**Working with Databricks Utilities (dbutils)**

**Objective**

* Learn to use dbutils commands for file management, secrets, widgets, and notebooks.
* Practice in a Databricks notebook environment.

**Pre-requisites**

1. Azure Databricks workspace & cluster.
2. Access to ADLS Gen2 or DBFS.
3. (Optional) Key Vault integration for Secrets scope.

**Step 1 – File System Utilities (dbutils.fs)**

**1.1 Create directories**

# Create a folder in DBFS

dbutils.fs.mkdirs("/mnt/lab/dbutils\_demo")

**1.2 Write & read files**

# Write a file

dbutils.fs.put("/mnt/lab/dbutils\_demo/sample.txt", "Hello, Databricks Utils Lab!", overwrite=True)

# Read file

display(dbutils.fs.head("/mnt/lab/dbutils\_demo/sample.txt"))

**1.3 List files & delete**

# List files

display(dbutils.fs.ls("/mnt/lab/dbutils\_demo"))

# Remove file

dbutils.fs.rm("/mnt/lab/dbutils\_demo/sample.txt")

**Step 2 – Notebook Utilities (dbutils.notebook)**

1. Create two notebooks:
   * MainNotebook
   * ChildNotebook

**ChildNotebook code:**

dbutils.notebook.exit("Hello from Child Notebook!")

**MainNotebook code:**

result = dbutils.notebook.run("ChildNotebook", 60)

print("Result from child:", result)

**Step 3 – Secrets Utilities (dbutils.secrets)**

**3.1 Setup**

* In Azure Databricks, create a **Secret Scope** linked to **Azure Key Vault**.
* Store a secret (e.g., "sql-password").

**3.2 Use in notebook**

# Get secret

password = dbutils.secrets.get(scope="kv-scope", key="sql-password")

print("Password retrieved securely (masked).")

**Step 4 – Widgets Utilities (dbutils.widgets)**

Widgets allow parameterization of notebooks.

**4.1 Create input widgets**

# Create widgets

dbutils.widgets.text("input\_text", "default", "Enter a Value")

dbutils.widgets.dropdown("region", "North", ["North","South","East","West"], "Select Region")

dbutils.widgets.combobox("year", "2025", ["2024","2025","2026"], "Choose Year")

**4.2 Access values**

# Get widget values

val1 = dbutils.widgets.get("input\_text")

region = dbutils.widgets.get("region")

year = dbutils.widgets.get("year")

print(f"Text: {val1}, Region: {region}, Year: {year}")

**4.3 Remove widgets**

dbutils.widgets.removeAll()

**Step 5 – Library Utilities (dbutils.library)**

Install packages dynamically.

# Install a PyPI library

dbutils.library.installPyPI("pandas")

# Restart cluster for changes

dbutils.library.restartPython()

**Step 6 – Jobs Utilities (dbutils.jobs)**

Simulate job task parameters.

# If running as a Databricks Job

param1 = dbutils.jobs.taskValues.get(taskKey="previous\_task", key="result", debugValue="default")

print("Value from previous task:", param1)

**Step 7 – Misc Utilities (dbutils.jobs, dbutils.fs.help)**

# Show available commands in each utility

dbutils.fs.help()

dbutils.widgets.help()

dbutils.secrets.help()