**Session 1**

**Data**

* Data is a collection of raw, unorganized facts and details like text, observations, figures etc.
* Data does not carry any specific purpose.
* Data can be recorded and doesn’t have any meaning unless processed.

**Information**

* Information is an organized, processed form of data.
* It provides context of the data and enables decision making.

**Data vs Information**

* Data is a collection of facts while information puts those facts into context.
* Data is raw and unorganized while information is organized.
* Data isn’t sufficient for decision making while you can make decision based on information.

**Database**

* Database is a system where data is stored in a way that it can be easily accessed, managed, and updated.

**Database Management System(DBMS)**

* Database Management Systems (DBMS) are software systems used to store, retrieve, and run queries on data.

**Database Model**

* **Hierarchical Model:** The Hierarchical Model was the first database management system model. This concept uses a hierarchical tree structure to organize the data. The hierarchy begins at the root, which contains root data, and then grows into a tree as child nodes are added to the parent node.
* **Network Model:** This model is the generalization of the hierarchical model. This model can consist of multiple parent segments and these segments are grouped as levels but there exists a logical association between the segments belonging to any level.
* **Relational Model:** The relational model represents how data is stored in Relational Databases. A relational database consists of a collection of tables, each of which is assigned a unique name.
* **Object Oriented Model:** The object-oriented data model, is based on the object-oriented-programming paradigm, which is now in wide use.
* Inheritance, object-identity, and encapsulation (information hiding), with methods to provide an interface to objects, are
* among the key concepts of object-oriented programming that have found applications in data modelling.
* **Distributed Database Model:** A distributed database is a database that runs and stores data across multiple computers, as opposed to doing everything on a single machine. Typically, distributed databases operate on two or more interconnected servers on a computer network.

**Session 2**

**SQL:**

SQL is the most widely used database query language. It is designed for retrieving and managing data in a relational database. SQL can be used to perform different types of operations in the database such as accessing data, describing data, manipulating data and setting users roles and privileges (permissions).

**Commands in SQL:**

The SQL Commands are grouped into four categories known as DDL, DML, DCL and TCL depending on their functionality, namely the type of operation they’re used to perform.  Let’s explore these commands in greater detail.

**Data Definition Language (DDL)**

The SQL DDL category provides commands for defining, deleting and modifying tables in a database. Use the following commands in this category.

**CREATE Command**

Purpose: To create the database or tables inside the database

Syntax to create a table with three columns:

**CREATE TABLE table\_name (column\_name1 datatype(size), column\_name2 datatype(size), column\_name3 datatype(size));**

**DROP Command**

Purpose: To delete a database or a table inside the database.

Syntax to drop a table:

**DROP TABLE table\_name;**

**ALTER Command**

Purpose: To change the structure of the tables in the database such as changing the name of a table, adding a primary key to a table, or adding or deleting a column in a table.

1. Syntax to add a column into a table:

**ALTER TABLE table\_name ADD (column\_name datatype(size));**

2. Syntax to add a primary key to a table:

**ALTER TABLE table\_name ADD primary key (column\_name);**

**TRUNCATE Command**

Purpose: To remove all records from a table, which will empty the table but not delete the table itself.

Syntax to truncate a table:

**TRUNCATE TABLE table\_name;**

**COMMENT Command**

Purpose: To add comments to explain or document SQL statements by using double dash (--) at the start of the line. Any text after the double dash will not be executed as part of the SQL statement. These comments are not there to build the database. They are only for your own use.

Syntax to COMMENT a line in SQL:

**--Retrieve all data from a table SELECT \* FROM table\_name;**

**Data Manipulation Language (DML)**

The SQL DML commands provide the ability to query, delete and update data in the database.  Use the following commands in this category.

**SELECT Command**

Purpose: To retrieve data from tables in the database.

Syntax to select data from a table:

**SELECT \* FROM table\_name;**

**INSERT Command**

Purpose: To add records of data into an existing table. Syntax to insert data into three columns in a table:

**INSERT INTO table\_name (column1, column2, column3) VALUES (value1, value2, value3);**

**UPDATE Command**

Purpose: To modify or update data contained within a table in the database.

Syntax to update data in two columns:

**UPDATE table\_name SET column1 = value1, column2 = value2 WHERE condition;**

**DELETE Command**

Purpose: To delete data from a table in the database.

Syntax to delete data:

**DELETE FROM table\_name WHERE condition;**

**Data Control Language (DCL)**

You use DCL to deal with the rights and permissions of users of a database system. You can execute SQL commands to perform different types of operations such as create and drop tables. To do this, you need to have user rights set up. This is called user privileges. This category deals with advanced functions or operations in the database. Note that this category can have a generic description of the two main commands. Use the following commands in this category:

**GRANT** Command to provide the user of the database with the privileges required to allow users to access and manipulate the database.

**REVOKE** Command to remove permissions from any user.

**Transaction Control Language (TCL)**

The TCL commands are used to manage transactions in the database. These are used to manage the changes made to the data in a table by utilizing the DML commands. It also allows SQL statements to be grouped together into logical transactions. This category deals with advanced functions or operations in a database. Note that this category can have a generic description of the two main commands. Use the following commands in this category:

**COMMIT** Command to save all the work you have already done in the database.

**ROLLBACK** Command to restore a database to the last committed state.

**Data types in SQL**

Data types tells a database management system how to interpret the value of a column. Following data types are mentioned below:

* Numeric:
  + Tiny int (max value is 255)
  + Int (over four billion number)
  + Decimal i.e. 4.00
* String
  + TINYTEXT: Less than 255 characters.
  + TEXT: Less than 65,000 characters.
  + MEDIUMTEXT: Supports 16.7 million characters.
  + LONGTEXT: Supports up to 4 gigabytes of text data.
  + CHAR: Best when we have predefined size of variable.
  + VARCHAR: Stands for variable length and does not sure how many characters might be inserted.
* Date and time
* MISCELLANEOUS: BLOB

Database Constraints:

* NOT NULL
* DEFAULT
* PRIMARY KEY
* FOREIGN KEY

**CREATING TABLES**

Creating a table which contains:

* CustomerId
* FirstName
* LastName
* Company
* Address
* City
* State
* Country
* PostalCode
* Phone
* Fax
* Email

Syntax:

**CREATE TABLE customers(**

**CustomerId INT,**

**FirstName VARCHAR(40),**

**LastName VARCHAR(20),**

**Company VARCHAR(80),**

**Address VARCHAR(70),**

**City VARCHAR(40),**

**State VARCHAR(40),**

**Country VARCHAR(40),**

**PostalCode VARCHAR(10),**

**Phone VARCHAR(24),**

**Fax VARCHAR(24),**

**Email VARCHAR(60)**

**);**

**Showing the table by using the following syntax:**

**SELECT \* FROM customers;**