USN NUMBER: 1RVU23CSE227
NAME: KRUTHIK S

Ex No: 4 **Date: 11-09-2025**	**Building and automating a pipeline in Databricks for both E-commerce and Healthcare datasets**

Objective

This lab provides hands-on experience in implementing the **data engineering lifecycle** using **Databricks**. Participants will simulate responsibilities of data engineers, data scientists, and business analysts by ingesting raw datasets (E-commerce & Healthcare), cleaning and transforming them, aggregating for analytics, and finally creating dashboards with automation and alerts.

Outcomes

- Identify and describe each stage of the **data engineering lifecycle** (Ingestion, Transformation, Aggregation, Visualization, Automation).
- Implement the **Architecture** (Bronze → Silver → Gold) in Databricks using PySpark and SQL.
- Collaborate to define a **business problem** using raw datasets:
  E-commerce: Revenue by product category, daily revenue trends.

  Healthcare: Service category performance, daily hospital revenue trends.

Materials

- **Raw datasets**:
  ecommerce_orders.csv → E-commerce sales transactions

  healthcare_orders.csv → Healthcare services transactions
- **Tools & Environment**:
  Databricks Workspace
  PySpark & SQL
- **Artifacts**:
  Bronze Layer → Raw ingestion tables
  Silver Layer → Cleaned, transformed tables (silver_ecommerce_orders, silver_healthcare_orders)
  Gold Layer → Aggregated analytics tables (gold_ecommerce_category_sales, gold_healthcare_service_sales, etc.)

USN NUMBER: 1RVU23CSE227
NAME: KRUTHIK S

- **Dashboard & Visualization**:
  Bar charts (Revenue by category/service)
  Line charts (Daily revenue trends)
  Filters (City, Payment Method)

- **Automation Setup**:
  Databricks Jobs for scheduled refresh
  Email notifications for success/failure alerts

Lab Procedure:

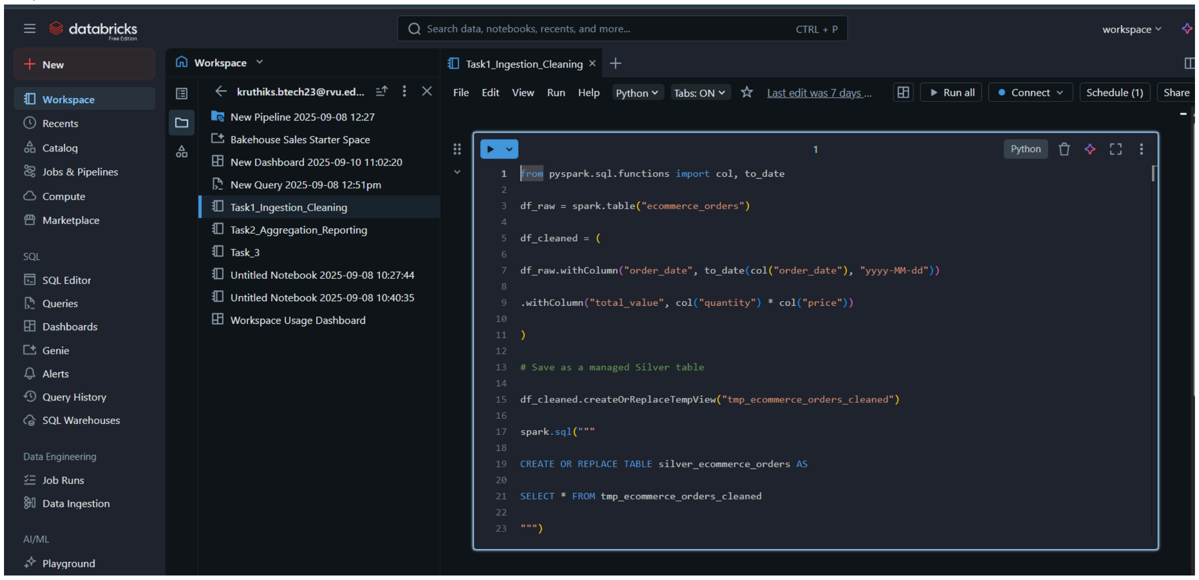
Step 1: Ingestion & Cleaning (Bronze → Silver)

```
from pyspark.sql.functions import col, to_date

# Bronze layer: raw table (already uploaded as healthcare_orders.csv)
df_raw = spark.read.csv("/FileStore/tables/healthcare_orders.csv", header=True, inferSchema=True)

# Clean & transform (Silver layer)
df_cleaned = (
   df_raw.withColumn("order_date", to_date(col("order_date"), "yyyy-MM-dd"))
        .withColumn("total_value", col("quantity") * col("price"))
)

# Save as Silver table
df_cleaned.createOrReplaceTempView("tmp_healthcare_orders_cleaned")
spark.sql("""
CREATE OR REPLACE TABLE silver_healthcare_orders AS
SELECT * FROM tmp_healthcare_orders_cleaned
""")
```


