Movie Ratings Data Analysis Using Apache Sqoop and Apache Hive

Project submitted to the
SRM UNIVERSITY-AP, Andhra Pradesh
for partial fulfilment of the requirements to award the degree
of
Master of Technology
in
Data Science

Submitted by
BANDLA RAVI - AP24122060008
THOTA TARUN KUMAR - AP24122060021
POLAROWTHU JASWANTH -AP24122060023

SRM University–AP
Neerukonda, Mangalagiri, Guntur
Andhra Pradesh – 522 240
[DECEMBER, 2024]

Certificate

Date: 28-Nov-24

This is to certify that the work presented in this Project entitled "Movie Ratings Data Analysis Using Apache Scoop and Apache Hive" has been carried out by Ravi, TarunKumar, Jaswanth under my supervision. The work is genuine, original, and suitable for submission to SRM University for the award of Master of Technology.

Supervisor

DR. Rajiv Senapati

Assistant Professor

Department of Computer Science and Engineering

❖ Acknowledgments

This project was brought to fruition with invaluable support and resources from several key platforms and communities. The Hortonworks Sandbox served as an exceptional environment for big data processing, offering a robust platform for experimenting with tools like HDFS, HIVE and Sqoop. GitHub played a pivotal role by providing a reliable and accessible repository for managing and sharing project data. The open-source community deserves heartfelt gratitude for its unwavering dedication to creating, maintaining, and documenting tools and frameworks, which formed the backbone of this project. Their collective efforts continue to inspire innovation and make advanced technologies accessible to all.

Table of Contents:

❖ Abstract

The project showcases a comprehensive end-to-end workflow for big data processing, demonstrating the seamless integration of various tools and platforms to manage and analyze data effectively. The process begins with transferring a dataset from GitHub, a widely used version control and collaboration platform, to the Hadoop Distributed File System (HDFS). This step ensures efficient data storage and scalability, leveraging the distributed architecture of HDFS.

Once stored, the dataset is ingested into MySQL, a robust relational database management system, to facilitate structured storage and relational querying. Apache Sqoop is then utilized as a bridge to move the data seamlessly from MySQL into Apache Hive. Hive, a powerful data warehousing tool, allows the execution of SQL-like queries on the dataset, providing an intuitive interface for big data analytics.

The project's final stage focuses on performing insightful queries in Hive, extracting meaningful patterns and trends from the data. This workflow exemplifies how big data tools collaborate to simplify complex data operations, streamline workflows, and enhance organizational capabilities in deriving actionable insights. It underscores the importance of integrated ecosystems in big data analytics, empowering users to harness the full potential of their data efficiently and effectively for decision-making and innovation.

❖ Introduction

Data-driven decision-making has become a cornerstone of modern organizations, enabling them to derive actionable insights from vast amounts of data. This project focuses on integrating components of the Hadoop ecosystem to demonstrate how big data tools can simplify and enhance the management and analysis of user movie ratings. These components—HDFS, MySQL, Sqoop, and Hive—work together to create an end-to-end data processing pipeline that handles tasks ranging from data storage and transfer to querying and insights generation.

Efficient Data Transfer Across Platforms:

Efficiently transferring data between platforms is a critical aspect of big data workflows. In this project, the dataset is initially sourced from GitHub, a version control platform that allows collaboration and secure storage of data.

1. **Transfer to HDFS**
   The dataset is first moved to the Hadoop Distributed File System (HDFS), which ensures scalability and fault tolerance. HDFS divides the data into blocks and replicates them across different nodes in a cluster, guaranteeing availability even if some nodes fail. This feature is particularly crucial for managing large datasets that traditional file systems might struggle to handle.
2. **Ingestion into MySQL**
   The dataset is then ingested into MySQL, a relational database management system (RDBMS). MySQL is used to store data in structured tables with predefined schemas. This stage facilitates a systematic organization of data, allowing relational operations such as joins, filtering, and aggregation. Relational storage is advantageous for datasets requiring strict schema enforcement and transaction support.
3. **Data Movement Using Apache Sqoop**
   Apache Sqoop bridges the gap between relational databases like MySQL and big data tools like Hive. By transferring structured data from MySQL to Hive, Sqoop allows the dataset to transition seamlessly into the Hadoop ecosystem. This transfer is automated, ensuring minimal manual intervention while maintaining data integrity. Sqoop can handle incremental imports, making it efficient for keeping datasets synchronized between systems.

Integration of Hadoop Ecosystem Components:

Each component of the Hadoop ecosystem plays a specific role in the workflow, and their integration showcases the ecosystem's power:

HDFS ➔ensures reliable storage of large datasets.
MySQL➔provides structured relational storage for initial data organization.
Sqoop ➔automates data transfer, bridging traditional databases with big data tools.
Hive➔ facilitates querying and insights generation, enabling non-programmers to interact with big data effectively.

By integrating HDFS, MySQL, Sqoop, and Hive, this project exemplifies the power of the Hadoop ecosystem in managing and analyzing large datasets. It provides a roadmap for leveraging these tools to gain actionable insights, underscoring their importance in the era of data-driven decision-making. This workflow not only highlights the strengths of each component but also demonstrates their synergy in creating robust data pipelines.

❖ Dataset Description

The ratings.data dataset is a rich source of information that captures user behavior and movie preferences. With over 100,000 rows of user ratings, it provides a comprehensive view of interactions between users and movies. The dataset is tab-separated (TSV) and structured as follows:

Dataset Structure

1. user_id
    - A unique identifier for each user in the system.
    - Represents the individual providing the movie rating.
    - Use cases:
        - Analyzing user behavior and preferences.
        - Identifying active and passive users based on rating frequency.
2. movie_id
    - A unique identifier for each movie rated by users.
    - Represents the entity receiving the rating.
    - Use cases:
        - Measuring the popularity of movies.
        - Grouping ratings by genre or release year (if additional metadata is available).
3. rating
    - A numerical score (1–5) indicating the user's opinion of the movie.
    - Represents the quality or enjoyment level perceived by the user.
    - Use cases:
        - Sentiment analysis of user feedback.
        - Identifying top-rated movies or poorly rated ones.
4. timestamp
    - The Unix timestamp of when the rating was submitted.
    - Represents the temporal aspect of the data.
    - Use cases:
        - Trend analysis over time.
        - Evaluating the impact of events (e.g., promotions or new releases) on user activity.

❖ Implementation

1. Data Transfer to HDFS:

Objective:

To move the dataset from GitHub to the Hadoop Distributed File System (HDFS) for scalable and fault-tolerant storage.
Steps:

1. Download the Dataset from GitHub
    - Use the wget command to fetch the dataset directly from the specified GitHub repository.
    - The wget command retrieves files from the web using HTTP, HTTPS, or FTP protocols.

Command Used:
wget https://raw.githubusercontent.com/PJ07-09-2003/Big-Data/refs/heads/main/ratings.data

After executing the above commands, ensure the dataset (ratings.data) is present in the local directory by listing files with the ls command.

Command Used:
ls

2. Create a Directory in HDFS

- Create a dedicated directory in HDFS to store the dataset. The mkdir command initializes the directory structure.

Command Used:
hdfs dfs -mkdir /user/maria_dev/ratings_data

The path /user/maria_dev/ratings_data ensures that the dataset is stored under the user-specific directory in HDFS.

3. Transfer the Dataset to HDFS

- Use the put command to move the ratings.data file from the local system to the newly created HDFS directory.

Command Used:
hdfs dfs -put ratings.data /user/maria_dev/ratings_data

This command ensures that the dataset is now accessible in the distributed storage environment.

4. Verify the File in HDFS

- List the contents of the HDFS directory to confirm the successful transfer of the dataset.

Command Used:

hdfs dfs -ls /user/maria_dev/ratings_data

2. **MySQL Integration**

Objective:

To temporarily move the dataset to a local directory and ingest it into a MySQL database for relational storage and validation.

Steps:

1. **Retrieve Dataset from HDFS to Local System**
    - Use the get command to move the dataset back to the local directory for MySQL ingestion.

Command Used:
hdfs dfs -get /user/maria_dev/ratings_data/ratings.data /tmp/ratings.data

The /tmp/ratings.data path ensures the file is stored in a temporary local directory.

2. **Open MySQL Shell**

To open the MySQL shell in the Hadoop sandbox, you can use the following command:

Command Used:

mysql -u root -p

Once executed, you will be prompted to enter the MySQL root user's password. In the Hadoop sandbox, the default password is usually:

Password is: Hadoop

After entering the password, you'll enter the MySQL shell where you can execute SQL commands.

3. **Create and Use a Database in MySQL**

- Launch MySQL and create a database named bigdata_db to store the ratings data.

Command Used:

CREATE DATABASE bigdata_db;
USE bigdata_db;

4. **Create a Table and Load Data**

- Create a table named ratings to store the data. The table schema matches the dataset structure:
    - user_id (INT)
    - movie_id (INT)
    - rating (FLOAT)
    - timestamp (BIGINT)

- Load the data into the table using the LOAD DATA LOCAL INFILE command.

Command Used:

```
LOAD DATA LOCAL INFILE '/tmp/ratings.data'
INTO TABLE ratings
FIELDS TERMINATED BY '\t'
LINES TERMINATED BY '\n'
(user_id, movie_id, rating, timestamp);
```

5. **Validate Data Ingestion**

- Check the total number of records ingested into the table using the COUNT(*) query.

Command Used:

```
SELECT COUNT(*) FROM ratings;
```

This ensures that the dataset has been successfully loaded into MySQL.

3. **Data Import with Sqoop**

Objective:

To move data from MySQL to Hive for distributed storage and querying using Apache Sqoop.

Steps:

1. **Set Up Sqoop Import Command**
    - Use the sqoop import command to transfer data from the MySQL database to Hive.

Command Used:
sqoop import \
--connect "jdbc:mysql://sandbox.hortonworks.com:3306/bigdata_db" \
--username root --password hadoop \
--table ratings --hive-import \
--hive-table ratings --delete-target-dir --m 1

Explanation:

- --connect: Specifies the JDBC URL to connect to the MySQL database.
- --username and --password: Provide authentication credentials for MySQL.
- --table: Specifies the table to be imported (ratings).
- --hive-import: Indicates that data should be imported into Hive.
- --hive-table: Names the Hive table where data will be stored.
- --delete-target-dir: Ensures the target directory is cleared before importing.
- --m 1: Specifies a single mapper job for simplicity.

2. **Verify Data in Hive Using Ambari Dashboard**

1. **Log into Ambari Dashboard:** Open your web browser and go to the Ambari dashboard URL.
2. **Access Hive View:** In the Ambari dashboard, find the **Hive** service in the left-hand navigation panel. Click on it to open the Hive service view.
3. **Go to the Files View:** In the Hive view, navigate to the **Files** tab or **Files View** section. This view will give you a file browser interface where you can see all the files and directories in HDFS.

1.1 Listing Databases in MySQL Using Sqoop

To identify the available databases in a MySQL server, the sqoop list-databases command is used. This helps users understand the data structure and available tables before performing operations.

Command Used:
sqoop list-databases --connect jdbc:mysql://localhost/bigdata_db --username root -P

Purpose: Lists all databases in the MySQL server.

1.2 Executing Queries Using Sqoop Eval
The sqoop eval command allows you to execute SQL queries on a database directly from Sqoop. This is useful for validating data before importing or exporting.

Command Used:
sqoop eval --connect jdbc:mysql://localhost/bigdata_db --username root -P --query "select * from ratings;"

Purpose: Executes the query SELECT * FROM ratings; to display all records from the ratings table in MySQL.

1.3 Importing Full Data from MySQL to HDFS
The sqoop import command transfers an entire table from MySQL to HDFS.

Command Used:
sqoop import --connect jdbc:mysql://localhost/bigdata_db --username root --password hadoop --table ratings --warehouse-dir /user --driver com.mysql.jdbc.Driver --m 1

Key Options:

- --connect: Specifies the JDBC URL to connect to the database.
- --username and --password: Authentication details for MySQL.
- --table: Specifies the table to import (ratings).
- --warehouse-dir: Defines the HDFS directory to store the data.
- --m 1: Uses a single mapper for the task.

Significance:

- Ensures the ratings table is imported into HDFS for processing.
- Creates an HDFS directory (/user/ratings) containing the imported data.

1.3.1 Validating Data in HDFS

a) Listing the Imported Data
To check the files created during import:

Command Used:
hdfs dfs -ls /user/ratings

b) Viewing Sample Data
To inspect the first few records in the file:

Command Used:
hdfs dfs -cat /user/ratings/part-m-00000 | head

c) **Checking HDFS File Count**
To validate the file count and sizes:

Command Used:
hdfs dfs -count /user/ratings

1.4 Importing Incremental Data (Append Mode)

Incremental imports allow adding only new or updated records to HDFS. This is critical for regularly updated datasets.

Command Used:
sqoop import --connect jdbc:mysql://localhost/bigdata_db --username root --password hadoop --table ratings --warehouse-dir /user --driver com.mysql.jdbc.Driver --m 1 --incremental append --check-column rating --last-value 5

- **Key Options**:
  - --incremental append: Adds new records based on the check-column.
  - --check-column rating: Uses the rating column to identify new records.
  - --last-value 5: Specifies the last imported value for rating.

Significance: Incremental imports are efficient for synchronizing only new data into HDFS without re-importing the entire table.

1.5 Exporting Data from HDFS to MySQL

To export processed or transformed data from HDFS back to MySQL, the sqoop export command is used.

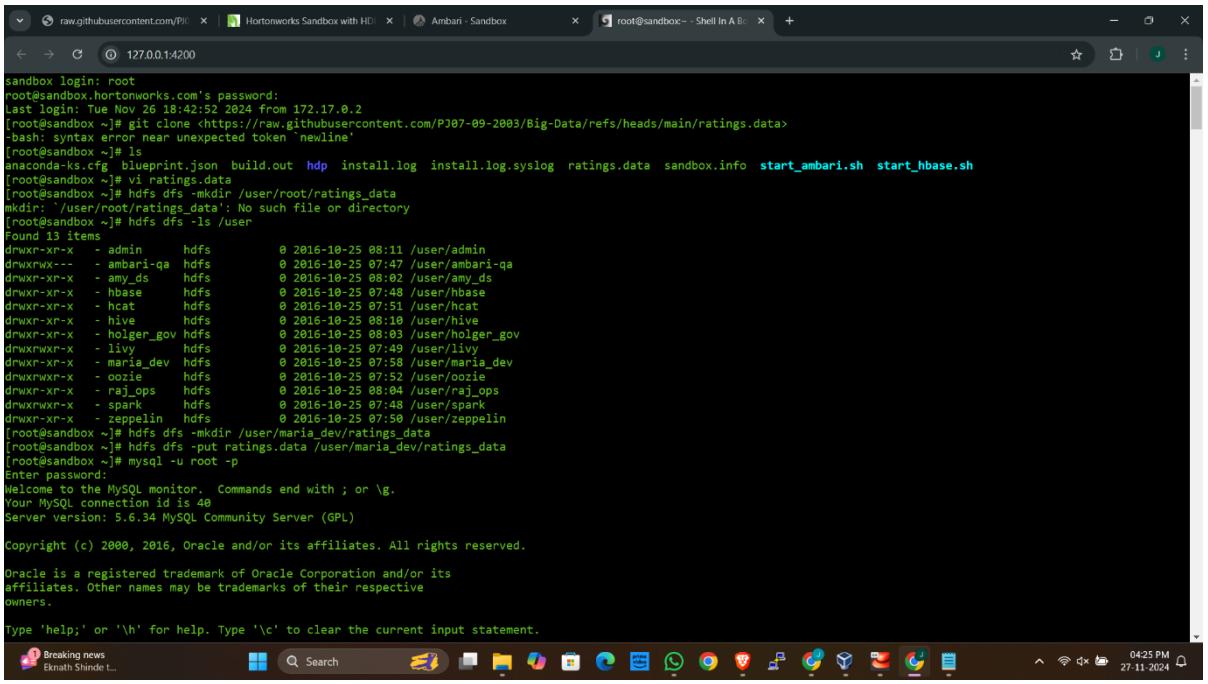
Command Used:
sqoop export --connect jdbc:mysql://localhost/bigdata_db --username root --password hadoop --table ratings --export-dir /user/ratings/part-m-00000 --driver com.mysql.jdbc.Driver --m 1

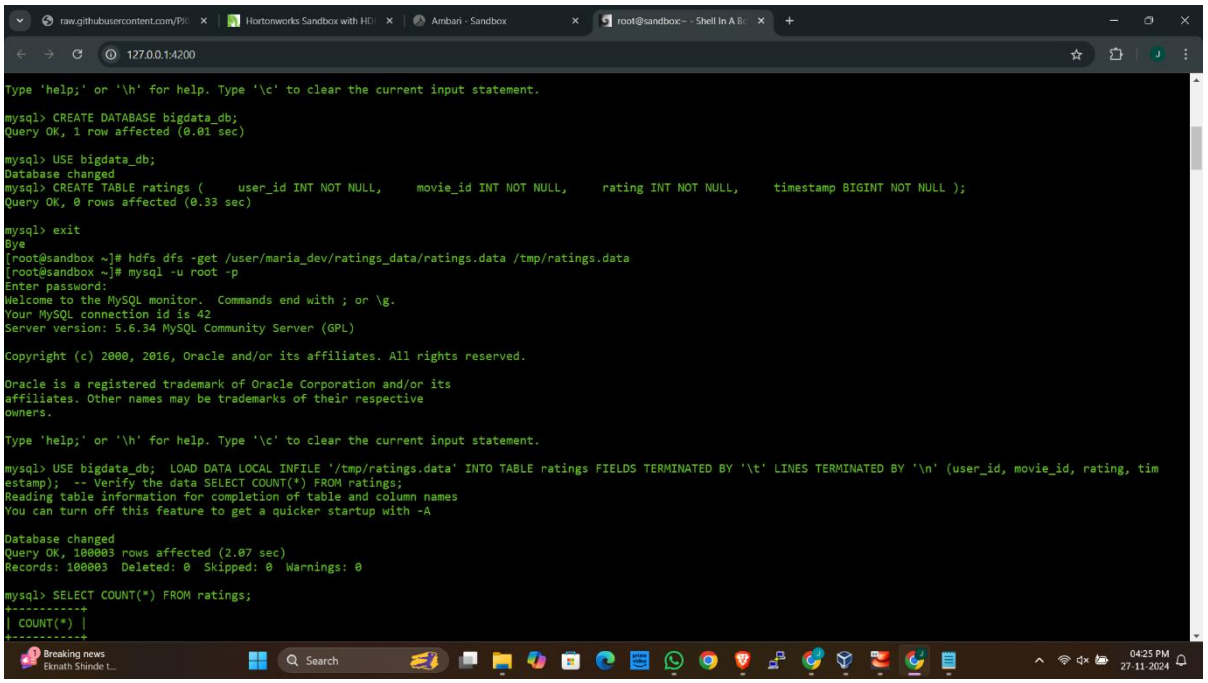
- **Key Options**:
  - --export-dir: Specifies the HDFS directory containing the data to export.
  - --table: The destination table in MySQL.

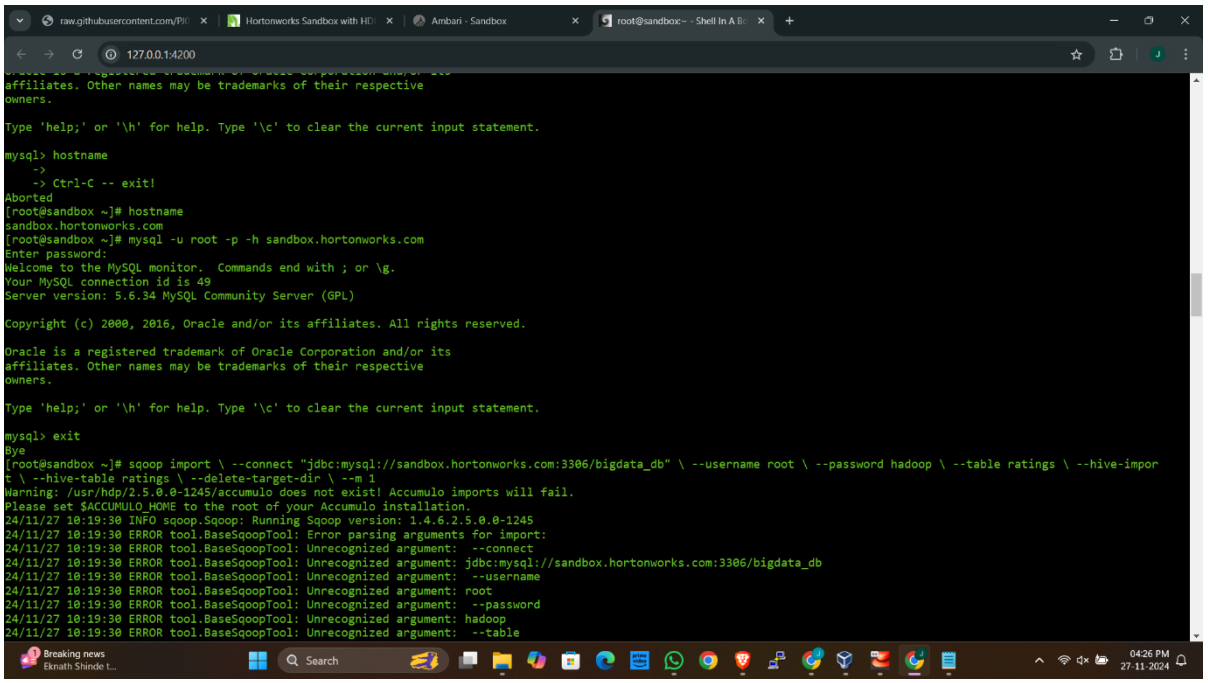
Significance:

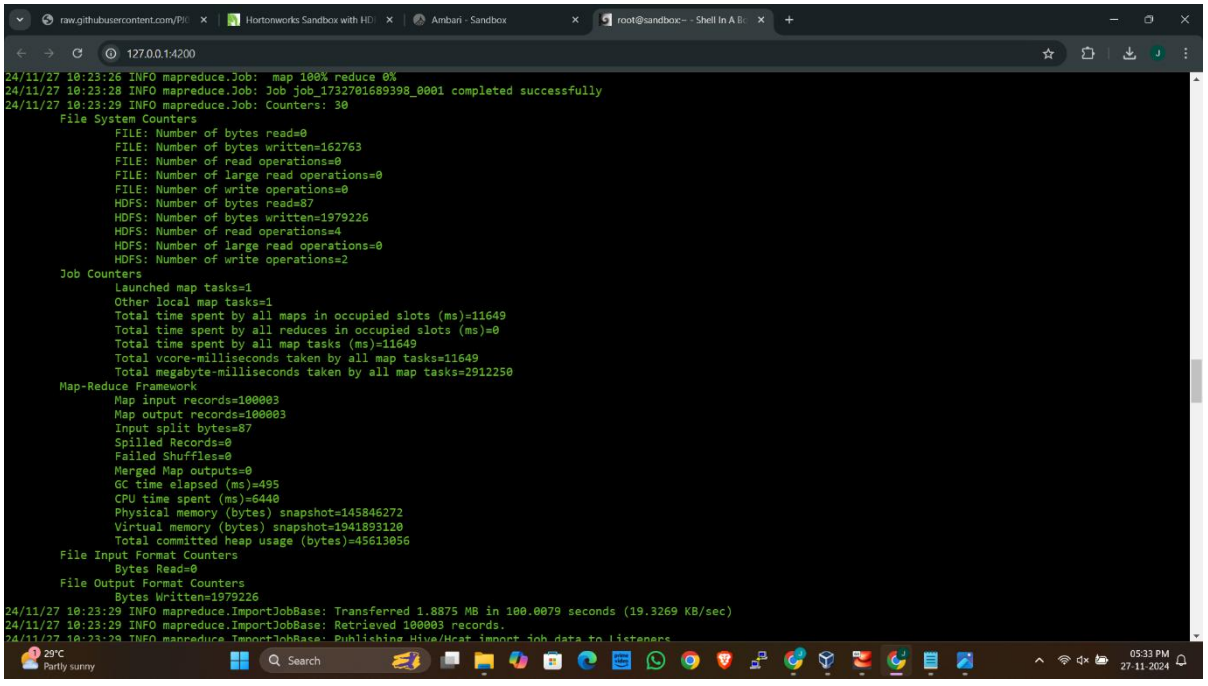
- Enables sharing processed data with downstream systems or applications.
- Demonstrates a complete data pipeline from MySQL to HDFS and back.

