Familiarize with Database Systems

Data

Data is raw fact figure or entity. When activities in the organization take place, the effect of these activities need to be recorded which is known as data. Data is the known facts and figures about a person, place, event or thing which can be recorded in computer in the form of number, text, picture, audio, Video etc., For example, '101', Dayaram Yadav', MCA' etc. are some of the examples of data. Data is the basic element or raw fact of database as it provides the information after processing. Data is the raw facts to generate information and data itself only is not very meaningful. It is processed to produce meaningful and useful information.

Information

Information is stimuli that has meaning in some context for its receiver. When information is entered into and stored in a computer, it is generally referred to as data. After processing (such as formatting and printing), output data can again be perceived as information. It is organized or classified data which has some meaningful values for the receiver. It is also known as the processed data on which decisions and actions are based.

For the decision to be meaningful, the processed data must qualify for the following characteristics:

- Timely - Information should be available when required.
- Accuracy - Information should be accurate.
- Completeness - Information should be complete.

Difference between Data and Information

	Data	Information
Definition	Facts and statistics collected together for reference or analysis	Facts provided or learned about something or someone Data as processed, stored, or transmitted by a computer
Refers to	Raw Data	Analyzed Data
Description	Qualitative Or Quantitative Variables that can be used to make ideas or conclusions	A group of data which carries news and meaning
In the form of	Numbers, letters, or a set of characters.	Ideas and inferences
Collected via	Measurements, experiments, etc.	Linking data and making inferences
Represented in	A structure, such as tabular data, data tree, a data graph, etc.	Language, ideas, and thoughts based on the data
Analysis	Not analyzed	Always analyzed
Meaning	Carries no specific meaning	Carries meaning that has been assigned by interpreting data
Interrelation	Information that is collected	Data that has been processed

Key Differences Between Data and Information

The points given below are substantial, so far as the difference between data and information is concerned:

1 Raw facts gathered about a condition, event, idea, entity or anything else which is bare and random, is called data. Information refers to facts concerning a particular event or subject, which are refined by processing.
2 Data are simple text and numbers, while information is processed and interpreted data.
3 Data is in an unorganized form, i.e., it is randomly collected facts and figures which are processed to draw conclusions. On the other hand, when the data is organized, it becomes information, which presents data in a better way and gives meaning to it.
4 Data is based on observations and records, which are stored in computers or simply remembered by a person. As against this, information is considered more reliable than data, as a proper analysis is conducted to convert data into information by the researcher or investigator.
5 The data collected by the researcher, may or may not be useful to him, as when the data is gathered, it is not known what they are about or what they represent? Conversely, information is valuable and useful to the researcher because it is presented in the given context and so readily available to the researcher for use.
6 Data is not always specific to the need of the researcher, but information is always specific to his requirements and expectations, because all the irrelevant facts and figures are eliminated, during the transformation of data into information.
7 When it comes to dependency, data does not depend on information. However, information cannot exist without data.

Database

It is a repository for collection of data which is organized in such a way that it is can be accessed by wide variety of different application programs. It is actually a single organized collection of structured data. The data in the database are always common to all the users of the system. A very simple example of database is telephone diary. Thus, database is a collection of related data.

DBMS

It is a collection of programs that enables user to create and maintain a database DBMS is a general-purpose software system that facilitates the process of defining, constructing and manipulating database for various application. A DBMS handles user request for database actions and allows for control of security and data integrity requirement DBASE, MS-ACCESS, SQL etc. are some software packages to work with DBMS.

Advantages of database over flat file or file-based system.

i. Reduction of data redundancies.
ii. Shared data.
iii. Improve security.
iv. Efficient data access.
v. Multiple user interface.
vi. Improved backup and recovery.
vii. Support for concurrent transaction.
viii. Unforeseen queries can be answered.

Functions of DBMS

The major functions of the DBMS are as follows:

i. **Creating database file:** Before we store data in the database, it is necessary to define the structure of the database by specifying the required tables, field names, field types, field width, validation rules for the field etc. then we can save the file as database file.
ii. **Entering database record:** After the database has been created, we need DBMS to enter data in the database.
iii. **Sorting the database records:** After the database records are stored in the database, the users need to arrange the records in order. It can be done on the basis of numerical or alphabetical in ascending or descending order.
iv. **Deleting records:** DBMS is used to delete the unnecessary records from the database.
v. **Updating records:** Another function of the DBMS is to modify the existing records according to the requirement of the users.

vi. **Searching records:** DBMS searches the specific record according to the user's need. The database contains a lot of records and but the user does not need all the records at once. The DBMS searches the required records and provides it to the user.
vii. **Merging database file:** DBMS is used to merge the records from two different files into a single database file.
viii. **Copying records:** It is used to make a duplicate copy of the complete database or copy only required records to the next database file.
ix. **Printing reports:** DBMS is also used to print the required records for reports.
x. **Back up database:** It is also used to back up the database for the recovery of the database from the accidental data loss or corruption.
xi. **Provide security to data:** DBMS protects the data in the database from the unauthorized access or modification.