databricks
Search data, notebooks, recents, and more...
workspace
New
Workspace
Recents
Catalog
Jobs & Pipelines
Compute
Marketplace
SQL
SQL Editor
Queries
Dashboards
Genie
Alerts
Query History
SQL Warehouses
Data Engineering
Job Runs
Data Ingestion
AI/ML
Playground
kruthiks.btech23@rvu.ed...
New Pipeline 2025-09-08 12:27
Bakehouse Sales Starter Space
New Dashboard 2025-09-10 11:02:20
New Query 2025-09-08 12:51pm
Task1_Ingestion_Cleaning
Task2_Aggregation_Reporting
Task_3
Untitled Notebook 2025-09-08 10:27:44
Untitled Notebook 2025-09-08 10:40:35
Workspace Usage Dashboard
File Edit View Run Help Python Tabs: ON
Last edit was 7 days ...
Run all
Connect
Schedule (1)
Share

```
from pyspark.sql.functions import col, to_date

df_raw = spark.table("ecommerce_orders")

df_cleaned = (

df_raw.withColumn("order_date", to_date(col("order_date"), "yyyy-MM-dd"))

.withColumn("total_value", col("quantity") * col("price"))

)

# Save as a managed Silver table

df_cleaned.createOrReplaceTempView("tmp_ecommerce_orders_cleaned")

spark.sql("""

CREATE OR REPLACE TABLE silver_ecommerce_orders AS

SELECT * FROM tmp_ecommerce_orders_cleaned

""")
```

Step 2: Aggregation & Enrichment (Silver → Gold)

