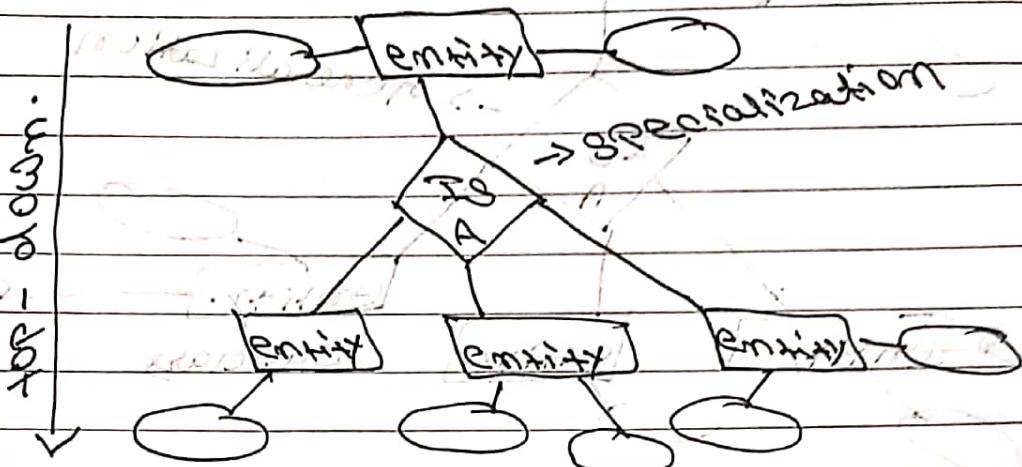


Example:-

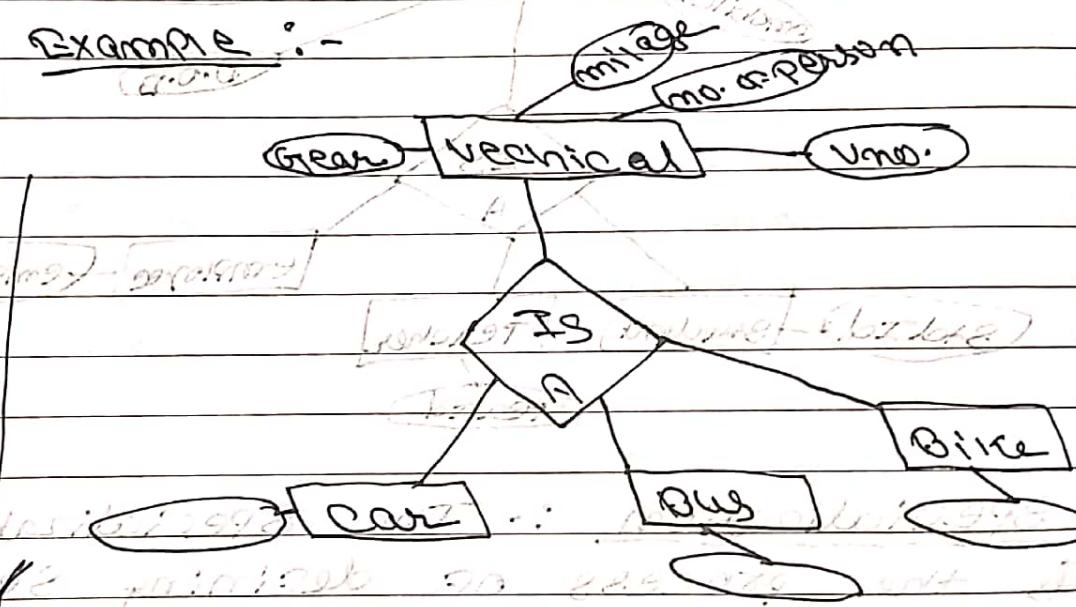


Entity Specialization :- The Specialization is the process of defining several lower level entity set from a high level entity set. Basically, Specialization is the reverse process of Generalization. It is also known as Top-down approach which is used to identify the subset ^{entity} from a subset entity that share some distinguishing characteristics. In Specialization, an Superset entity is broken down into two or more subset entity.

The specialization is represented by



Example :-

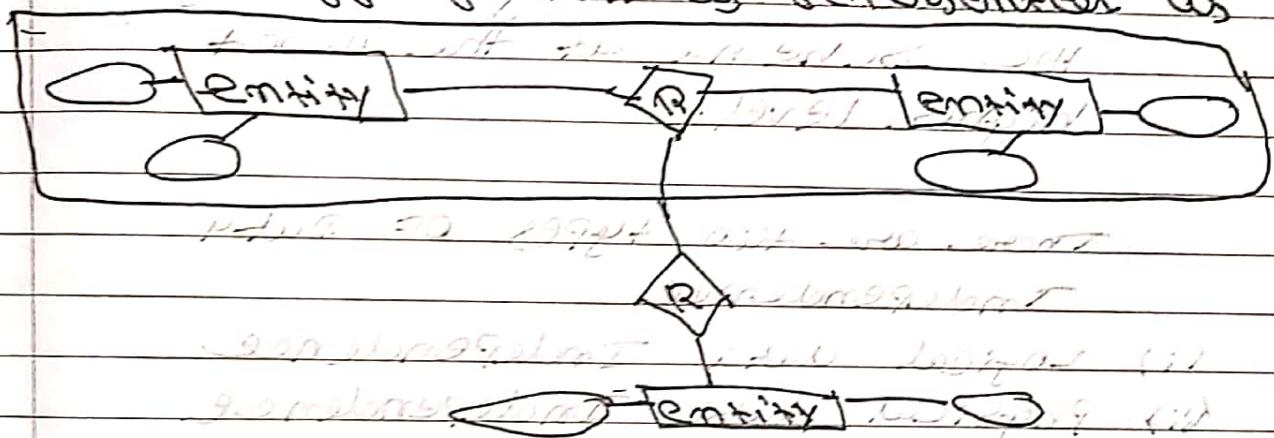


Aggregation \Rightarrow The aggregation is the process of combining relationships within a relationship. In ER diagram, it is difficult to represent the nested relationship. To overcome from such problem, the concept of aggregation is introduced. In the

