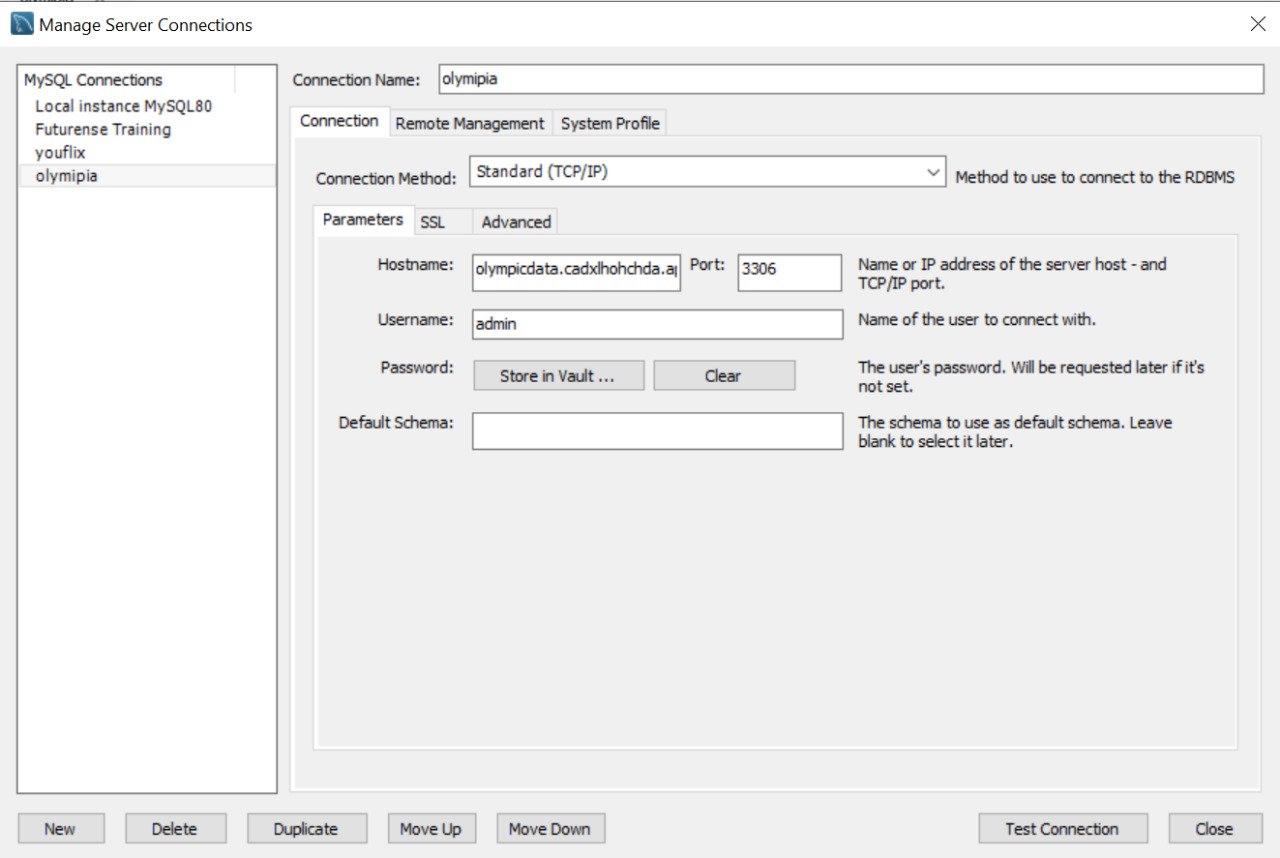
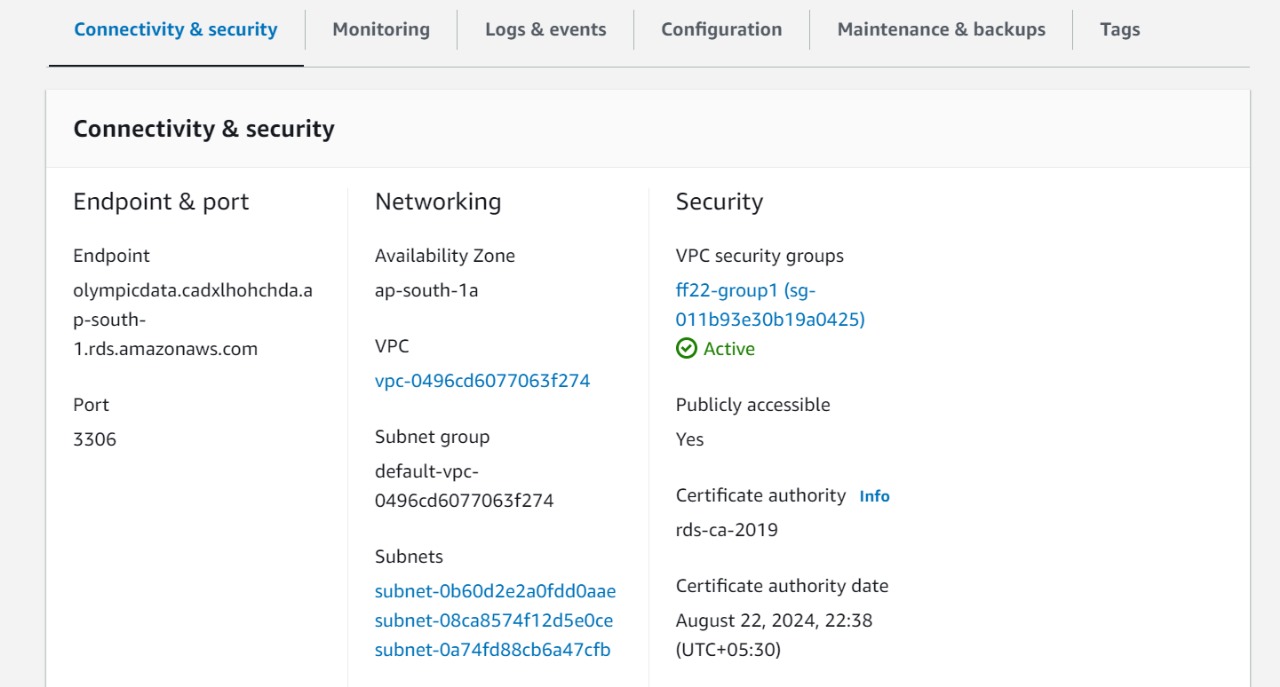
