**Mckesson Databricks Column Masking Workflow**

Databricks column masking uses a combination of Excel for the masking input parameters which is hosted in an Azure Dev Ops repository. There is an Azure Dev Ops (ADO) pipeline which starts and copies the Excel input parameters template into an Azure Storage Account container folder. Once the file lands there a Databricks workflow is triggered to update an ‘audit\_rules’ managed Unity Catalog Delta table with new masking rules, and the masking rules will be applied to the appropriate tables and columns within a Domain Environment.

Masking Repo: <https://dev.azure.com/mckessontech/mtdatamesh/_git/mtdatamesh_databricks?path=%2F&version=GBDatamasking_Upload>

A screenshot of a computer

Description automatically generated

**What are each of the following files / folders in the Repo for:**

1. **ddm**, **frm**, **mm**, **pm**, and **sm** folder represent each Domain, and within each of these folders you will find a **dev**, **qa**, **uat**, and **prod** folder. Within the environment folders you will find a separate Excel “datamasking\_template.xlsx” which has masking rules for each Domain Environment.
2. The **data\_masking.py** is the **main** Databricks notebook is called from ADO by the datamasking\_workflow in Databricks that orchestrates all the column masking. It also calls the **import\_data\_from\_excel.py”** notebook.
3. **datamasking\_template.xlsx** is an example file to show an end user what user input is needed to use the datamasking workflow.
4. The **mount\_storage\_account.py** is a notebook which is run in order to create a mount point to the storage account container where the **datamasking\_template.xlsx** is copied into.
5. The **domain\_manifest.json** contains environment specific information (e.g. resource group name, region, subscription, etc) and it is read by the **azure-pipelines.yml** file. This manifest file is important to reduce or dependency on needing so many variable groups and allows us to us a single ADO pipeline to deploy column masking between Domain Environments.
6. The **azure-pipelines.yml** file is called by an ADO Pipeline, and it copies the **datamasking\_template.xlsx** and it copies the **datamasking\_template.xlsx** into the appropriate Storage Account Container.

**What Storage Account Containers and folders are used?**

A screenshot of a computer

Description automatically generated

**What Azure Dev Ops (ADO) Pipelines are used?**

<https://dev.azure.com/mckessontech/mtdatamesh/_build?definitionId=52>

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**Where is the data masking workflow in Databricks?**

<https://adb-8619450367578475.15.azuredatabricks.net/?o=8619450367578475#job/666395697975975>

**A screenshot of a computer

Description automatically generated**

Update workflow parameters to reflect Domain Environment

**A screenshot of a computer

Description automatically generated**

**What improvements can be made?**

A new file trigger should be setup on the data masking workflow using an external location so when the ADO pipeline copies the **datamasking\_template.xlsx** in the storage account container the data masking workflow will automatically start.

**Appendix**

File Arrival Triggers Documentation:

<https://learn.microsoft.com/en-us/azure/databricks/workflows/jobs/file-arrival-triggers>

Row and Column Filters in Unity Catalog:

<https://learn.microsoft.com/en-us/azure/databricks/data-governance/unity-catalog/row-and-column-filters>

Unity Catalog Volumes (Replaces DBFS Mount Points):

<https://learn.microsoft.com/en-us/azure/databricks/data-governance/unity-catalog/create-volumes>