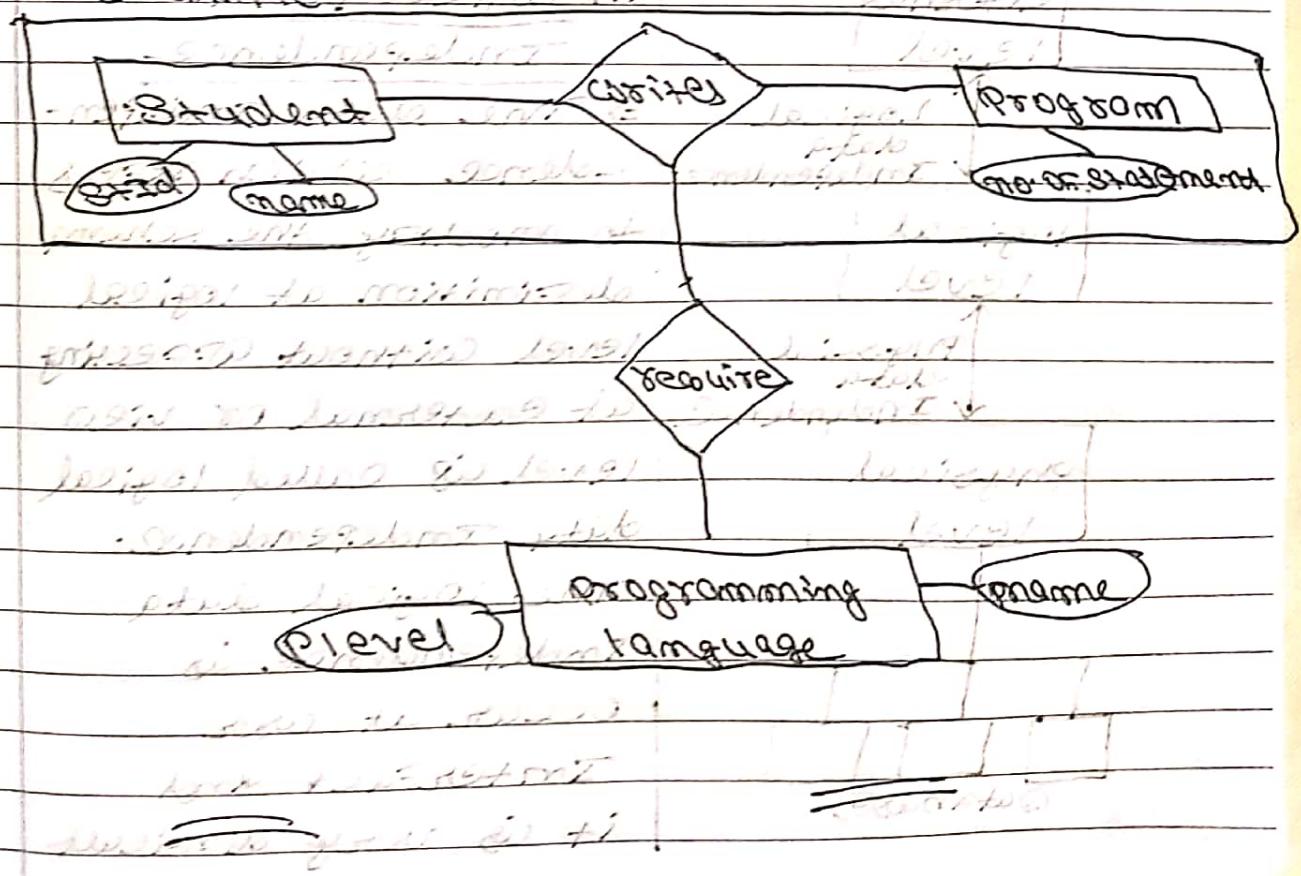
aggregation relationship b/w two entity is treated as a ~~single~~ ~~entity~~ higher level single entity. In aggregation one type of relation is grouped together to be as a single entity and then represent a relation with another entity.

