Q1)What is OLTP and OLAP?

A) OLTP: OLTP(Online Transaction Processing) is a type of data processing that consists of executing several transactions occurring simultaneously like online banking, shopping, order entry, or sending text messages.

Ex) Examples of OLTP databases include Oracle, SQL Server, MySQL, and PostgreSQL.

**OLAP:** OLAP (Online analytical processing) is software technology you can use to analyze

business data from different points of view. Organizations collect and store data from

multiple data sources, such as websites, applications, smart meters, and internal

systems.

Ex) Examples of OLAP tools include Microsoft Analysis Services, IBM Cognos, SAP

BW (Business Warehouse), and Tableau.

Q2) Differences between OLTP and OLAP?

|  |  |
| --- | --- |
| **OLTP systems** | **OLAP systems** |
| Enable the real-time execution of large numbers of database transactions by large numbers of people | Usually involve querying many records (even all records) in a database for analytical purposes |
| Require lightning-fast response times | Require response times that are orders of magnitude slower than those required by OLTP |
| Modify small amounts of data frequently and usually involve a balance of reads and writes | Do not modify data at all; workloads are usually read-intensive |
| Use indexed data to improve response times | Store data in columnar format to allow easy access to large numbers of records |
| Require frequent or concurrent database backups | Require far less frequent database backup |
| Require relatively little storage space | Typically have significant storage space requirements because they store large amounts of historical data |
| Usually run simple queries involving just one or a few records | Run complex queries involving large numbers of records |

Q3) What are Database Normal Forms (5 Normal forms)?

A) **Types of Normal Forms in DBMS**

## 1NF (First Normal Form): Ensures that the database table is organized such that each column contains indivisible values, and each record is unique. This eliminates repeating groups, thereby structuring data into tables and columns.

## 2NF (Second Normal Form): Builds on 1NF but we need to remove duplicate data from a table that is being applied to multiple rows. and placing them in separate tables. It requires all non-key attributes to be fully functional on the primary key.\*(Primary Key: a column in a relational database table that's distinctive for each record. It's a unique identifier number)

## 3NF (Third Normal Form): Extends 2NF by ensuring that all non-key attributes are not only fully functional on the primary key but also independent of each other. This eliminates \*transitive dependency(Indirect dependency relationship between software components)

1. **BCNF (Boyce-Codd Normal Form):** A refinement of 3NF that addresses anomalies not handled by 3NF. It requires every determinant to be a candidate key, ensuring even stricter adherence to normalization rules.
2. **4NF (Fourth Normal Form):** Addresses multi-valued dependencies. It ensures that there are no multiple independent multi-valued facts about an entity in a record.
3. **5NF (Fifth Normal Form)**: Also known as “Projection-Join Normal Form” (PJNF), It pertains to the reconstruction of information from smaller, differently arranged data pieces.

Q4) Dimension vs Fact Table and Types of Dimensions?

A)

a)

|  |  |
| --- | --- |
| Dimension Table | Fact Table |
| Contains descriptive attributes that provide context to the measurements stored in the fact table | Contains quantitative measurements, metrics, or facts that are typically numeric in nature. |
| These attributes are used for filtering, grouping, and aggregating data during analysis. | These measurements represent business transactions, events, or activities that are being analyzed. |
| Dimension tables are typically narrower but taller compared to fact tables, as they contain fewer rows but more columns. | Fact tables usually have many rows and fewer columns compared to dimension tables. |
| Examples of dimension tables include Customer, Product, Time, Geography, and Employee tables. | Examples of fact tables include Sales, Orders, Inventory, and Financial Transactions tables. |

b) Types of Dimensions:

1. **Conformed Dimension**: A conformed dimension is shared across multiple fact tables in a data warehouse or across different data marts, ensuring consistency and enabling integration and comparison of data from various sources or business processes.
2. **Junk Dimension**: A junk dimension is a compact, denormalized table that consolidates low-cardinality attributes or flags unsuitable for main dimension tables. Junk dimensions streamline queries by merging multiple flags or attributes into one table, reducing the number of dimension tables.
3. **Degenerate Dimension**: A degenerate dimension is an attribute stored directly within the fact table, bypassing a separate dimension table. Usually holding transactional attributes or identifiers, degenerate dimensions lack their independent dimension table.
4. **Role-Playing Dimension:** A role-playing dimension is a table referenced multiple times in a fact table, each time portraying a distinct role or viewpoint.

For instance, a Time dimension might appear several times in a fact table, portraying various dates like Order Date, Ship Date, and Delivery Date.

Q5)Differentiate between Snowflake Vs Star Schema?

A)

|  |  |  |
| --- | --- | --- |
| **Feature** | **Snowflake Schema** | **Star Schema** |
| Structure | Normalized | Denormalized |
| Table Relationships | Complex relationships | Simple relationships |
| Query Performance | Potential impact on performance | Generally better performance |
| Storage Efficiency | Better for large dimensions | Redundant storage |
| Scalability | More scalable | May face challenges |
| Maintenance | Requires more maintenance | Requires less maintenance |
| Flexibility | More flexible | Less flexible |