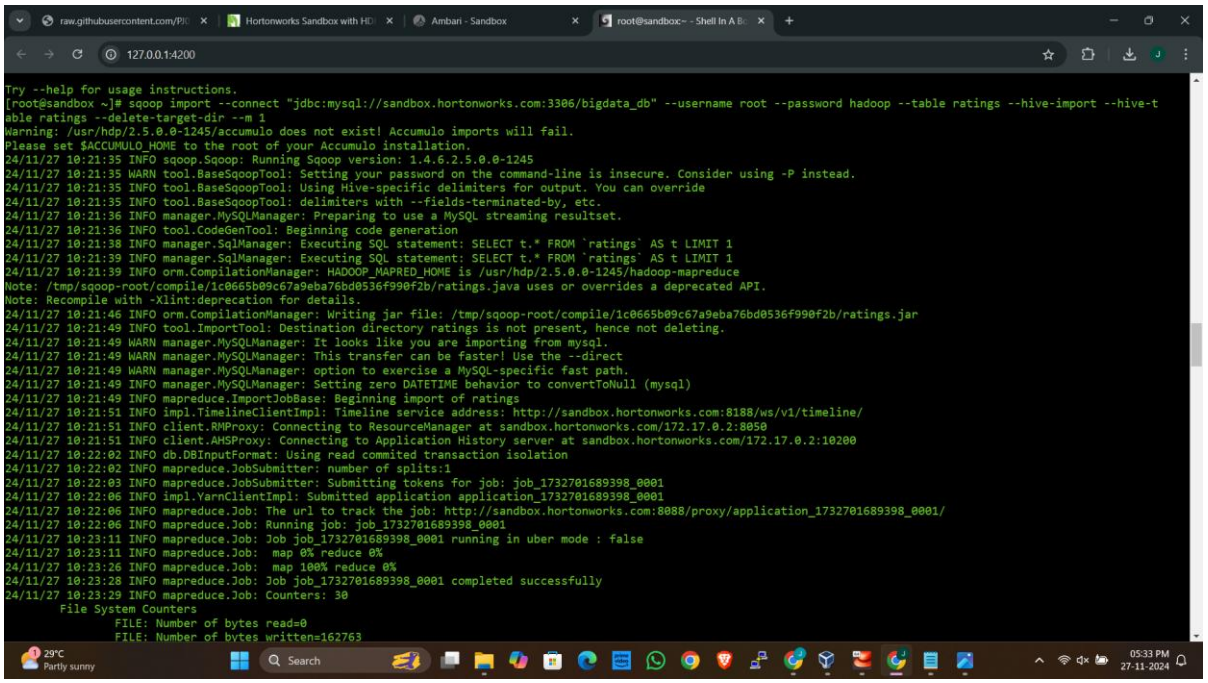
Overall Outputs of Execution:

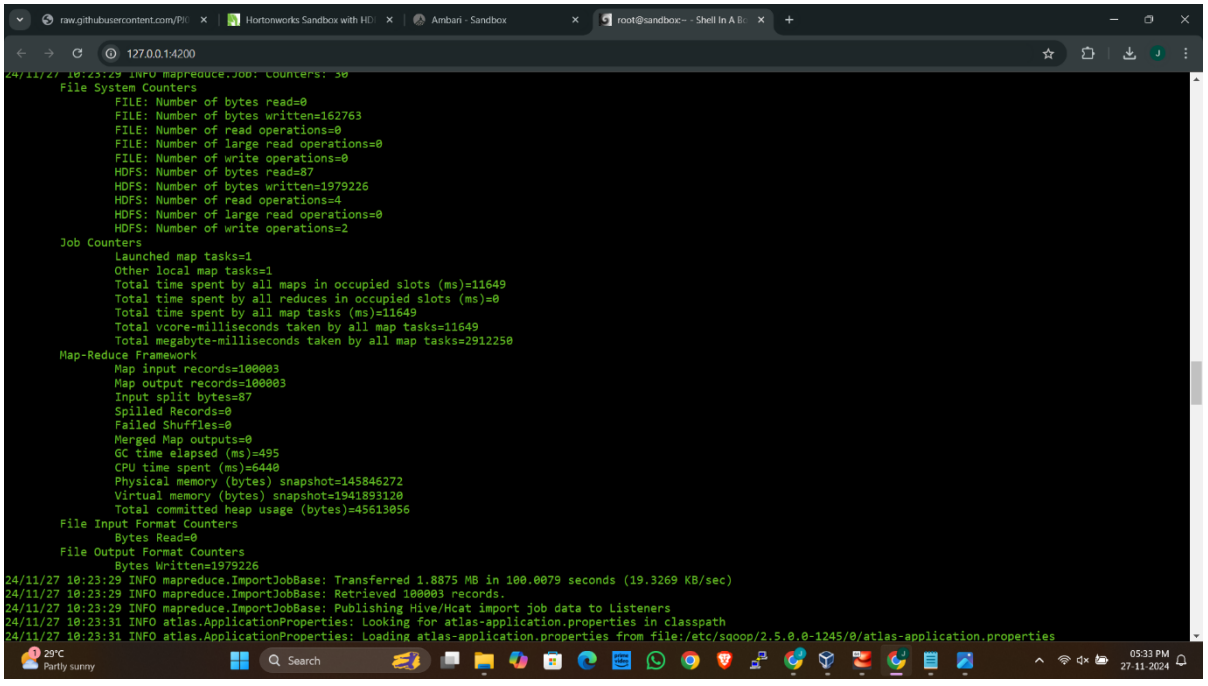


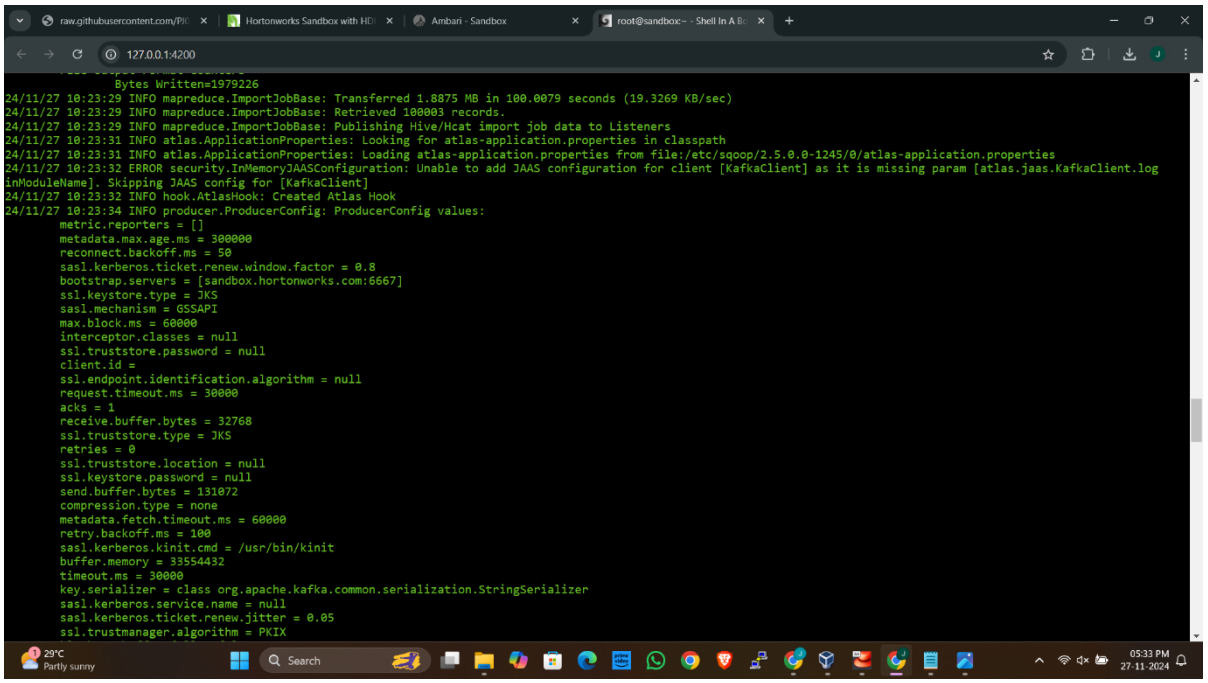


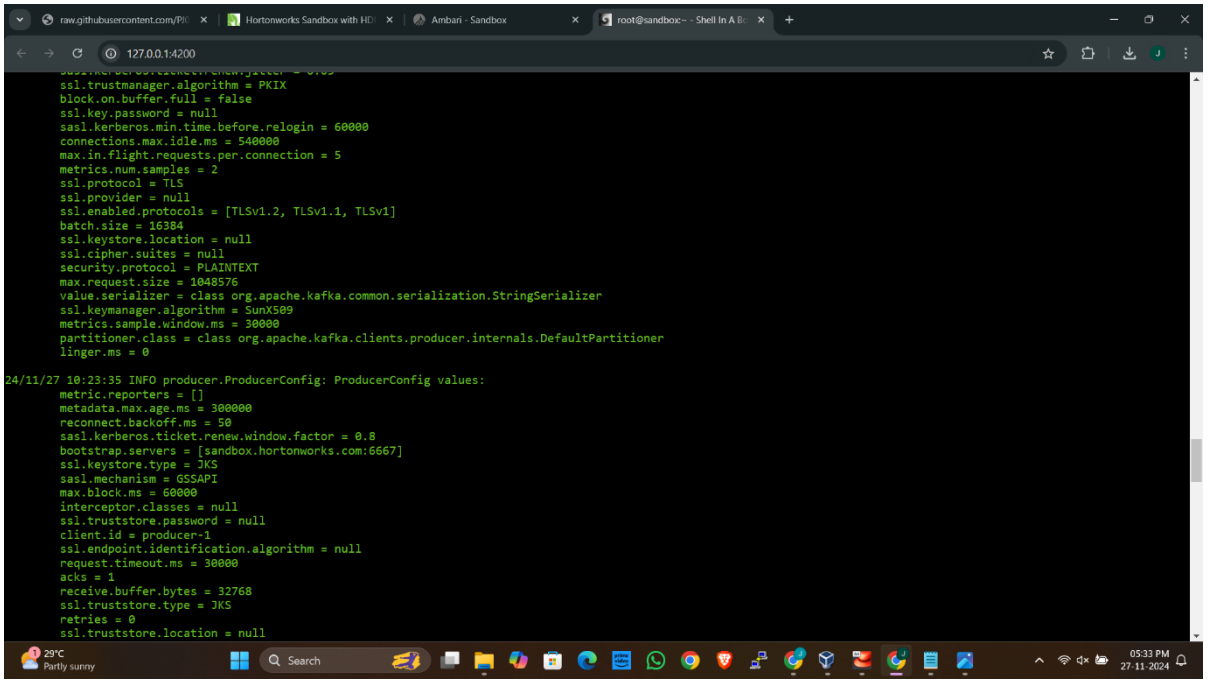

raw.githubusercontent.com/Pi
Hortonworks Sandbox with HD
Ambari - Sandbox
root@sandbox:~ - Shell In A B
127.0.0.1:4200
affiliates. Other names may be trademarks of their respective
owners.
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
mysql> hostname
-> Ctrl-C -- exit!
Aborted
[root@sandbox ~]# hostname
sandbox.hortonworks.com
[root@sandbox ~]# mysql -u root -p -h sandbox.hortonworks.com
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 49
Server version: 5.6.34 MySQL Community Server (GPL)
Copyright (c) 2000, 2016, Oracle and/or its affiliates. All rights reserved.
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
mysql> exit
Bye
[root@sandbox ~]# sqoop import \ --connect "jdbc:mysql://sandbox.hortonworks.com:3306/bigdata_db" \ --username root \ --password hadoop \ --table ratings \ --hive-impor
t \ --hive-table ratings \ --delete-target-dir \ --m 1
Warning: /usr/hdp/2.5.0.0-1245/accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
24/11/27 10:19:30 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6.2.5.0.0-1245
24/11/27 10:19:30 ERROR tool.BaseSqoopTool: Error parsing arguments for import:
24/11/27 10:19:30 ERROR tool.BaseSqoopTool: Unrecognized argument: --connect
24/11/27 10:19:30 ERROR tool.BaseSqoopTool: Unrecognized argument: jdbc:mysql://sandbox.hortonworks.com:3306/bigdata_db
24/11/27 10:19:30 ERROR tool.BaseSqoopTool: Unrecognized argument: --username
24/11/27 10:19:30 ERROR tool.BaseSqoopTool: Unrecognized argument: root
24/11/27 10:19:30 ERROR tool.BaseSqoopTool: Unrecognized argument: --password
24/11/27 10:19:30 ERROR tool.BaseSqoopTool: Unrecognized argument: hadoop
24/11/27 10:19:30 ERROR tool.BaseSqoopTool: Unrecognized argument: --table
Breaking news
Eknath Shinde t...
Search
04:26 PM
27-11-2024

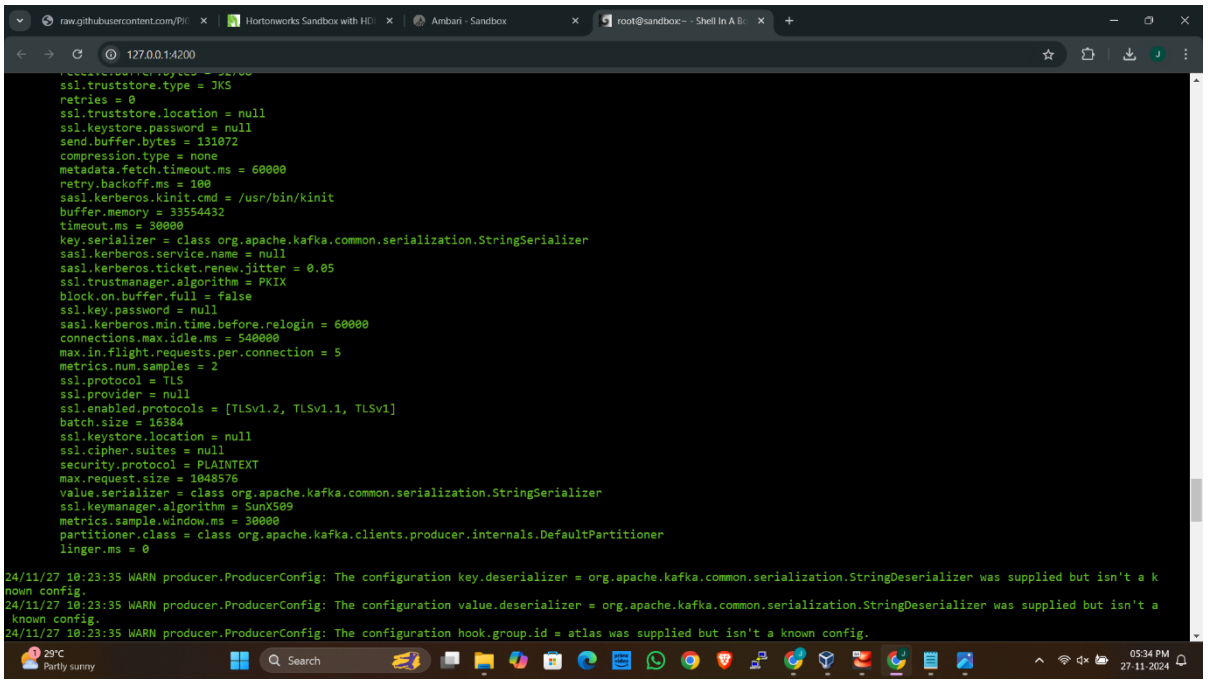

raw.githubusercontent.com/Pi
Hortonworks Sandbox with HD
Ambari - Sandbox
root@sandbox:~ - Shell In A B
127.0.0.1:4200
24/11/27 10:23:26 INFO mapreduce.Job: map 100% reduce 0%
24/11/27 10:23:28 INFO mapreduce.Job: Job job_1732701689398_0001 completed successfully
24/11/27 10:23:29 INFO mapreduce.Job: Counters: 30
File System Counters
FILE: Number of bytes read=0
FILE: Number of bytes written=162763
FILE: Number of read operations=0
FILE: Number of large read operations=0
FILE: Number of write operations=0
HDFS: Number of bytes read=87
HDFS: Number of bytes written=1979226
HDFS: Number of read operations=4
HDFS: Number of large read operations=0
HDFS: Number of write operations=2
Job Counters
Launched map tasks=1
Other local map tasks=1
Total time spent by all maps in occupied slots (ms)=11649
Total time spent by all reduces in occupied slots (ms)=0
Total time spent by all map tasks (ms)=11649
Total vcore-milliseconds taken by all map tasks=11649
Total megabyte-milliseconds taken by all map tasks=2912250
Map-Reduce Framework
Map input records=100003
Map output records=100003
Input split bytes=87
Spilled Records=0
Failed Shuffles=0
Merged Map outputs=0
GC time elapsed (ms)=495
CPU time spent (ms)=6440
Physical memory (bytes) snapshot=145846272
Virtual memory (bytes) snapshot=1941893120
Total committed heap usage (bytes)=45613056
File Input Format Counters
Bytes Read=0
File Output Format Counters
Bytes Written=1979226
24/11/27 10:23:29 INFO mapreduce.ImportJobBase: Transferred 1.8875 MB in 100.0079 seconds (19.3269 KB/sec)
24/11/27 10:23:29 INFO mapreduce.ImportJobBase: Retrieved 100003 records.
29°C
Partly sunny
Search
05:33 PM
27-11-2024

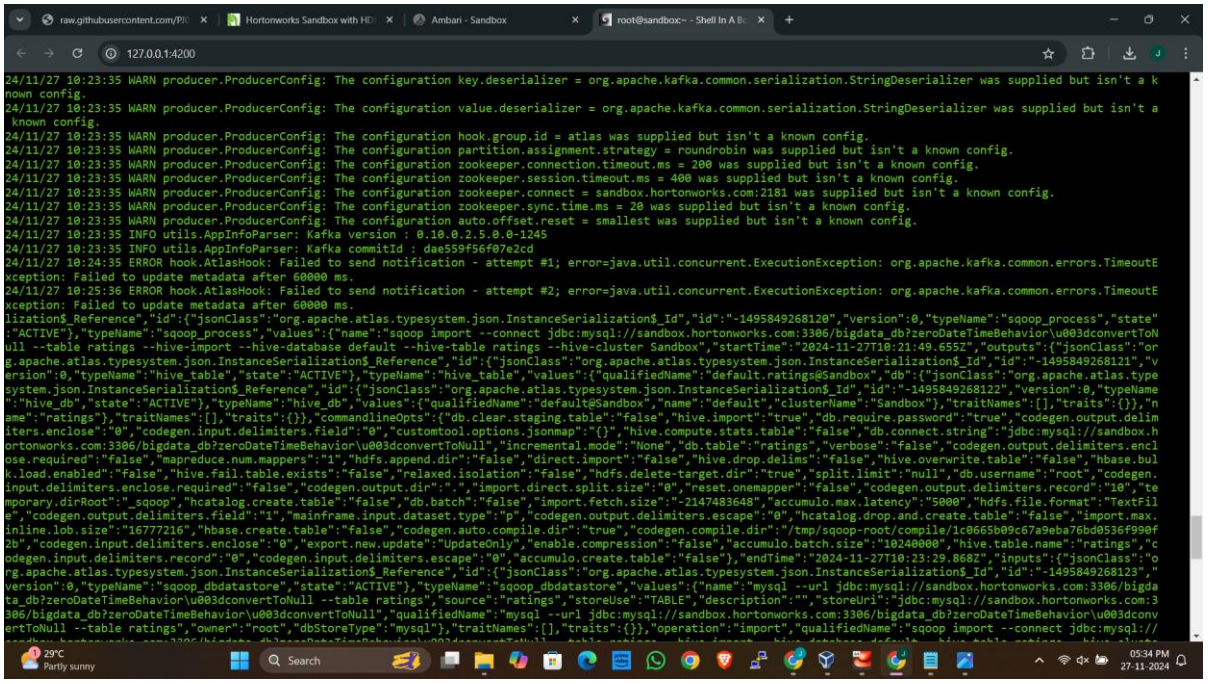

raw.githubusercontent.com/Pl
Hortonworks Sandbox with HD
Ambari - Sandbox
root@sandbox:~ - Shell In A B
127.0.0.1:4200
Try --help for usage instructions.
[root@sandbox ~]# sqoop import --connect "jdbc:mysql://sandbox.hortonworks.com:3306/bigdata_db" --username root --password hadoop --table ratings --hive-import --hive-t
able ratings --delete-target-dir --m 1
Warning: /usr/hdp/2.5.0.0-1245/accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
24/11/27 10:21:35 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6.2.5.0.0-1245
24/11/27 10:21:35 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.
24/11/27 10:21:35 INFO tool.BaseSqoopTool: Using Hive-specific delimiters for output. You can override
24/11/27 10:21:35 INFO tool.BaseSqoopTool: delimiters with --fields-terminated-by, etc.
24/11/27 10:21:36 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
24/11/27 10:21:36 INFO tool.CodeGenTool: Beginning code generation
24/11/27 10:21:38 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `ratings` AS t LIMIT 1
24/11/27 10:21:39 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `ratings` AS t LIMIT 1
24/11/27 10:21:39 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/hdp/2.5.0.0-1245/hadoop-mapreduce
Note: /tmp/sqoop-root/compile/1c0665b09c67a9eba76bd0536f990f2b/ratings.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
24/11/27 10:21:46 INFO orm.CompilationManager: Writing jar file: /tmp/sqoop-root/compile/1c0665b09c67a9eba76bd0536f990f2b/ratings.jar
24/11/27 10:21:49 INFO tool.ImportTool: Destination directory ratings is not present, hence not deleting.
24/11/27 10:21:49 WARN manager.MySQLManager: It looks like you are importing from mysql.
24/11/27 10:21:49 WARN manager.MySQLManager: This transfer can be faster! Use the --direct
24/11/27 10:21:49 WARN manager.MySQLManager: option to exercise a MySQL-specific fast path.
24/11/27 10:21:49 INFO manager.MySQLManager: Setting zero DATETIME behavior to convertToNull (mysql)
24/11/27 10:21:49 INFO mapreduce.ImportJobBase: Beginning import of ratings
24/11/27 10:21:51 INFO impl.TimelineClientImpl: Timeline service address: http://sandbox.hortonworks.com:8188/ws/v1/timeline/
24/11/27 10:21:51 INFO client.RMProxy: Connecting to ResourceManager at sandbox.hortonworks.com/172.17.0.2:8050
24/11/27 10:21:51 INFO client.AHSProxy: Connecting to Application History server at sandbox.hortonworks.com/172.17.0.2:10200
24/11/27 10:22:02 INFO db.DBInputFormat: Using read commited transaction isolation
24/11/27 10:22:02 INFO mapreduce.JobSubmitter: number of splits:1
24/11/27 10:22:03 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1732701689398_0001
24/11/27 10:22:06 INFO impl.YarnClientImpl: Submitted application application_1732701689398_0001
24/11/27 10:22:06 INFO mapreduce.Job: The url to track the job: http://sandbox.hortonworks.com:8088/proxy/application_1732701689398_0001/
24/11/27 10:22:06 INFO mapreduce.Job: Running job: job_1732701689398_0001
24/11/27 10:23:11 INFO mapreduce.Job: Job job_1732701689398_0001 running in uber mode : false
24/11/27 10:23:11 INFO mapreduce.Job: map 0% reduce 0%
24/11/27 10:23:26 INFO mapreduce.Job: map 100% reduce 0%
24/11/27 10:23:28 INFO mapreduce.Job: Job job_1732701689398_0001 completed successfully
24/11/27 10:23:29 INFO mapreduce.Job: Counters: 30
File System Counters
FILE: Number of bytes read=0
FILE: Number of bytes written=162763
29°C
Partly sunny
Search
05:33 PM
27-11-2024