(M.9 + P.9) points to 19.9.1

The aggregation is represented as



Example:-

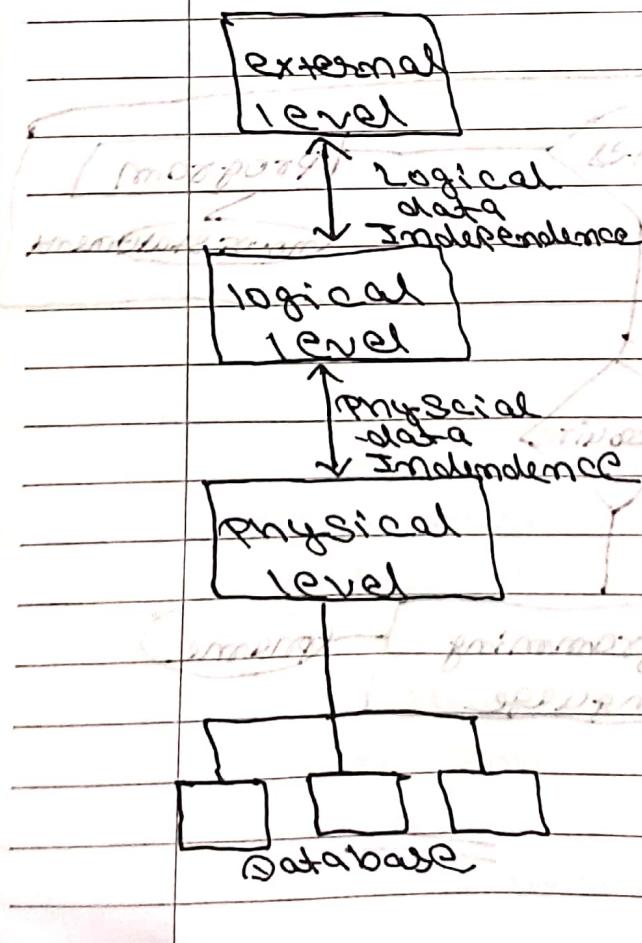


What is Data Independence in DBMS? Describe it.

⇒ The data independence is defined as a property of DBMS that helps you to change the database schema at one level of database system without requiring to change the schema at the next higher level.

There are two types of data independence

- (i) logical data independence
- (ii) physical data independence



(i) Logical data independence

⇒ The data independence which refers to modify the schema definition at logical level without affecting at external or view level is called logical data independence.

The logical data independence is occur at user interface and it is very difficult

to achieve because the application program are highly dependent on the logical structure of the database. The logical data independence is used to separate the external level from the conceptual view.

Example :- (i) Add / modify / delete a new attribute, entity or relationship.

(ii) merging of two records into one

(iii) Breaking an existing records into two or more records.

(iv) Physical data independence :-

⇒ The data independence which refers to modify the schema definition at physical level without affecting the schema definition at logical level.

The physical data independence helps you to separate conceptual level from the internal / physical level. compare to logical data independence, it is easy to achieve.

It occurs at the logical interface level. Any physical change like using new storage device, changing the data structure, altering index etc. is considered as the physical data independence.

Example :- (i) Using / change a storage device. (ii) modify the file organisation technique in the database (iii) Change the location of database from one drive to another drive.

8. What is ER Diagram? Draw the ER Diagrams of the following:-

(a) School Management System.

(b) Bank Management System.

Ans:- The ER Diagram stands for Entity Relationship Diagram. The ER Diagram is a type of flowchart which is used to represent an entity & their attribute and how entities relate with each other. The ER Diagram helps to explain the logical structure of the database in easy way. So, that they can be also easily understood by the non-technical users.

There were different types of Geometrical Shapes and Symbol used to represent the different part of the ER diagram like oval for attribute, rectangle for entity etc.

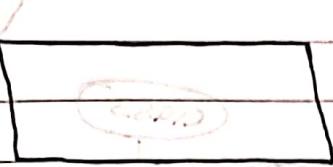
Notation for ER Diagram.

There are many type of shape and symbols are used to draw a ER Diagram. The most common basic symbols of ER Diagram are :-

Entity :- An entity is a thing or object which has its own properties.

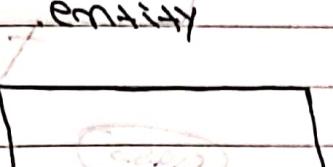
(i) entity or Strong entity

entity

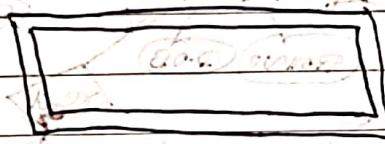


(ii)

weak entity



(iii) composite entity



(iv) Attribute

Attribute

(vii)

relationship

(ix)

weak relationship

(x)

1:1 relation

(xi)

1:m one to many

(xii)

m:1 many to one

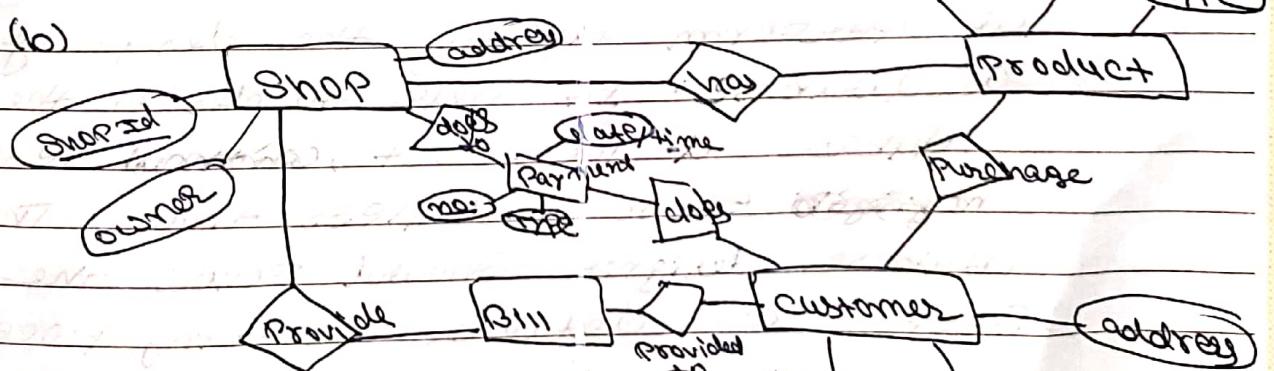
(xiii)