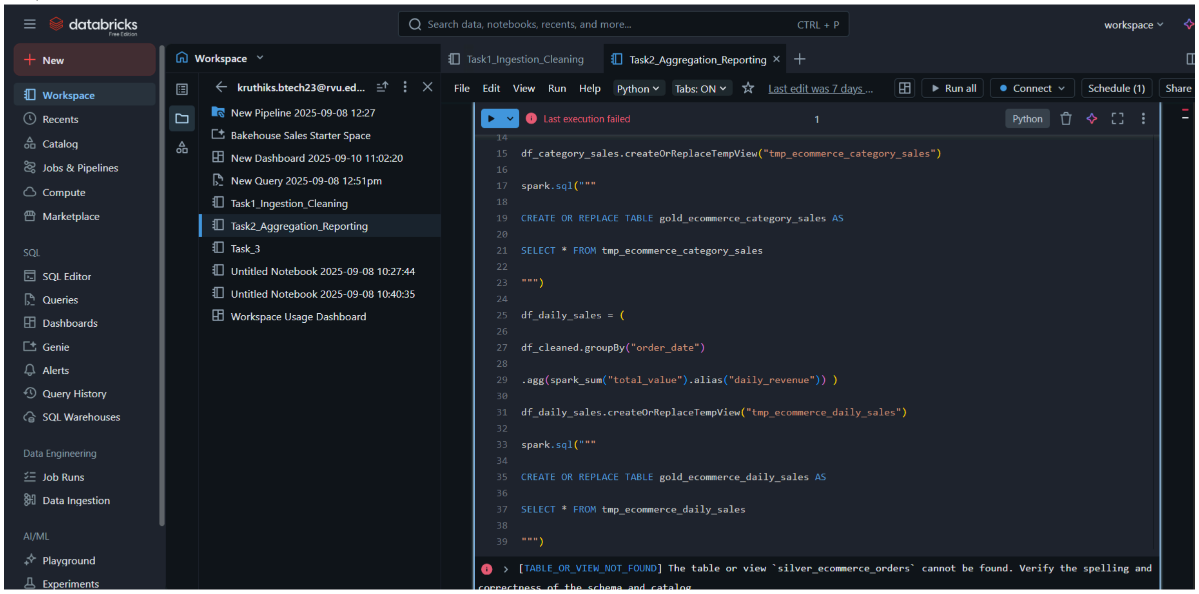
USN NUMBER: 1RVU23CSE227
NAME: KRUTHIK S

```
from pyspark.sql.functions import sum as spark_sum

# Load Silver table
df_cleaned = spark.table("silver_healthcare_orders")

# Gold 1: Revenue by service category
df_service_sales = (
    df_cleaned.groupBy("service_category")
              .agg(spark_sum("total_value").alias("total_revenue"))
)
df_service_sales.createOrReplaceTempView("tmp_healthcare_service_sales")
spark.sql("""
CREATE OR REPLACE TABLE gold_healthcare_service_sales AS
SELECT * FROM tmp_healthcare_service_sales
""")

# Gold 2: Daily revenue trends
df_daily_sales = (
    df_cleaned.groupBy("order_date")
              .agg(spark_sum("total_value").alias("daily_revenue"))
)
df_daily_sales.createOrReplaceTempView("tmp_healthcare_daily_sales")
spark.sql("""
CREATE OR REPLACE TABLE gold_healthcare_daily_sales AS
SELECT * FROM tmp_healthcare_daily_sales
""")
```



```
df_category_sales.createOrReplaceTempView("tmp_ecommerce_category_sales")

spark.sql("""

CREATE OR REPLACE TABLE gold_ecommerce_category_sales AS

SELECT * FROM tmp_ecommerce_category_sales

""")

df_daily_sales = (

df_cleaned.groupBy("order_date")

.agg(spark_sum("total_value").alias("daily_revenue")) )

df_daily_sales.createOrReplaceTempView("tmp_ecommerce_daily_sales")

spark.sql("""

CREATE OR REPLACE TABLE gold_ecommerce_daily_sales AS

SELECT * FROM tmp_ecommerce_daily_sales

""")
```

```
[TABLE_OR_VIEW_NOT_FOUND] The table or view `silver_ecommerce_orders` cannot be found. Verify the spelling and
correctness of the schema and catalog
```