Objectives of DBMS

i. Making access to the data easy for the user.
ii. Provide quick response to the user's request for the data.
iii. Making the latest modification to the database available immediately.
iv. Eliminate redundant (duplicate) data from the database.
v. Allow multiple users to share the database at one time.
vi. Allow the growth of database system.
vii. Provide data security by protecting the data from physical harm and unauthorized access.

RDBMS

It is a relational database is a database that has a collection of tables of data items, all of which is formally described and organized according to the relational model. The term is in contrast to only one table as the database, and in contrast to other models which also have many tables in one database. In the relational model, each table schema must identify a column or group of columns, called the primary key, to uniquely identify each row. Rows in one table can relate to rows in another table by establishing a foreign key, a column or group of columns in one table that points to the primary key of another table. The relational model offers various levels of refinement of table organization and reorganization called database normalization. The database management system (DBMS) of a relational database is called an RDBMS, and is the software of a relational database.

Flat file or File based system

Before the database system has come in use, people used to keep records in the file based or a flat file system. It is the early attempt to computerize the manual filing system. The flat file-based system works well when the number of records to be stored is small and there are not many limitations. As a result, it has been replaced by the database system.

Limitation of file based/flat file system

a. Duplication of data.
b. Inconsistent data.
c. Data dependence.
d. Incompatible file formats.
e. Fixed queries

Database Model.

The description of database is called database schema and it is specified during database design. The process of designing the database schema is called data modeling. The data modeling describes the structure of database such as no. of tables, data types, relationship, constraint etc. that should hold-on the data.

Hence, data modeling is also known as database modeling are as follows:

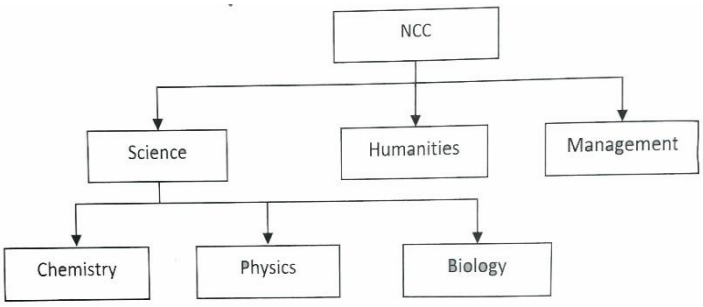
Purpose of data modeling

i. To design the fully normalized database.
ii. To reduce the data redundancy.

iii. To improve faster sorting and indexing.
iv. To improve the performance of database system.
v. To establish proper relationship between the tables in the database.
vi. To simplify the design of structure of table.
vii. To make the database reliable and provide high security of stored data.

The various types of database models are follows:

Hierarchical Model



Hierarchical Model It is one of the oldest database models. This model arranges the files used in the database top-down structures which is similar to an upside-down tree.

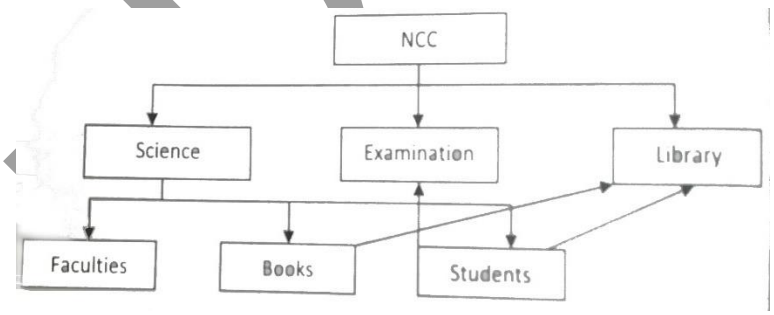
Advantages

i. It is the easiest model of database.
ii. Searching is fast and easy if parent is known.
iii. This model is very efficient in handling 'one-to-many' relationship.

Disadvantages

i. It is old and outdated database model.
ii. Modification and addition of the child node is very hard. Hence, it is non-flexible database model.
iii. It can't handle 'many-to-many' relationship.
iv. It increases the redundancy because same data is to be stored in different places.

Network Model



The network database model looks similar to the hierarchical model but it is more flexible to access the data. Unlike the hierarchical model, each child can be linked with more than one parents. So the records can be accessed from more than one parents, which are linked. This model is more flexible and has multidimensional connection.

Advantages

i. More flexible than hierarchical model.
ii. Reduces data redundancy because similar data is not stored in more than one file.
iii. Searching is faster because of multidimensional pointers.

Disadvantages

i. It is very complex to design.
ii. Needs long program to handle the relationship.
iii. Pointers, needed in the database, model increases overhead of database storage.
iv. Less security model because data can be accessed from any parents.