m:m

(v) key attribute

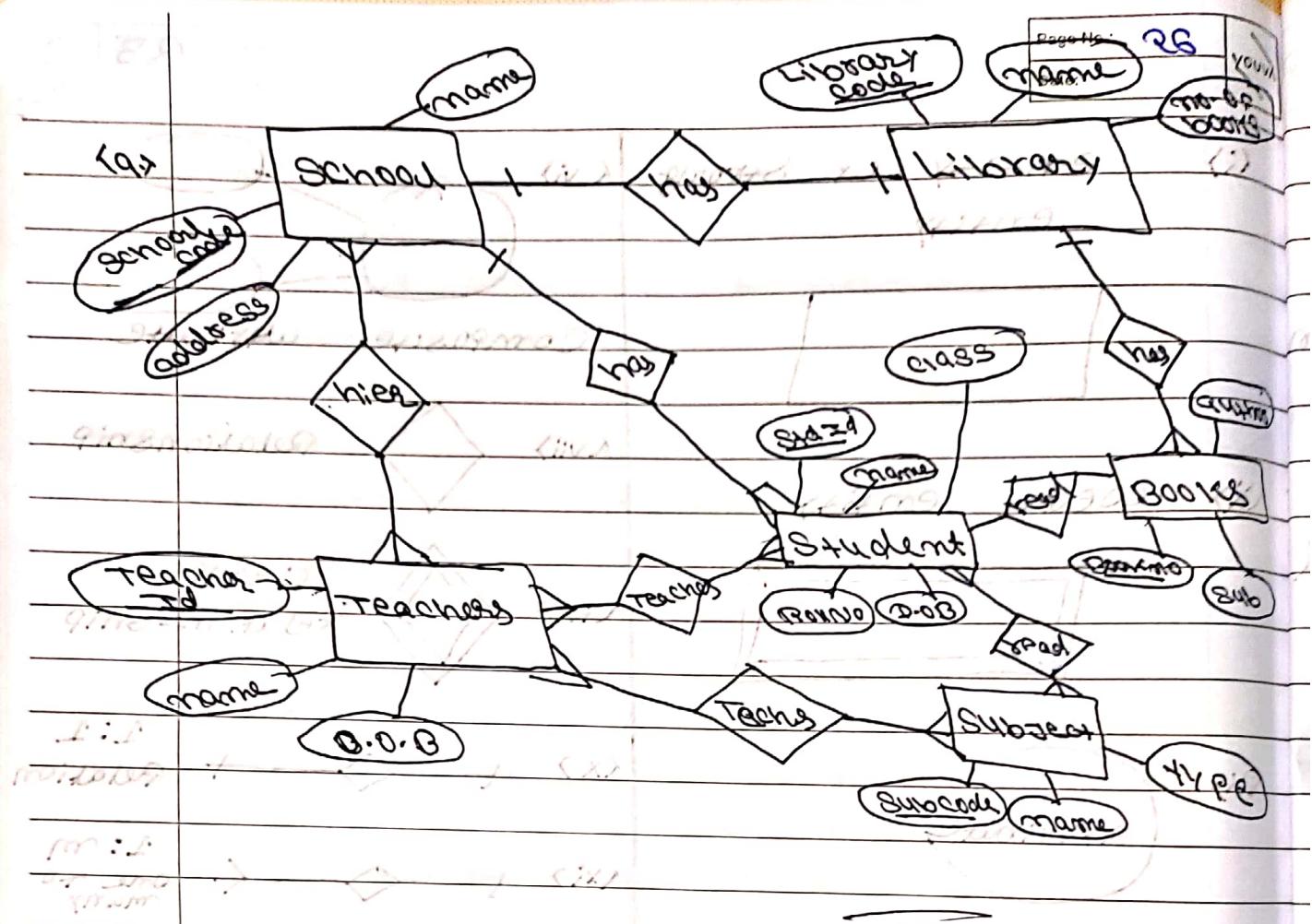
key attribute

attribute



Customer Shop ER diagram

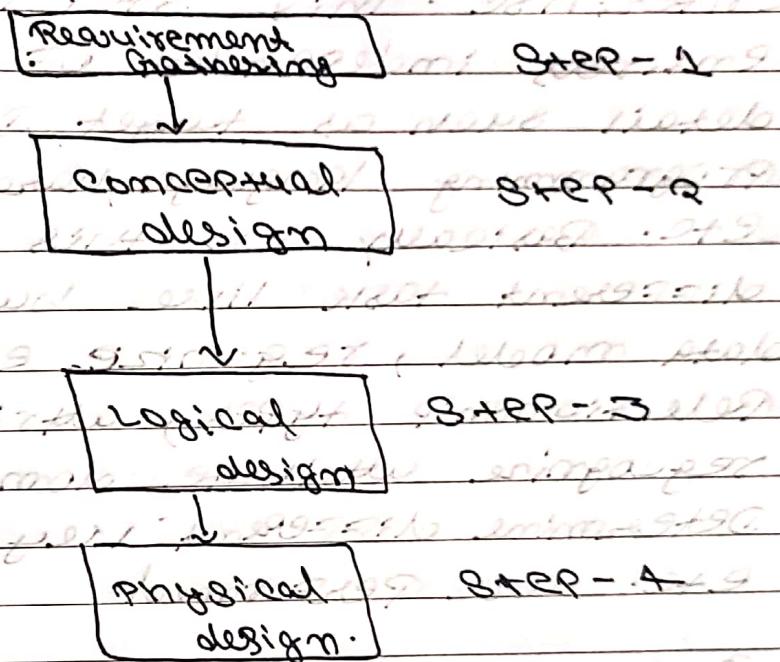
Customer no. 1001
Name: John Doe
Address: 123 Main St
Phone: 555-1234
Email: john.doe@example.com



Q. Describe the step by step process used while database designing process.

In database designing process, collection of task which is perform during the designing a database. It help to design the database at low cost, less error and less time. The database designer should follow the steps of database designing process to design a proper database. The main objective of the database designing process is to produce physical and logical design model of proposed database system.