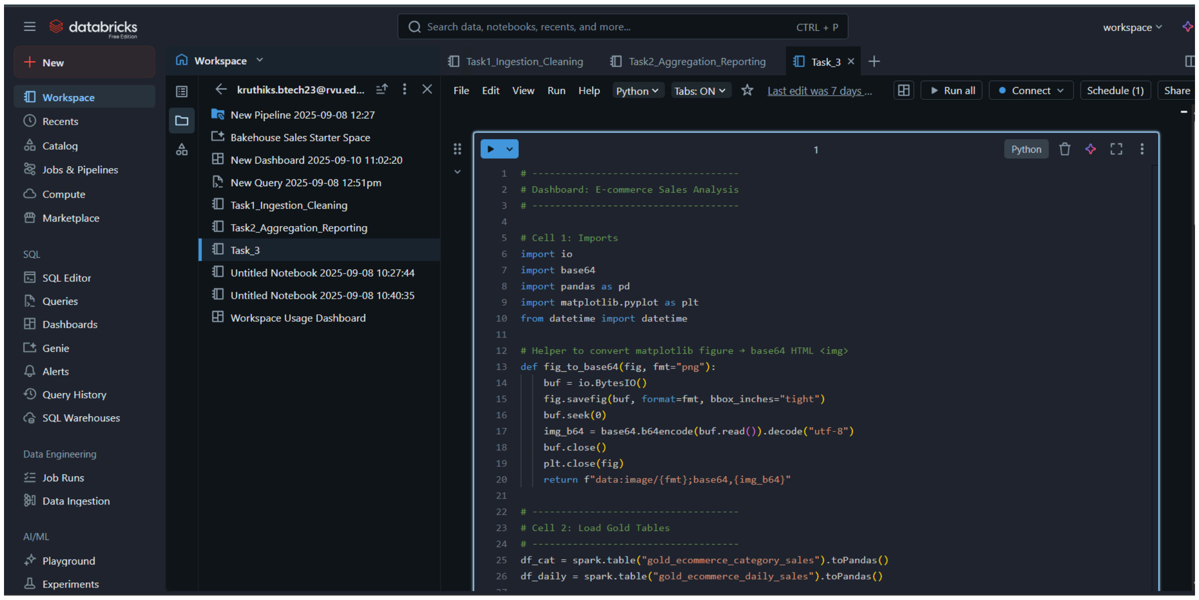
Step 3: Dashboard & Visualization

In Databricks Notebook:

- **Bar Chart**: gold_healthcare_service_sales → service_category vs total_revenue

USN NUMBER: 1RVU23CSE227
NAME: KRUTHIK S

- **Line Chart**: gold_healthcare_daily_sales → order_date vs daily_revenue
- Add **filters**: city, payment_method