Relational Model

It is the best and the most common database model developed by E.F. Codd. The relational database model basically defines the structure or organization of data and a set of operations on that data. It is a simple model in which database is represented as a collection of 'Relation', where relation is represented by a two-dimensional table. Because of its simplicity, it becomes commonly used database model. The following table represents a simple relation.

Roll no.	First Name	Middle Name	Last Name	Address	Mobile
1	Utsab	Raj	Adhikari	Balkot	9849715751
2	Kriti		Adhikari	Thimi	9864110222

3	Girisha		Dhakal	Sallaghari	9807624092

Advantages

i. Since one table is linked to other tables with some common fields, rules implemented on one table can easily be implemented to another table.
ii. Referential integrity can be easily implemented.
iii. The Database has very less data redundancy.
iv. Normalization of the database is possible.
v. Rapid database processing and searching is possible.

Disadvantages

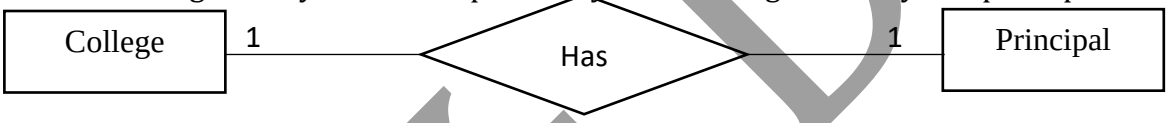
i. It is more complex than other models due to relationship (link) with other tables.
ii. Too many rules make the database not very user-friendly.

Relationship and Its Types

A relationship is a link or association between two or more entities in a database. For example, a relationship exists between the 'Teacher' entity and 'Student' entity. Relationship plays very important role in database design and implementation. There should be a common field between the two entities for relationship and it is established by linking the foreign key in a child table with the primary key in a master table. '

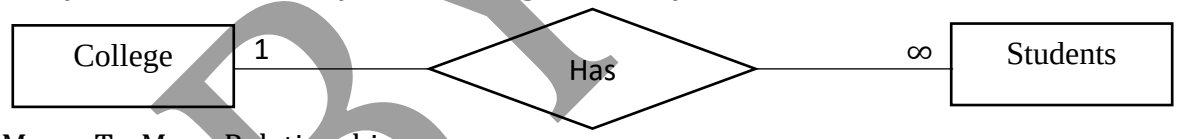
There are three types of the relationship, these are as follows:

1 One-To-One Relationship. If one record of an entity is related with only one record of another entity then such type of relationship is called One-To-One relationship. For example, the relationship between 'College' entity and 'Principal' entity is one college has only one principal.



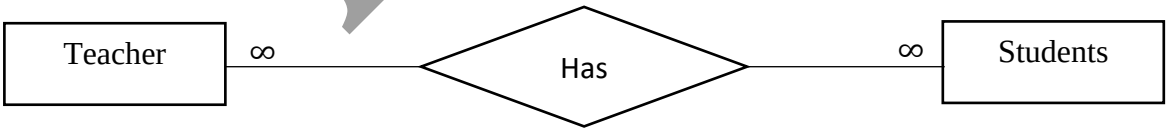
2 One-To-Many Relationship

If one record of an entity is related with one or more instances of another entity, then such type of relationship is called One-To-Many relationship. For example, the relationship between 'College' entity and 'Student' entity as one college has many students.



3 Many -To-Many Relationship

If many instances of one entity are related with many instances of another entity then it is called Many-To-Many relationship. For example, the relationship between 'Teacher' entity and 'Student', considered as many-to-many relationship.



One teacher teaches many students and a student is also taught by many teachers in college. But many-to-many relationship is not possible directly between the 2 tables. Hence, a third table, called a link table, is used to maintain many-to-many relationship.

Keys in DBMS

Keys are, as their name suggests, a key part of a relational database and a vital part of the structure of a table. They ensure each record within a table can be uniquely identified by one or a combination of fields within the table. They help enforce integrity and help identify the relationship between tables. There are three main types of keys, candidate keys, primary keys and foreign keys. There is also an

alternative key or secondary key that can be used, as the name suggests, as a secondary or alternative key to the primary key

Types of Keys:

I. Super Key
A Super key is any combination of fields within a table that uniquely identifies each record within that table.

II. Candidate Key
A candidate is a subset of a super key. A candidate key is a single field or the least combination of fields that uniquely identifies each record in the table. The least combination of fields distinguishes a candidate key from a super key. Every table must have at least one candidate key but at the same time can have several.
In order to be eligible for a candidate key it must pass certain criteria.
- It must contain unique values
- It must not contain null values
- It contains the minimum number of fields to ensure uniqueness
- It must uniquely identify each record in the table

Once your candidate keys have been identified you can now select one to be your primary key

III. Primary Key
A primary key is a candidate key that is most appropriate to be the main reference key for the table. As its name suggests, it is the primary key of reference for the table and is used throughout the database to help establish relationships with other tables. As with any candidate key the primary key must contain unique values, must never be null and uniquely identify each record in the table.
As an example, a student id might be a primary key in a student table, a department code in a table of all departments in an organization. This module has the code DH3D 35 that is no doubt used in a database somewhere to identify RDBMS as a unit in a table of modules. In the table below we have selected the candidate key student_id to be our most appropriate primary key

IV. Composite Key
Key that consist of two or more attributes that uniquely identify an entity occurrence is called Composite key. But any attribute that makes up the Composite key is not a simple key in its own. It is a primary key that consists of two or more attributes.

