***NAME: Aniket Biyani***

***DATA ENGINEERING- BATCH 1***

***DAY 2 ASSIGNMENT***

***>>OLAP :-***

*🡪* OLAP,or Online Analytical Processing, is a category of software tools and systems that enable users to interactively analyze and explore multidimensional data from different perspectives.

🡪 Unlike OLTP (Online Transaction Processing), which is focused on transactional data processing, OLAP is designed for complex querying, reporting, and data analysis.

🡪 Key features of OLAP include:

1. Multidimensional Data Model:

OLAP systems organize data into a multidimensional model, typically represented as cubes. This allows users to analyze data along multiple dimensions, such as time, geography, and product categories.

1. Dimension Hierarchies:

Dimensions in OLAP systems are often organized in hierarchies, allowing users to drill down into more detailed levels of data or roll up to higher-level summaries. For example, a time dimension might have hierarchies like year, quarter, month, and day.

1. Aggregation and Summarization:

OLAP systems provide the ability to aggregate and summarize data at different levels of granularity. This facilitates quick analysis of large datasets without the need to retrieve and process every individual record.

1. Fast Query Response Times:

OLAP systems are optimized for fast query response times, enabling users to interactively explore and analyze data. This is in contrast to OLTP systems, which prioritize quick processing of individual transactions.

1. Decision Support and Business Intelligence:

OLAP is a fundamental technology for decision support and business intelligence applications. It allows users to gain insights, discover trends, and make informed decisions based on the analysis of historical and current data.

1. Slice-and-Dice Operations:

Users can "slice" the data by selecting a subset of dimensions, "dice" it by selecting a specific intersection of values, and "pivot" to view the data from different perspectives. These operations provide flexibility in exploring data.

🡪Examples of OLAP applications include financial analysis, sales forecasting, business performance monitoring, and data mining.

🡪OLAP systems are complementary to OLTP systems, together providing a comprehensive solution for managing and analyzing enterprise data.

***>>OLTP:-***

🡪OLTP stands for Online Transaction Processing, and it refers to a class of systems and processes that support and manage transaction-oriented applications.

🡪In an OLTP system, the emphasis is on quickly processing a large number of short, simple transactions. These transactions can include inserting, updating, or deleting records in a database.

🡪Key characteristics of OLTP systems include:

1. Concurrency Control: OLTP systems often involve multiple users simultaneously accessing and modifying the data. Concurrency control mechanisms are employed to ensure that transactions are executed in a way that maintains data consistency.
2. Atomicity, Consistency, Isolation, Durability (ACID): OLTP systems adhere to the ACID properties to ensure the reliability of transactions. ACID guarantees that transactions are processed in a way that maintains the integrity of the data.

*Atomicity*: Transactions are treated as a single, indivisible unit of work. Either all the changes in a transaction are applied, or none of them are.

*Consistency:* Transactions bring the database from one consistent state to another. The database should satisfy certain integrity constraints before and after the transaction.

*Isolation:* Transactions are executed in isolation from each other. The intermediate state of a transaction is not visible to other transactions until it is committed.

*Durability:* Once a transaction is committed, its changes are permanent and survive system failures.

1. High Throughput: OLTP systems are designed for high transaction volume and quick response times. They are optimized for read and write operations on individual records.
2. Normalized Data Structures: OLTP databases are typically normalized to reduce data redundancy and improve data integrity. Normalization involves organizing data in such a way that minimizes redundancy and avoids update anomalies.
3. Data Integrity: Ensuring the accuracy and reliability of data is crucial in OLTP systems. Constraints, such as unique keys and foreign keys, are commonly used to maintain data integrity.

🡪Examples of OLTP applications include online banking systems, order processing systems, airline reservation systems, and retail point-of-sale systems.

🡪In contrast, OLAP (Online Analytical Processing) systems are designed for complex queries and data analysis, often involving large volumes of historical data.

***>>RDBMS:-***

🡪RDBMS stands for Relational Database Management System. It is a type of database management system that is based on the principles of the relational model, as proposed by E.F. Codd in 1970.

🡪RDBMS is a widely used technology for managing and organizing data in a structured and efficient manner.

