**Azure databricks coding challenge Q-2**

**2. EXPLAIN OVERVIEW OF 3 LEVEL NAMESPACE AND CREATING UNITY CATALOG OBJECTS.**

Unity Catalog provides centralized access control, auditing, lineage, and data discovery capabilities across Azure Databricks workspaces.

Metastore is the top-level container of the objects in the azure storage account.

**Catalog:**

Catalog is the first layer in the 3-level namespace, it provides access privilege to the user.

**Schema:**

Schema is the second layer of the 3-level namespace, we generally use the default schema. Default schema will be present in the main catalog.

**Tables:**

Tables are the third layer of the 3-level namespace, it contains rows of data. Tables can be managed or external.

**Managed tables:**

Default tables that are created in unity catalog. They are stored in root storage location that is connected to the metastore. No open-source tools should be used to change the schema or data in the table.

They always use delta table format.

**External Tables:**

External Tables are tables in which whole data lifecycles and file layout are not managed by unity catalog and will be stored in the tabular data. Externally managed data file format could be DELTA, CSV, JSON, AVRO, PARAQUET, ORC, TEXT.

**Advantages of a Three-Level Namespace:**

**More granular organization:** The primary advantage of using a three-level namespace in Unity Catalogue is that it provides a more granular organization of data. This is especially important when dealing with large datasets and complex data requirements.

**Prevent naming conflicts:** Another advantage of using a three-level namespace is that it helps to prevent naming conflicts. With a three-level namespace, each object within a catalog and schema has a unique name, making it easier to manage data and reduce conflicts.

**Manage security and access control:** You can assign permissions at each level of the namespace, allowing users to access specific catalogs, schemas, or objects based on their roles or responsibilities.

**CREATING UNITY CATALOG OBJECTS:**

To create a unity catalog object, there are some of the steps to be followed:

* Attach your workspace to a Unity Catalog metastore:

Your workspace must be enabled for Unity Catalog, which means

it must be attached to a Unity Catalog metastore

* Add users and assign the workspace admin role:

As a workspace admin, you can add and invite users to the workspace, can assign the workspace admin role to other users, and can create service principals and groups.

* Create clusters or SQL warehouses that users can use to run queries and create objects:

We can create cluster or run SQL warehouses in the data bricks. After that we can run the queries.

* Grant privileges to users:

If your workspace includes the automatically-provisioned main catalog, users have some privileges on that catalog by default.

* Create new catalogs and schemas

Finally, we can create a new catalog and run them.