V. Foreign Key
A foreign key is an attribute or set of attributes in a relation whose values match a primary key in another relation. The relation in which foreign key is created is known as Dependent Table or Child Table. The relation to which the foreign key refers is known as Parent Table. The key connects to another relation when a relationship is established between two relations. A relation may contain more than one foreign key.

VI. Unique Key
In database relational modeling and implementation, a unique key is a set of zero, one, or more attributes. The value(s) of these attributes are required to be unique for each tuple (row) in a relation. The value, or combination of values, of unique key attributes for any tuple should not be repeated for any other tuple in that relation.

The Difference Between Primary Key and Unique Key:

Primary Key		Unique Key	
1.	Only one key within an entity is selected to be the primary key.	1.	More than one key within an entity is selected to be the unique key.
2.	Primary key constraints are used to ensure that data is in a relation.	2.	Unique key constraints are used to ensure that data is not duplicated in two rows in the database.
3.	Primary key can perform the constraints of Unique key	3.	Unique key is not able to perform all the functions of Primary key.
4.	It is also worth knowing that, by default, many DBMS index and physically order tables on disk using the PK.	4.	Index created on unique key not as faster as primary key.
5.	It is mostly used for relationship between tables	5.	It is used to manage data consistency.

Normalization

It is the process of organizing data in a database to reduce the redundancy and inconsistency. The data is said to be redundant if the same data is found is more than one place in a table. Normalization is the process of breaking down a single big table into many smaller tables with fewer fields. Normalization also reduces the inconsistent dependencies between the various fields in a table. Normalization is used to promote efficient maintenance, storage and updating of data tables.

Advantages

i. It reduces the data redundancy.
ii. It improves faster sorting and index creation.
iii. It improves the performance of the database.
iv. It simplifies the structure of tables.
v. It avoids the loss of information.

Disadvantages

i. It is complex to design due to the relationship between different tables.
ii. Requires more CPU cycles, large memory and input/output devices to data process software.
iii. Requires more joins to get the desired result. A poorly-written query can bring the database down.
iv. Maintenance overhead. At the higher level of normalization, a large number of tables are created in the database.

Types of Normalization

Many normal forms are defined; the 1NF, 2NF, 3NF, Boyce-Codd Normal Form (BCNF), 4NF, 5NF, Domain Key Normal Form (DKNF) and 6NF. If a database is in 6NF, then the database is fully normalized. Among the several normal forms, the most used three are as follows:

i. First Normal Form (1NF)
ii. Second Normal Form (2NF)
iii. Third Normal Form (3NF).

The following table is in denormalized form:

RN	Name	Address	Phone	DOB	TID	Dept	Tname	Tphone
1	Hari	Kathmandu	014444446	2049/08/15	12	Management	Ratna	9845654789 9851100029
2	Kriti	Janakpur	041569874	2063/01/07	11	Science	Sundar	9851104445 9844040779
3	Sneha	Biratnagar	047564789	2061/05/09	10	Computer	Daya	9864110222 9807624092

This table is not a normalized table. The data field Tphone consist multiple data on it, this violation the rule of normalization.

First Normal Form 1NF

The above table, the field Tphone holds multiple values in one cell. In this case the primary key is RN With the design like this table, we can have the insert, update, delete and select anomalies. Note that only update and select anomalies deal with redundancy. They are problematic because the table contains redundant data. Insert and delete anomalies deals with incompleteness of the table.

A relation is said to be in **First Normal Form (1NF)** if and only if each attribute of the relation is atomic. Atomic means, the smallest piece of data which cannot be divided further. More simply, to be in 1NF, each column must contain only a single value and each row must contain the same columns. The purpose of the 1NF is to eliminate repeating groups of attributes in an entity. The relation to be in 1NF can be represented by the following schema and corresponding table.

After applying the rule of INF, we get the above denormalized table as follow:

RN	Name	Address	Phone	DOB	TID	Dept	Tname	Tphone
1	Hari	Kathmandu	014444446	2049/08/15	12	Management	Ratna	9845654789
1	Hari	Kathmandu	014444446	2049/08/15	12	Management	Ratna	9851100029
2	Kriti	Janakpur	041569874	2063/01/07	11	Science	Sundar	9851104445
2	Kriti	Janakpur	041569874	2063/01/07	11	Science	Sundar	9844040779
3	Sneha	Biratnagar	047564789	2061/05/09	10	Computer	Daya	9864110222
3	Sneha	Biratnagar	047564789	2061/05/09	10	Computer	Daya	9807624092

This table is in 1NF because every field of the table is atomic, i.e. small piece of data consists in the table held. Further, there are no duplication of rows or columns.