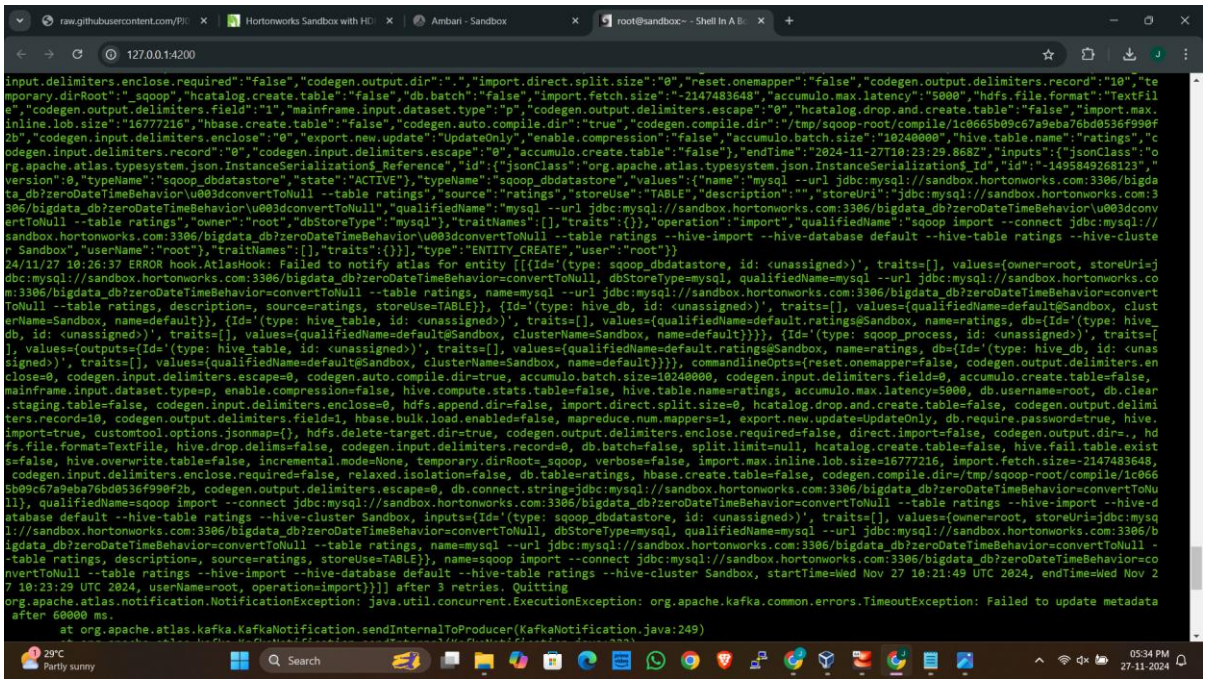

raw.githubusercontent.com/Pl
Hortonworks Sandbox with HD
Ambari - Sandbox
root@sandbox:~ - Shell In A B
127.0.0.1:4200
24/11/27 10:23:29 INFO mapreduce.Job: Counters: 30
File System Counters
FILE: Number of bytes read=0
FILE: Number of bytes written=162763
FILE: Number of read operations=0
FILE: Number of large read operations=0
FILE: Number of write operations=0
HDFS: Number of bytes read=87
HDFS: Number of bytes written=1979226
HDFS: Number of read operations=4
HDFS: Number of large read operations=0
HDFS: Number of write operations=2
Job Counters
Launched map tasks=1
Other local map tasks=1
Total time spent by all maps in occupied slots (ms)=11649
Total time spent by all reduces in occupied slots (ms)=0
Total time spent by all map tasks (ms)=11649
Total vcore-milliseconds taken by all map tasks=11649
Total megabyte-milliseconds taken by all map tasks=2912250
Map-Reduce Framework
Map input records=100003
Map output records=100003
Input split bytes=87
Spilled Records=0
Failed Shuffles=0
Merged Map outputs=0
GC time elapsed (ms)=495
CPU time spent (ms)=6440
Physical memory (bytes) snapshot=145846272
Virtual memory (bytes) snapshot=1941893120
Total committed heap usage (bytes)=45613056
File Input Format Counters
Bytes Read=0
File Output Format Counters
Bytes Written=1979226
24/11/27 10:23:29 INFO mapreduce.ImportJobBase: Transferred 1.8875 MB in 100.0079 seconds (19.3269 KB/sec)
24/11/27 10:23:29 INFO mapreduce.ImportJobBase: Retrieved 100003 records.
24/11/27 10:23:29 INFO mapreduce.ImportJobBase: Publishing Hive/Hcat import job data to Listeners
24/11/27 10:23:31 INFO atlas.ApplicationProperties: Looking for atlas-application.properties in classpath
24/11/27 10:23:31 INFO atlas.ApplicationProperties: Loading atlas-application.properties from file:/etc/sqoop/2.5.0.0-1245/0/atlas-application.properties
29°C
Partly sunny
Search
05:33 PM
27-11-2024

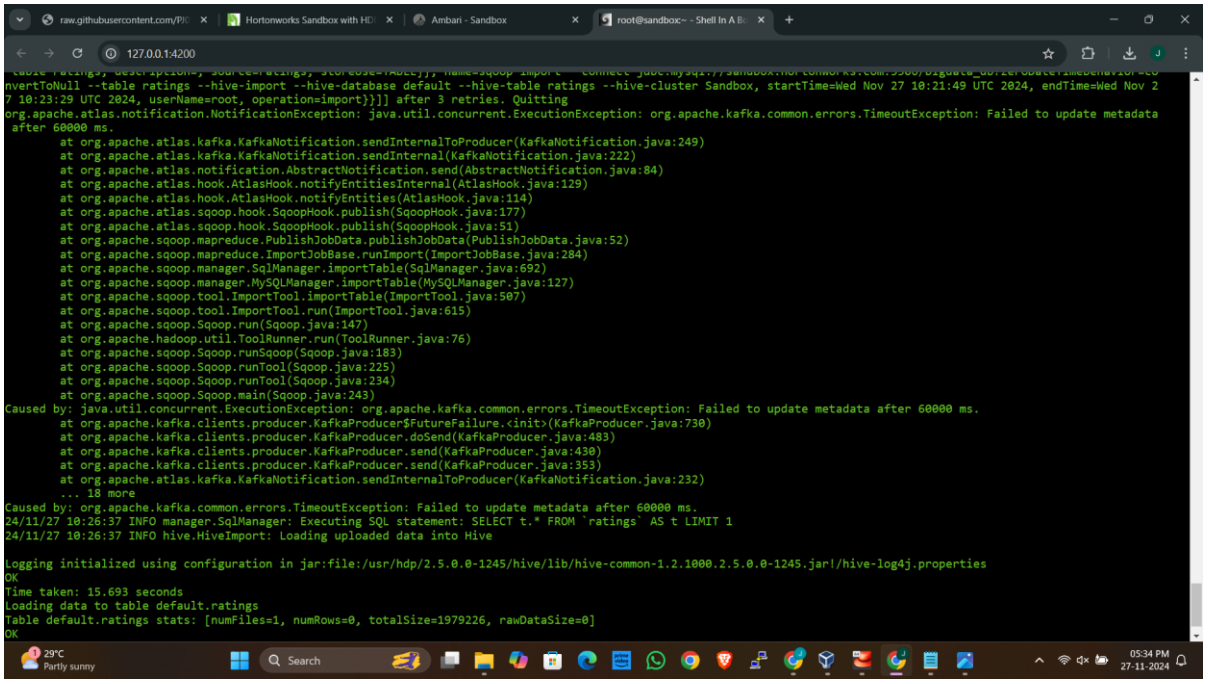

raw.githubusercontent.com/PI
Hortonworks Sandbox with HD
Ambari - Sandbox
root@sandbox:~ - Shell In A B
127.0.0.1:4200
Bytes Written=1979226
24/11/27 10:23:29 INFO mapreduce.ImportJobBase: Transferred 1.8875 MB in 100.0079 seconds (19.3269 KB/sec)
24/11/27 10:23:29 INFO mapreduce.ImportJobBase: Retrieved 100003 records.
24/11/27 10:23:29 INFO mapreduce.ImportJobBase: Publishing Hive/Hcat import job data to Listeners
24/11/27 10:23:31 INFO atlas.ApplicationProperties: Looking for atlas-application.properties in classpath
24/11/27 10:23:31 INFO atlas.ApplicationProperties: Loading atlas-application.properties from file:/etc/sqoop/2.5.0.0-1245/0/atlas-application.properties
24/11/27 10:23:32 ERROR security.InMemoryJAASConfiguration: Unable to add JAAS configuration for client [KafkaClient] as it is missing param [atlas.jaas.KafkaClient.log
inModuleName]. Skipping JAAS config for [KafkaClient]
24/11/27 10:23:32 INFO hook.AtlasHook: Created Atlas Hook
24/11/27 10:23:34 INFO producer.ProducerConfig: ProducerConfig values:
metric.reporters = []
metadata.max.age.ms = 300000
reconnect.backoff.ms = 50
sasl.kerberos.ticket.renew.window.factor = 0.8
bootstrap.servers = [sandbox.hortonworks.com:6667]
ssl.keystore.type = JKS
sasl.mechanism = GSSAPI
max.block.ms = 60000
interceptor.classes = null
ssl.truststore.password = null
client.id =
ssl.endpoint.identification.algorithm = null
request.timeout.ms = 30000
acks = 1
receive.buffer.bytes = 32768
ssl.truststore.type = JKS
retries = 0
ssl.truststore.location = null
ssl.keystore.password = null
send.buffer.bytes = 131072
compression.type = none
metadata.fetch.timeout.ms = 60000
retry.backoff.ms = 100
sasl.kerberos.kinit.cmd = /usr/bin/kinit
buffer.memory = 33554432
timeout.ms = 30000
key.serializer = class org.apache.kafka.common.serialization.StringSerializer
sasl.kerberos.service.name = null
sasl.kerberos.ticket.renew.jitter = 0.05
ssl.trustmanager.algorithm = PKIX
29°C
Partly sunny
Search
05:33 PM
27-11-2024

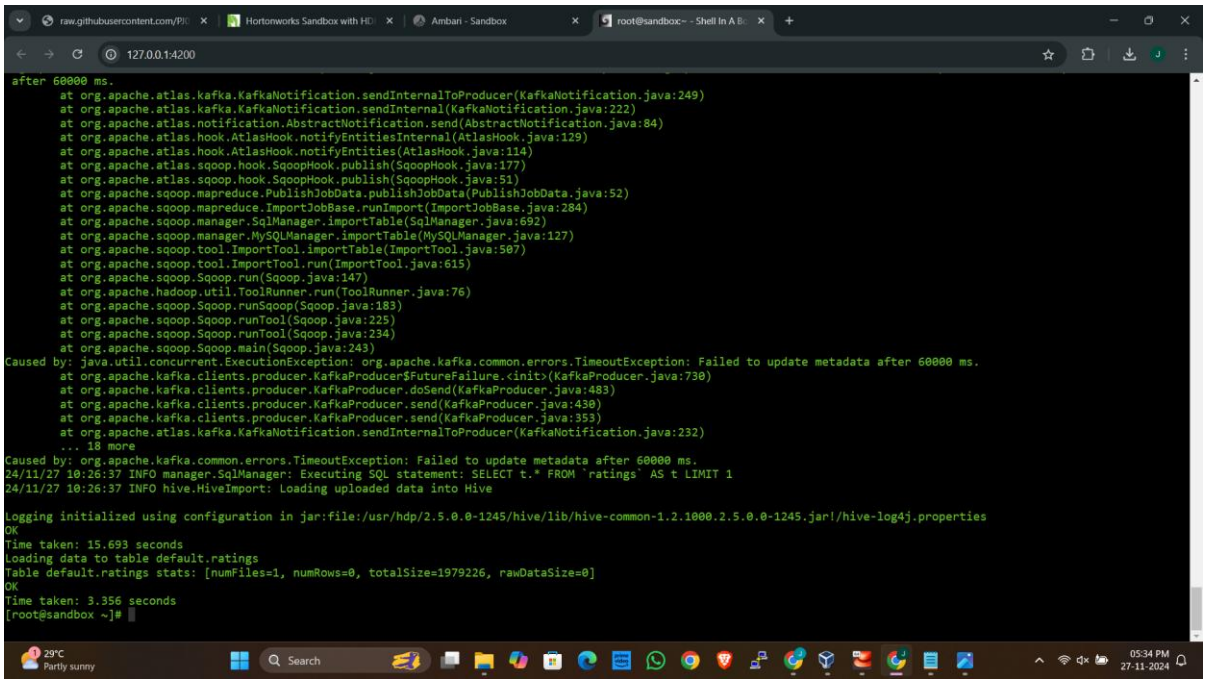

raw.githubusercontent.com/PI
Hortonworks Sandbox with HD
Ambari - Sandbox
root@sandbox:~ - Shell In A B
127.0.0.1:4200
ssl.trustmanager.algorithm = PKIX
block.on.buffer.full = false
ssl.key.password = null
sasl.kerberos.min.time.before.relogin = 60000
connections.max.idle.ms = 540000
max.in.flight.requests.per.connection = 5
metrics.num.samples = 2
ssl.protocol = TLS
ssl.provider = null
ssl.enabled.protocols = [TLSv1.2, TLSv1.1, TLSv1]
batch.size = 16384
ssl.keystore.location = null
ssl.cipher.suites = null
security.protocol = PLAINTEXT
max.request.size = 1048576
value.serializer = class org.apache.kafka.common.serialization.StringSerializer
ssl.keymanager.algorithm = SunX509
metrics.sample.window.ms = 30000
partitioner.class = class org.apache.kafka.clients.producer.internals.DefaultPartitioner
linger.ms = 0
24/11/27 10:23:35 INFO producer.ProducerConfig: ProducerConfig values:
metric.reporters = []
metadata.max.age.ms = 300000
reconnect.backoff.ms = 50
sasl.kerberos.ticket.renew.window.factor = 0.8
bootstrap.servers = [sandbox.hortonworks.com:6667]
ssl.keystore.type = JKS
sasl.mechanism = GSSAPI
max.block.ms = 60000
interceptor.classes = null
ssl.truststore.password = null
client.id = producer-1
ssl.endpoint.identification.algorithm = null
request.timeout.ms = 30000
acks = 1
receive.buffer.bytes = 32768
ssl.truststore.type = JKS
retries = 0
ssl.truststore.location = null
29°C
Partly sunny
Search
05:33 PM
27-11-2024

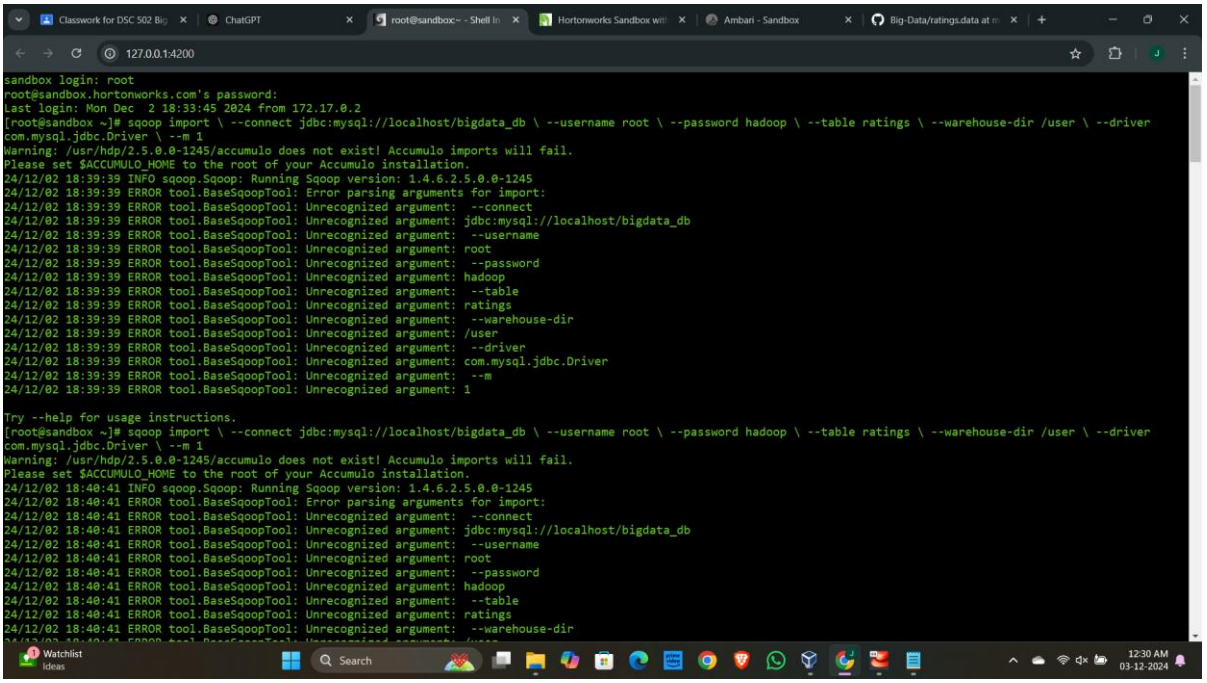

raw.githubusercontent.com/PK
Hortonworks Sandbox with HD
Ambari - Sandbox
root@sandbox:~ - Shell In A B
127.0.0.1:4200
ssl.truststore.type = JKS
retries = 0
ssl.truststore.location = null
ssl.keystore.password = null
send.buffer.bytes = 131072
compression.type = none
metadata.fetch.timeout.ms = 60000
retry.backoff.ms = 100
sasl.kerberos.kinit.cmd = /usr/bin/kinit
buffer.memory = 33554432
timeout.ms = 30000
key.serializer = class org.apache.kafka.common.serialization.StringSerializer
sasl.kerberos.service.name = null
sasl.kerberos.ticket.renew.jitter = 0.05
ssl.trustmanager.algorithm = PKIX
block.on.buffer.full = false
ssl.key.password = null
sasl.kerberos.min.time.before.relogin = 60000
connections.max.idle.ms = 540000
max.in.flight.requests.per.connection = 5
metrics.num.samples = 2
ssl.protocol = TLS
ssl.provider = null
ssl.enabled.protocols = [TLSv1.2, TLSv1.1, TLSv1]
batch.size = 16384
ssl.keystore.location = null
ssl.cipher.suites = null
security.protocol = PLAINTEXT
max.request.size = 1048576
value.serializer = class org.apache.kafka.common.serialization.StringSerializer
ssl.keymanager.algorithm = SunX509
metrics.sample.window.ms = 30000
partitioner.class = class org.apache.kafka.clients.producer.internals.DefaultPartitioner
linger.ms = 0
24/11/27 10:23:35 WARN producer.ProducerConfig: The configuration key.deserializer = org.apache.kafka.common.serialization.StringDeserializer was supplied but isn't a k
nown config.
24/11/27 10:23:35 WARN producer.ProducerConfig: The configuration value.deserializer = org.apache.kafka.common.serialization.StringDeserializer was supplied but isn't a
known config.
24/11/27 10:23:35 WARN producer.ProducerConfig: The configuration hook.group.id = atlas was supplied but isn't a known config.
29°C
Partly sunny
Search
05:34 PM
27-11-2024

