1. **Zero copy cloning**

Zero copy cloning is a feature in snowflake which creates exact copy of the database, table or schema without the need for extra storage space. Through zero-copy cloning, Snowflake makes it simple to make backup copies of our data without increasing our storage needs or requiring us to wait for extended periods.

1. **Time Travel**

It is a feature that enables restoring the data that might have been accidentally or intentionally deleted at any point within the defined period. We can access the historical data using TIME\_TRAVEL function. Data upto 1- 90 days can be recovered using this feature.

1. **Transient table vs permanent table vs temporary table vs external table**

Transient table: It persists until user drops or deletes them and can be accessed by multiple users. Time travel feature is for 0 – 1 day but doesn’t support Fail safe feature.

Permanent table: Permanent tables are created in database so they persist until we delete or drop them from database. So the data that requires highest level of protection and recovery are present in these tables. It supports time travel feature for 90 days and also fail safe mechanism.

Temporary table: Temporary tables persist for shorter duration like for a particular session and when the session is completes the table is dropped. It is not fail safe and time travel is applicable for 0 – 1 day only.

External table: This table persists until removed but they can’t be dropped. This is only meant for reading where data can be accessed via an external stage. This doesn’t support time travel and fail safe mechanism.

1. **Fail safe and retention implementation on all the tables**

Fail safe provides 7-day period during which historical data can be recovered, it basically starts after the Time travel retention period ends. Data recovery through fail safe can take several hours to several days.

1. **What will happen in zero copy cloning when you change the data on source**

Cloned objects will not be effected when there is change in data source because once the zero copy clone is created it will automatically become independent entity with its own metadata and storage.

1. **What will happen to source when there is change in zero copy cloning**

Cloned objects are separate from the original table so we can update them at any point without having effect on source.

1. **What is task in snowflake**

Snowflake tasks are basically the schedulers that can assist in scheduling a single SQL query or stored procedure. Tasks can be combined with table streams for continuous ELT workflows to process recently changed table rows or they also can be used independently to generate periodic reports by inserting or merging rows into a report table or perform other periodic work.

1. **How to load 10k records from salesforce to snowflake**

We can use salesforce BULK API to extract data from salesforce as it can be used to query a large number of records from a few thousand to millions in the background and also supports different file formats and create batch so that data can be retrieved. In snowflake we should create stage and we can use them to bulk load the generated Salesforce JSON file and load data into Snowflake

1. **How do you perform or increase the performance tuning in your project**

To increase the performance tuning the first thing that needs to be taken care of is choosing the right virtual machine/warehouse (compute size) based on the workload so that we don’t waste any CPU or memory resources unnecessarily. And for the large data sets the data should be distributed across multiple storage containers to enable parallel processing and improve query performance. SQL queries in snowflake can also be optimized using EXPLAIN PLAN and QUALIFY commands. Addressing data skew issues by redistributing or reorganizing data to ensure even distribution across nodes.

1. **How do you handle duplicates in kafka**

We can handle duplicates in Kafka by configuring Kafka producer to be idempotent so that even if message is sent multiple times, the resulting state is same as if it had been sent only once.

And one more way is by adding timestamp to each message either in message payload or headers. At the consumer side we can keep track of latest processed message timestamp so that older timestamp messages can be ignored.

By implementing message deduplication at the producer side before sending messages to Kafka. You can assign a unique identifier (message ID) to each message and use this ID to detect and eliminate duplicates before producing the message.

1. **How do you read data from external api in json format and send notifications to specific business units**

In my project I can configure logic apps to retrieve data from Aetna, regulatory warehouse, and store this data. So first I use a schedule based trigger to receive data every day at 4 AM to automate the execution pipeline. I configure Logic apps by making HTTP Requests where we can provide APIs to retrieve data using HTTP Connector providing necessary end point URL, authentication details. After retrieving the data from external source I will add **Insert row** action to store the data into Azure SQL database. Then after this I added **publish event** action to send notifications to specific business event.