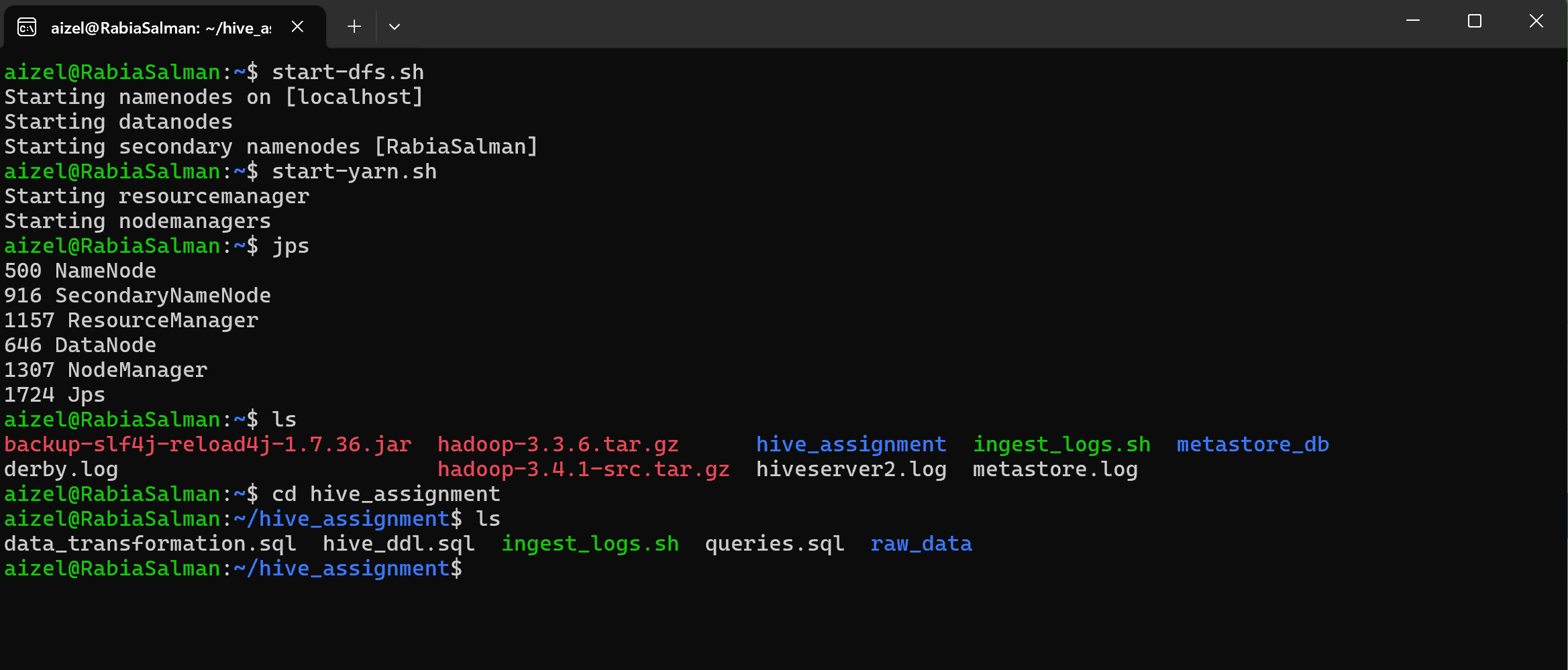
**Hadoop Setup**

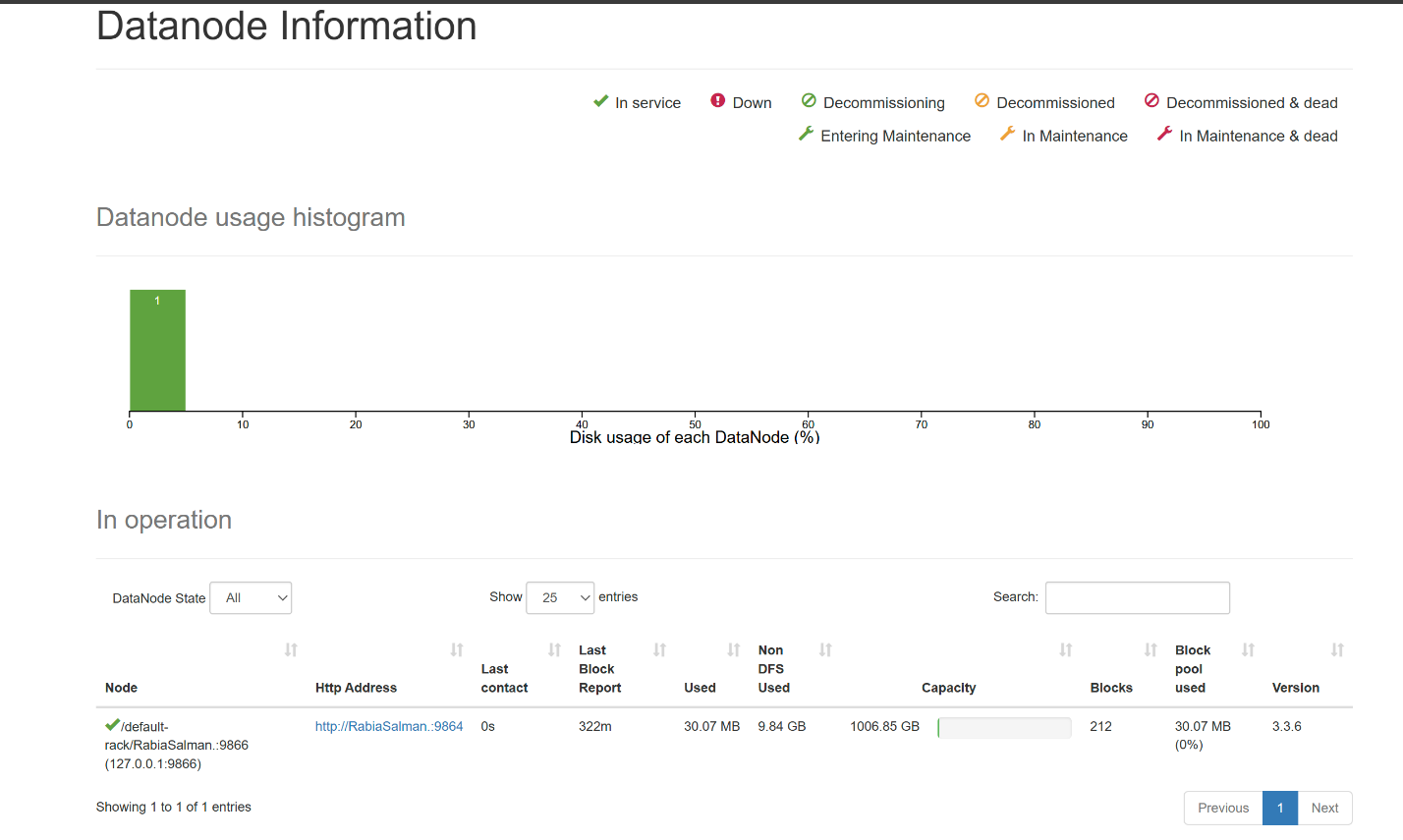
****

**Directories at Localhost**

**A screenshot of a computer

AI-generated content may be incorrect.**

**DataNode Information**

****

**File Information**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Partitions in Hadoop**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Startup Progress**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Hive Tables**

1. **User\_activity\_logs**

CREATE EXTERNAL TABLE IF NOT EXISTS user\_activity\_logs (

user\_id STRING,

action STRING,

`timestamp` STRING,

details STRING

)