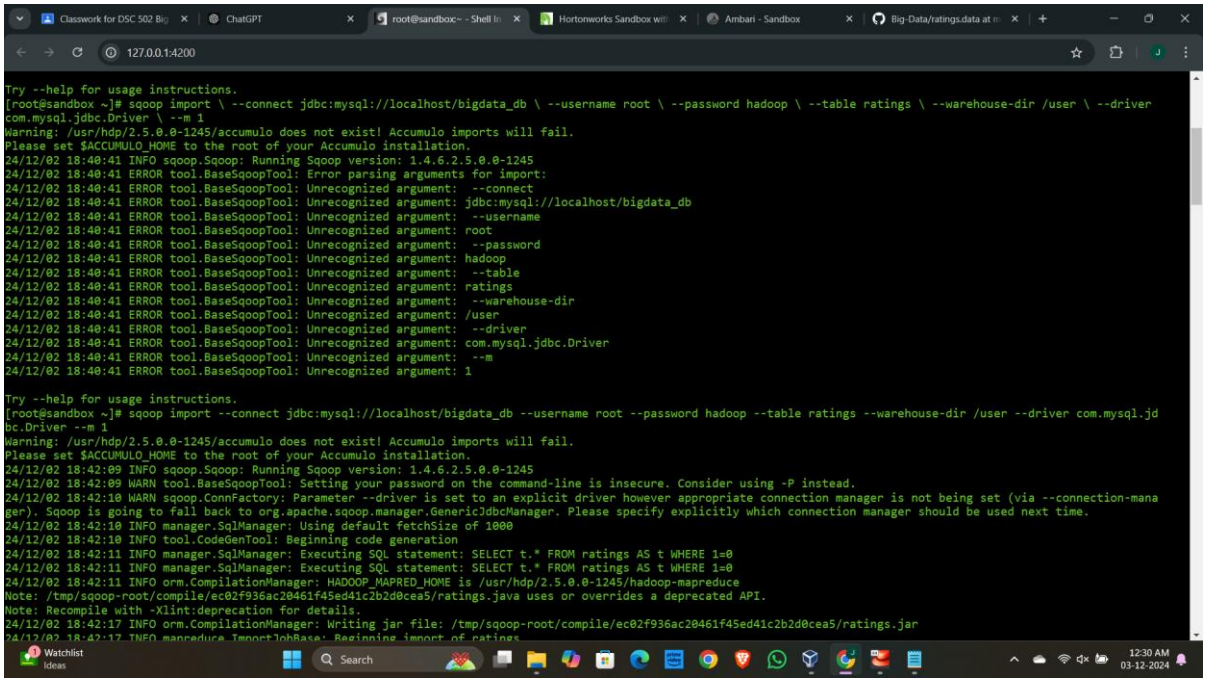

raw.githubusercontent.com/PK
Hortonworks Sandbox with HD
Ambari - Sandbox
root@sandbox:~ - Shell In A B
127.0.0.1:4200
24/11/27 10:23:35 WARN producer.ProducerConfig: The configuration key.deserializer = org.apache.kafka.common.serialization.StringDeserializer was supplied but isn't a k
nown config.
24/11/27 10:23:35 WARN producer.ProducerConfig: The configuration value.deserializer = org.apache.kafka.common.serialization.StringDeserializer was supplied but isn't a
known config.
24/11/27 10:23:35 WARN producer.ProducerConfig: The configuration hook.group.id = atlas was supplied but isn't a known config.
24/11/27 10:23:35 WARN producer.ProducerConfig: The configuration partition.assignment.strategy = roundrobin was supplied but isn't a known config.
24/11/27 10:23:35 WARN producer.ProducerConfig: The configuration zookeeper.connection.timeout.ms = 200 was supplied but isn't a known config.
24/11/27 10:23:35 WARN producer.ProducerConfig: The configuration zookeeper.session.timeout.ms = 400 was supplied but isn't a known config.
24/11/27 10:23:35 WARN producer.ProducerConfig: The configuration zookeeper.connect = sandbox.hortonworks.com:2181 was supplied but isn't a known config.
24/11/27 10:23:35 WARN producer.ProducerConfig: The configuration zookeeper.sync.time.ms = 20 was supplied but isn't a known config.
24/11/27 10:23:35 WARN producer.ProducerConfig: The configuration auto.offset.reset = smallest was supplied but isn't a known config.
24/11/27 10:23:35 INFO utils.AppInfoParser: Kafka version : 0.10.0.2.5.0.0-1245
24/11/27 10:23:35 INFO utils.AppInfoParser: Kafka commitId : dae559f56f07e2cd
24/11/27 10:24:35 ERROR hook.AtlasHook: Failed to send notification - attempt #1; error=java.util.concurrent.ExecutionException: org.apache.kafka.common.errors.TimeoutE
xception: Failed to update metadata after 60000 ms.
24/11/27 10:25:36 ERROR hook.AtlasHook: Failed to send notification - attempt #2; error=java.util.concurrent.ExecutionException: org.apache.kafka.common.errors.TimeoutE
xception: Failed to update metadata after 60000 ms.
29°C
Partly sunny
Search
05:34 PM
27-11-2024

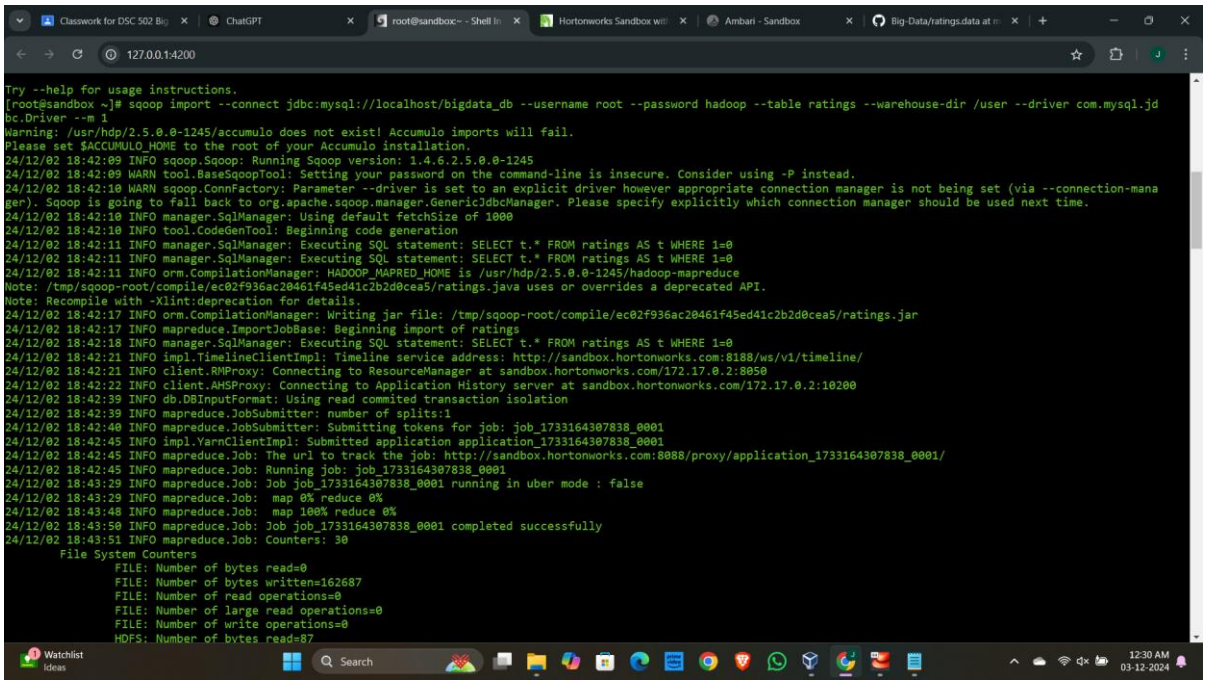

raw.githubusercontent.com/PJ
Hortonworks Sandbox with HD
Ambari - Sandbox
root@sandbox:~ - Shell In A B
127.0.0.1:4200
input.delimiters.enclose.required":"false","codegen.output.dir":".","import.direct.split.size":"0","reset.onemapper":"false","codegen.output.delimiters.record":"10","te
mporary.dirRoot":"_sqoop","hcatalog.create.table":"false","db.batch":"false","import.fetch.size":"-2147483648","accumulo.max.latency":"5000","hdfs.file.format":"TextFil
e","codegen.output.delimiters.field":"1","mainframe.input.dataset.type":"p","codegen.output.delimiters.escape":"0","hcatalog.drop.and.create.table":"false","import.max.
inline.lob.size":"16777216","hbase.create.table":"false","codegen.auto.compile.dir":"true","codegen.compile.dir":"/tmp/sqoop-root/compile/1c0665b09c67a9eba76bd0536f990f
2b","codegen.input.delimiters.enclose":"0","export.new.update":"UpdateOnly","enable.compression":"false","accumulo.batch.size":"10240000","hive.table.name":"ratings","c
odegen.input.delimiters.record":"0","codegen.input.delimiters.escape":"0","accumulo.create.table":"false"},"endTime":"2024-11-27T10:23:29.868Z","inputs":{"jsonClass":"o
rg.apache.atlas.typesystem.json.InstanceSerialization$_Reference","id":{"jsonClass":"org.apache.atlas.typesystem.json.InstanceSerialization$_Id","id":"-1495849268123","
version":0,"typeName":"sqoop_dbdatastore","state":"ACTIVE"},"typeName":"sqoop_dbdatastore","values":{"name":"mysql --url jdbc:mysql://sandbox.hortonworks.com:3306/bigda
ta_db?zeroDateTimeBehavior\u003dconvertToNull --table ratings","source":"ratings","storeUse":"TABLE","description":"","storeUri":"jdbc:mysql://sandbox.hortonworks.com:3
306/bigdata_db?zeroDateTimeBehavior\u003dconvertToNull","qualifiedName":"mysql --url jdbc:mysql://sandbox.hortonworks.com:3306/bigdata_db?zeroDateTimeBehavior\u003dconv
ertToNull --table ratings","owner":"root","dbStoreType":"mysql"},"traitNames":[],"traits":{}},"operation":"import","qualifiedName":"sqoop import --connect jdbc:mysql://
sandbox.hortonworks.com:3306/bigdata_db?zeroDateTimeBehavior\u003dconvertToNull --table ratings --hive-import --hive-database default --hive-table ratings --hive-cluste
r Sandbox","userName":"root"},"traitNames":[],"traits":{}}],"type":"ENTITY_CREATE","user":"root"}}
24/11/27 10:26:37 ERROR hook.AtlasHook: Failed to notify atlas for entity [[{Id='(type: sqoop_dbdatastore, id: <unassigned>)', traits=[], values={owner=root, storeUri=j
dbc:mysql://sandbox.hortonworks.com:3306/bigdata_db?zeroDateTimeBehavior=convertToNull, dbStoreType=mysql, qualifiedName=mysql --url jdbc:mysql://sandbox.hortonworks.co
m:3306/bigdata_db?zeroDateTimeBehavior=convertToNull --table ratings, name=mysql --url jdbc:mysql://sandbox.hortonworks.com:3306/bigdata_db?zeroDateTimeBehavior=convert
ToNull --table ratings, description=, source=ratings, storeUse=TABLE}}, {Id='(type: hive_db, id: <unassigned>)', traits=[], values={qualifiedName=default@Sandbox, clust
erName=Sandbox, name=default}}, {Id='(type: hive_table, id: <unassigned>)', traits=[], values={qualifiedName=default.ratings@Sandbox, name=ratings, db={Id='(type: hive_
db, id: <unassigned>)', traits=[], values={qualifiedName=default@Sandbox, clusterName=Sandbox, name=default}}}}, {Id='(type: sqoop_process, id: <unassigned>)', traits=[
], values={outputs={Id='(type: hive_table, id: <unassigned>)', traits=[], values={qualifiedName=default.ratings@Sandbox, name=ratings, db={Id='(type: hive_db, id: <unas
signed>)', traits=[], values={qualifiedName=default@Sandbox, clusterName=Sandbox, name=default}}}}, commandlineOpts={reset.onemapper=false, codegen.output.delimiters.en
close=0, codegen.input.delimiters.escape=0, codegen.auto.compile.dir=true, accumulo.batch.size=10240000, codegen.input.delimiters.field=0, accumulo.create.table=false,
mainframe.input.dataset.type=p, enable.compression=false, hive.compute.stats.table=false, hive.table.name=ratings, accumulo.max.latency=5000, db.username=root, db.clear
.staging.table=false, codegen.input.delimiters.enclose=0, hdfs.append.dir=false, import.direct.split.size=0, hcatalog.drop.and.create.table=false, codegen.output.delimi
ters.record=10, codegen.output.delimiters.field=1, hbase.bulk.load.enabled=false, mapreduce.num.mappers=1, export.new.update=UpdateOnly, db.require.password=true, hive.
import=true, customtool.options.jsonmap={}, hdfs.delete-target.dir=true, codegen.output.delimiters.enclose.required=false, direct.import=false, codegen.output.dir=., hd
fs.file.format=TextFile, hive.drop.delims=false, codegen.input.delimiters.record=0, db.batch=false, split.limit=null, hcatalog.create.table=false, hive.fail.table.exist
s=false, hive.overwrite.table=false, incremental.mode=None, temporary.dirRoot=_sqoop, verbose=false, import.max.inline.lob.size=16777216, import.fetch.size=-2147483648,
codegen.input.delimiters.enclose.required=false, relaxed.isolation=false, db.table=ratings, hbase.create.table=false, codegen.compile.dir=/tmp/sqoop-root/compile/1c066
5b09c67a9eba76bd0536f990f2b, codegen.output.delimiters.escape=0, db.connect.string=jdbc:mysql://sandbox.hortonworks.com:3306/bigdata_db?zeroDateTimeBehavior=convertToNu
ll}, qualifiedName=sqoop import --connect jdbc:mysql://sandbox.hortonworks.com:3306/bigdata_db?zeroDateTimeBehavior=convertToNull --table ratings --hive-import --hive-d
atabase default --hive-table ratings --hive-cluster Sandbox, inputs={Id='(type: sqoop_dbdatastore, id: <unassigned>)', traits=[], values={owner=root, storeUri=jdbc:mysq
l://sandbox.hortonworks.com:3306/bigdata_db?zeroDateTimeBehavior=convertToNull, dbStoreType=mysql, qualifiedName=mysql --url jdbc:mysql://sandbox.hortonworks.com:3306/b
igdata_db?zeroDateTimeBehavior=convertToNull --table ratings, name=mysql --url jdbc:mysql://sandbox.hortonworks.com:3306/bigdata_db?zeroDateTimeBehavior=convertToNull -
-table ratings, description=, source=ratings, storeUse=TABLE}}, name=sqoop import --connect jdbc:mysql://sandbox.hortonworks.com:3306/bigdata_db?zeroDateTimeBehavior=co
nvertToNull --table ratings --hive-import --hive-database default --hive-table ratings --hive-cluster Sandbox, startTime=Wed Nov 27 10:21:49 UTC 2024, endTime=Wed Nov 2
7 10:23:29 UTC 2024, userName=root, operation=import}}]] after 3 retries. Quitting
org.apache.atlas.notification.NotificationException: java.util.concurrent.ExecutionException: org.apache.kafka.common.errors.TimeoutException: Failed to update metadata
after 60000 ms.
at org.apache.atlas.kafka.KafkaNotification.sendInternalToProducer(KafkaNotification.java:249)
29°C
Partly sunny
Search
05:34 PM
27-11-2024

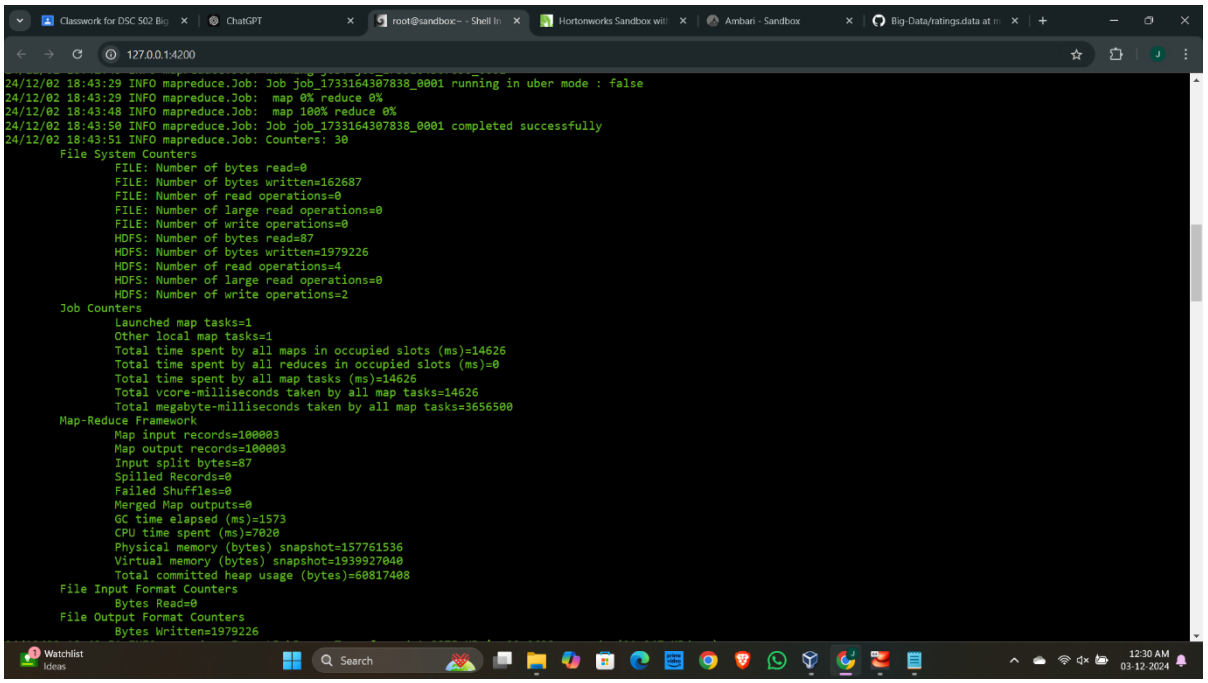

raw.githubusercontent.com/PJ
Hortonworks Sandbox with HD
Ambari - Sandbox
root@sandbox:~ - Shell In A B
127.0.0.1:4200
nvertToNull --table ratings --hive-import --hive-database default --hive-table ratings --hive-cluster Sandbox, startTime=Wed Nov 27 10:21:49 UTC 2024, endTime=Wed Nov 2
7 10:23:29 UTC 2024, userName=root, operation=import}}]] after 3 retries. Quitting
org.apache.atlas.notification.NotificationException: java.util.concurrent.ExecutionException: org.apache.kafka.common.errors.TimeoutException: Failed to update metadata
after 60000 ms.
at org.apache.atlas.kafka.KafkaNotification.sendInternalToProducer(KafkaNotification.java:249)
at org.apache.atlas.kafka.KafkaNotification.sendInternal(KafkaNotification.java:222)
at org.apache.atlas.notification.AbstractNotification.send(AbstractNotification.java:84)
at org.apache.atlas.hook.AtlasHook.notifyEntitiesInternal(AtlasHook.java:129)
at org.apache.atlas.hook.AtlasHook.notifyEntities(AtlasHook.java:114)
at org.apache.atlas.sqoop.hook.SqoopHook.publish(SqoopHook.java:177)
at org.apache.atlas.sqoop.hook.SqoopHook.publish(SqoopHook.java:51)
at org.apache.sqoop.mapreduce.PublishJobData.publishJobData(PublishJobData.java:52)
at org.apache.sqoop.mapreduce.ImportJobBase.runImport(ImportJobBase.java:284)
at org.apache.sqoop.manager.SqlManager.importTable(SqlManager.java:692)
at org.apache.sqoop.manager.MySQLManager.importTable(MySQLManager.java:127)
at org.apache.sqoop.tool.ImportTool.importTable(ImportTool.java:507)
at org.apache.sqoop.tool.ImportTool.run(ImportTool.java:615)
at org.apache.sqoop.Sqoop.run(Sqoop.java:147)
at org.apache.hadoop.util.ToolRunner.run(ToolRunner.java:76)
at org.apache.sqoop.Sqoop.runSqoop(Sqoop.java:183)
at org.apache.sqoop.Sqoop.runTool(Sqoop.java:225)
at org.apache.sqoop.Sqoop.runTool(Sqoop.java:234)
at org.apache.sqoop.Sqoop.main(Sqoop.java:243)
Caused by: java.util.concurrent.ExecutionException: org.apache.kafka.common.errors.TimeoutException: Failed to update metadata after 60000 ms.
at org.apache.kafka.clients.producer.KafkaProducer$FutureFailure.<init>(KafkaProducer.java:730)
at org.apache.kafka.clients.producer.KafkaProducer.doSend(KafkaProducer.java:483)
at org.apache.kafka.clients.producer.KafkaProducer.send(KafkaProducer.java:430)
at org.apache.kafka.clients.producer.KafkaProducer.send(KafkaProducer.java:353)
at org.apache.atlas.kafka.KafkaNotification.sendInternalToProducer(KafkaNotification.java:232)
... 18 more
Caused by: org.apache.kafka.common.errors.TimeoutException: Failed to update metadata after 60000 ms.
24/11/27 10:26:37 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `ratings` AS t LIMIT 1
24/11/27 10:26:37 INFO hive.HiveImport: Loading uploaded data into Hive
Logging initialized using configuration in jar:file:/usr/hdp/2.5.0.0-1245/hive/lib/hive-common-1.2.1000.2.5.0.0-1245.jar!/hive-log4j.properties
OK
Time taken: 15.693 seconds
Loading data to table default.ratings
Table default.ratings stats: [numFiles=1, numRows=0, totalSize=1979226, rawDataSize=0]
OK
29°C
Partly sunny
Search
05:34 PM
27-11-2024