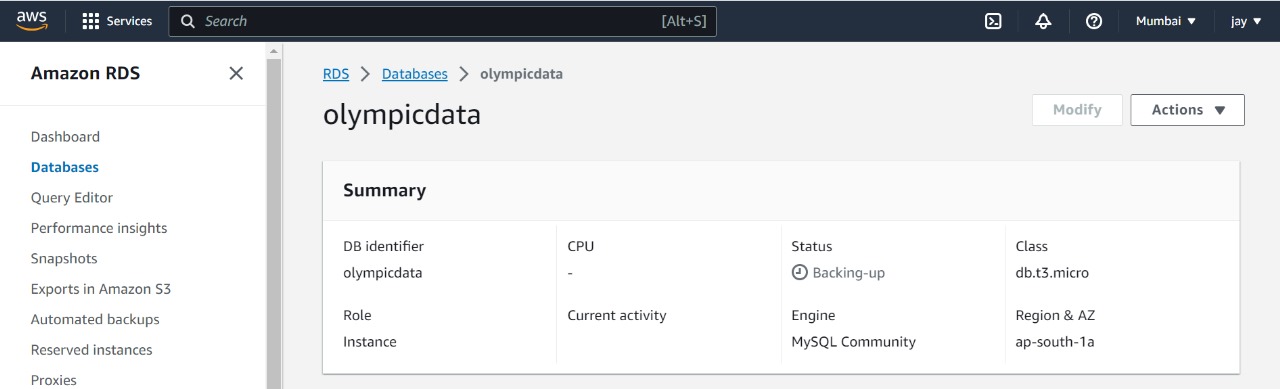
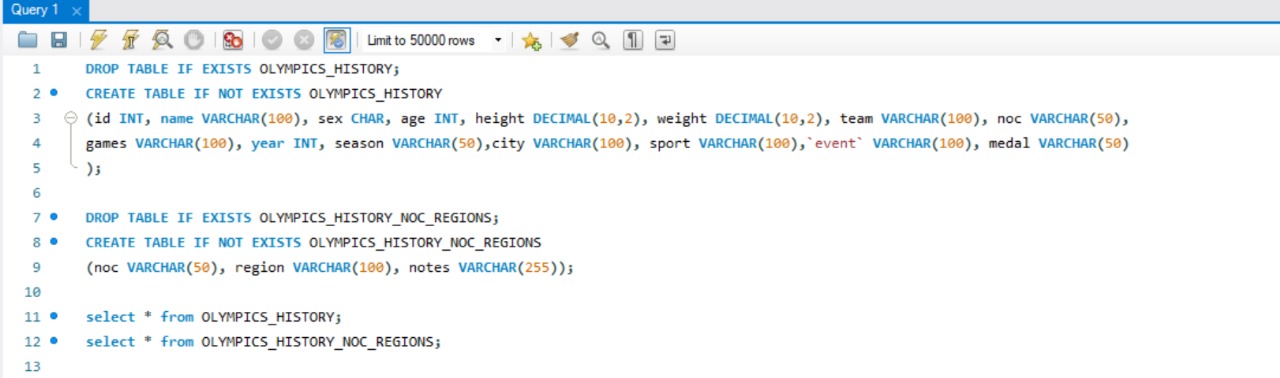
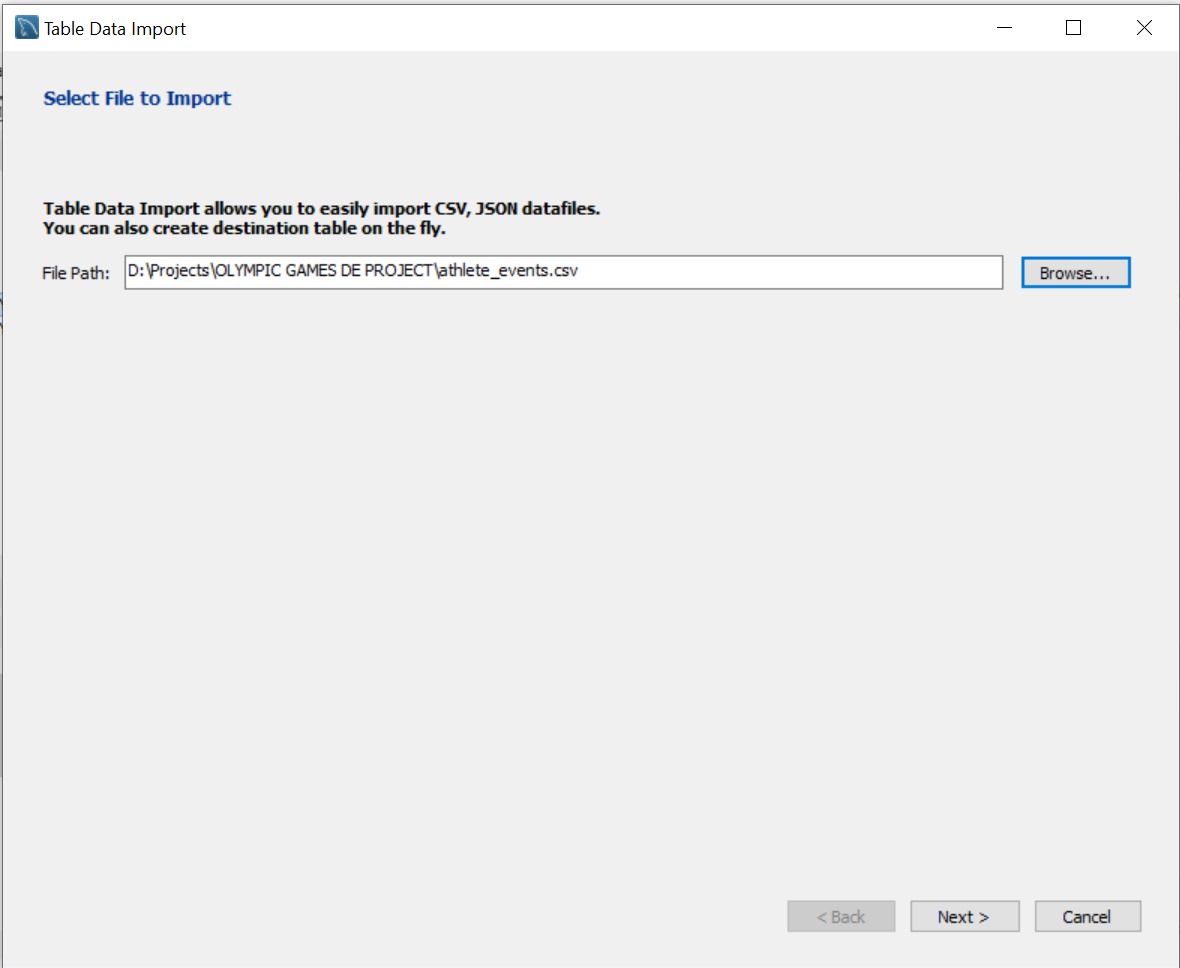
Step 1:- Creating Extraction pipelines from multiple client data sources to our data warehouse.