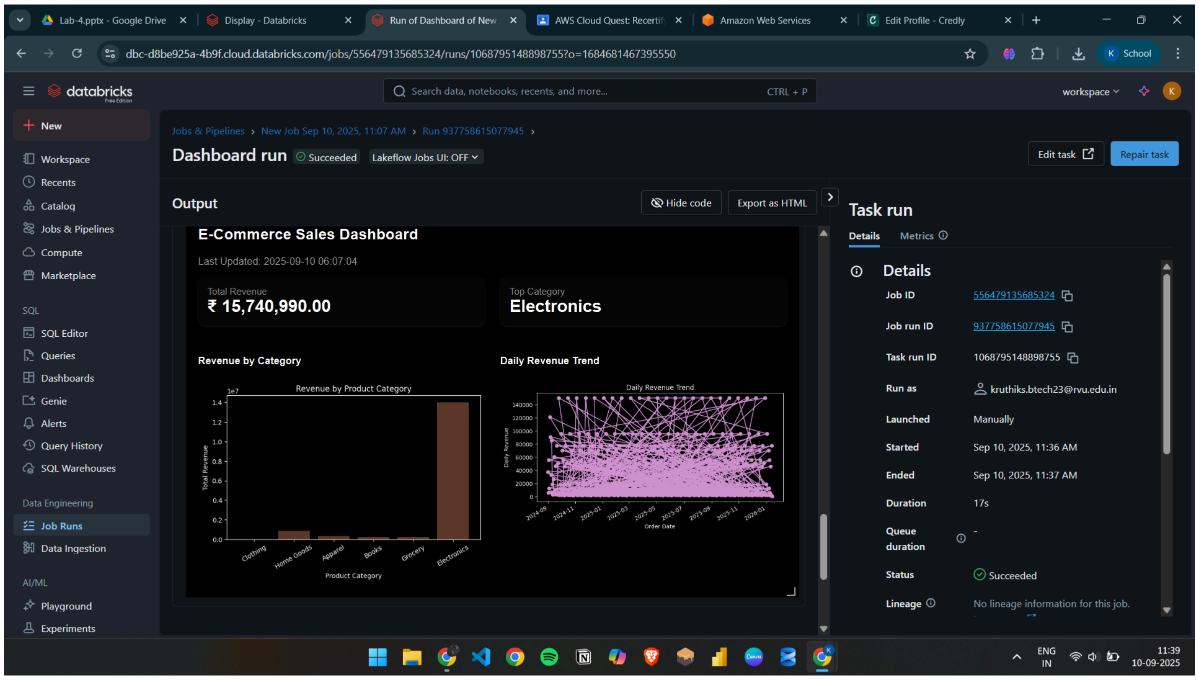

databricks
New
Workspace
Recents
Catalog
Jobs & Pipelines
Compute
Marketplace
SQL
SQL Editor
Queries
Dashboards
Genie
Alerts
Query History
SQL Warehouses
Data Engineering
Job Runs
Data Ingestion
AI/ML
Playground
Experiments
kruthiks.btech23@rvu.ed...
New Pipeline 2025-09-08 12:27
Bakehouse Sales Starter Space
New Dashboard 2025-09-10 11:02:20
New Query 2025-09-08 12:51pm
Task1_Ingestion_Cleaning
Task2_Aggregation_Reporting
Task_3
Untitled Notebook 2025-09-08 10:27:44
Untitled Notebook 2025-09-08 10:40:35
Workspace Usage Dashboard
File Edit View Run Help Python Tabs: ON Last edit was 7 days ...
Run all Connect Schedule (1) Share

```python
# -----------------------------------
# Dashboard: E-commerce Sales Analysis
# -----------------------------------

# Cell 1: Imports
import io
import base64
import pandas as pd
import matplotlib.pyplot as plt
from datetime import datetime

# Helper to convert matplotlib figure → base64 HTML <img>
def fig_to_base64(fig, fmt="png"):
    buf = io.BytesIO()
    fig.savefig(buf, format=fmt, bbox_inches="tight")
    buf.seek(0)
    img_b64 = base64.b64encode(buf.read()).decode("utf-8")
    buf.close()
    plt.close(fig)
    return f"data:image/{fmt};base64,{img_b64}"

# -----------------------------------
# Cell 2: Load Gold Tables
# -----------------------------------
df_cat = spark.table("gold_ecommerce_category_sales").toPandas()
df_daily = spark.table("gold_ecommerce_daily_sales").toPandas()
```



Task run

Details

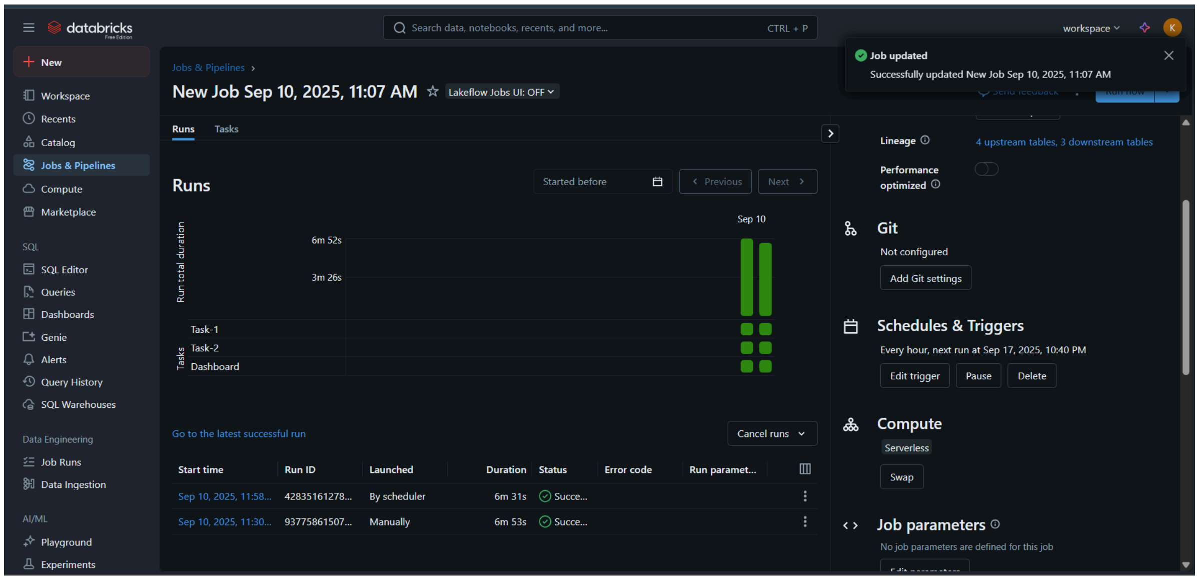
Job ID	556479135685324
Job run ID	937758615077945
Task run ID	1068795148898755
Run as	kruthiks.btech23@rvu.edu.in
Launched	Manually
Started	Sep 10, 2025, 11:36 AM
Ended	Sep 10, 2025, 11:37 AM
Duration	17s
Queue duration	-
Status	Succeeded
Lineage	No lineage information for this job.