The Database designing process involves mainly 4 steps:-



~~Step 1~~ Requirement Gathering & Analysis

⇒ It is the first step of database designing process. In this step different types of analysis and feasibility is performed. The different required material and the demand of the user are gathered. In this step when the requirement gathering & analysis is over the next step of the database designing process can start.

~~Step 2~~ Conceptual design

⇒ It is the second step in the database designing process. The conceptual database design phase

Start with the formation of a conceptual data model of the enterprise. This step is entirely independent of implementation detail such as target DBMS, programming language, hardware platform, etc. Basically in this step different task like build conceptual data model, recognize entity types, relationship types, attribute, recognize attribute domains, determine different key attribute etc. are performed.

~~Step 3~~

Step 3: Logical design: It is the 3rd step in database designing process. The main purpose of this step is to translate the conceptual data model into a logical data model then authorize this model to check whether it is structurally correct and able to support the required transaction or not. In this step different tasks like Authorize relation using normalization, check integrity control and its limitation, validate those relation against user transactions, evaluate logical data model with users are performed.

~~QUESTION~~ Physical design \Rightarrow It is the 4th and last step in database designing process. In this step the logical database design is converted or translated into Physical database design by listing them, target chosen DBMS. In this step suitable storage space is also selected. Basically in physical database design steps tasks like Design base relation, general constraints, design choice, file organization, user requirements, Design security mechanisms etc. are performed.

10. Write short notes on the following :-

- (a) Foreign key (b) data & information
- (c) Database integrity (d) data - Encryption
- (e) Instance & Schema (f) Constraints

(a) Foreign Key :- A foreign key is the type of key which is mainly used to link two table together.

The purpose of the foreign key is to ensure or maintain referential integrity of the data.

The column of one table points to the primary key attribute of another table.

The Foreign key is not used frequently because

Ques: In a relational database, in each table primary key is linked with the same primary key.

Explain with example, answer in "Relational model".

Ans: Student contains following marks.

Rollno	Name	Marks	Rollno	Subject	Per cent
1	Ramit	987654321	1	Maths	48%
2	Babul	898765432	2	Physics	32%
3	Rohit	94323289	3	Chemistry	25%
4	Amita	8821470	4	English	19%
5	Salmi	7610861	5	Computer	28%

In these two tables (Rollno), in the foreign key because it links the both table.

SQL Syntax

CREATE TABLE Tablename

Columnname type constraint

FORIGN KEY (column name)

column is REFERENCES existingtablename

(column name you want to

referenced)

);

Ex:- CREATE TABLE MARKS

Rollno int(5) PRIMARY KEY

FOREIGN KEY

(Rollno) REFERENCES

STUDENT (Rollno)

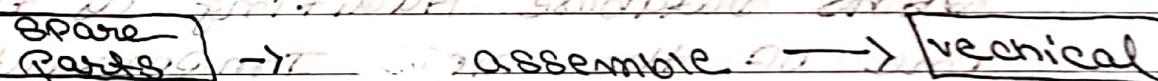
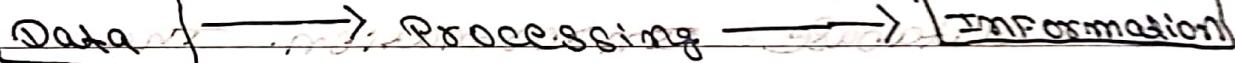
);

predominant

also

- b) Data and Information
- Data :- The collection of raw facts is called Data. The word "data" is derived from Greek word διάτατον which means facts. It is unorganized facts that required to be processed to make it meaningful.

Information :- Information is the organized set of data which is processed in a meaningful way according to the given requirement. The information is used for decision making because information has meaningful facts.



* Difference b/w data & information

Data	Information
raw & unorganized facts	organized & meaningful
i) Data is raw facts & unorganized	It is meaningful data.
ii) It does not depend upon information	It depends upon data.
iii) It does not directly help in decision making.	It directly helps in decision making.

DataInformation

- (i) Data are text and information is (a)
 (ii) numeric values in required form
 "code" box is actual data
 box may not be used
 (iv) It is low-level knowledge is the second level of knowledge
 (v) It is low-level knowledge is the second level of knowledge

(c) Database Auditing

⇒ The Database Auditing is a facility of the database management system that enables the database administrator to track the use of database resources and authority.

When database auditing is enable the database management system will produce an audit trail of database information. The

Database Auditing is the one of the most important security and tracking feature of the database management system.

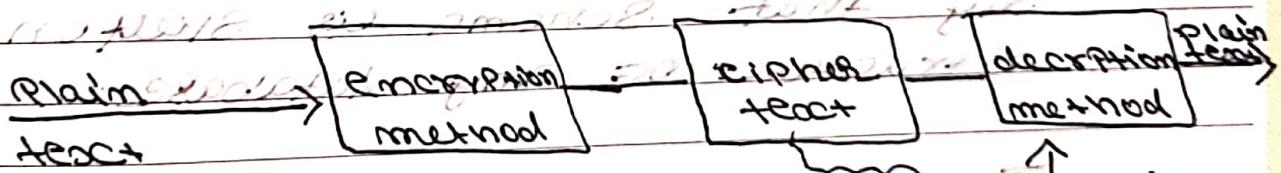
Different DBMS provide different auditing capability. Some of the most common

that can be audited by DBMS are:-
 (i) successful or unsuccessful login attempts
 (ii) maintenance requests

(iii) various modifications made on data

- (i) database server report
 - (ii) Select, Insert, Update and delete operations.
 - (iii) Change to system catalog table
 - (iv) Unsuccessful attempt to access database
 - (v) unsuccessful attempts to access table
- Ques 2)** Data Encryption: - The data encryption is a security method where information is encoded in such a way that only authorized user can access it. A Database management system can use encryption to protect the information in certain situations where the normal security mechanism of the DBMS is not enough.