Pipeline 1- Amazon RDS or client relational database to Hive/HBase.

Ingestion Tool used: Sqoop





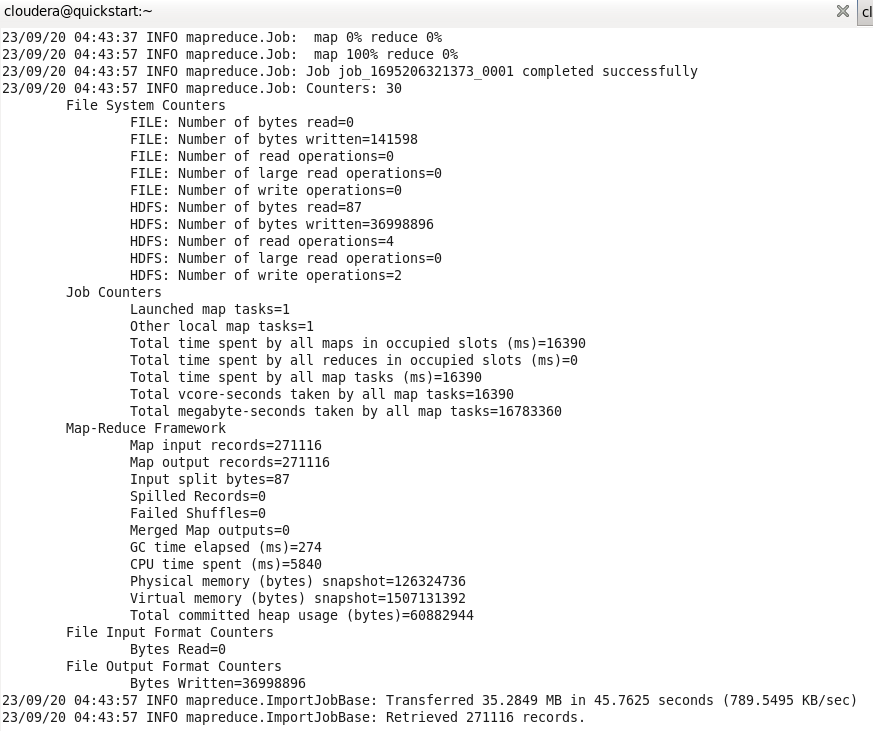




sqoop import-all-tables --connect jdbc:mysql://olympicdata.cadxlhohchda.ap-south-1.rds.amazonaws.com:3306/olympicdata --username admin --password olympia1 --hive-import --fields-terminated-by ',' -m 1;