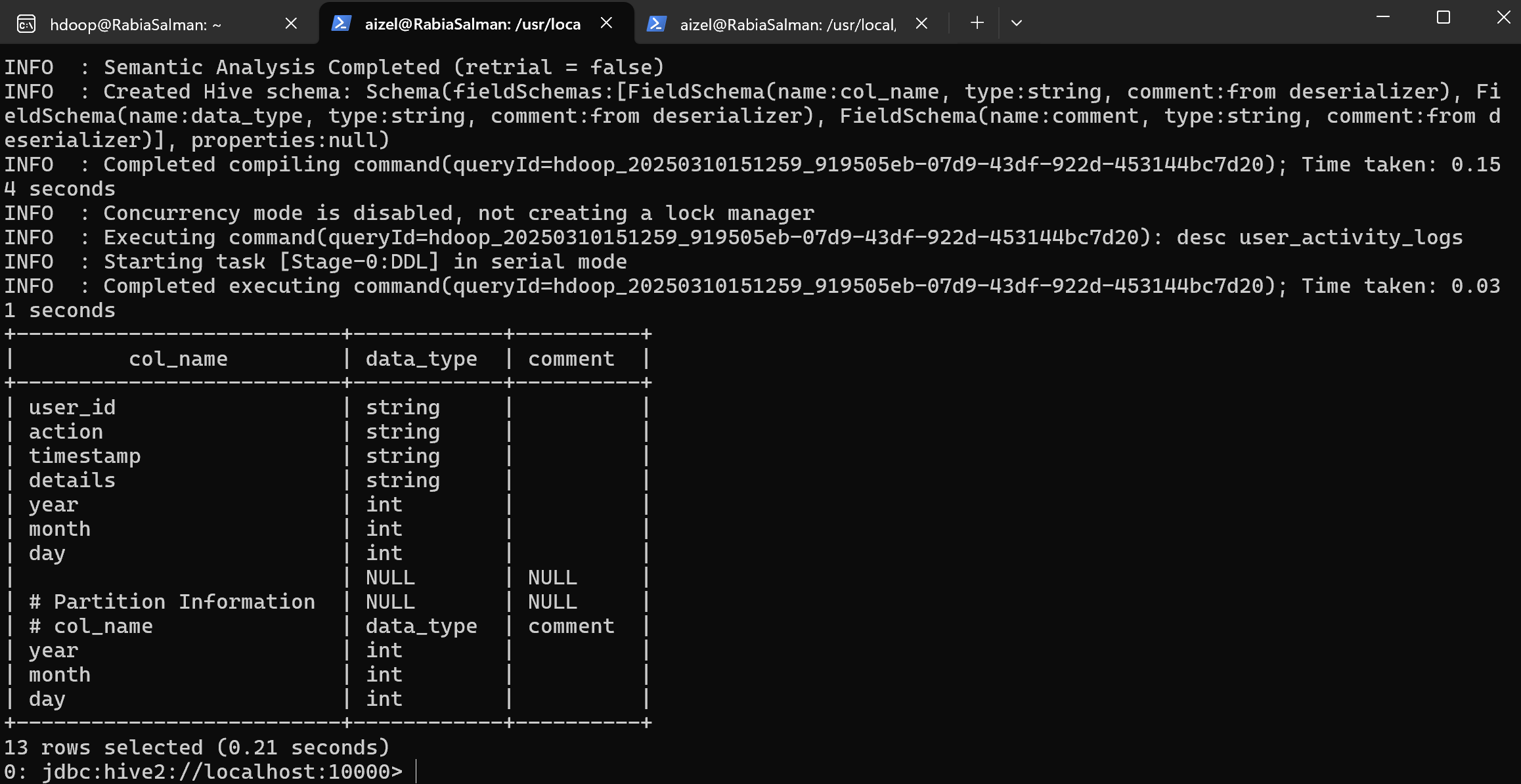
PARTITIONED BY (year INT, month INT, day INT)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ','

STORED AS TEXTFILE

LOCATION 'hdfs://localhost:9000/raw/logs';

****

**Showing Partitions**

SHOW PARTITIONS aizel\_hadoop\_hive.user\_activity\_logs;

**A screenshot of a computer

AI-generated content may be incorrect.**

**Injestion Script**

**#!/bin/bash**

**# File: project-root/scripts/ingest\_logs.sh**

**# Usage: ./ingest\_logs.sh <YYYY-MM-DD>**

**# This script ingests user activity log CSV files into HDFS.**

**if [ "$#" -ne 1 ]; then**

**echo "Usage: $0 <YYYY-MM-DD>"**

**exit 1**

**fi**

**DATE="$1"**

**YEAR=$(echo "$DATE" | cut -d'-' -f1)**

**MONTH=$(echo "$DATE" | cut -d'-' -f2)**

**DAY=$(echo "$DATE" | cut -d'-' -f3)**

**# Local directories for logs and metadata (adjust paths as needed)**

**LOCAL\_LOG\_DIR="C:/Users/aizel/Downloads/LUMS OFFICIAL DATA/Data Eng/labs/data-ingestion-hive/project-root/user\_activity\_logs"**

**LOCAL\_META\_DIR="C:/Users/aizel/Downloads/LUMS OFFICIAL DATA/Data Eng/labs/data-ingestion-hive/project-root/raw\_data/metadata"**

**# HDFS target directories**

**HDFS\_LOG\_DIR="/raw/logs/$YEAR/$MONTH/$DAY"**

**HDFS\_META\_DIR="/raw/metadata/$YEAR/$MONTH/$DAY"**

**# Create HDFS directories**

**hdfs dfs -mkdir -p "$HDFS\_LOG\_DIR"**

**hdfs dfs -mkdir -p "$HDFS\_META\_DIR"**

**# Ingest log file (if exists) into HDFS**

**LOG\_FILE="$LOCAL\_LOG\_DIR/user\_activity\_logs\_${DATE}.csv"**

**if [ -s "$LOG\_FILE" ]; then**

**hdfs dfs -put "$LOG\_FILE" "$HDFS\_LOG\_DIR/"**

**echo "Ingested logs from $LOG\_FILE into $HDFS\_LOG\_DIR"**

**else**

**echo "Warning: Log file $LOG\_FILE is empty or does not exist."**

**fi**

**# Ingest metadata file into HDFS (assuming one file for all dates)**

**META\_FILE="C:/Users/aizel/Downloads/LUMS OFFICIAL DATA/Data Eng/labs/data-ingestion-hive/project-root/content\_metadata.csv"**