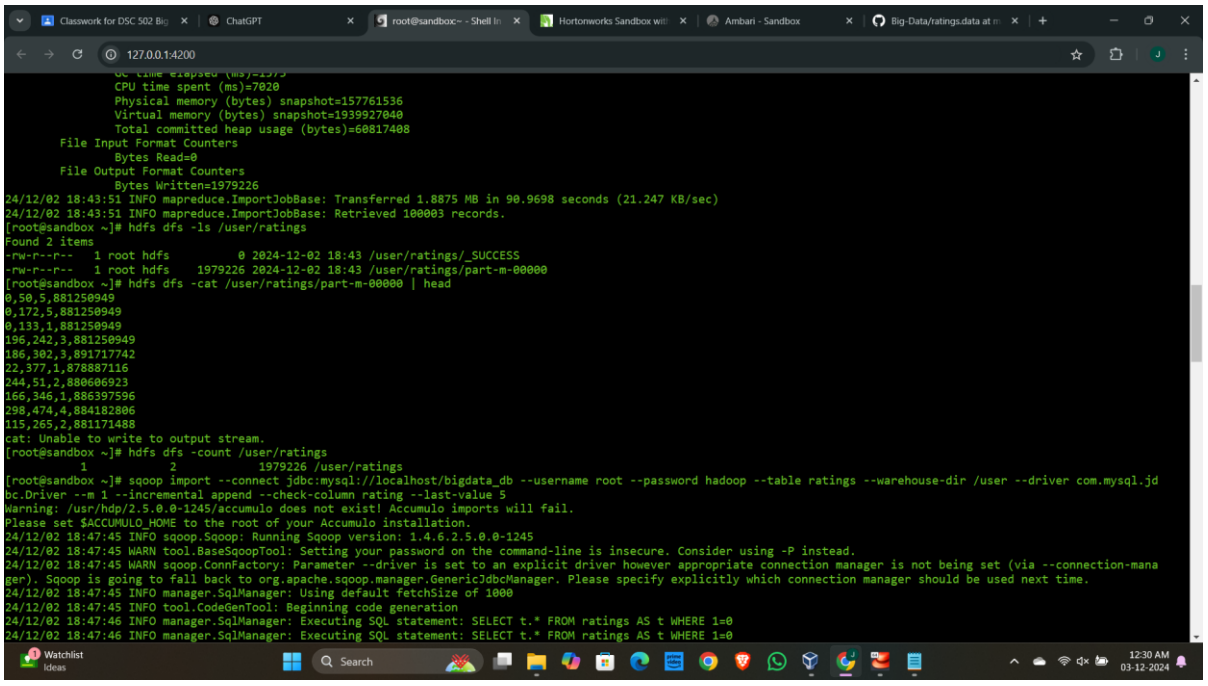

raw.githubusercontent.com/PJ
Hortonworks Sandbox with HD
Ambari - Sandbox
root@sandbox:~ - Shell In A B
127.0.0.1:4200
after 60000 ms.
at org.apache.atlas.kafka.KafkaNotification.sendInternalToProducer(KafkaNotification.java:249)
at org.apache.atlas.kafka.KafkaNotification.sendInternal(KafkaNotification.java:222)
at org.apache.atlas.notification.AbstractNotification.send(AbstractNotification.java:84)
at org.apache.atlas.hook.AtlasHook.notifyEntitiesInternal(AtlasHook.java:129)
at org.apache.atlas.hook.AtlasHook.notifyEntities(AtlasHook.java:114)
at org.apache.atlas.sqoop.hook.SqoopHook.publish(SqoopHook.java:177)
at org.apache.atlas.sqoop.hook.SqoopHook.publish(SqoopHook.java:51)
at org.apache.sqoop.mapreduce.PublishJobData.publishJobData(PublishJobData.java:52)
at org.apache.sqoop.mapreduce.ImportJobBase.runImport(ImportJobBase.java:284)
at org.apache.sqoop.manager.SqlManager.importTable(SqlManager.java:692)
at org.apache.sqoop.manager.MySQLManager.importTable(MySQLManager.java:127)
at org.apache.sqoop.tool.ImportTool.importTable(ImportTool.java:507)
at org.apache.sqoop.tool.ImportTool.run(ImportTool.java:615)
at org.apache.sqoop.Sqoop.run(Sqoop.java:147)
at org.apache.hadoop.util.ToolRunner.run(ToolRunner.java:76)
at org.apache.sqoop.Sqoop.runSqoop(Sqoop.java:183)
at org.apache.sqoop.Sqoop.runTool(Sqoop.java:225)
at org.apache.sqoop.Sqoop.runTool(Sqoop.java:234)
at org.apache.sqoop.Sqoop.main(Sqoop.java:243)
Caused by: java.util.concurrent.ExecutionException: org.apache.kafka.common.errors.TimeoutException: Failed to update metadata after 60000 ms.
at org.apache.kafka.clients.producer.KafkaProducer$FutureFailure.<init>(KafkaProducer.java:730)
at org.apache.kafka.clients.producer.KafkaProducer.doSend(KafkaProducer.java:483)
at org.apache.kafka.clients.producer.KafkaProducer.send(KafkaProducer.java:430)
at org.apache.kafka.clients.producer.KafkaProducer.send(KafkaProducer.java:353)
at org.apache.atlas.kafka.KafkaNotification.sendInternalToProducer(KafkaNotification.java:232)
... 18 more
Caused by: org.apache.kafka.common.errors.TimeoutException: Failed to update metadata after 60000 ms.
24/11/27 10:26:37 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `ratings` AS t LIMIT 1
24/11/27 10:26:37 INFO hive.HiveImport: Loading uploaded data into Hive
Logging initialized using configuration in jar:file:/usr/hdp/2.5.0.0-1245/hive/lib/hive-common-1.2.1000.2.5.0.0-1245.jar!/hive-log4j.properties
OK
Time taken: 15.693 seconds
Loading data to table default.ratings
Table default.ratings stats: [numFiles=1, numRows=0, totalSize=1979226, rawDataSize=0]
OK
Time taken: 3.356 seconds
[root@sandbox ~]#
29°C
Partly sunny
Search
05:34 PM
27-11-2024

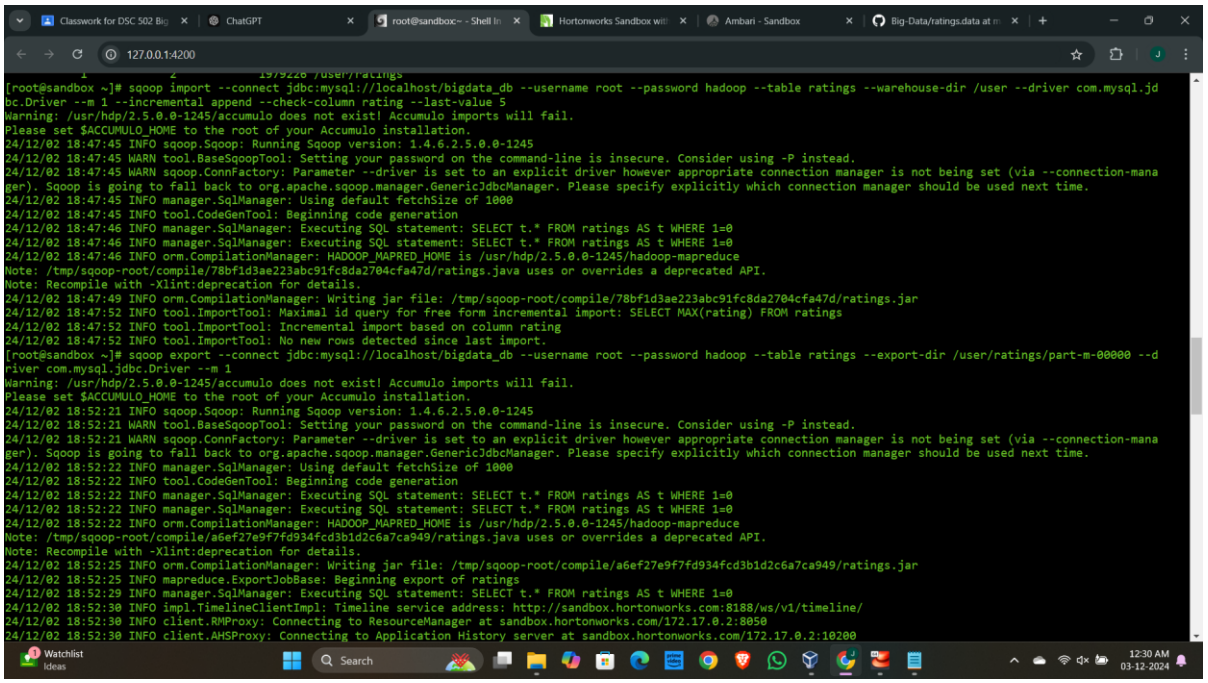

Classwork for DSC 502 Big
ChatGPT
root@sandbox:~ - Shell In
Hortonworks Sandbox wit
Ambari - Sandbox
Big-Data/ratings.data at m
127.0.0.1:4200
sandbox login: root
root@sandbox.hortonworks.com's password:
Last login: Mon Dec 2 18:33:45 2024 from 172.17.0.2
[root@sandbox ~]# sqoop import \ --connect jdbc:mysql://localhost/bigdata_db \ --username root \ --password hadoop \ --table ratings \ --warehouse-dir /user \ --driver
com.mysql.jdbc.Driver \ --m 1
Warning: /usr/hdp/2.5.0.0-1245/accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
24/12/02 18:39:39 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6.2.5.0.0-1245
24/12/02 18:39:39 ERROR tool.BaseSqoopTool: Error parsing arguments for import:
24/12/02 18:39:39 ERROR tool.BaseSqoopTool: Unrecognized argument: --connect
24/12/02 18:39:39 ERROR tool.BaseSqoopTool: Unrecognized argument: jdbc:mysql://localhost/bigdata_db
24/12/02 18:39:39 ERROR tool.BaseSqoopTool: Unrecognized argument: --username
24/12/02 18:39:39 ERROR tool.BaseSqoopTool: Unrecognized argument: root
24/12/02 18:39:39 ERROR tool.BaseSqoopTool: Unrecognized argument: --password
24/12/02 18:39:39 ERROR tool.BaseSqoopTool: Unrecognized argument: hadoop
24/12/02 18:39:39 ERROR tool.BaseSqoopTool: Unrecognized argument: --table
24/12/02 18:39:39 ERROR tool.BaseSqoopTool: Unrecognized argument: ratings
24/12/02 18:39:39 ERROR tool.BaseSqoopTool: Unrecognized argument: --warehouse-dir
24/12/02 18:39:39 ERROR tool.BaseSqoopTool: Unrecognized argument: /user
24/12/02 18:39:39 ERROR tool.BaseSqoopTool: Unrecognized argument: --driver
24/12/02 18:39:39 ERROR tool.BaseSqoopTool: Unrecognized argument: com.mysql.jdbc.Driver
24/12/02 18:39:39 ERROR tool.BaseSqoopTool: Unrecognized argument: --m
24/12/02 18:39:39 ERROR tool.BaseSqoopTool: Unrecognized argument: 1
Try --help for usage instructions.
[root@sandbox ~]# sqoop import \ --connect jdbc:mysql://localhost/bigdata_db \ --username root \ --password hadoop \ --table ratings \ --warehouse-dir /user \ --driver
com.mysql.jdbc.Driver \ --m 1
Warning: /usr/hdp/2.5.0.0-1245/accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
24/12/02 18:40:41 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6.2.5.0.0-1245
24/12/02 18:40:41 ERROR tool.BaseSqoopTool: Error parsing arguments for import:
24/12/02 18:40:41 ERROR tool.BaseSqoopTool: Unrecognized argument: --connect
24/12/02 18:40:41 ERROR tool.BaseSqoopTool: Unrecognized argument: jdbc:mysql://localhost/bigdata_db
24/12/02 18:40:41 ERROR tool.BaseSqoopTool: Unrecognized argument: --username
24/12/02 18:40:41 ERROR tool.BaseSqoopTool: Unrecognized argument: root
24/12/02 18:40:41 ERROR tool.BaseSqoopTool: Unrecognized argument: --password
24/12/02 18:40:41 ERROR tool.BaseSqoopTool: Unrecognized argument: hadoop
24/12/02 18:40:41 ERROR tool.BaseSqoopTool: Unrecognized argument: --table
24/12/02 18:40:41 ERROR tool.BaseSqoopTool: Unrecognized argument: ratings
24/12/02 18:40:41 ERROR tool.BaseSqoopTool: Unrecognized argument: --warehouse-dir
Watchlist
Ideas
Search
12:30 AM
03-12-2024

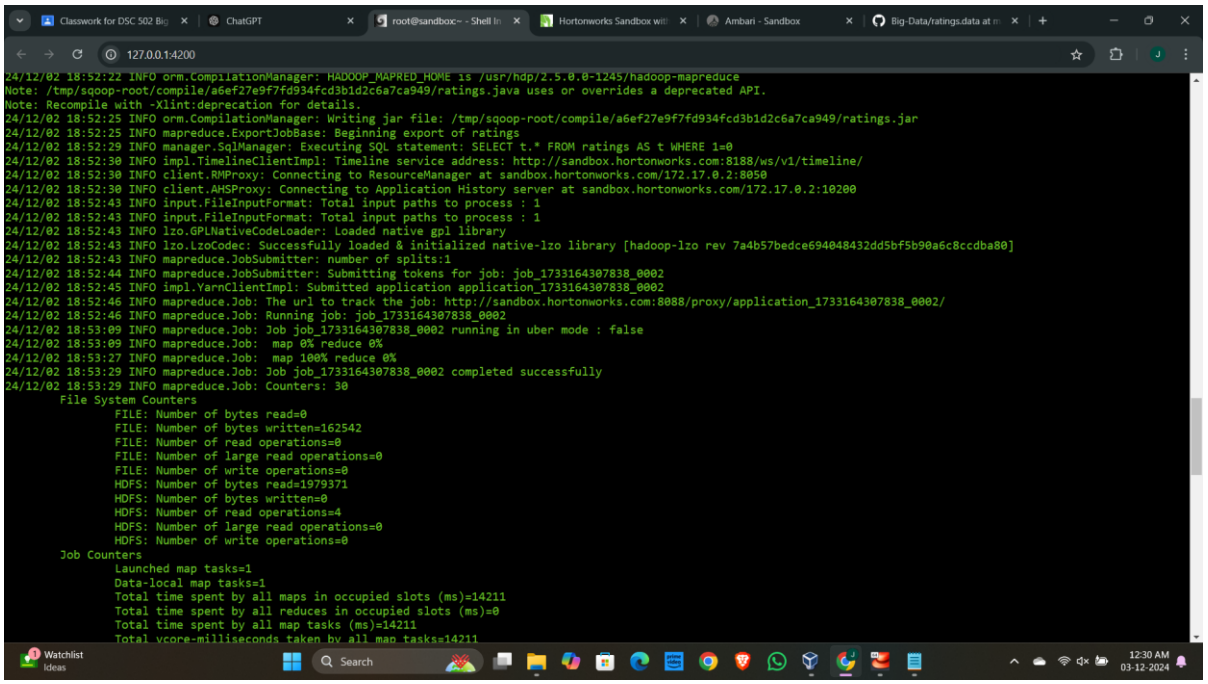

Classwork for DSC 502 Big
ChatGPT
root@sandbox:~ - Shell In
Hortonworks Sandbox with
Ambari - Sandbox
Big-Data/ratings.data at m
127.0.0.1:4200
Try --help for usage instructions.
[root@sandbox ~]# sqoop import \ --connect jdbc:mysql://localhost/bigdata_db \ --username root \ --password hadoop \ --table ratings \ --warehouse-dir /user \ --driver com.mysql.jdbc.Driver \ --m 1
Warning: /usr/hdp/2.5.0.0-1245/accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
24/12/02 18:40:41 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6.2.5.0.0-1245
24/12/02 18:40:41 ERROR tool.BaseSqoopTool: Error parsing arguments for import:
24/12/02 18:40:41 ERROR tool.BaseSqoopTool: Unrecognized argument: --connect
24/12/02 18:40:41 ERROR tool.BaseSqoopTool: Unrecognized argument: jdbc:mysql://localhost/bigdata_db
24/12/02 18:40:41 ERROR tool.BaseSqoopTool: Unrecognized argument: --username
24/12/02 18:40:41 ERROR tool.BaseSqoopTool: Unrecognized argument: root
24/12/02 18:40:41 ERROR tool.BaseSqoopTool: Unrecognized argument: --password
24/12/02 18:40:41 ERROR tool.BaseSqoopTool: Unrecognized argument: hadoop
24/12/02 18:40:41 ERROR tool.BaseSqoopTool: Unrecognized argument: --table
24/12/02 18:40:41 ERROR tool.BaseSqoopTool: Unrecognized argument: ratings
24/12/02 18:40:41 ERROR tool.BaseSqoopTool: Unrecognized argument: --warehouse-dir
24/12/02 18:40:41 ERROR tool.BaseSqoopTool: Unrecognized argument: /user
24/12/02 18:40:41 ERROR tool.BaseSqoopTool: Unrecognized argument: --driver
24/12/02 18:40:41 ERROR tool.BaseSqoopTool: Unrecognized argument: com.mysql.jdbc.Driver
24/12/02 18:40:41 ERROR tool.BaseSqoopTool: Unrecognized argument: --m
24/12/02 18:40:41 ERROR tool.BaseSqoopTool: Unrecognized argument: 1
Try --help for usage instructions.
[root@sandbox ~]# sqoop import --connect jdbc:mysql://localhost/bigdata_db --username root --password hadoop --table ratings --warehouse-dir /user --driver com.mysql.jdbc.Driver --m 1
Warning: /usr/hdp/2.5.0.0-1245/accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
24/12/02 18:42:09 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6.2.5.0.0-1245
24/12/02 18:42:09 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.
24/12/02 18:42:10 WARN sqoop.ConnFactory: Parameter --driver is set to an explicit driver however appropriate connection manager is not being set (via --connection-manager). Sqoop is going to fall back to org.apache.sqoop.manager.GenericJdbcManager. Please specify explicitly which connection manager should be used next time.
24/12/02 18:42:10 INFO manager.SqlManager: Using default fetchSize of 1000
24/12/02 18:42:10 INFO tool.CodeGenTool: Beginning code generation
24/12/02 18:42:11 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM ratings AS t WHERE 1=0
24/12/02 18:42:11 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM ratings AS t WHERE 1=0
24/12/02 18:42:11 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/hdp/2.5.0.0-1245/hadoop-mapreduce
Note: /tmp/sqoop-root/compile/ec02f936ac20461f45ed41c2b2d0cea5/ratings.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
24/12/02 18:42:17 INFO orm.CompilationManager: Writing jar file: /tmp/sqoop-root/compile/ec02f936ac20461f45ed41c2b2d0cea5/ratings.jar
Watchlist
Ideas
Search
12:30 AM
03-12-2024

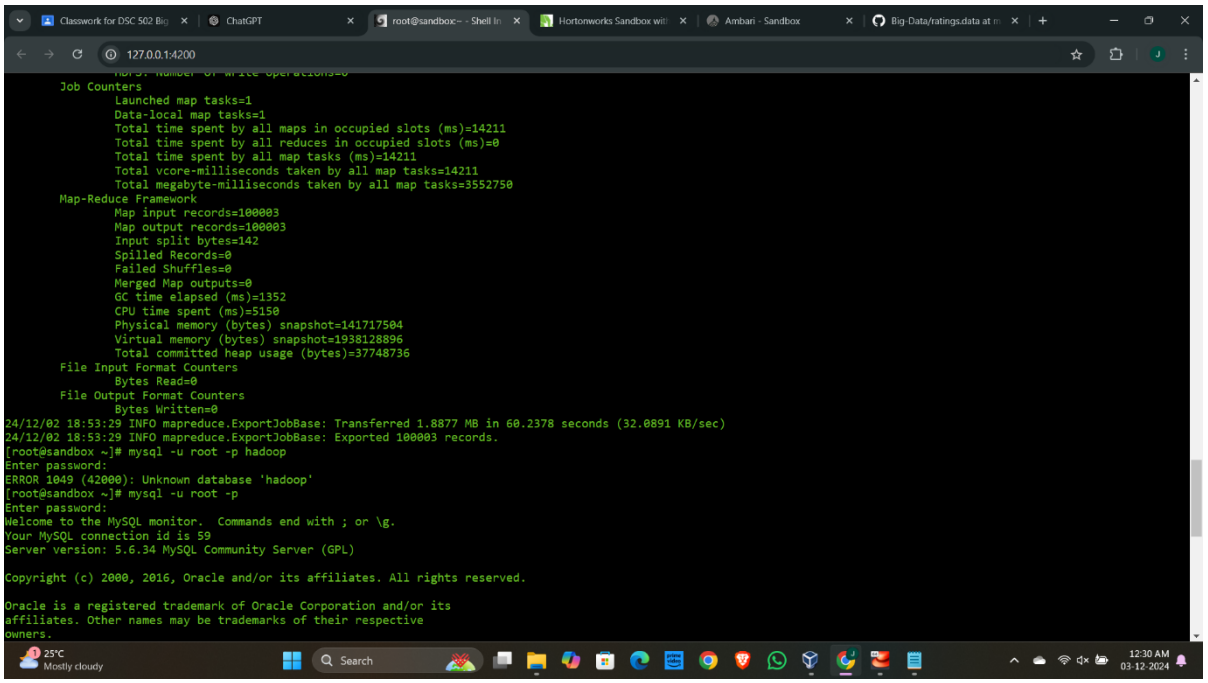

Classwork for DSC 502 Big
ChatGPT
root@sandbox:~ - Shell In
Hortonworks Sandbox with
Ambari - Sandbox
Big-Data/ratings.data at m
127.0.0.1:4200
Try --help for usage instructions.
[root@sandbox ~]# sqoop import --connect jdbc:mysql://localhost/bigdata_db --username root --password hadoop --table ratings --warehouse-dir /user --driver com.mysql.jdbc.Driver --m 1
Warning: /usr/hdp/2.5.0.0-1245/accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
24/12/02 18:42:09 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6.2.5.0.0-1245
24/12/02 18:42:09 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.
24/12/02 18:42:10 WARN sqoop.ConnFactory: Parameter --driver is set to an explicit driver however appropriate connection manager is not being set (via --connection-manager). Sqoop is going to fall back to org.apache.sqoop.manager.GenericJdbcManager. Please specify explicitly which connection manager should be used next time.
24/12/02 18:42:10 INFO manager.SqlManager: Using default fetchSize of 1000
24/12/02 18:42:10 INFO tool.CodeGenTool: Beginning code generation
24/12/02 18:42:11 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM ratings AS t WHERE 1=0
24/12/02 18:42:11 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM ratings AS t WHERE 1=0
24/12/02 18:42:11 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/hdp/2.5.0.0-1245/hadoop-mapreduce
Note: /tmp/sqoop-root/compile/ec02f936ac20461f45ed41c2b2d0cea5/ratings.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
24/12/02 18:42:17 INFO orm.CompilationManager: Writing jar file: /tmp/sqoop-root/compile/ec02f936ac20461f45ed41c2b2d0cea5/ratings.jar
24/12/02 18:42:17 INFO mapreduce.ImportJobBase: Beginning import of ratings
24/12/02 18:42:18 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM ratings AS t WHERE 1=0
24/12/02 18:42:21 INFO impl.TimelineClientImpl: Timeline service address: http://sandbox.hortonworks.com:8188/ws/v1/timeline/
24/12/02 18:42:21 INFO client.RMProxy: Connecting to ResourceManager at sandbox.hortonworks.com/172.17.0.2:8050
24/12/02 18:42:22 INFO client.AHSProxy: Connecting to Application History server at sandbox.hortonworks.com/172.17.0.2:10200
24/12/02 18:42:39 INFO db.DBInputFormat: Using read commited transaction isolation
24/12/02 18:42:39 INFO mapreduce.JobSubmitter: number of splits:1
24/12/02 18:42:40 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1733164307838_0001
24/12/02 18:42:45 INFO impl.YarnClientImpl: Submitted application application_1733164307838_0001
24/12/02 18:42:45 INFO mapreduce.Job: The url to track the job: http://sandbox.hortonworks.com:8088/proxy/application_1733164307838_0001/
24/12/02 18:42:45 INFO mapreduce.Job: Running job: job_1733164307838_0001
24/12/02 18:43:29 INFO mapreduce.Job: Job job_1733164307838_0001 running in uber mode : false
24/12/02 18:43:29 INFO mapreduce.Job: map 0% reduce 0%
24/12/02 18:43:48 INFO mapreduce.Job: map 100% reduce 0%
24/12/02 18:43:50 INFO mapreduce.Job: Job job_1733164307838_0001 completed successfully
24/12/02 18:43:51 INFO mapreduce.Job: Counters: 30
File System Counters
FILE: Number of bytes read=0
FILE: Number of bytes written=162687
FILE: Number of read operations=0
FILE: Number of large read operations=0
FILE: Number of write operations=0
HDFS: Number of bytes read=87
Watchlist
Ideas
Search
12:30 AM
03-12-2024