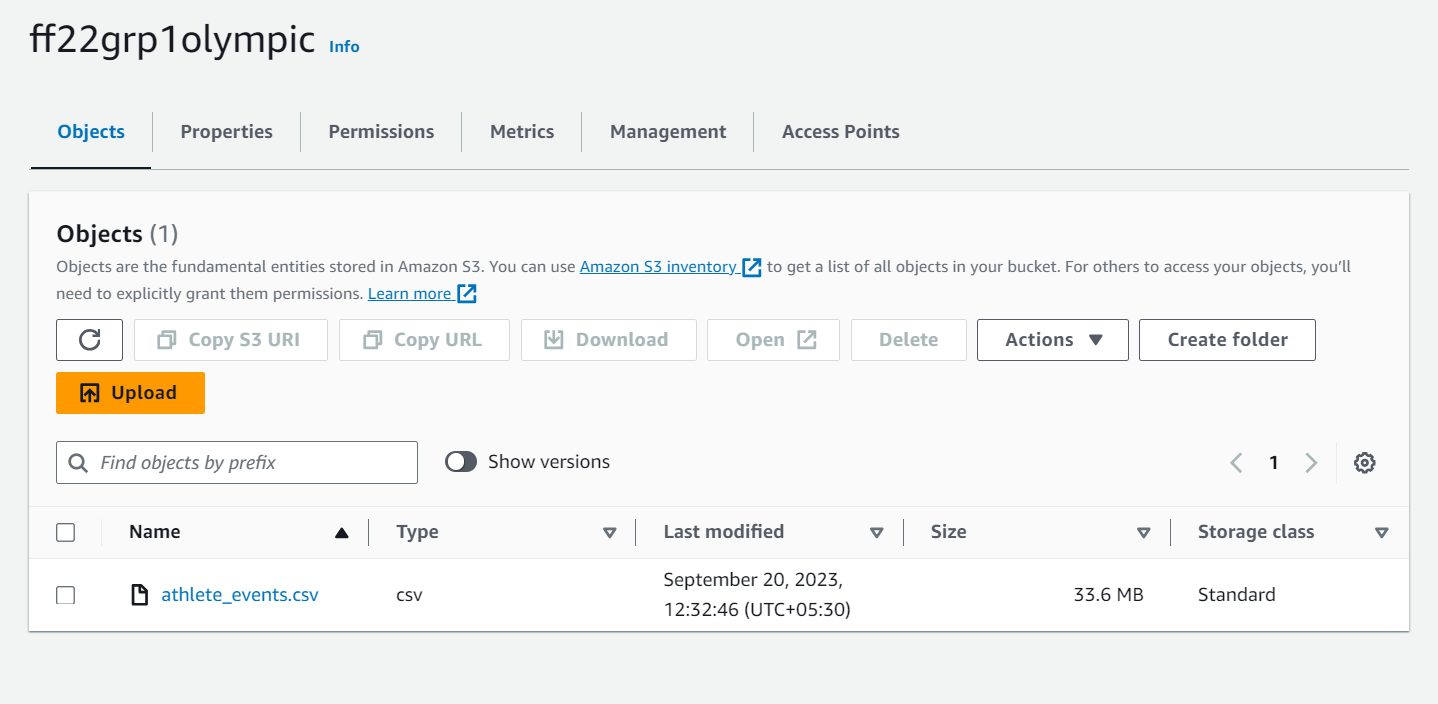


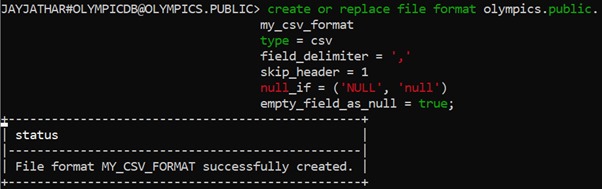
sqoop import-all-tables --connect jdbc:mysql://localhost:3306/olympicdb --username root --password cloudera –hive-import –fields-terminated-by ‘,’ -m 1;

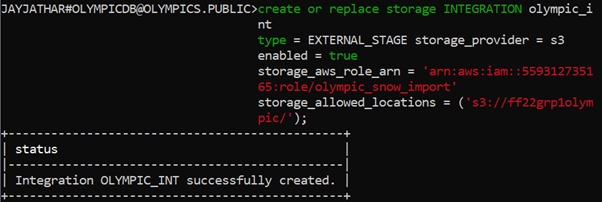


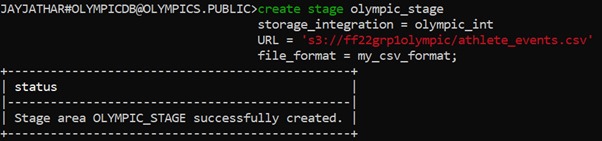
Pipeline 2- Amazon S3 Bucket to SnowFlake data warehouse.