Second Normal Form 2NF

In the above table, it is in 1NF, we still have the insert, update and delete anomalies. To remove the above anomalies, the table should be changed into 2NF. A relation is said to be in 2NF if the relation must first fulfill the requirements to be in First Normal Form and each non-key attribute in the relation must be functionally dependent upon the primary key.

In the above example, the RN and Tid with Tphone field can be used as primary keys. Similarly, the corresponding key field depends on its primary keys like Name, Address, Phone and DOB depends on RN field and Dept, Tname and Tphone depends on Tid. To change the above table in to 2NF, we need to decompose the table into multiple tables. The relation to be in 2NF can be represented by the following schema and corresponding table Multidimensional connections.

Student(RN, Name, Address, Phone, DOB) Teacher(Tid, Tname, Tphone) Std_Tea(RN, Tid)

Student

RN	Name	Address	Phone	DOB
1	Hari	Kathmandu	014444446	2049/08/15
2	Kriti	Janakpur	041569874	2063/01/07
3	Sneha	Biratnagar	047564789	2061/05/09

Teacher

TID	Dept	Tname	Tphone
12	Management	Ratna	9845654789
12	Management	Ratna	9851100029
11	Science	Sundar	9851104445
11	Science	Sundar	9844040779
10	Computer	Daya	9864110222
10	Computer	Daya	9807624092

Std_Tea

RN	TID
1	12
2	11
3	10

After decomposing to multiple tables, we must have some common values among these tables that enable us to join tables in queries. Otherwise, we would lose some information from table. For example, in last table Std_Tea, both candidate keys are connected to maintain relationship between two tables. This process indicates the relational database.

Third Normal Form 3NF

The above tables are in 2NF, but still have Insert and Delete anomalies. To remove these anomalies, these tables should be changed into 3NF. A relation is said to be in 3NF, a table must be in 2NF and non-prime attribute of table is non-transitively dependent (i.e. directly dependent) on every candidate key. Additionally, all attributes that are not dependent upon the primary key must be eliminated. For example, in above table student (RN, Name, Address, Phone, DOB), Name, Address, Phone are fully dependent on its primary key 'RN'. But the field DOB is dependent on student name not in primary key. The schema and corresponding table after 3NF are as follows:

Student1

RN	DOB
1	2049/08/15
2	2063/01/07
3	2061/05/09

This schema is free from insert, update, delete and select anomalies. Still, there are some rules for decomposing the table depending upon higher order normal form.

SQL

Structured Query Language is the universal standard database language for accessing and managing data in the database. At first SQL was called 'SEQUEL' which means 'Structured English Query Language and it was designed and implemented at IBM Research for relational database system. SQL-92 is the certified version of SQL by ANSI and ISO. Different companies have used SQL for their relational database. for example, Microsoft has used SQL as T-SQL for SQL Server, Oracle has used SQL as P-SQL etc.

SQL provides an interface where user can specify **'what'** are the expected results? The query execution plan and optimization is performed by the DBMS. The query plan and optimization determine how a query needs to be executed. SQL is a non-procedural language as it just specifies what is to be done rather than how it is to be done. It is also a high-level query language, so it is closer to a language like English. Therefore, it is very user friendly.

SQL is the comprehensive database language which allows the user to query a database without getting depth knowledge of the design of the underlying tables. SQL has statements for data definition, data manipulation and data control. A query is a request to the DBMS for the retrieval, modification, insertion and deletion of the data from the database.

'SELECT rollno, name, phone from Student where class ='JCT' is an example of SQL query which displays the roll no, name and phone of the students who study in class JCT.

Why SQL

- Allow users to access data in relational database management systems.
- Allow users to describe the data.
- Allow users to define the data in database and manipulate that data.
- Allow to embed within other languages using SQL modules, libraries & pre-compilers.
- Allow users to create and drop databases and tables.
- Allow users to create view, stored procedure, functions in a database.
- Allow users to set permissions on tables, procedures, and views

SQL Constraints

Constraints are the rules enforced on data columns on 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 column level or table level. Column level constraints are applied only to one column whereas table level constraints are applied to the whole table.

- NOT NULL Constraint: Ensures that a column cannot have NULL value.
- DEFAULT Constraint: Provides a default value for a column when none is specified.
- UNIQUE Constraint: Ensures that all values in a column are different.
- PRIMARY Key: Uniquely identified each rows/records in a database table.
- FOREIGN Key: Uniquely identified a rows/records in any another database table.
- CHECK Constraint: The CHECK constraint ensures that all values in a column satisfy certain conditions.
- INDEX: Use to create and retrieve data from the database very quickly.