🡪 Here are the key components and characteristics of an RDBMS:

Tables: Data in an RDBMS is organized into tables, which are two-dimensional structures consisting of rows and columns. Each row in a table represents a record, and each column represents an attribute or field of the record.

Relationships: RDBMS allows the establishment of relationships between tables. These relationships are defined based on common attributes (columns) in different tables. Primary keys and foreign keys are used to enforce integrity and ensure data consistency across related tables.

Data Integrity: RDBMS enforces data integrity through various constraints. Common constraints include primary key constraints (ensuring uniqueness of a record), foreign key constraints (maintaining referential integrity between tables), and check constraints (specifying conditions that data must meet).

ACID Properties: RDBMS adheres to the ACID properties, which ensure the reliability and consistency of transactions. ACID stands for Atomicity, Consistency, Isolation, and Durability. These properties guarantee that database transactions are processed reliably, even in the presence of system failures.

Structured Query Language (SQL): RDBMS uses a standardized language called SQL (Structured Query Language) for defining, manipulating, and querying the database. SQL provides a set of commands for creating tables, inserting, updating, and deleting data, as well as retrieving information from the database.

Normalization: RDBMS supports normalization, a process that organizes data to reduce redundancy and dependency. Normalization helps in avoiding update anomalies and ensures efficient storage and maintenance of data.

🡪Popular examples of RDBMS include MySQL, PostgreSQL, Microsoft SQL Server, Oracle Database, and SQLite.

🡪 RDBMS is widely used in various applications, from business and finance to web development and data-driven applications, due to its ability to efficiently manage structured data and provide a robust framework for data manipulation and retrieval.

***>>SQL AND ITS FEATURES :-***

🡪SQL, or Structured Query Language, is a domain-specific language used for managing and manipulating relational databases.

🡪It is the standard language for interacting with relational database management systems (RDBMS) and is designed to perform tasks such as querying data, inserting, updating, and deleting records, as well as defining and managing the structure of databases.

🡪Here are key features of SQL:

Data Query Language (DQL): SQL provides a set of commands for querying data from a database. The primary DQL command is the SELECT statement, which allows users to retrieve data from one or more tables based on specified criteria.

Ex:- SELECT column1, column2 FROM table WHERE condition;

Data Definition Language (DDL): SQL includes commands for defining and managing the structure of a database. DDL commands are used to create, modify, and delete database objects such as tables, indexes, and constraints.

Example:CREATE TABLE table\_name (

column1 datatype,

column2 datatype,

...);

Data Manipulation Language (DML): SQL provides commands for manipulating data stored in a database. DML commands include INSERT (adding new records), UPDATE (modifying existing records), and DELETE (removing records).

Example :-INSERT INTO table\_name (column1, column2, ...) VALUES (value1, value2, ...);

Data Control Language (DCL): SQL includes commands for managing access to data. DCL commands control privileges and permissions, allowing administrators to grant or revoke access rights to users and roles.

Example :-GRANT SELECT, INSERT ON table\_name TO user;

Transaction Control Statements: SQL supports transactions, allowing users to group a set of SQL statements into a single unit of work. Transaction control statements like COMMIT and ROLLBACK ensure the integrity and consistency of the database.

Example:BEGIN TRANSACTION;

-- SQL statements

COMMIT;

Constraints: SQL allows the definition of constraints on tables to enforce data integrity. Common constraints include primary keys, foreign keys, unique constraints, and check constraints.

Example:CREATE TABLE example (

id INT PRIMARY KEY,

name VARCHAR(50) UNIQUE,

foreign\_key\_id INT REFERENCES another\_table(id)

);

Joins: SQL supports joins to combine rows from two or more tables based on related columns. The most common types of joins are INNER JOIN, LEFT JOIN, RIGHT JOIN, and FULL JOIN.

Example:SELECT \* FROM table1

INNER JOIN table2 ON table1.column = table2.column;

🡪SQL is a powerful language that plays a crucial role in database management. It allows users to interact with and manage data in a relational database system efficiently, providing a standardized and widely accepted way to work with structured data.