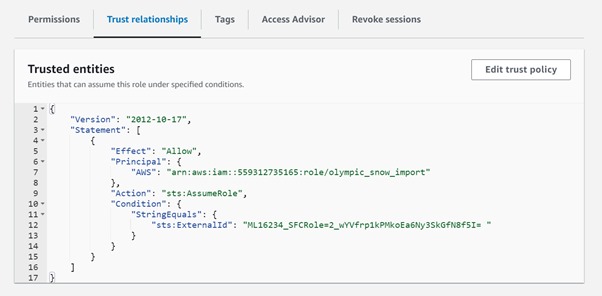
Ingestion Tool used: Copy into

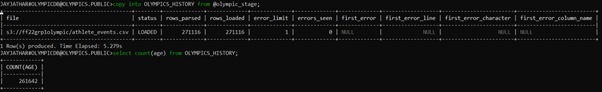






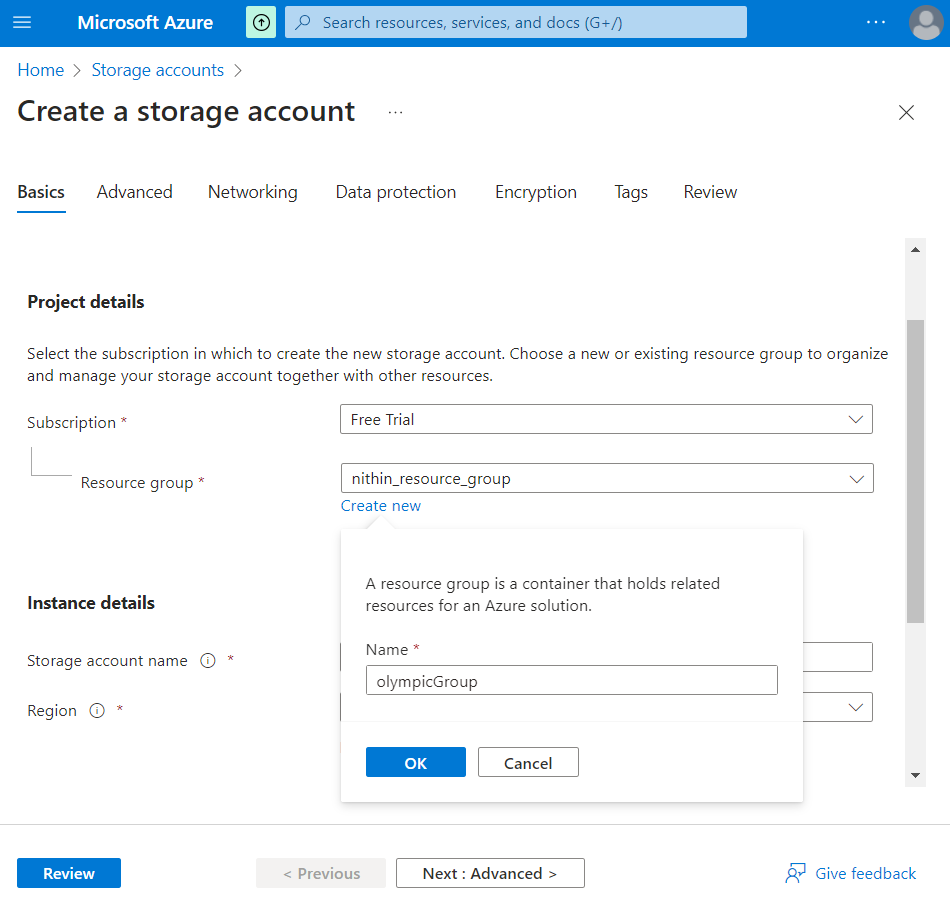


