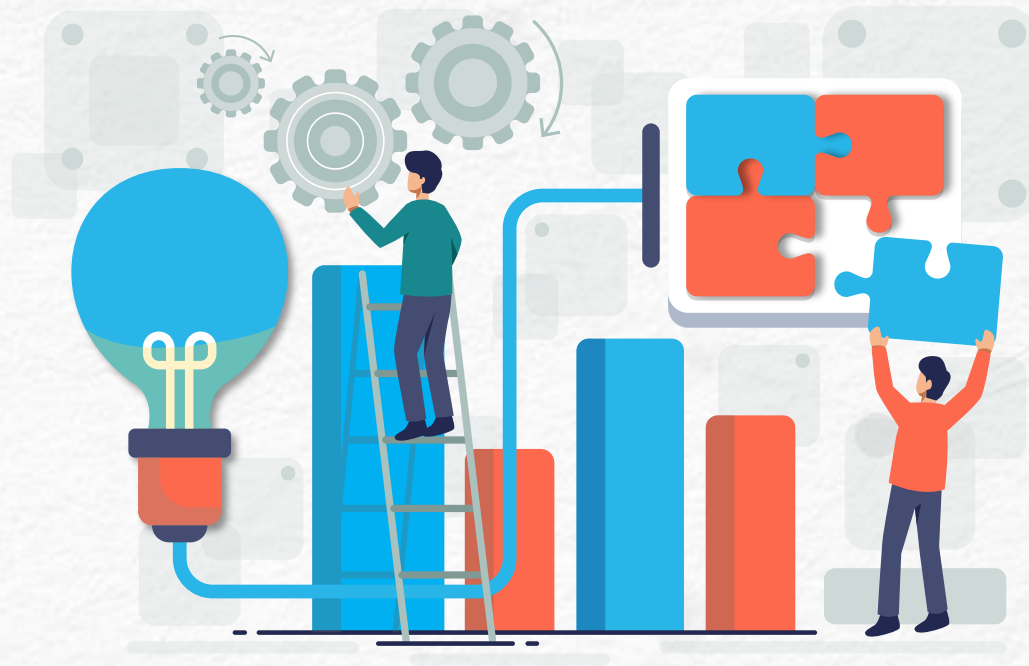


Types of Table In Snowflake



Snowflake offers various types of tables, each suited for specific use cases:

- **Permanent Tables**
- **Temporary Tables**
- **Transient Tables**
- **External Tables**
- **Dynamic Tables**
- **Iceberg Tables**
- **Hybrid Tables**

Permanent Tables:

Permanent tables store data permanently in Snowflake, with automatic handling of storage, backup, and recovery.

When to Use: Use for long-term, critical data that needs to be frequently queried and recoverable in case of failures.

Temporary Tables:

Temporary tables exist only during a session and are automatically dropped afterward. They do not persist across sessions or users.

When to Use: Use for session-specific data that doesn't need to be stored permanently, such as intermediate processing results.

Transient Tables:

Transient tables are similar to permanent tables but lack fail-safe and time travel features, making them more cost-effective for short-lived data. Data can be deleted without recovery options.

When to Use: Use for temporary data across sessions where fail-safe recovery isn't needed, especially for non-critical data where cost reduction is a priority.

External Tables:

External tables allow you to query data stored outside Snowflake (e.g., in cloud storage like S3, Azure, or Google Cloud) without importing it into Snowflake.

When to Use: Use when you need to query external data directly without loading it into Snowflake, ideal for integrating data from data lakes or cloud storage while reducing storage costs.

Dynamic Tables:

Dynamic tables in Snowflake automatically update and maintain real-time data from external sources, allowing efficient data synchronization.

When to Use: Use when you need near real-time data updates and want to reduce manual intervention for keeping tables in sync with external data sources.

Iceberg Tables:

Iceberg tables are a type of data structure for storing large datasets, designed for fast queries, data consistency, and scalable processing across distributed systems.

When to Use: Use for large-scale, high-performance analytics where managing large datasets with complex schema evolution and version control is required.

Hybrid Tables:

Hybrid tables combine features of both **external** and **internal** tables, allowing data to be queried directly from external sources while also storing some data internally for performance optimization.

When to Use: Use when you need to optimize query performance by combining external data with internal storage, offering a balance between cost and speed for large datasets.

Stay Tuned For Next Post...