Data encryption converts plain text into cipher text, easily accessible by an unauthorized person into an unreadable, unaccessible without key, and encoded format.



⇒ The message which is to be encrypted by using encryption is known as plain text over the plain text encryption is process and the encoded data which is the output of encryption method is known as cipher text. The cipher text

Page 58

is in encoded format and hence it is processed by the processor, converting the cipher text into plain text and over again unknown decryption method. Now the decryption method needs a decryption key for converting the cipher text to plain text. The output of the decryption method is plain text which can be read by anyone giving right key or password.

(e) Instances & Schemas: - An organization has many different types of data items. The overall organization of database which includes all various data items, their datatype, constraints, relation etc. is called Schema. The schema gives logical view of entire database. It also defines how the data is organized and how the relation among them is associated. In short we can say that Schema is structure of any database.

Instances:- The data stored in the database at a particular moment of time is called instance of the database. The instance of the database should be changed by certain operation over the data. These are addition, deletion, insertion, update, etc.

X. DIFFERENCES.

Schema and Instance

- (i) Schema refers to overall description of data information any given database stores at any particular time - ~~any particular time~~
- (ii) It does not change very frequently
- (iii) Schema is same for entire database.
- (iv) It defines how the data can be stored in the database.
- (v) Example.

empid	Name	Salary	Dept
010	Ramul	12000	D1
210	Babu	18000	E2

(F) Constraints → Constraints are the rules enforced on the data column of a tables. These are used to limit the type of data that can go into the column.

of the table. It helps in ensure the accuracy and reliability of the data in the database. The constraints ensure that whenever the data entered by the customers into column must be values within the criteria specified by the condition of that constraint.

Ex:- If we do not want to leave any record in the column as null then we define that column as NOT NULL constraint.

* The most commonly key constraints that are used are as:-

i) NOT NULL → It ensures that the column does not contain any null value.

ii) UNIQUE → It ensures that all the values in the column should be unique.

iii) DEFAULT → The default constraint provides a default value to a column when there is no value provided.

- (iv) Primary Key :- It ensure that the column cannot contain any null as well as duplicate value.
- (v) Foreign Key :- It ensure referential Integrity of the relation.
- (vi) Check :- Allow to apply a certain condition for insert
Ex - Sales > 50
- (vii) Auto Increment :- It provide auto increment value for a column.
- Q1. Write the differences b/w following :-
- DBMS and File System
 - Candidate key, Primary key, Superkey and Foreign Key

⇒ (a) DBMS and File System

File System	DBMS
(i) It is a software that manage and organize the files in a storage medium within a computer.	DBMS is a software for managing the database.
(ii) It has low level security features.	It has high level security features.

File SystemDBMS

(i) ~~Redundant data~~
can be present in
the file system.

(ii) It does not provide
backup and recovery
of data.

(iii) It stores data
into the Hard disk.

(iv) Less data consistency
in the File System.

(v) It is less expensive
than DBMS.

(vi) Not easy to do
operations over data.

(vii) It supports only
'single view'.

(viii) It is suitable for
small organization.

(ix) It cannot provide
concurrency facility.

(x) Ex - NTFS,
EXT

there is no data
consistency.

It provides backup
and recovery
of data.

It stores data
on the database.

more data
consistency in DBMS.

It is expensive
than File System.

Easy to do
operations over data.

It supports
multiple view.

It is suitable for
large organization.

It provides
concurrency facility.

Ex - MySQL
MS Access
Oracle

(i) Primary keys, Super keys, candidate keys and foreign keys

Primary key

- It is column which can uniquely identify each row.
- Primary key can have one or more primary keys.
- Primary key is column which can uniquely identify each row.

Super key

- It is the group of keys which can uniquely identify rows in a table.
- Primary key can have one or more super keys.

Candidate key

- It is a set which is not primary key but is definitely a super key.

Foreign key

- It is the key which uniquely links two tables together.

many.

not depended

dependent
super key

not alone

alone

alone

alone

no

alone

no & depends of one

one

one

more than one.

ex:- Rollno, emplid

Rollno, name

name, class, address

Roll no

(29)

Example in order to solve

student record candidate test

student record candidate test

(4)

Student

Rollno	Name	Class	Address	Mobile No.
RO1009	Ramya	XI	Burgess 2441	88692000
W00084	Rajesh	XII
R1891	Aditi	XII
G2104	Saranya	IX
G33245	Ranavijay	IX
A21020	Aditya	XI
G21012	Ramanish	XI

Rollno	Name	Class	Address	Mobile No.
W01009	Usha	IX
W00084	Praveen	XII
R210891	Umesh	XII
G21041	Praveen	XII
G21012	Umesh	XII

Rollno, Name, Class, Address, Mobile No.
 Rollno, Name, Class, Address, Mobile No.

Q. What are the different types of relation b/w tables in DBMS?

Ans:- Relation is an association b/w two or more data tables. It is used to identify how much record of one table is related with the record in another table.

There are 4 types of relation b/w tables.

(i) One to One Relation

(ii) One to many Relation

(iii) many to One Relation

(iv) many to many Relation

(i) One to One Relation

→ It is represented by 1:1. It is used to create a relationship between two tables in which a single row of the first table can not be related to one and only one record of the second table. Similarly the row of second table also be related to the only one record of the first table.