Some of the important features of SQL are:

i. It is a non-procedural language.
ii. It is English like language.
iii. It can process a single record as well as sets of records at a time.
iv. It is 4th GL as all SQL statements define what is to be done rather than how it is to be done.
v. SQL is a data sub-language consisting of three built in language: Data Definition Language (DDL), Data Manipulation Language (DML) and Data Control Language (DCL).
vi. It insulates the user from the underlying structure and algorithm.
vii. It has the facilities for defining database tables, views, security, integrity transaction control, index etc.

SQL has different statements for DDL DCL and DML.

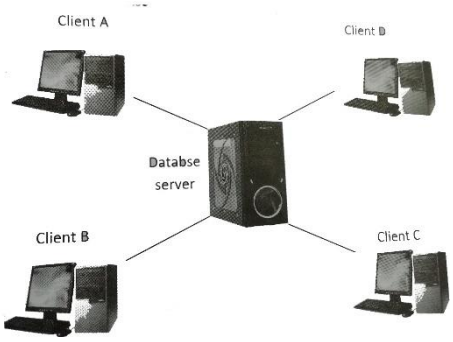
DCL stands for **'Data Control 'Language'** and **DCL** statements are used to change the permission associated with a database user or role. DCL contains the commands that allow system and data privileges to be passed to various users. The DCL commands are normally available to database administrator. The DCL statements are GRANT, DENY and REVOKE. DDL

DDL stands for **'Data Definition Language'** and **DDL** statements are used to create, modify and drop database and database objects like table, user-defined data types, index, views etc. CREATE, ALTER and DROP are the DDL statements of SQL. CREATE is used to make new database objects, ALTER is used to modify the existing database objects and DROP is used to remove the unnecessary database objects from the database.

DML stands for **'Data Manipulation Language'** and **DML** statements are used to work with data in the database. The DML statements are 'SELECT', 'lNSERT', 'UPDATE' and 'DELETE'. The DML statements are used to retrieve required data from database, to insert (add) new data into database, modify existing data and delete unnecessary data from the database.

Centralized and Distributed Database

Centralized Database As shown in the figure, centralized database works in a client-server system. The centralized database has one central computer, called database server to store all the data and files and it provides services to all the clients in the networks. Only the central computer or database server is responsible for processing the data.



Advantages of centralized database

i. Low cost to set up.
ii. High performance.
iii. Centralization of all data in a single computer called server.

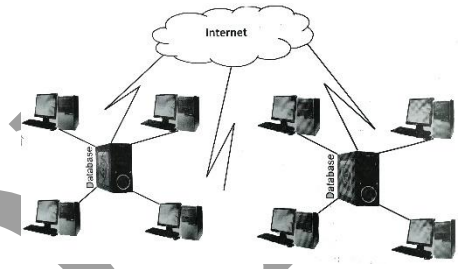
iv. Easier to manage to manage and manipulate data and database.
v. High security as a single Database Administrator can control the whole database system.
vi. Suitable for small organization with different departments.
vii. It is easier to manage and manipulate the data as data is stored only in a server computer.
viii. Easier data access.

Disadvantages of centralized database

i. Cannot cover larger area and not suitable for large organization.
ii. Database is location dependent, cannot be accessed from other places.
iii. It does not support globalize connection.

Distributed Database

Distributed database system is a set of database stored on multiple computers that appears to applications as a single database. The user can simultaneously access and modify the data in several databases in a network. The computers in a distributed system communicate with each other through various communication media, such as high speed buses or telephone line. These computers neither share the main memory nor a clock cycle of processor, although to work properly many applications on different computers might have to synchronize their clicks.



The main differences between centralized database and distributed database is that, in the centralized database, the data resides in one single centralized computer, while in the distributed database the data is stored in several sets under the control of local distributed DBMS components which are under the control of distributed database system.

Advantages of Distributed Database:

i. Data sharing and distributed control all over the world.
ii. Improved reliability for users.
iii. Improved availability of data.
iv. Economy on operation and data sharing.
v. Modular growth can support.

Disadvantages of Distributed Database:

i. Higher software development cost.
ii. Greater potential for bugs and hacked.
iii. Increased processing overhead for client and server computers.
iv. More Complexity in database design.
v. Less security model because data may travel continent to continent.
vi. More difficult for general integrity.

Data Security

Database is an organized collection of related data to provide the right information at right time collected data are the basic element of the database to provide meaningful and useful information for the users and the management. Data plays very important roles to give right information at right time, to take right decision and even to make right plans for future. The data stored in the database should always remain safe, correct and authentic. There should not be any damage or harm to the data in the database.

Information security is the protection of information against unauthorized disclosure, alteration or destruction. Database security is the protection of information that is maintained in a database. It deals with ensuring only the 'right people' get the right to access the 'right data'. By right people, we mean those people who have the right to access or update the data that they are requesting to access or update

with the database. This should also ensure the confidentiality of the data. For example, the personal information of the employees should be accessible only to the authorities concerned and not to everyone.