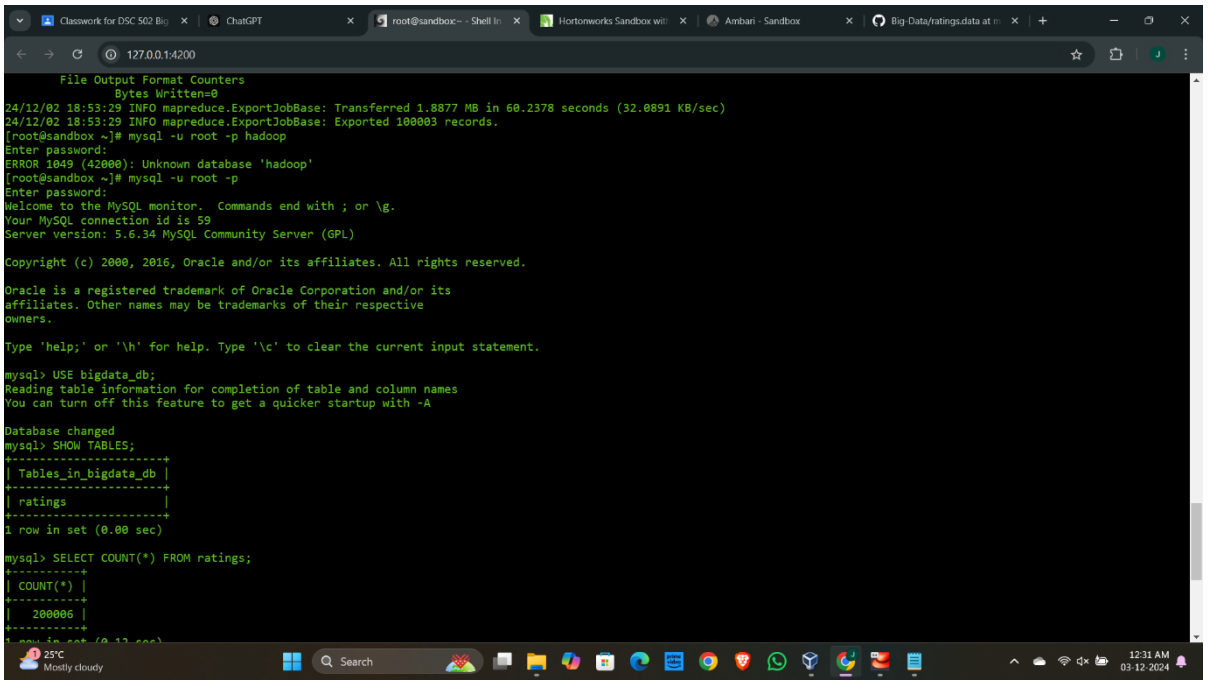

24/12/02 18:43:29 INFO mapreduce.Job: Job job_1733164307838_0001 running in uber mode : false
24/12/02 18:43:29 INFO mapreduce.Job: map 0% reduce 0%
24/12/02 18:43:48 INFO mapreduce.Job: map 100% reduce 0%
24/12/02 18:43:50 INFO mapreduce.Job: Job job_1733164307838_0001 completed successfully
24/12/02 18:43:51 INFO mapreduce.Job: Counters: 30
File System Counters
FILE: Number of bytes read=0
FILE: Number of bytes written=162687
FILE: Number of read operations=0
FILE: Number of large read operations=0
FILE: Number of write operations=0
HDFS: Number of bytes read=87
HDFS: Number of bytes written=1979226
HDFS: Number of read operations=4
HDFS: Number of large read operations=0
HDFS: Number of write operations=2
Job Counters
Launched map tasks=1
Other local map tasks=1
Total time spent by all maps in occupied slots (ms)=14626
Total time spent by all reduces in occupied slots (ms)=0
Total time spent by all map tasks (ms)=14626
Total vcore-milliseconds taken by all map tasks=14626
Total megabyte-milliseconds taken by all map tasks=3656500
Map-Reduce Framework
Map input records=100003
Map output records=100003
Input split bytes=87
Spilled Records=0
Failed Shuffles=0
Merged Map outputs=0
GC time elapsed (ms)=1573
CPU time spent (ms)=7020
Physical memory (bytes) snapshot=157761536
Virtual memory (bytes) snapshot=1939927040
Total committed heap usage (bytes)=60817408
File Input Format Counters
Bytes Read=0
File Output Format Counters
Bytes Written=1979226


CPU time spent (ms)=7020
Physical memory (bytes) snapshot=157761536
Virtual memory (bytes) snapshot=1939927040
Total committed heap usage (bytes)=60817408
File Input Format Counters
Bytes Read=0
File Output Format Counters
Bytes Written=1979226
24/12/02 18:43:51 INFO mapreduce.ImportJobBase: Transferred 1.8875 MB in 90.9698 seconds (21.247 KB/sec)
24/12/02 18:43:51 INFO mapreduce.ImportJobBase: Retrieved 100003 records.
[root@sandbox ~]# hdfs dfs -ls /user/ratings
Found 2 items
-rw-r--r--   1 root hdfs   0 2024-12-02 18:43 /user/ratings/_SUCCESS
-rw-r--r--   1 root hdfs   1979226 2024-12-02 18:43 /user/ratings/part-m-00000
[root@sandbox ~]# hdfs dfs -cat /user/ratings/part-m-00000 | head
0,50,5,881250949
0,172,5,881250949
0,133,1,881250949
196,242,3,881250949
186,302,3,891717742
22,377,1,878887116
244,51,2,880606923
166,346,1,886397596
298,474,4,884182806
115,265,2,881171488
cat: Unable to write to output stream.
[root@sandbox ~]# hdfs dfs -count /user/ratings
1   2   1979226 /user/ratings
[root@sandbox ~]# sqoop import --connect jdbc:mysql://localhost/bigdata_db --username root --password hadoop --table ratings --warehouse-dir /user --driver com.mysql.jd
bc.Driver --m 1 --incremental append --check-column rating --last-value 5
Warning: /usr/hdp/2.5.0.0-1245/accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
24/12/02 18:47:45 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6.2.5.0.0-1245
24/12/02 18:47:45 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.
24/12/02 18:47:45 WARN sqoop.ConnFactory: Parameter --driver is set to an explicit driver however appropriate connection manager is not being set (via --connection-mana
ger). Sqoop is going to fall back to org.apache.sqoop.manager.GenericJdbcManager. Please specify explicitly which connection manager should be used next time.
24/12/02 18:47:45 INFO manager.SqlManager: Using default fetchSize of 1000
24/12/02 18:47:45 INFO tool.CodeGenTool: Beginning code generation
24/12/02 18:47:46 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM ratings AS t WHERE 1=0
24/12/02 18:47:46 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM ratings AS t WHERE 1=0


Classwork for DSC 502 Big
ChatGPT
root@sandbox:~ - Shell In
Hortonworks Sandbox wit
Ambari - Sandbox
Big-Data/ratings.data at m
127.0.0.1:4200
[root@sandbox ~]# sqoop import --connect jdbc:mysql://localhost/bigdata_db --username root --password hadoop --table ratings --warehouse-dir /user --driver com.mysql.jdbc.Driver --m 1 --incremental append --check-column rating --last-value 5
Warning: /usr/hdp/2.5.0.0-1245/accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
24/12/02 18:47:45 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6.2.5.0.0-1245
24/12/02 18:47:45 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.
24/12/02 18:47:45 WARN sqoop.ConnFactory: Parameter --driver is set to an explicit driver however appropriate connection manager is not being set (via --connection-manager). Sqoop is going to fall back to org.apache.sqoop.manager.GenericJdbcManager. Please specify explicitly which connection manager should be used next time.
24/12/02 18:47:45 INFO manager.SqlManager: Using default fetchSize of 1000
24/12/02 18:47:45 INFO tool.CodeGenTool: Beginning code generation
24/12/02 18:47:46 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM ratings AS t WHERE 1=0
24/12/02 18:47:46 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM ratings AS t WHERE 1=0
24/12/02 18:47:46 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/hdp/2.5.0.0-1245/hadoop-mapreduce
Note: /tmp/sqoop-root/compile/78bf1d3ae223abc91fc8da2704cfa47d/ratings.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
24/12/02 18:47:49 INFO orm.CompilationManager: Writing jar file: /tmp/sqoop-root/compile/78bf1d3ae223abc91fc8da2704cfa47d/ratings.jar
24/12/02 18:47:52 INFO tool.ImportTool: Maximal id query for free form incremental import: SELECT MAX(rating) FROM ratings
24/12/02 18:47:52 INFO tool.ImportTool: Incremental import based on column rating
24/12/02 18:47:52 INFO tool.ImportTool: No new rows detected since last import.
[root@sandbox ~]# sqoop export --connect jdbc:mysql://localhost/bigdata_db --username root --password hadoop --table ratings --export-dir /user/ratings/part-m-00000 --driver com.mysql.jdbc.Driver --m 1
Warning: /usr/hdp/2.5.0.0-1245/accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
24/12/02 18:52:21 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6.2.5.0.0-1245
24/12/02 18:52:21 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.
24/12/02 18:52:21 WARN sqoop.ConnFactory: Parameter --driver is set to an explicit driver however appropriate connection manager is not being set (via --connection-manager). Sqoop is going to fall back to org.apache.sqoop.manager.GenericJdbcManager. Please specify explicitly which connection manager should be used next time.
24/12/02 18:52:22 INFO manager.SqlManager: Using default fetchSize of 1000
24/12/02 18:52:22 INFO tool.CodeGenTool: Beginning code generation
24/12/02 18:52:22 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM ratings AS t WHERE 1=0
24/12/02 18:52:22 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM ratings AS t WHERE 1=0
24/12/02 18:52:22 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/hdp/2.5.0.0-1245/hadoop-mapreduce
Note: /tmp/sqoop-root/compile/a6ef27e9f7fd934fcd3b1d2c6a7ca949/ratings.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
24/12/02 18:52:25 INFO orm.CompilationManager: Writing jar file: /tmp/sqoop-root/compile/a6ef27e9f7fd934fcd3b1d2c6a7ca949/ratings.jar
24/12/02 18:52:25 INFO mapreduce.ExportJobBase: Beginning export of ratings
24/12/02 18:52:29 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM ratings AS t WHERE 1=0
24/12/02 18:52:30 INFO impl.TimelineClientImpl: Timeline service address: http://sandbox.hortonworks.com:8188/ws/v1/timeline/
24/12/02 18:52:30 INFO client.RMProxy: Connecting to ResourceManager at sandbox.hortonworks.com/172.17.0.2:8050
24/12/02 18:52:30 INFO client.AHSProxy: Connecting to Application History server at sandbox.hortonworks.com/172.17.0.2:10200
Watchlist
Ideas
Search
12:30 AM
03-12-2024


Classwork for DSC 502 Big
ChatGPT
root@sandbox:~ - Shell In
Hortonworks Sandbox wit
Ambari - Sandbox
Big-Data/ratings.data at m
127.0.0.1:4200
24/12/02 18:52:22 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/hdp/2.5.0.0-1245/hadoop-mapreduce
Note: /tmp/sqoop-root/compile/a6ef27e9f7fd934fcd3b1d2c6a7ca949/ratings.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
24/12/02 18:52:25 INFO orm.CompilationManager: Writing jar file: /tmp/sqoop-root/compile/a6ef27e9f7fd934fcd3b1d2c6a7ca949/ratings.jar
24/12/02 18:52:25 INFO mapreduce.ExportJobBase: Beginning export of ratings
24/12/02 18:52:29 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM ratings AS t WHERE 1=0
24/12/02 18:52:30 INFO impl.TimelineClientImpl: Timeline service address: http://sandbox.hortonworks.com:8188/ws/v1/timeline/
24/12/02 18:52:30 INFO client.RMProxy: Connecting to ResourceManager at sandbox.hortonworks.com/172.17.0.2:8050
24/12/02 18:52:30 INFO client.AHSProxy: Connecting to Application History server at sandbox.hortonworks.com/172.17.0.2:10200
24/12/02 18:52:43 INFO input.FileInputFormat: Total input paths to process : 1
24/12/02 18:52:43 INFO input.FileInputFormat: Total input paths to process : 1
24/12/02 18:52:43 INFO lzo.GPLNativeCodeLoader: Loaded native gpl library
24/12/02 18:52:43 INFO lzo.LzoCodec: Successfully loaded & initialized native-lzo library [hadoop-lzo rev 7a4b57bedce694048432dd5bf5b90a6c8ccdba80]
24/12/02 18:52:43 INFO mapreduce.JobSubmitter: number of splits:1
24/12/02 18:52:44 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1733164307838_0002
24/12/02 18:52:45 INFO impl.YarnClientImpl: Submitted application application_1733164307838_0002
24/12/02 18:52:46 INFO mapreduce.Job: The url to track the job: http://sandbox.hortonworks.com:8088/proxy/application_1733164307838_0002/
24/12/02 18:52:46 INFO mapreduce.Job: Running job: job_1733164307838_0002
24/12/02 18:53:09 INFO mapreduce.Job: Job job_1733164307838_0002 running in uber mode : false
24/12/02 18:53:09 INFO mapreduce.Job: map 0% reduce 0%
24/12/02 18:53:27 INFO mapreduce.Job: map 100% reduce 0%
24/12/02 18:53:29 INFO mapreduce.Job: Job job_1733164307838_0002 completed successfully
24/12/02 18:53:29 INFO mapreduce.Job: Counters: 30
File System Counters
FILE: Number of bytes read=0
FILE: Number of bytes written=162542
FILE: Number of read operations=0
FILE: Number of large read operations=0
FILE: Number of write operations=0
HDFS: Number of bytes read=1979371
HDFS: Number of bytes written=0
HDFS: Number of read operations=4
HDFS: Number of large read operations=0
HDFS: Number of write operations=0
Job Counters
Launched map tasks=1
Data-local map tasks=1
Total time spent by all maps in occupied slots (ms)=14211
Total time spent by all reduces in occupied slots (ms)=0
Total time spent by all map tasks (ms)=14211
Total vcore-milliseconds taken by all map tasks=14211
Watchlist
Ideas
Search
12:30 AM
03-12-2024


Classwork for DSC 502 Big
ChatGPT
root@sandbox:~ - Shell In
Hortonworks Sandbox with
Ambari - Sandbox
Big-Data/ratings.data at m
127.0.0.1:4200
Job Counters
Launched map tasks=1
Data-local map tasks=1
Total time spent by all maps in occupied slots (ms)=14211
Total time spent by all reduces in occupied slots (ms)=0
Total time spent by all map tasks (ms)=14211
Total vcore-milliseconds taken by all map tasks=14211
Total megabyte-milliseconds taken by all map tasks=3552750
Map-Reduce Framework
Map input records=100003
Map output records=100003
Input split bytes=142
Spilled Records=0
Failed Shuffles=0
Merged Map outputs=0
GC time elapsed (ms)=1352
CPU time spent (ms)=5150
Physical memory (bytes) snapshot=141717504
Virtual memory (bytes) snapshot=1938128896
Total committed heap usage (bytes)=37748736
File Input Format Counters
Bytes Read=0
File Output Format Counters
Bytes Written=0
24/12/02 18:53:29 INFO mapreduce.ExportJobBase: Transferred 1.8877 MB in 60.2378 seconds (32.0891 KB/sec)
24/12/02 18:53:29 INFO mapreduce.ExportJobBase: Exported 100003 records.
[root@sandbox ~]# mysql -u root -p hadoop
Enter password:
ERROR 1049 (42000): Unknown database 'hadoop'
[root@sandbox ~]# mysql -u root -p
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 59
Server version: 5.6.34 MySQL Community Server (GPL)
Copyright (c) 2000, 2016, Oracle and/or its affiliates. All rights reserved.
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.
25°C
Mostly cloudy
Search
12:30 AM
03-12-2024


Classwork for DSC 502 Big
ChatGPT
root@sandbox:~ - Shell In
Hortonworks Sandbox with
Ambari - Sandbox
Big-Data/ratings.data at m
127.0.0.1:4200
File Output Format Counters
Bytes Written=0
24/12/02 18:53:29 INFO mapreduce.ExportJobBase: Transferred 1.8877 MB in 60.2378 seconds (32.0891 KB/sec)
24/12/02 18:53:29 INFO mapreduce.ExportJobBase: Exported 100003 records.
[root@sandbox ~]# mysql -u root -p hadoop
Enter password:
ERROR 1049 (42000): Unknown database 'hadoop'
[root@sandbox ~]# mysql -u root -p
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 59
Server version: 5.6.34 MySQL Community Server (GPL)
Copyright (c) 2000, 2016, Oracle and/or its affiliates. All rights reserved.
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
mysql> USE bigdata_db;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A
Database changed
mysql> SHOW TABLES;
+----------------------+
| Tables_in_bigdata_db |
+----------------------+
| ratings |
+----------------------+
1 row in set (0.00 sec)
mysql> SELECT COUNT(*) FROM ratings;
+----------+
| COUNT(*) |
+----------+
| 200006 |
+----------+
25°C
Mostly cloudy
Search
12:31 AM
03-12-2024

4. Hive Queries

Objective:

To perform analytical queries on the imported data in Hive to derive insights.

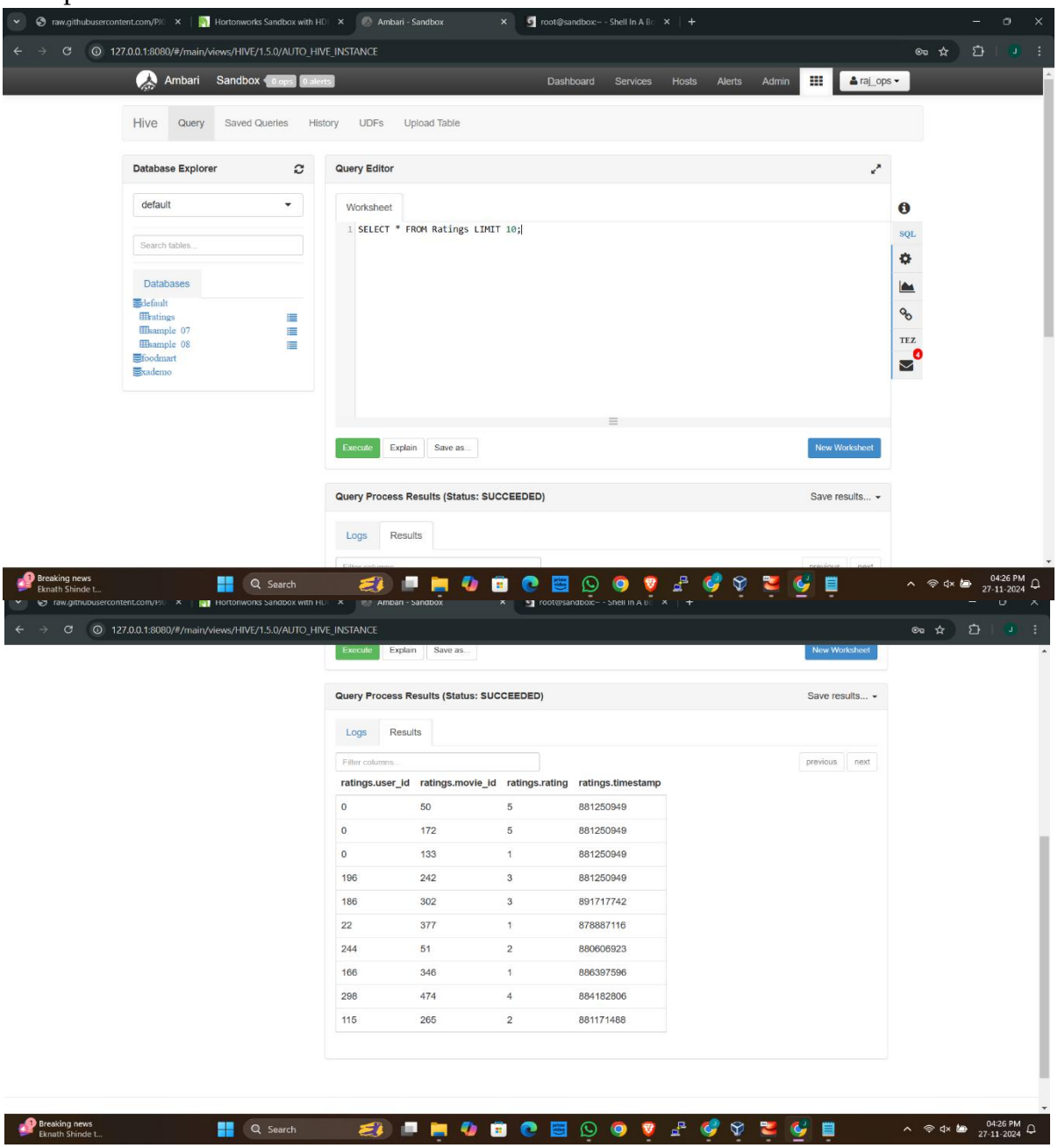
Queries:

1. **Basic Analysis:**

1.Fetch the first 10 rows to verify data accuracy.