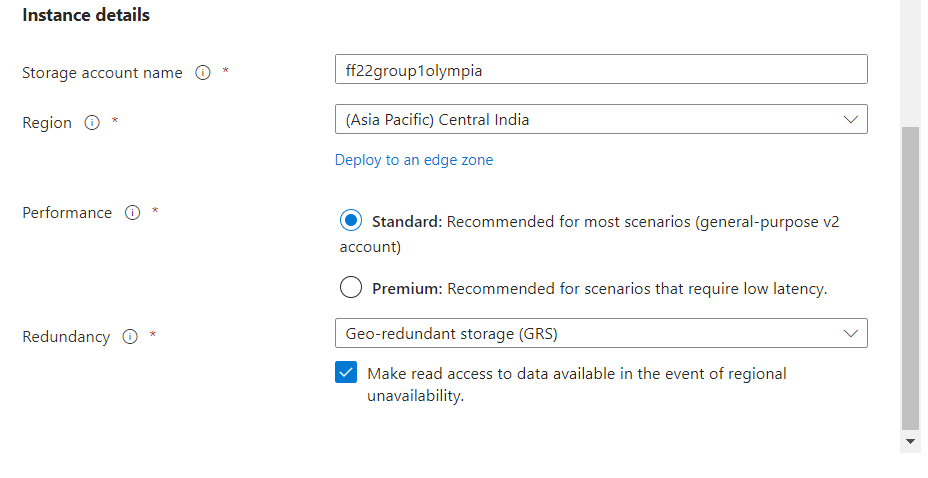


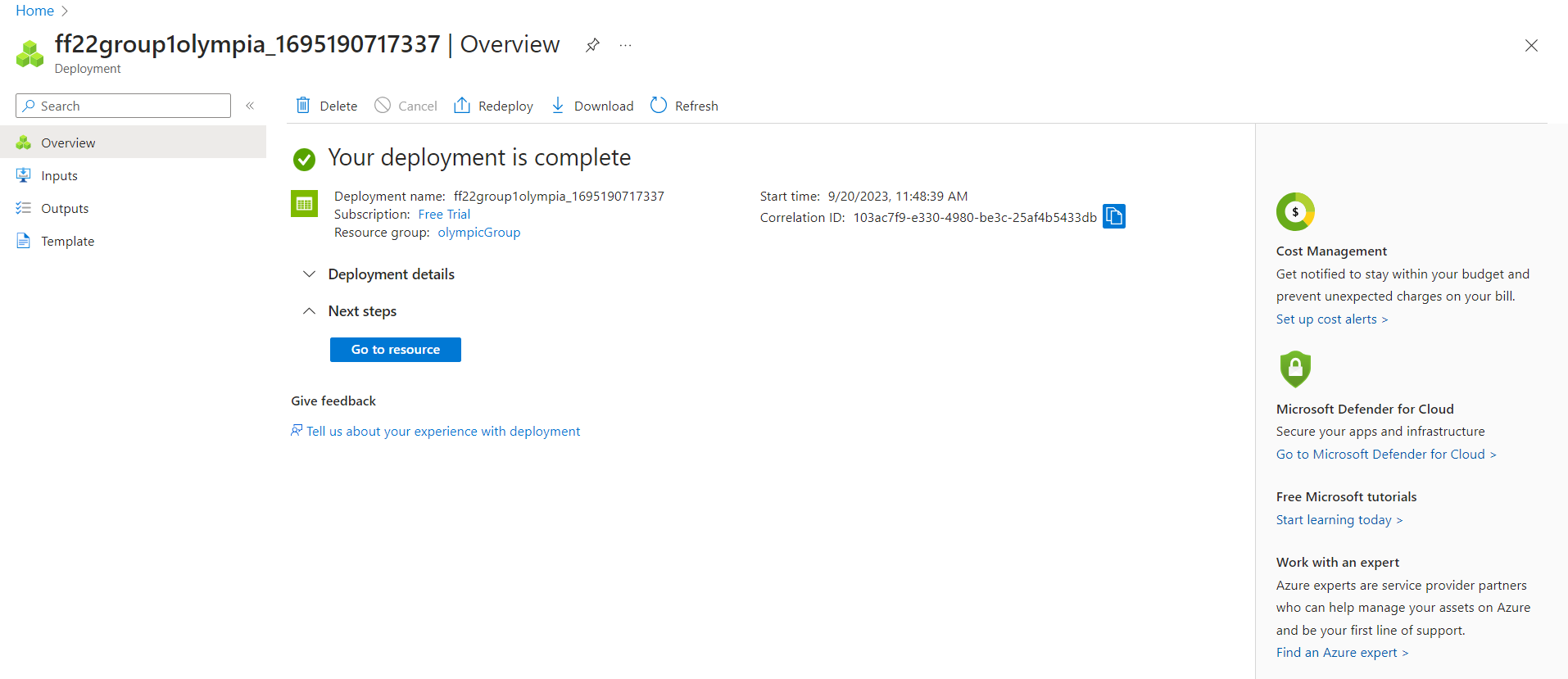


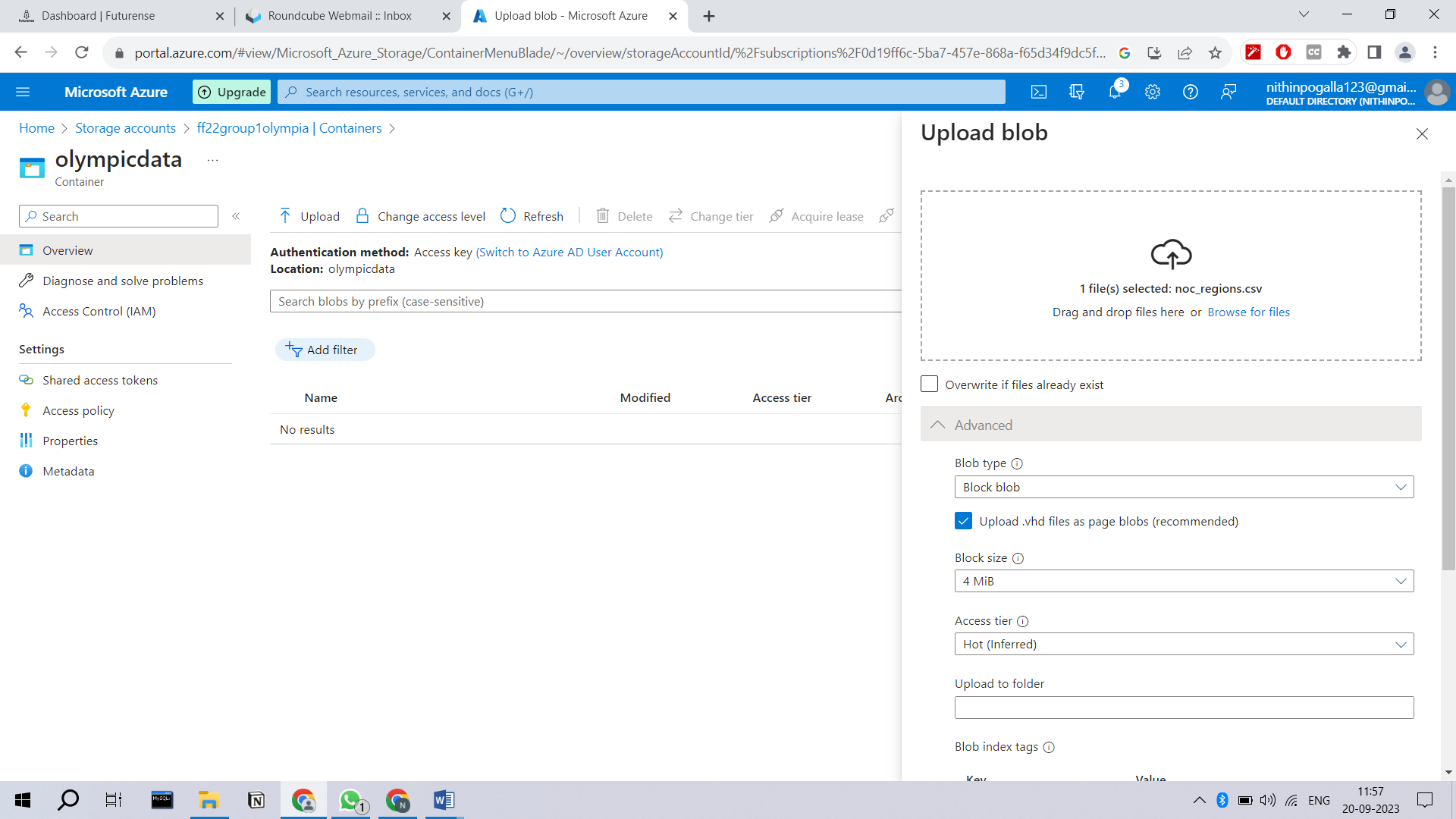
Pipeline 3- Azure Blob Container to SnowFlake data warehouse.