Thus, one of the concepts of database security is primarily a specification of access rules about who has what type of access to what information. In a wide concept, data security also covers physical security such as data file protection from heat, dust, power failure, theft etc. The important objective of data security is to prevent data loss, misuse of data and disclosure of the secret data.

The methods of data security are as follows:

i. By doing the regular backup of database in disks, tapes, optical disks etc. in order to prevent from the accidental loss.
ii. By using password log in system to prevent data from unauthorized access to the database.
iii. By specifying the specific roles to every user of the database for granting the appropriate permission to them.
iv. Making physical prevention by using stabilizer and UPS to supply a regular power through which we can prevent hardware and software from high electricity voltage and irregular power supply.
v. By keeping the system in safe room or place with lock and key or under the Supervision of watchman to prevent from theft.
vi. By implementing software protection like antivirus, firewalls, antispyware etc.

Data Integrity

Integrity means 'honesty' or 'correctness'. Similarly data integrity refers to the consistency and accuracy of data of stored in a database. Collected data should be consistent and accurate as it is the basic element of database to provide right information at right time, to take right decision and to make right planning for future.

The data integrity is an important step in database planning and proper decision has to be made for the best way to enforce the integrity of the data.

If the data integrity is not maintained in a database, then the database may have the following drawbacks.

i. Unrelated data may be stored in the master and child table.
ii. Improper data may be stored in the column. For example, a column having text value which should have numerical value.
iii. Data may not come from a defined domain (set of defined data). For example, 'Age' field of 'Employee' table having 500 as the data value.
iv. Data may not match the business rules of the business house. For example, salary field having 50,000 where maximum salary given to an employee cannot exceed more than 25,000.

The changes made in one table may not bring the necessary changes in other tables. For example when 'Sold Quantity' of an item is inserted in 'Sales' table, then the 'Stock Quantity' it 'Stock' table may remain constant.

Types of Data Integrity

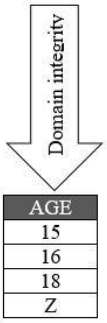
There are three types of data integrity.

i. Domain Integrity ii. Entity Integrity iii. Referential Integrity

Domain Integrity

Domain or field integrity specifies a set of data values that are valid for a column or field and determines whether null values are allowed or not. It is often enforced through the use of validity checking for a column or field and can also be enforced by restricting the data types, format, or range of possible values allowed in a field. The domain integrity is used to check validity of data for a field of the table. For example, domain integrity checks the marks of students in English must be in the range of 0 and 100 or not. Or, Age of students must consist numeric value rather than character.

Domain integrity is maintained in the database through 'DEFAULT', 'CHECK', and 'REFERENTIAL' Constraint.



Entity Integrity

It is related to the row of an entity. The entity integrity specifies that all rows in a table have a unique identifier, known as the primary key value. It also ensures that all the rows in a database have a not-null value and no table has any duplicate primary key values. When we change the primary key value in master table, then related key value in child table will be automatically changed. For example, entity integrity checks whether the 'Registration no' of two students have same value or not.

Entity integrity is maintained in the database through **Primary Key** and **Unique** constraint.

RN	Name	Address	AGE	Class
4	Sneha	Sabaila	7	3
4	Kriti	Janakpur	7	3

Entity Integrity rule violation

Referential Integrity

Teacher Table

ID	Name	Subject
1	Daya	Computer
2	Sundar	Science

Relation

ID	RN
1	4
1	
1	6

?

Student Table

RN	Name	Address	Age	Class
4	Sneha	Sabaila	7	5
5	Kriti	Janakpur	4	1
6	Priyanka	Kathmandu	17	12

It is related between the two tables in a database. It ensures that the relationship between the primary keys in the master table and foreign key in child table are always maintained. It ensures that all foreign key values in a database are valid. For a foreign key to be valid, it must contain either the value NULL, or an existing key value from the primary or unique key columns referenced by the foreign key. A foreign key value in a child table cannot be modified if it has the related primary key value in the master table.

In above table, data value ID of Teacher is referred to relation to Student table. The above tables are in one-to-many relationship because one teacher is related with multiple students. The value of RN on Relation table consist NULL value. This is the violation of referential integrity, i.e., some field has not reference key. The reference key is also called foreign key. The second row is pointed by Teacher table but not point to Student table.

