## JOB Runtime SLA Breach Databricks

### Scenario:

## Use Case: Job Runtime Logging & SLA Breach Dashboarding

Track and log job start time, end time, duration, and SLA status → store in Delta table → visualize in Power BI/Looker.

### ****Step 1: Define SLA Time (in minutes)****

python

CopyEdit

# Define SLA (expected runtime in minutes)

JOB\_SLA\_MINUTES = 30 # Customize per job

### ****Step 2: Log Start Time at Job Begin****

python

CopyEdit

from datetime import datetime, timezone

job\_id = dbutils.widgets.get("job\_id")

run\_id = dbutils.widgets.get("run\_id")

start\_time = datetime.now(timezone.utc)

### ****Step 3: Your Actual Job Logic****

(Replace this with your data processing logic)

python

CopyEdit

# Simulate work

import time

time.sleep(60) # Simulates 1-minute job

### ****Step 4: Capture End Time and Calculate Duration****

python

CopyEdit

end\_time = datetime.now(timezone.utc)

duration\_minutes = (end\_time - start\_time).total\_seconds() / 60

### ****Step 5: SLA Evaluation + Logging to Delta Table****

python

CopyEdit

from pyspark.sql import SparkSession

from pyspark.sql.functions import lit

spark = SparkSession.builder.getOrCreate()

log\_data = [{

"job\_id": job\_id,

"run\_id": run\_id,

"start\_time": start\_time.isoformat(),

"end\_time": end\_time.isoformat(),

"duration\_minutes": round(duration\_minutes, 2),

"sla\_minutes": JOB\_SLA\_MINUTES,

"breached": duration\_minutes > JOB\_SLA\_MINUTES,

"logged\_at": datetime.now(timezone.utc).isoformat()

}]

log\_df = spark.createDataFrame(log\_data)

log\_df.write.mode("append").format("delta").saveAsTable("monitoring.job\_runtime\_log")

### ****Step 6: Create Power BI/Looker Dashboard (Summary)****

**Connect Power BI or Looker to Delta Lake** via Databricks SQL endpoint or connector.

**Suggested Columns to Use in Dashboard**:

* job\_id, run\_id
* start\_time, end\_time
* duration\_minutes
* breached (Boolean → SLA violated)
* Add visualizations:
  + Bar chart: Avg duration vs SLA
  + Filter: breached == True
  + Trend line: Daily average runtime

### Optional: Email Alert / Slack on SLA Breach

You can trigger an alert with dbutils.notebook.exit() and pass breach info to orchestration (e.g., Azure Data Factory, Airflow, or Alert API).

### Delta Table Schema Created:

sql

CopyEdit

DESCRIBE TABLE monitoring.job\_runtime\_log;

-- Columns:

-- job\_id STRING

-- run\_id STRING

-- start\_time TIMESTAMP

-- end\_time TIMESTAMP

-- duration\_minutes DOUBLE

-- sla\_minutes INT

-- breached BOOLEAN

-- logged\_at TIMESTAMP