Step 4: Automation & Scheduling

- Go to **Databricks Jobs** → create a Job for this Notebook.
- Set **schedule** = daily/weekly refresh.

USN NUMBER: 1RVU23CSE227
NAME: KRUTHIK S

- Configure **task order**: Bronze → Silver → Gold → Dashboard.



Job updated
Successfully updated New Job Sep 10, 2025, 11:07 AM

New Job Sep 10, 2025, 11:07 AM

Runs

Sep 10
6m 52s
3m 26s
Task-1
Task-2
Dashboard

Go to the latest successful run

Start time	Run ID	Launched	Duration	Status	Error code	Run paramet...
Sep 10, 2025, 11:58...	42835161278...	By scheduler	6m 31s	Succe...		
Sep 10, 2025, 11:30...	93775861507...	Manually	6m 53s	Succe...		

Lineage: 4 upstream tables, 3 downstream tables
Performance optimized

Git
Not configured

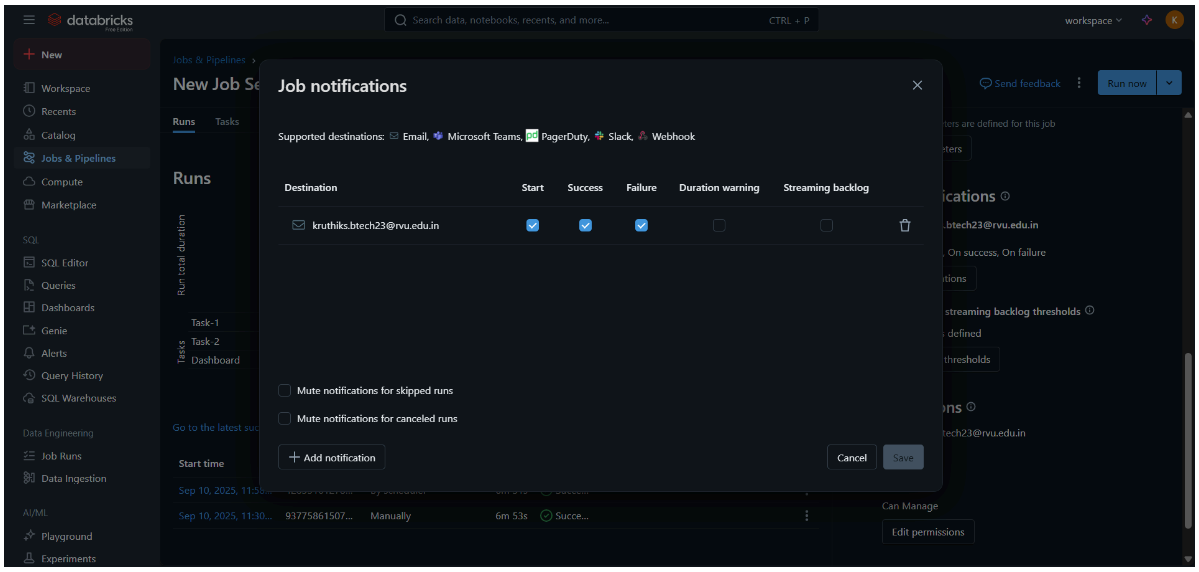
Schedules & Triggers
Every hour, next run at Sep 17, 2025, 10:40 PM