**if [ -s "$META\_FILE" ]; then**

**hdfs dfs -put "$META\_FILE" "$HDFS\_META\_DIR/"**

**echo "Ingested metadata from $META\_FILE into $HDFS\_META\_DIR"**

**else**

**echo "Warning: Metadata file $META\_FILE is empty or does not exist."**

**fi**

**echo "Ingestion process complete."**

1. **Content\_metadata**

CREATE TABLE content\_metadata (

content\_id INT PRIMARY KEY,

title TEXT,

category TEXT,

length INT,

artist TEXT

);

**A screenshot of a computer

AI-generated content may be incorrect.**

**Query: 1. Get All User Activities with Content Details**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

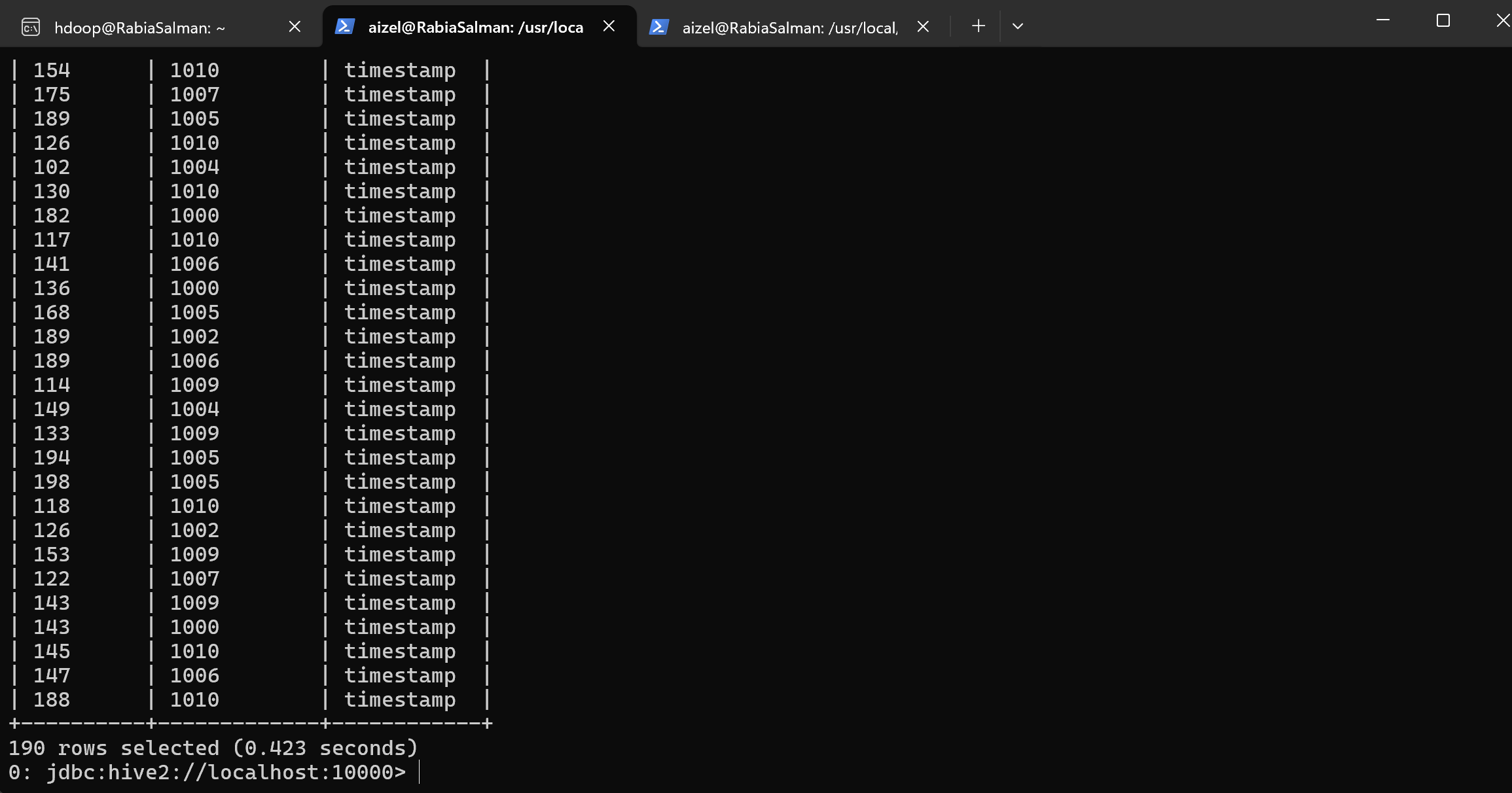
AI-generated content may be incorrect.**

