**Types of Data and Storage:**

• Data can come many forms and sizes.

1. **Unstructured Data: •** Documents, Files, Videos, Images etc.

• It can be physical or digital copy.

• For physical storage we can use file storage mechanism.

• File Storage:

• Filing Cabinets – Hard Copies.

• Hard Drives – Digital Copies.

1. **Semi-Structured Data:** • It is also file based.

• File based storage (JSON, XML, etc.)

• NoSQL Database

• MongoDB

• Azure Cosmos DB

• Digital Documents.

1. **Structured Relational Data:** • It stored data in form of tables, rows and columns.

**Databases: •** Microsoft SQL Server.

• Azure SQL.

• MySQL.

• PostgreSQL.

• Oracle DB.

**Microsoft SQL Server:**

• A powerful relational database management system provided by Microsoft.

• Used in many companies for simple and complex database operations.

**Features:**

**•** Perfect for beginners.

• Employable skills.

• Learn best practice.

• Industry Standard Database Development and Management Software.

**Install SQL Server & SSMS:**

• SQL Server is database engine.

• SSMS is user interface to interact with engine.

• SQL Server is managed by SSMS.

**Steps:**

1. Google SQL Server Express Installation.
2. Download Express edition.
3. Click on Basic installation.
4. Setup SQL Server.
5. Click on install SSMS.
6. Download SSMS latest version.
7. Click on “Install” option.
8. Close.

**Data:**

• Data can be fact related to any object.

• For example, our name, age, height, weight etc. are some data related to you.

• A picture, image, file, pdf etc can also be considered data.

**Database:**

**•** Database is systematic collection of data.

• Databases support storage and manipulation of data.

• Databases make data management easy.

**SQL:**

**•** SQL stands for Structured Query Language.

• SQL is used to communicate with database.

• SQL lets us access and manipulate databases.

• SQL became a standard of the American National Standards Institute (ANSI) in 1986, and of the International Organization for Standardization (ISO) in 1987.

**What can SQL do?**

**•** SQL can execute queries against a database.

• SQL can retrieve queries from a database.

• SQL can insert records in a database.

• SQL can update records in a database.

• SQL can delete records from a database.

**•** SQL can create new databases.

• SQL can create new tables in a database.

• SQL can create stored procedures in a database.

• SQL can create views in a database.

• SQL can set permissions on tables, stored procedures and views in a database.

**T-SQL:**

**•** T-SQL stands for Transact-SQL.

• T-SQL is a set of programming extensions from Sybase and Microsoft that add several features to the Structured Query Language (SQL), including transaction control, exception and error handling, row processing and declared variables.

• All applications that communicate with SQL Server do so by sending T-SQL statements to the server.

• T-SQL queries include the SELECT statement, selecting columns, labeling output columns, restricting rows and modifying a search condition.

**SQL Server:**

**•** SQL Server is RDBMS, developed and marketed by Microsoft.

• SQL Server works exclusively on Windows environment for more than 20 years. In 2016, Microsoft made it available on Linux.

## **SQL Common Table Expression (CTE):**

• TheWITH clause in MySQL is used to specify a Common Table Expression.

• A Common Table Expression (CTE) in SQL is a one-time result set, i.e. it is a temporary table that exists only during the execution of a single query. It allows us to work with data specifically within that query, such as using it in SELECT, UPDATE, INSERT, DELETE, CREATE, VIEW, OR MERGE statements.

• CTE is temporary because it cannot be stored anywhere for later use; once the query is executed, it is lost.

Syntax: WITH CTE\_NAME (column name) AS (query)

SELECT \* FROM CTE\_NAME;

Here,

* CTE\_NAME **−** It is the name assigned to the CTE.
* column name **−** It is the column names for the CTE, which can be useful for improving query readability.
* query **−** It defines the CTE, and it can be any valid SQL query.
* After defining the CTE, you can reference it in subsequent queries within the same session.

**Advantages:**

* CTE makes the code maintenance easier.
* It increases the readability of the code.
* It increases the performance of the query.
* CTE allows for the simple implementation of recursive queries.

**Disadvantages:**

* CTE can only be referenced once by the recursive member.
* We cannot use the table variables and CTEs as parameters in a stored procedure.
* A CTE can be used in place of a view, but a CTE cannot be nested while views can.