Compute
Serverless

Job parameters
No job parameters are defined for this job

Step 5: Notifications & Alerts

- In Job settings → **Notifications**.
- Add emails for *success* and *failure events*.
- Ensures admins are alerted if ingestion/transform fails.



Job notifications

Supported destinations: Email, Microsoft Teams, PagerDuty, Slack, Webhook

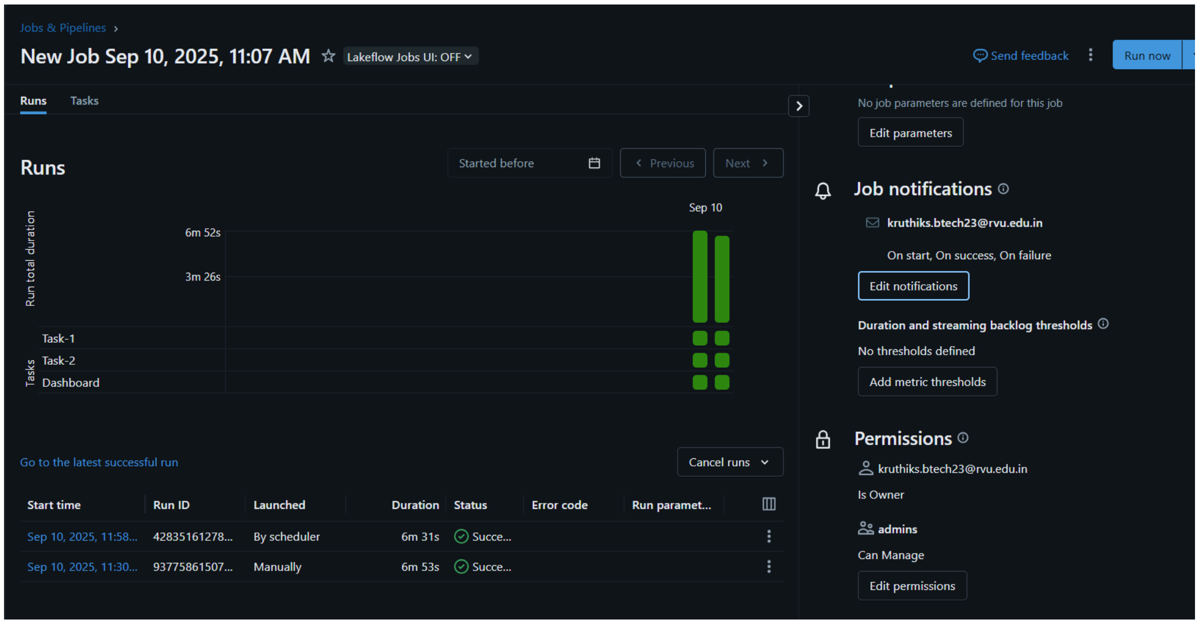
Destination	Start	Success	Failure	Duration warning	Streaming backlog
kruthiks.btech23@rvu.edu.in	✓	✓	✓		

Mute notifications for skipped runs
Mute notifications for canceled runs

Sep 10, 2025, 11:30... 93775861507... Manually 6m 53s Succe...