Ex:-

Table A

Customer

	Customer	Order
1	S101	1001011X
2	G1003	1013314
3	F104	08981331
4	A100	9132111

Table B

order

order no.

1001011X

1013314

08981331

9132111

One many Relation

⇒ 1 to many is represented by 1:m.

1:m is used to create a relationship between two tables in which any single row of a first table can be related to one or more than one row of the second table. But one or more than one row of second table can relate with only one record of the first table.

Ex:- In 1 student can play many games.

Student

	Student
1	S101
2	G100
3	F104
4	A100

Games

Game

Cricket

Football

Hockey

Volleyball

Badminton

Tennis

(iii) Many to one relation

→ It is represented by m:1. It is the reverse of one to many relation. It is used to create a relationship b/w two table in which ^{one or} ~~more than~~ one record of 1st table can relate with only one record of the second table. but only one record of the second table can relate with ^{one or} ~~more than~~ one record of the first table.

Ex:-

Student

Student
SG101
RV102
CV302
TS009
VC001

Course

Course
COURSE CODE
TDC101
TDC009
TDC051
TDC111
TDC202
TDC001

(iv) Many to many relation

→ It is represented by N:N. It is used to create a relationship b/w two tables in which any no. of records of the first table can relate to any no. of records of the second table. It means any no. of

records from both the table

can relate with each other.

From DB, we can see that there are

9 student in student list & 10 subjects

Student	Subcode
GS 301	R01
CV 264	R02
PN 101	R03
AP 404	R04
AA 401	R05
DA 904	R06
CO 441	R07
EG 104	R08
	R09
	R10
	R11

Q3. SQL Command Related Ques.

1. Create table "employee".

2. Insert new record in employee table.

EmpId	EmployeeName	City	Salary	Designation
100007				

(i) Create table

(ii) Insert new record in employee table.

(iii) Add new column 'bmi' in employee table.

(iv) Update the salary of employee having empId 100007 to 20000.

(v) Update the name of the employee having empId - 2010 with "manish".

(vi) List all employee from city Pune.

- (i) Delete the whole structure
- (ii) Delete all records
- (iii) Rename tablename with Emp
- (iv) Rename columnname empid to empno
- (v) Delete the record of empid - 303.

(vi) CREATE TABLE Employee

```

    (
        emp_id int(4) PRIMARY KEY,
        emp_name varchar(30) NOT NULL,
        city varchar(30) NOT NULL,
        designation varchar(20)
    );
    
```

(vii) → INSERT INTO Employee
VALUES (209, 'Babul', 'Patna', 10000,
'worker');

→ INSERT INTO Employee
VALUES (211, 'Ranul', 'Pune', 25000,
'supervisor');

(viii) ALTER TABLE Employee
Add bim int(5) UNIQUE;

(ix) SELECT * FROM Employee
WHERE salary > 10000 AND
salary < 20000;

- (vi) `UPDATE Employee
SET EmpName = 'manish'
WHERE Emp_id = 201;`
- (vii) `SELECT * FROM Employee
WHERE City = 'Pune';`
- (viii) `DROP TABLE Employee;`
- (ix) ~~`TRUNCATE TABLE Employee;`~~
- (x) ~~`RENAME TABLE employee TO emp;`~~
- (xi) `ALTER TABLE Employee
RENAME COLUMN Emp_id TO Empno;`
- (xii) ~~`DELETE FROM Employee
WHERE Emp_id = 203;`~~

14. Illustrate with the help of diagram three levels of data abstraction:

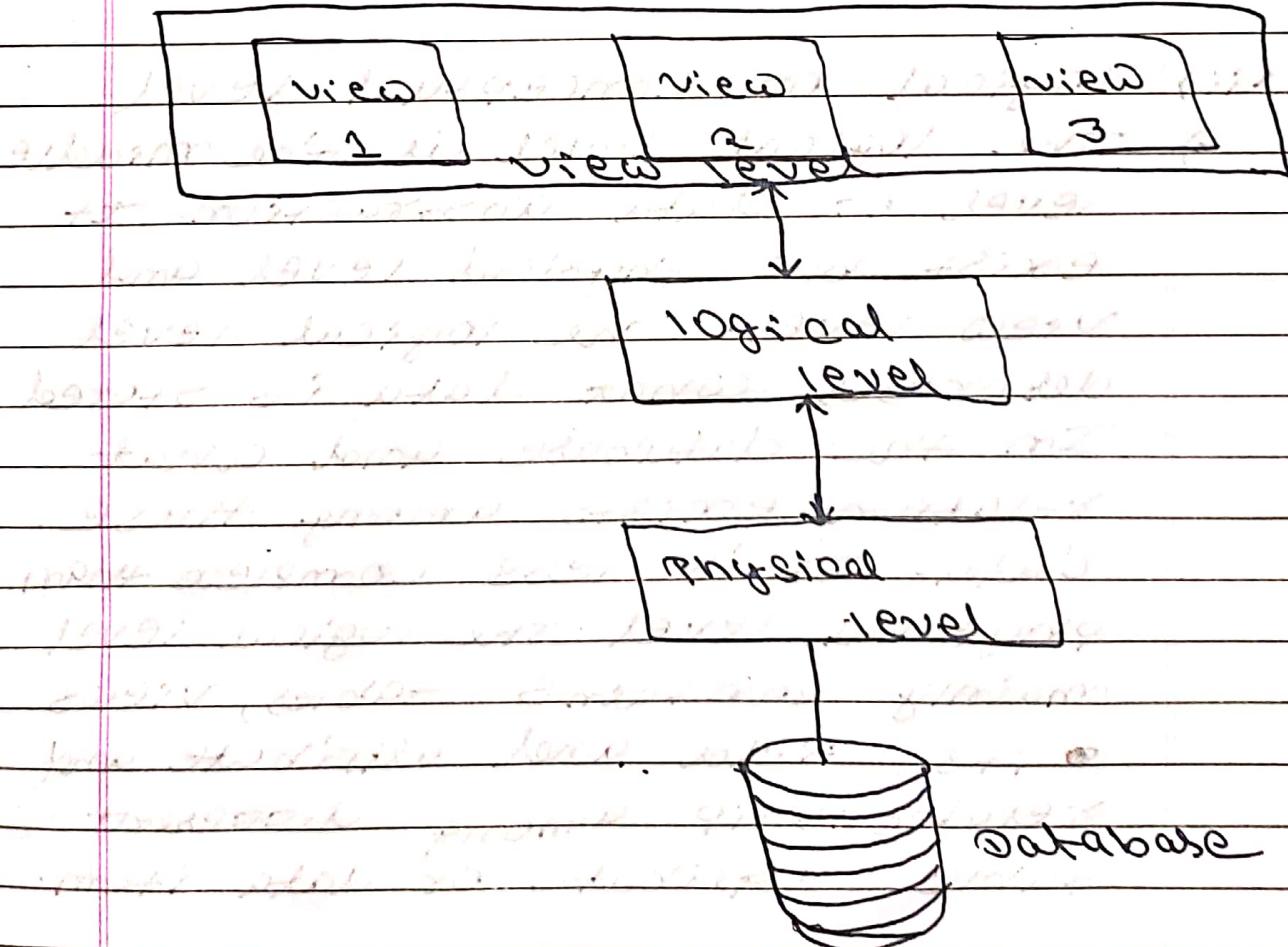
⇒ Data abstraction is the process of hiding unwanted or irrelevant details from the end users. It provides different view and helps in achieving data independence which is used to enhance the security of data.