Data Dictionary

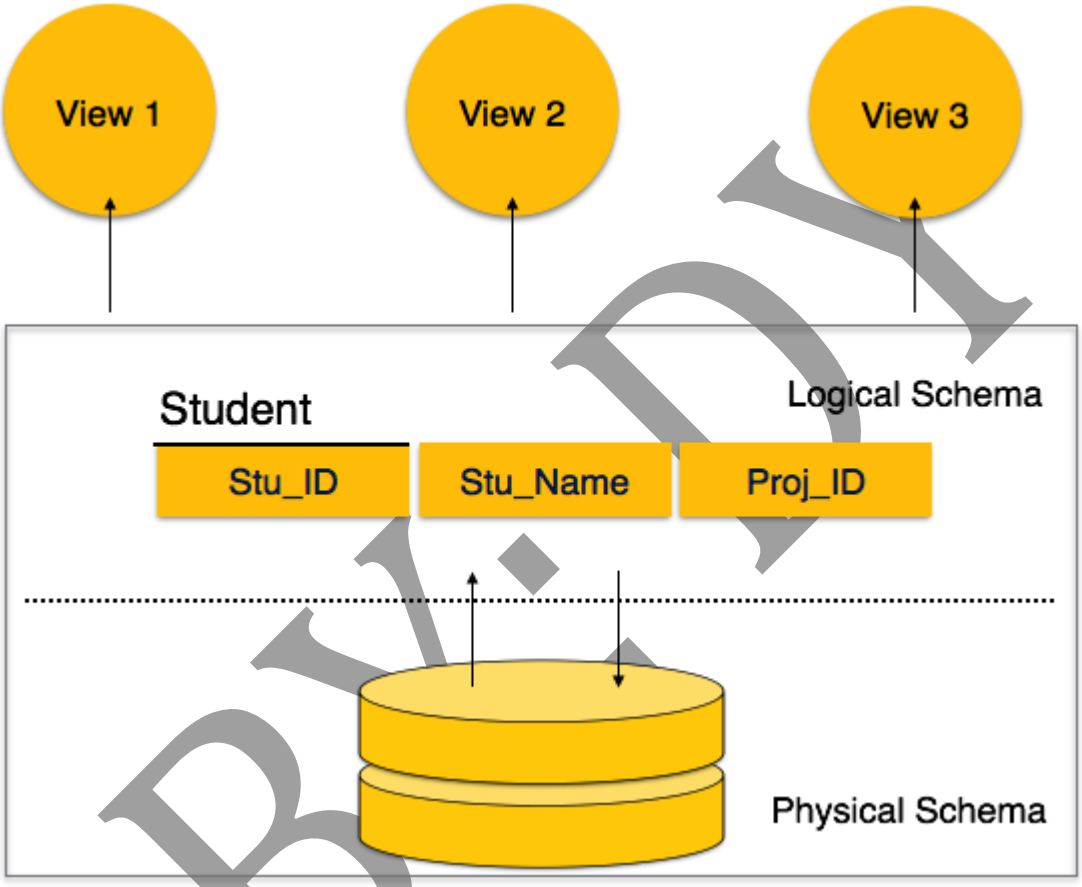
A data dictionary stores information about the structures of the database. The data dictionary defines each term encountered during the analysis and design of a new system. A good data dictionary should have a good design and efficient implementation.

It is seen that when a program becomes somewhat large in size, keeping track of all the available names that are used and purpose for which they were used, becomes more and more difficult. After sometimes, if the same or another programmer has to modify the program, then it becomes extremely difficult.

Data dictionary provides the definition of data items, how they fit into the da structure and how they relate to other entities in the database.
The DBA uses the data dictionary in every phase of a database life cycle. An ideal data dictionary should include everything a DBA wants to know about the database.

Database Schema

A database schema is the skeleton structure that represents the logical view of the entire database. It defines how the data is organized and how the relations among them are associated. It formulates all the constraints that are to be applied on the data.
A database schema defines its entities and the relationship among them. It contains a descriptive detail of the database, which can be depicted by means of schema diagrams. It's the database designers who design the schema to help programmers understand the database and make it useful.



A database schema can be divided broadly into two categories –

- **Physical Database Schema** – This schema pertains to the actual storage of data and its form of storage like files, indices, etc. It defines how the data will be stored in a secondary storage.
- **Logical Database Schema** – This schema defines all the logical constraints that need to be applied on the data stored. It defines tables, views, and integrity constraints.

Database Administrator (DBA)

In any organization, a chief administrator is needed to oversee and manage the resource of database because the same data resources of the database are shared by all the staffs and users. Such a chief person, who is assigned to take care database and DBMS software is called Database Administrator (DBA) Or simply, a DBA is a person who is responsible for maintaining the RDBMS in an organization.

Responsibilities of a DBA

Although DBA has many duties to do, s/he is mainly responsible for authorizing access to the database, for coordinating and monitoring its uses and for acquiring software and hardware resources as needed. The DBA makes sure that the database is protected from physical harm, virus, hackers, and data loss is minimized by maintaining data security.

The main responsibilities of DBA are as follows:

i. Installing and Upgrading database server.
ii. Monitoring the database server and tuning accordingly.
iii. Using the database storage properly.
iv. Performing backup and recovery of data.
v. Managing database users and security.
vi. Working with program developers.
vii. Transferring data from one computer to another.
viii. Scheduling jobs and events.
ix. Replicating data in multiple servers.

Qualities of a good DBA

i. The DBA should have depth knowledge of OS in which database server is running.
ii. S/he should have sound knowledge of SQL.
iii. S/he should have sound knowledge of good database design.
iv. S/he should have sound understanding of network architecture.
v. Good knowledge of database server.