**Query 2.**

Select user\_id, action, `timestamp` from user\_activity\_logs where year=2025 and month=2**A screenshot of a computer

AI-generated content may be incorrect.**

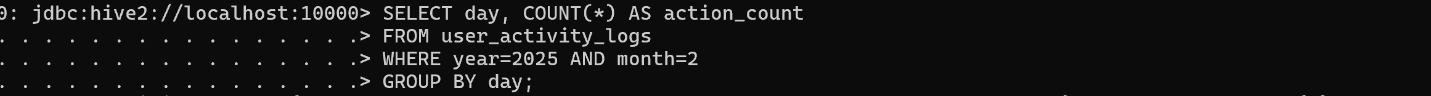
**A screenshot of a computer

AI-generated content may be incorrect.**

**Query 3**

**Count Actions per Month (here for month=2)**

Select day, count(\*) as action\_count from user\_activity\_logs where year=2025 and month=2 group by day;

****

**Query Results for data injested in Content\_metadata table**

****

**Query Results for Limit 5 data injested in User\_activity\_logs table**

**A screen shot of a computer

AI-generated content may be incorrect.**

**Query Results for active users in second month**

**A screen shot of a computer

AI-generated content may be incorrect.**

**Query Count of Unique Active Users per Day**

SELECT year, month, day, COUNT(DISTINCT user\_id) AS active\_users

FROM user\_activity\_logs

GROUP BY year, month, day

ORDER BY year DESC, month DESC, day DESC;****

**Query Results for Top Played Content**

SELECT content\_id, COUNT(\*) AS play\_count

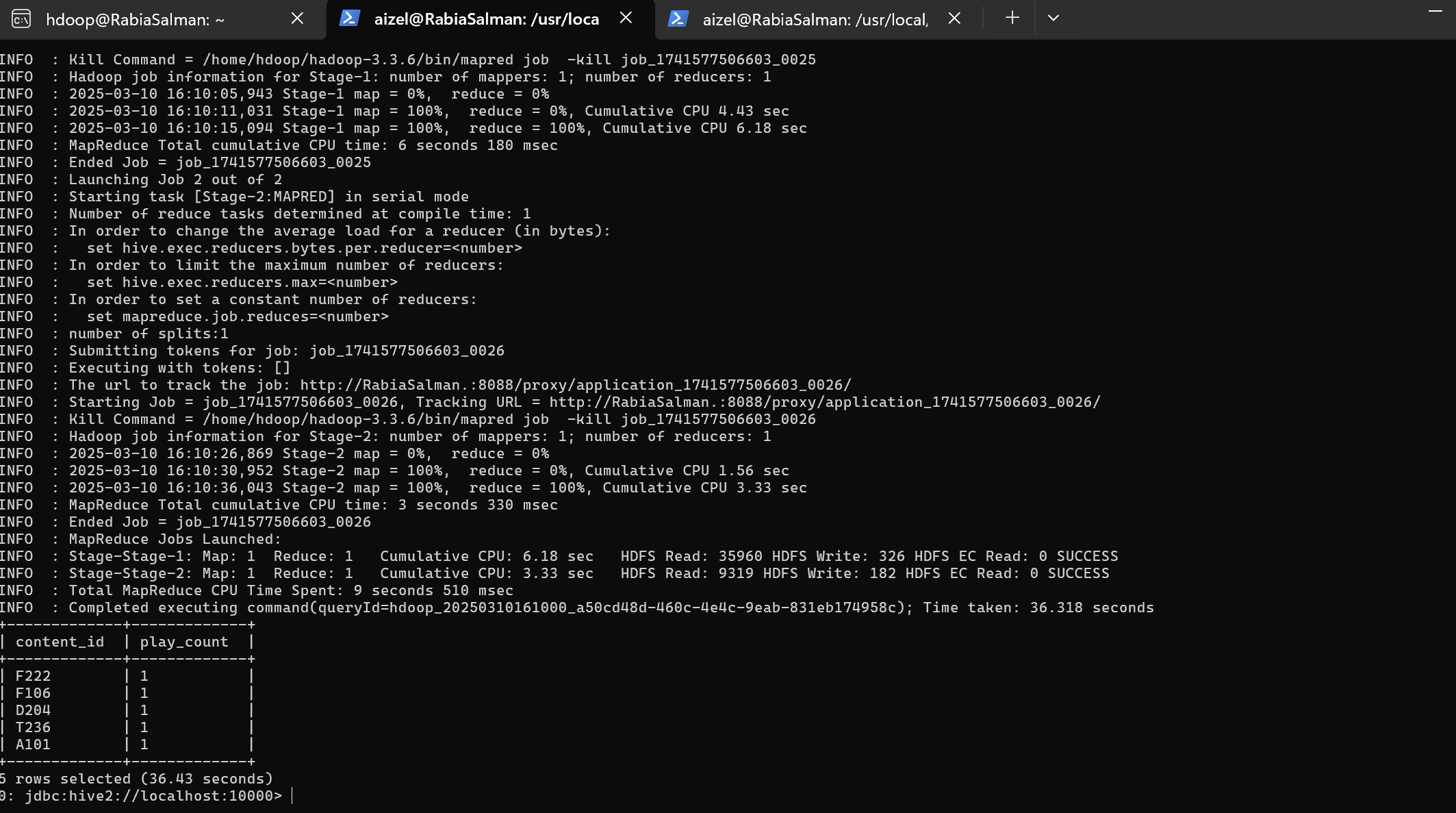
FROM user\_activity\_logs

WHERE action = 'play'

GROUP BY content\_id

ORDER BY play\_count DESC

LIMIT 5;

****

**Pause-to-Play Ratio**

**This query calculates how often users pause compared to play for a given date.**

SELECT year, month, day,

SUM(CASE WHEN action = 'pause' THEN 1 ELSE 0 END) AS pause\_count,

SUM(CASE WHEN action = 'play' THEN 1 ELSE 0 END) AS play\_count,

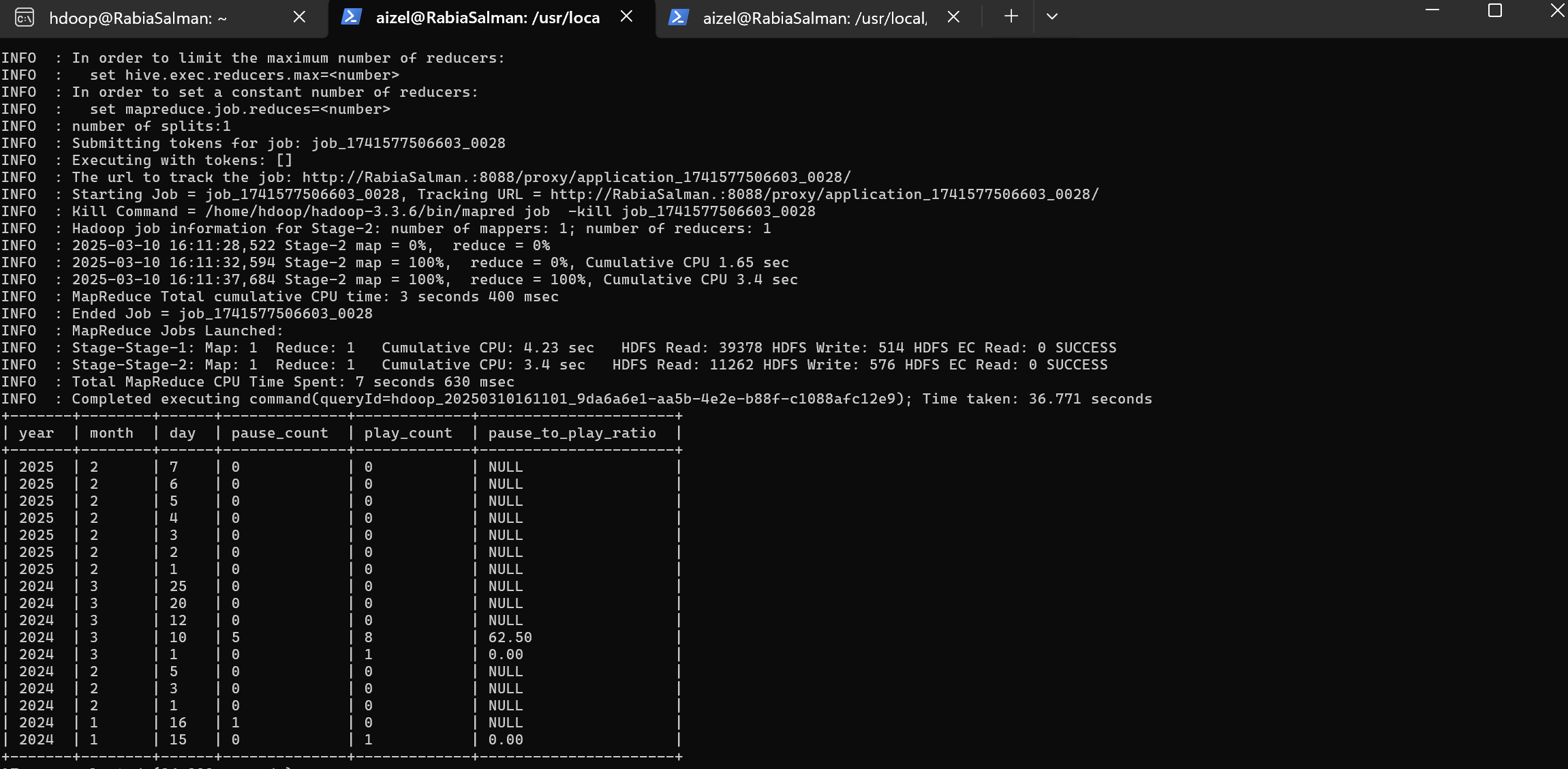
ROUND(SUM(CASE WHEN action = 'pause' THEN 1 ELSE 0 END) \* 100.0 /