Can Manage

USN NUMBER: 1RVU23CSE227
NAME: KRUTHIK S


Jobs & Pipelines ›

New Job Sep 10, 2025, 11:07 AM

Lakeflow Jobs UI: OFF

Runs | Tasks

Runs

No job parameters are defined for this job

Job notifications

kruthiks.btech23@rvu.edu.in

On start, On success, On failure

Duration and streaming backlog thresholds

No thresholds defined

Permissions

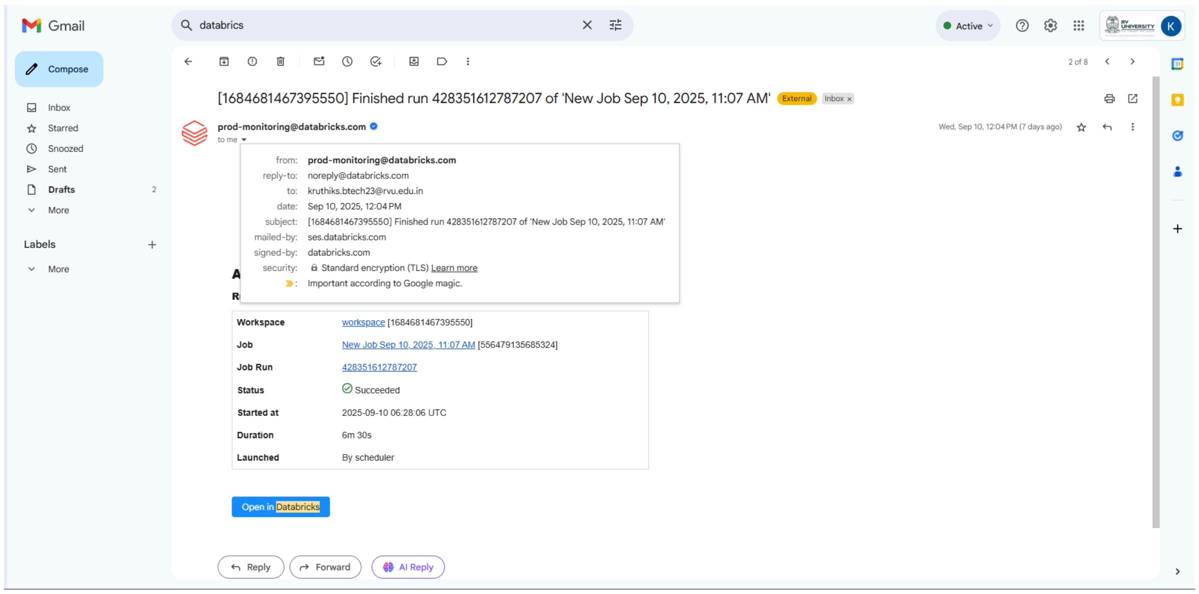
kruthiks.btech23@rvu.edu.in

Is Owner

admins

Can Manage

Start time	Run ID	Launched	Duration	Status	Error code	Run paramet...
Sep 10, 2025, 11:58...	42835161278...	By scheduler	6m 31s	Succe...		
Sep 10, 2025, 11:30...	93775861507...	Manually	6m 53s	Succe...		


[1684681467395550] Finished run 428351612787207 of 'New Job Sep 10, 2025, 11:07 AM'

prod-monitoring@databricks.com
to me

Wed, Sep 10, 12:04 PM (7 days ago)

from:	prod-monitoring@databricks.com
reply-to:	noreply@databricks.com
to:	kruthiks.btech23@rvu.edu.in
date:	Sep 10, 2025, 12:04 PM
subject:	[1684681467395550] Finished run 428351612787207 of 'New Job Sep 10, 2025, 11:07 AM'
mailed-by:	ses.databricks.com
signed-by:	databricks.com
security:	Standard encryption (TLS) Learn more
:	Important according to Google magic.

Workspace	workspace [1684681467395550]
Job	New Job Sep 10, 2025, 11:07 AM [556479135685324]
Job Run	428351612787207
Status	✓ Succeeded
Started at	2025-09-10 06:28:06 UTC
Duration	6m 30s
Launched	By scheduler

GitHub Link: https://github.com/kruth-s/Data-Engg-Lab