

Exercise: Mini Project Using Unity Catalog and Data Governance

Objective:

Develop a mini project using Unity Catalog to demonstrate key data governance capabilities such as **Data Discovery**, **Data Audit**, **Data Lineage**, and **Access Control**.

Part 1: Setting Up the Environment

Task 1: Create a Metastore

- Set up a Unity Catalog metastore that will act as the central location to manage all catalogs and schemas.

Task 2: Create Department-Specific Catalogs

- Create separate catalogs for the following departments:
 - Marketing
 - Engineering
 - Operations

Task 3: Create Schemas for Each Department

- Inside each catalog, create specific schemas to store different types of data, e.g.:
 - For the Marketing catalog, create schemas such as `ads_data` and `customer_data`.
 - For the Engineering catalog, create schemas such as `projects` and `development_data`.
 - For the Operations catalog, create schemas such as `logistics_data` and `supply_chain`.
-

Part 2: Loading Data and Creating Tables

Task 4: Prepare Datasets

- Use sample datasets for each schema (create CSV or JSON files if required):
 - **Marketing - Ads Data:** Contains columns such as `ad_id`, `impressions`, `clicks`, `cost_per_click`.
 - **Engineering - Projects:** Contains columns such as `project_id`, `project_name`, `start_date`, `end_date`.
 - **Operations - Logistics:** Contains columns such as `shipment_id`, `origin`, `destination`, `status`.

Task 5: Create Tables from the Datasets

- Load the datasets into their respective schemas as tables.
 - Example: Create a table for `ads_data` in the `marketing` catalog.
 - Example: Create a table for `projects` in the `engineering` catalog.
-

Part 3: Data Governance Capabilities

Data Access Control

Task 6: Create Roles and Grant Access

- Create specific roles for each department and grant access to the relevant catalogs and schemas.
 - For example: create roles such as `marketing_role`, `engineering_role`, and `operations_role`.

Task 7: Configure Fine-Grained Access Control

- Set up fine-grained access control, where users in the marketing department can only access customer-related data, while engineers can only access project data. Define permissions accordingly.

Data Lineage

Task 8: Enable and Explore Data Lineage

- Enable data lineage for the tables created in Part 2.
- Perform some queries (e.g., aggregate queries) on the datasets and examine how the data lineage feature traces the origin of data and tracks transformations.

Data Audit

Task 9: Monitor Data Access and Modifications

- Set up audit logging to track who is accessing or modifying the datasets. Access the audit logs to view data access patterns and identify who performed which actions on the data.

Data Discovery

Task 10: Explore Metadata in Unity Catalog

- Explore the metadata of the tables you've created. Document information such as table schema, number of rows, and table properties for each department.
- Make sure that the appropriate descriptions and properties are added to each catalog, schema, and table.

Deliverables:

- Department catalogs, schemas, and tables created in Unity Catalog.
- Access roles and controls in place for each department.
- Demonstrations of data governance capabilities such as **Data Lineage**, **Data Audit**, and **Data Discovery**.