NULLIF(SUM(CASE WHEN action = 'play' THEN 1 ELSE 0 END), 0), 2) AS pause\_to\_play\_ratio

FROM user\_activity\_logs

GROUP BY year, month, day

ORDER BY year DESC, month DESC, day DESC;

**A black and white screen

AI-generated content may be incorrect.**

**Create the Dimension Table (dim\_content)**

**This stores content metadata, which will be joined with the fact table for analysis.**

CREATE TABLE dim\_content (

content\_id STRING,

title STRING,

category STRING,

length INT,

artist STRING

)

STORED AS PARQUET;

**A black and white screen

AI-generated content may be incorrect.**

**Create the Fact Table (fact\_user\_actions)**

**This stores user activity logs and is partitioned by year, month, and day.**

CREATE TABLE fact\_user\_actions (

user\_id STRING,

action STRING,

timestamp STRING,

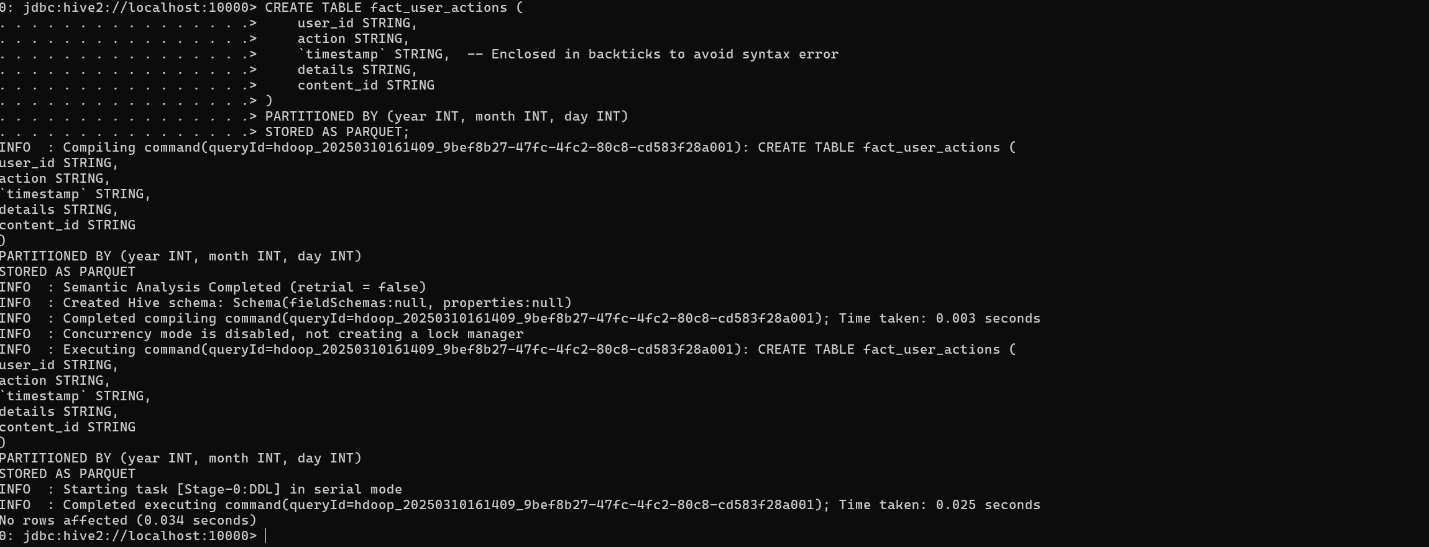
details STRING,

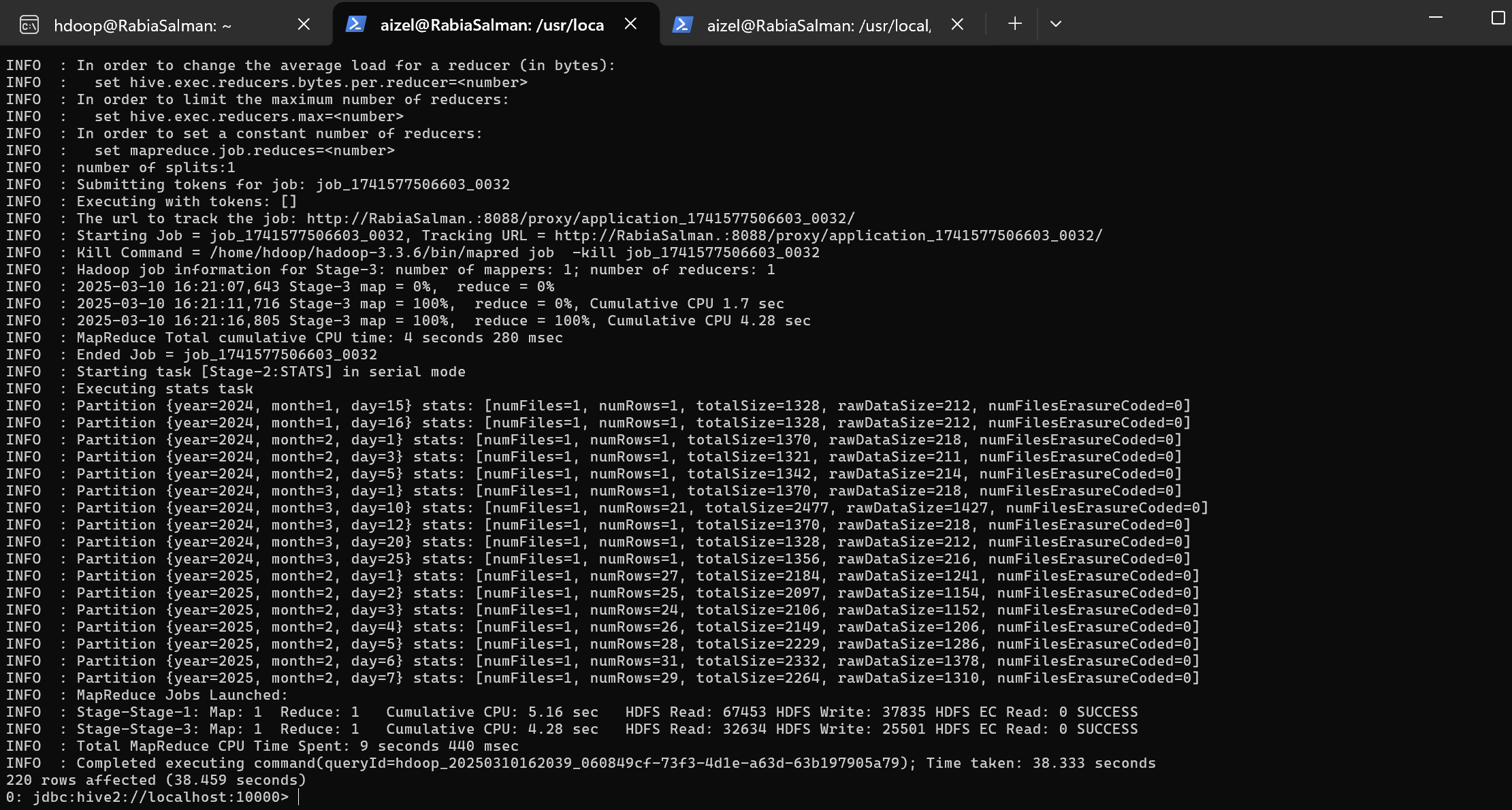
content\_id STRING

)

PARTITIONED BY (year INT, month INT, day INT)

STORED AS PARQUET;

****

****

**Query: Showing data from Star Schema (Fact\_user\_actions)**

**A black screen with white text

AI-generated content may be incorrect.**