The database system consists of complicated data structure and relations. For users, to access the data easily, these complications are kept hidden, and only the relevant part of the database is made accessible to the users through data abstraction.

Mainly, there are three levels of data abstraction

~~at present, there are three levels of schema.~~

- (i) Physical or Internal level
- (ii) Logical or Conceptual level
- (iii) View or External level



(i) Physical or Internal level

→ It is the lowest level of data abstraction which defines how the data is actually stored. It defines data-structure to store data and access method used by the database. The physical level is defined by the developer or database application programmer that how to store the data in the database. So, overall the entire database is described in this level. It is very complex level to understand.

(ii) Logical or Conceptual level

→ The logical level is the middle level of data abstraction. It exists between physical level and view level. The logical level describes what data is stored in the database and what relation exists among those data. It is less complex than physical level. The logical level mainly contains tables, views, its field and attribute and relationship among different table attribute or data item.

(iii) View or External level

→ It is the highest level of data abstraction. The view level provide an interaction with the user and also many views or multiple views of the same database. View level can be used by the users and it is least complex and easy to understand. In this level only a part of actual database is viewed by the user and the user view data in the form of rows and columns.

15. What is log file? Explain.

Ans:- Log is a sequence of log records to keep track of database transaction, the DBMS maintain a specified file called log file. The log file contain information about all update executed on the database. The log file contain not only all update but also the records inserted or deleted. Any type of operation which is performed on the database is recorded in the log file. The log or record of each transaction is maintained in some stable storage so that if any failure

they can be recovered from it. There, the log file contains all the data about the transaction that is Start of transaction, Transaction ID, record ID, type of operation (Insert, delete, update), old value, new value, end of transaction that is commit or aborted.

In log file the log records are represented by different terms:-

(i) $\langle Ti \rangle \rightarrow$ Transaction Identifier

\Rightarrow It is the unique

Identifier of the transaction.

(ii) $\langle RID \rangle \rightarrow$ Data item

\Rightarrow It represent the data item which transaction is performed on.

(iii) $\langle VD \rangle \rightarrow$ Old value \Rightarrow It represent

the value of the data item before transaction.

(iv) $\langle NV \rangle \rightarrow$ New value \Rightarrow It represent

the value of the data item after a transaction.

- (v) $\langle Ti \text{ Start} \rangle \rightarrow$ It contains information about when a transaction $\langle Ti \rangle$ starts.
- (vi) $\langle Ti \text{ commit} \rangle \rightarrow$ It contains the information about when a transaction $\langle Ti \rangle$ commits.
- (vii) $\langle Ti \text{ abort} \rangle \rightarrow$ It contains the information about when a transaction $\langle Ti \rangle$ aborts.

The log file is mainly used to recover the database from the point of system failure.

The recovery technique which uses the log file to recover the database is known as log based recovery. After a system crash has occurred, the system consults the log to determine which transaction need to be redone and need to be undone.

\Rightarrow Transaction $\langle Ti \rangle$ need to be undone, if the log contains the record $\langle Ti \text{ Start} \rangle$ but does not contain either the $\langle Ti \text{ commit} \rangle$ or $\langle Ti \text{ abort} \rangle$.

→ Transaction (Ti) need to be
redone, if the log contains
(Ti - Start) and either (Ti
(Commit)) or the record (Ti
End). Start is followed by End
in order to commit
the transaction.

~~DBMS~~ completed

↳ If a transaction has been started
and committed, then it can't be
rolled back. It is called a
committed transaction.

~~DBMS~~ ~~transaction~~

↳ If a transaction has been started
and rolled back, then it can't be
committed. It is called an uncom-