Query Used:
SELECT * FROM ratings LIMIT 10;

Output:

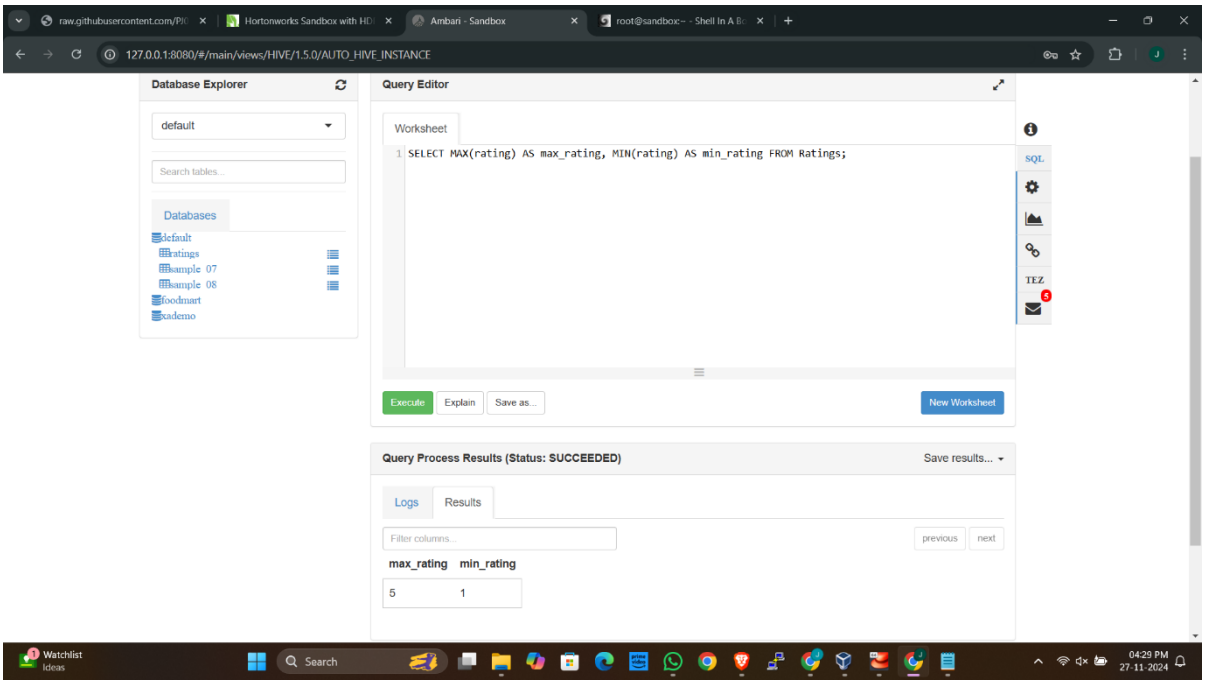


2. Determine the maximum and minimum ratings.

Query Used:

SELECT MAX(rating) AS max_rating, MIN(rating) AS min_rating FROM ratings;

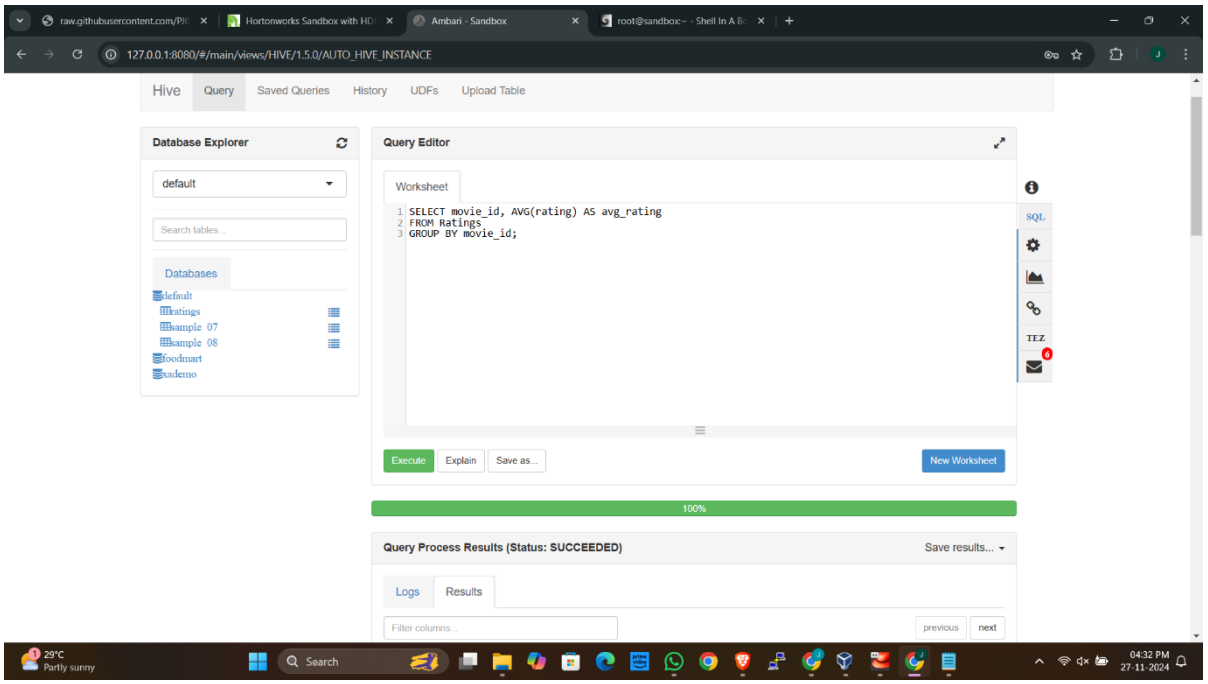
Output:

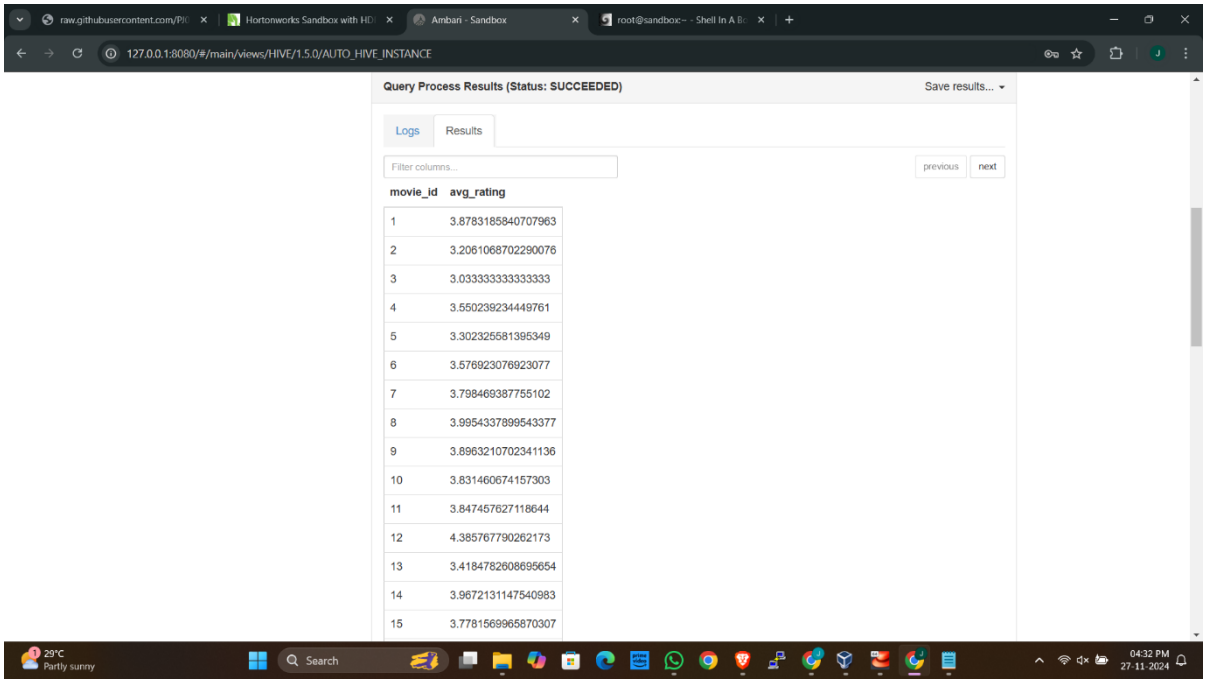


3. Calculate the average rating for each movie.

Query Used:
SELECT movie_id, AVG(rating) AS avg_rating
FROM ratings
GROUP BY movie_id;

Output:

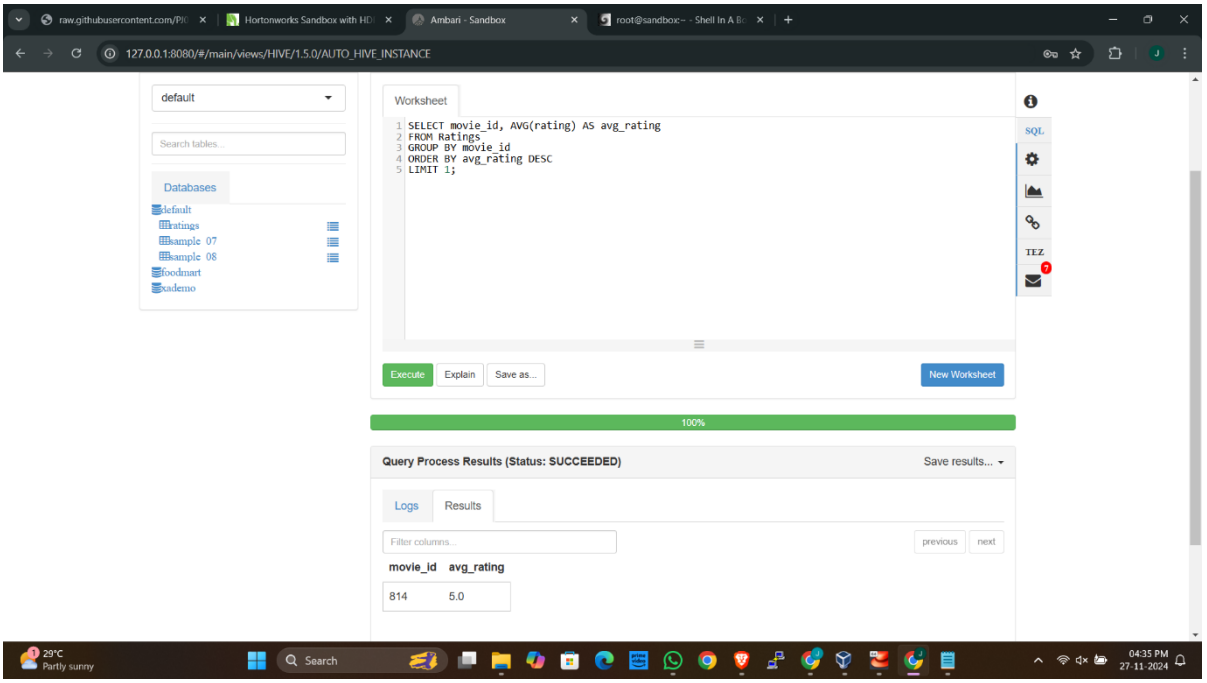




4. Identify the highest-rated movie based on average ratings.

Query Used:
SELECT movie_id, AVG(rating) AS avg_rating
FROM ratings
GROUP BY movie_id
ORDER BY avg_rating DESC
LIMIT 1;

Output:

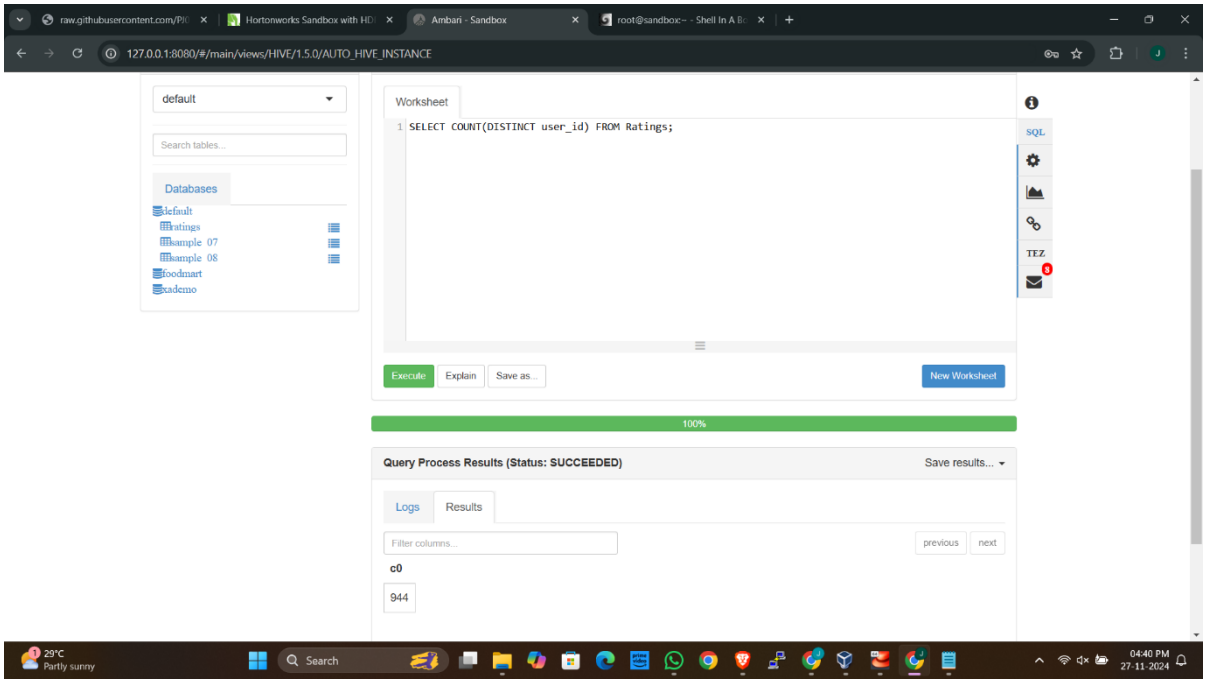


5. Identify the highest-rated movie based on average ratings.

Query Used:

```
SELECT movie_id, AVG(rating) AS avg_rating
FROM ratings
GROUP BY movie_id
ORDER BY avg_rating DESC
LIMIT 1;
```

Output:



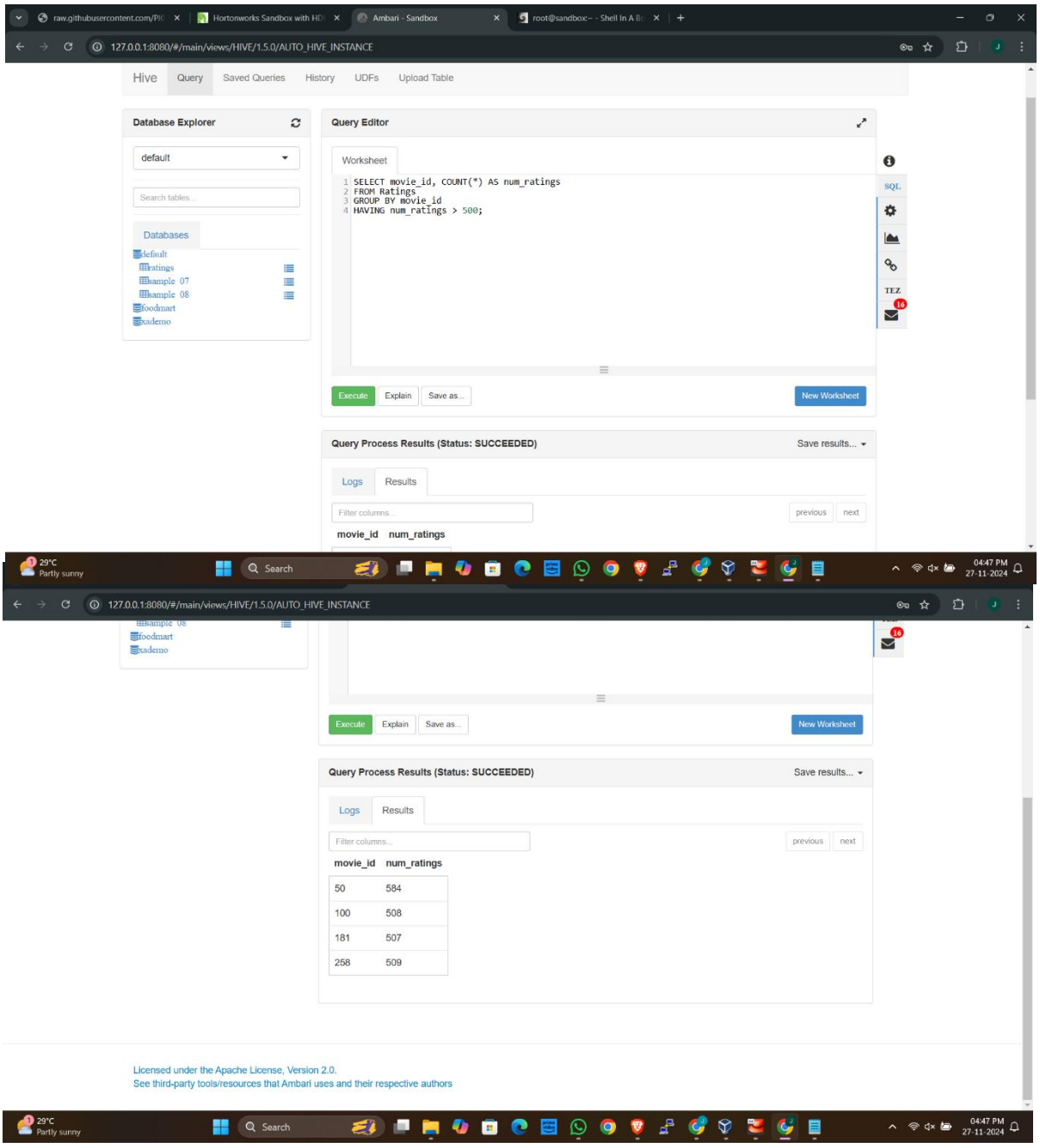
2. Complex Queries

1. Find movies with more than 500 ratings.

Query Used:

```
SELECT movie_id, COUNT(*) AS num_ratings
FROM ratings
GROUP BY movie_id
HAVING num_ratings > 500;
```

Output:

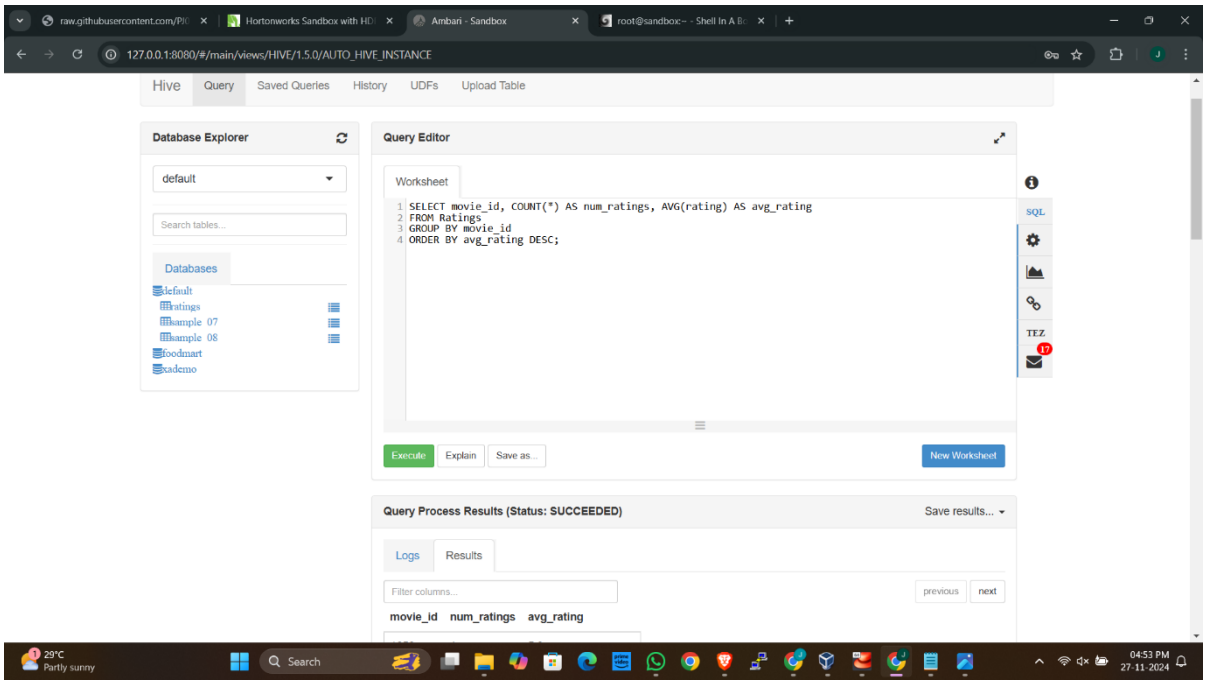


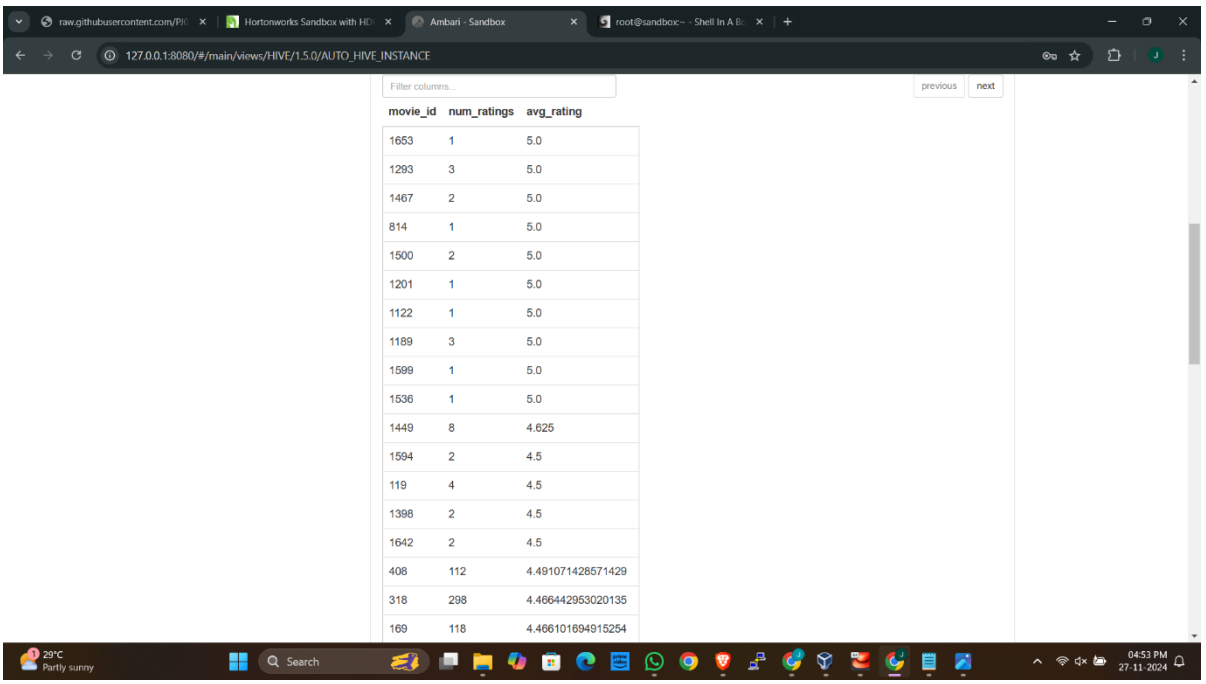
2. Get the average rating for each movie, along with the number of ratings:

Query Used:

```
SELECT movie_id, COUNT(*) AS num_ratings, AVG(rating) AS avg_rating
FROM Ratings
GROUP BY movie_id
ORDER BY avg_rating DESC;
```

Output:



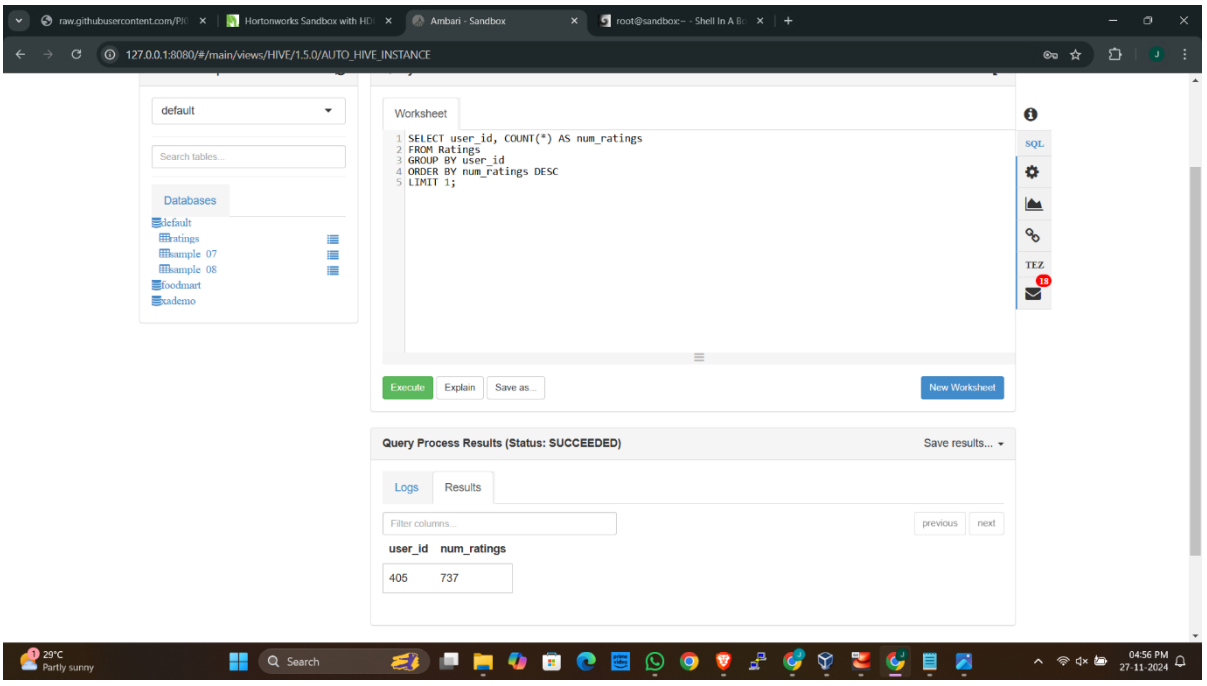


3. Find the user with the most ratings

Query Used:

```
SELECT user_id, COUNT(*) AS num_ratings
FROM Ratings
GROUP BY user_id
ORDER BY num_ratings DESC
LIMIT 1;
```

Output:

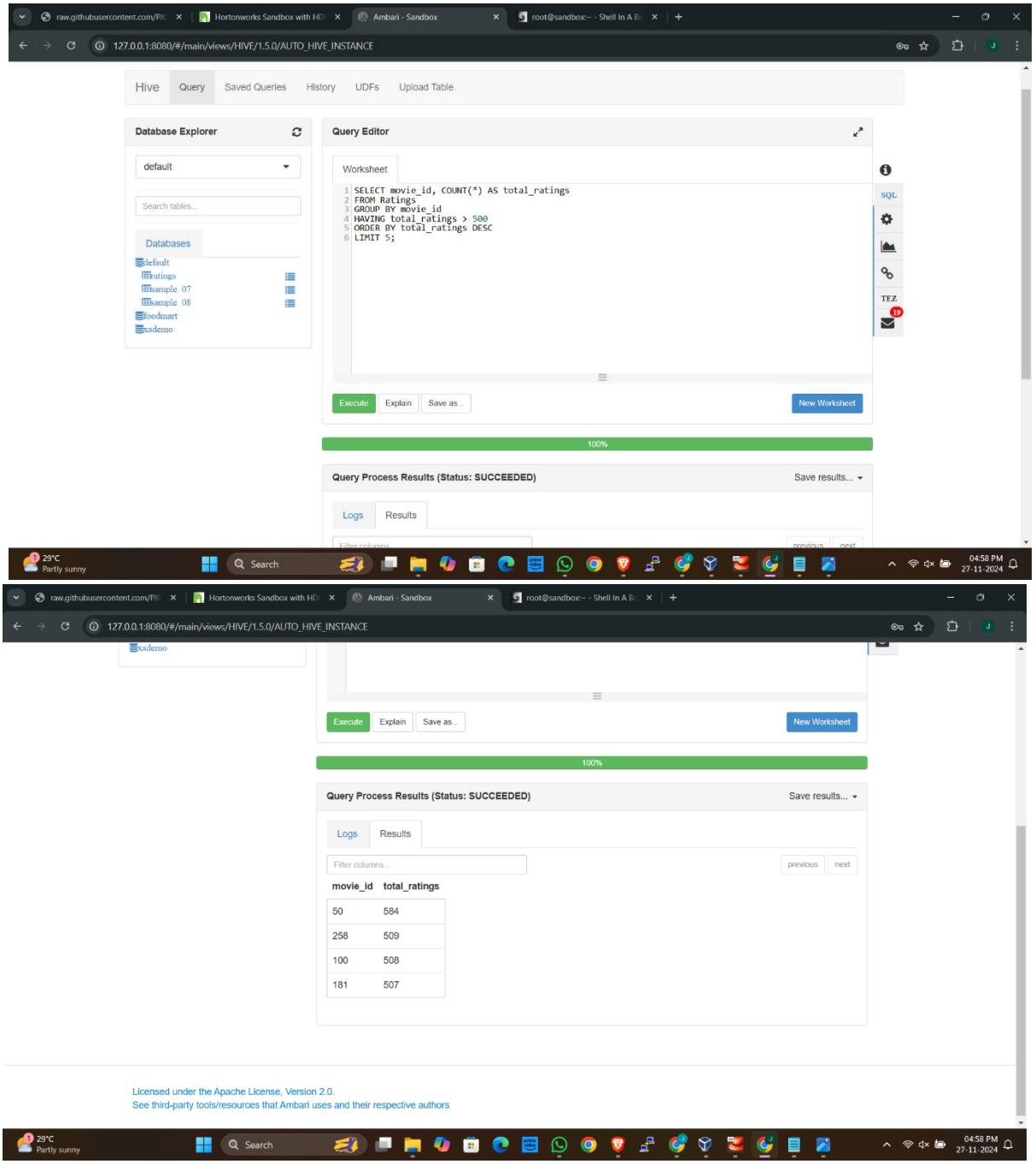


4. Find the most rated movies (with more than 500 ratings)

Query Used:

```
SELECT movie_id, COUNT(*) AS total_ratings
FROM Ratings
GROUP BY movie_id
HAVING total_ratings > 500
ORDER BY total_ratings DESC
LIMIT 5;
```

Output:

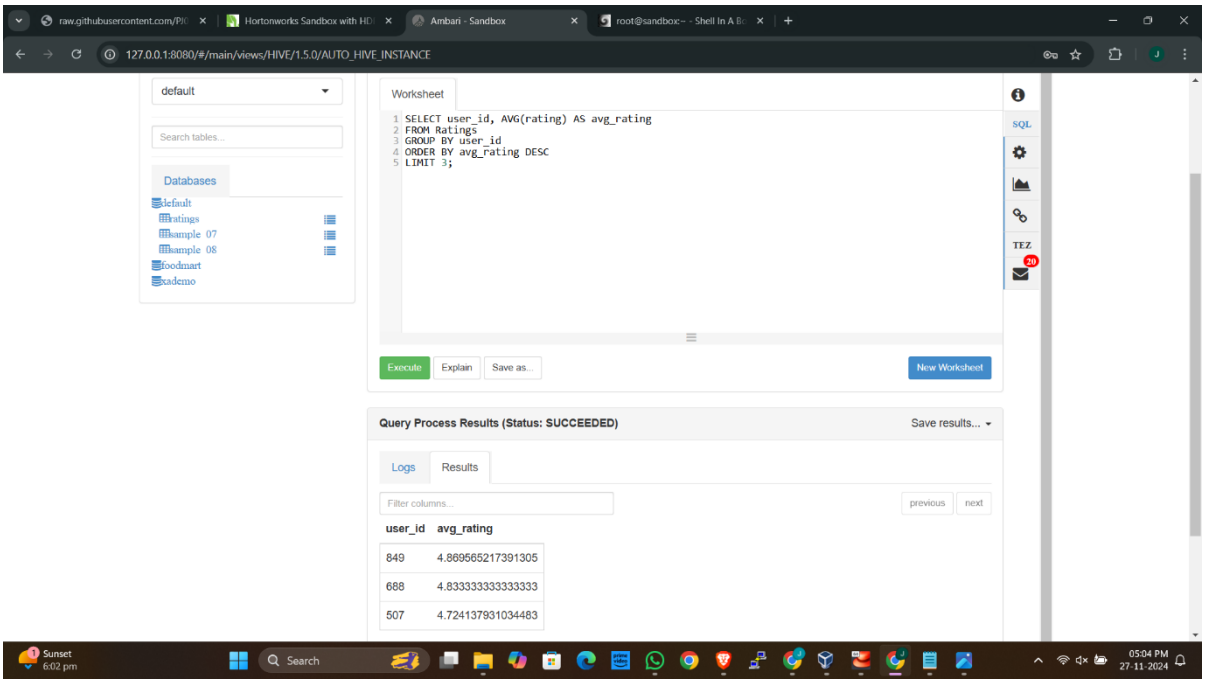


5. Get the top 3 users who have given the highest average ratings

Query Used:

```
SELECT user_id, AVG(rating) AS avg_rating
FROM Ratings
GROUP BY user_id
ORDER BY avg_rating DESC
LIMIT 3;
```

Output:

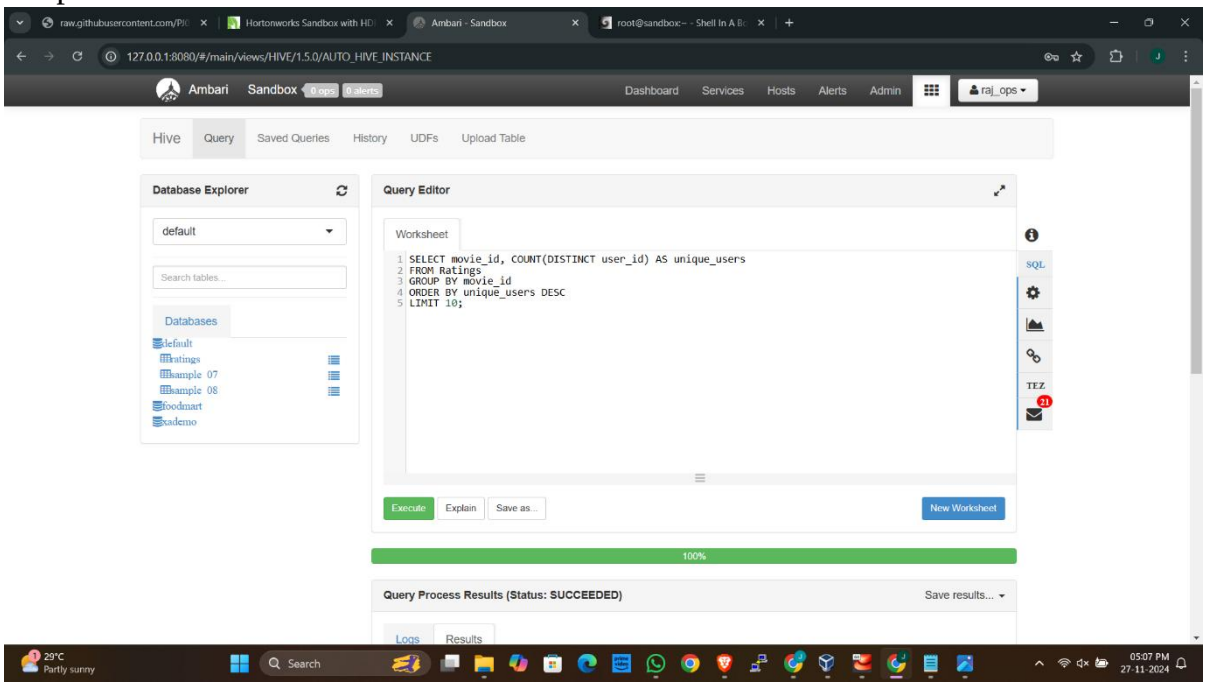


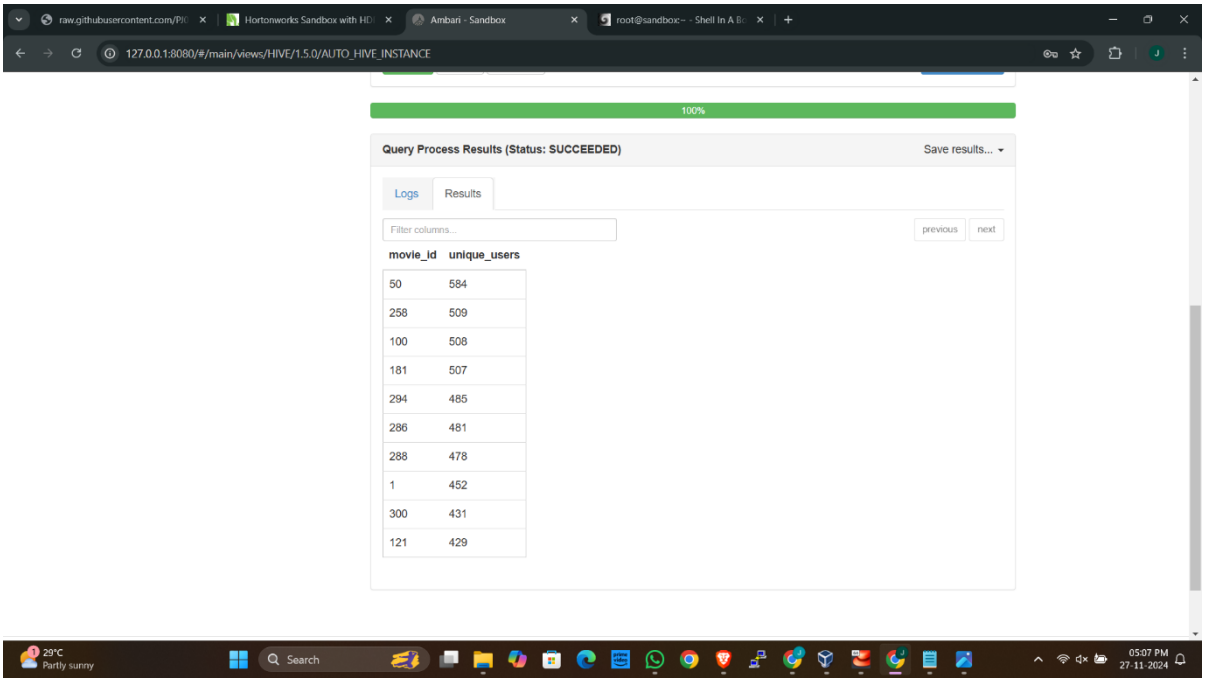
6. Find movies that have ratings from the most number of distinct users

Query Used:

```
SELECT movie_id, COUNT(DISTINCT user_id) AS unique_users
FROM Ratings
GROUP BY movie_id
ORDER BY unique_users DESC
LIMIT 10;
```

Output:



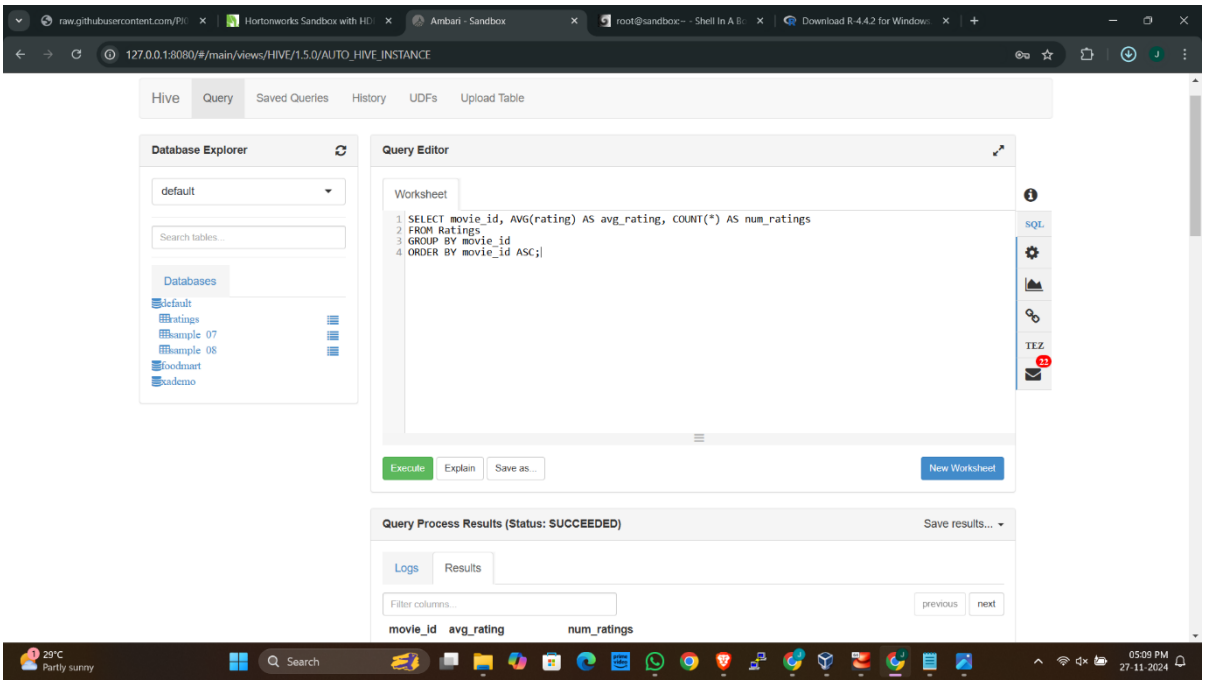


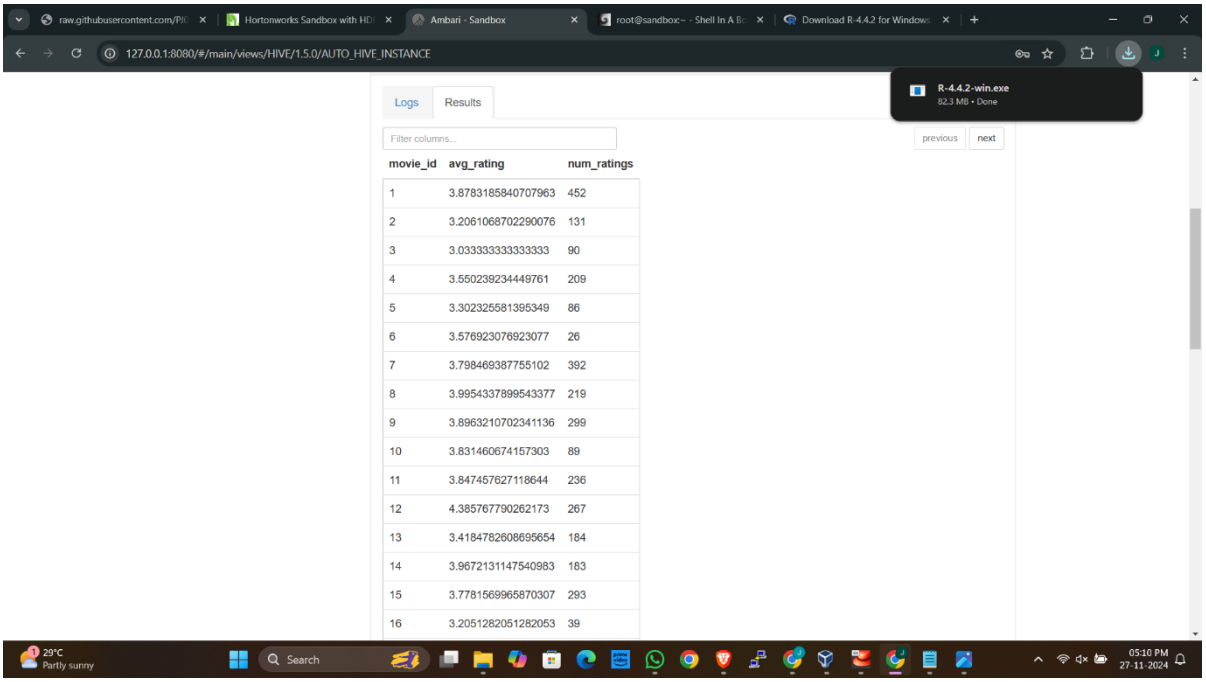
7. Get the average rating for each movie, with user count, and order by movie ID

Query Used:

```
SELECT movie_id, AVG(rating) AS avg_rating, COUNT(*) AS num_ratings
FROM Ratings
GROUP BY movie_id
ORDER BY movie_id ASC;
```

Output:





❖ Conclusion

This project provides a comprehensive end-to-end demonstration of how different tools within the Hadoop ecosystem can work together to manage, store, and analyze large datasets efficiently. By employing Hadoop Distributed File System (HDFS), MySQL, Apache Sqoop, and Apache Hive, it highlights the powerful capabilities of big data technologies in solving complex data challenges, especially in the context of user ratings for movies.

This implementation serves as an excellent demonstration of how the tools in the Hadoop ecosystem can be leveraged in tandem to manage, analyze, and derive insights from large-scale datasets. The ability to seamlessly transfer data across platforms, from GitHub to HDFS, then into MySQL and finally into Hive, underscores the flexibility and scalability of the Hadoop ecosystem.

Key Takeaways:

1. Data Movement and Integration: The use of tools like Sqoop facilitates smooth data movement across different platforms (relational to big data), ensuring that data from various sources can be integrated efficiently and stored in the most suitable environment for processing.
2. Scalable Storage Solutions: By using HDFS for scalable storage and MySQL for structured data management, the project highlights how both systems complement each other. While MySQL handles transactional and structured data, HDFS offers fault tolerance and scalability for larger datasets.
3. Insightful Data Analysis: Hive's SQL-like interface allows for efficient querying of large datasets, enabling data exploration and analysis without the need for complex coding. Its ability to handle petabytes of data across distributed systems makes it an ideal tool for big data analysis.
4. Practical Application: The ability to generate insights from large datasets, such as identifying trends, the highest-rated movies, and active users, demonstrates the practical value of big data tools in real-world applications such as movie recommendations, user profiling, and content popularity analysis.

Final Conclusion:

This project effectively showcases the potential of big data tools to solve real-world data challenges. It not only highlights the technical capabilities of each component (HDFS, MySQL, Sqoop, Hive) but also illustrates the collaborative power of the Hadoop ecosystem. From data ingestion to analysis, the project demonstrates how organizations can leverage big data technologies to handle vast amounts of information, streamline operations, and generate actionable insights for better decision-making. The approach is flexible, scalable, and efficient, making it a robust solution for handling large-scale data analysis in today's data-driven world.

❖ **References**

https://www.proquest.com/openview/dcb49cc3706d2057b05f1285783cd77d/1?pq-origsite=gscholar&cbl=2035897

https://www.proquest.com/openview/dcb49cc3706d2057b05f1285783cd77d/1?pq-origsite=gscholar&cbl=2035897

https://books.google.co.in/books?hl=en&lr=&id=ufjJDAAAQBAJ&oi=fnd&pg=PP1&dq=papers+on+hdfs+sqoop+apache+hive&ots=WkwpkyGrBT&sig=KOmT_a2oB6BOXDxsbfEwQc6_Hhc&redir_esc=y#v=onepage&q=papers%20on%20hdfs%20sqoop%20apache%20hive&f=false

https://dl.acm.org/doi/abs/10.1145/3299869.3314045

https://ieeexplore.ieee.org/abstract/document/7453400

https://books.google.co.in/books?hl=en&lr=&id=IMNiDwAAQBAJ&oi=fnd&pg=PP1&dq=papers+on+apache+hive&ots=BHRdelfJh4&sig=dbYZ2yJxF9h3d2V98oMHDN09DIk&redir_esc=y#v=onepage&q=papers%20on%20apache%20hive&f=false