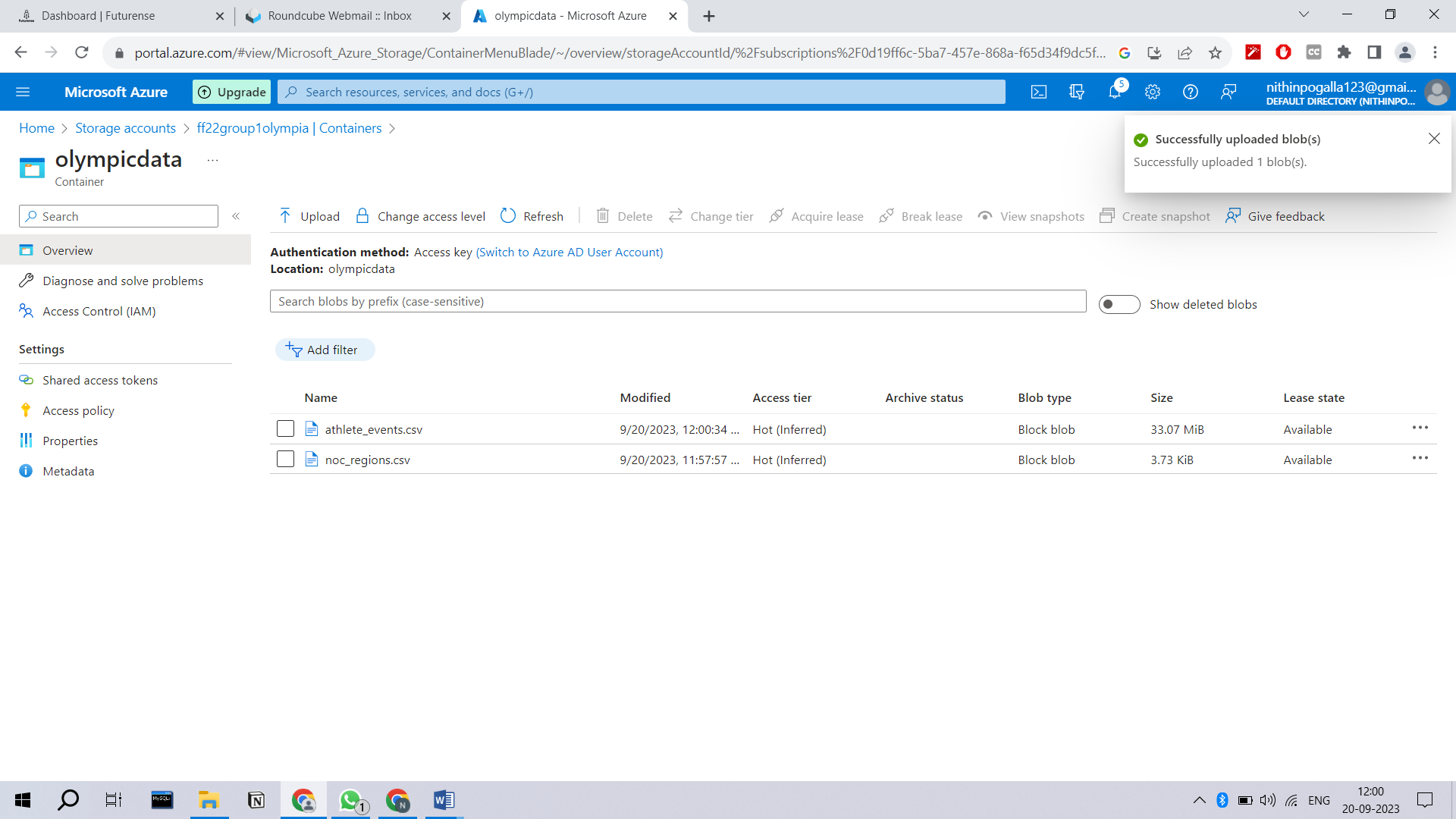
Ingestion Tool used: Snow Pipe

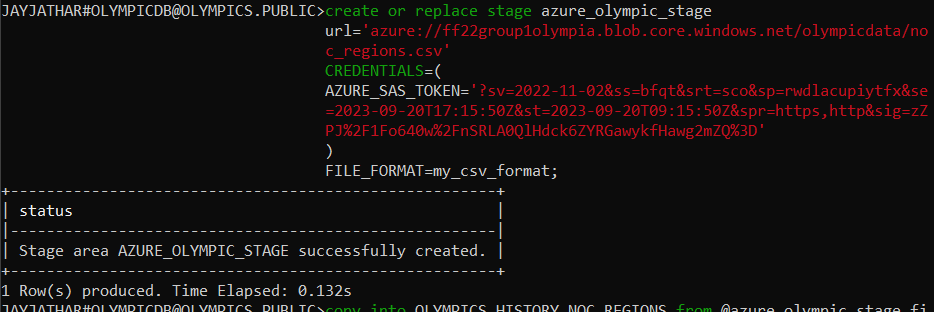


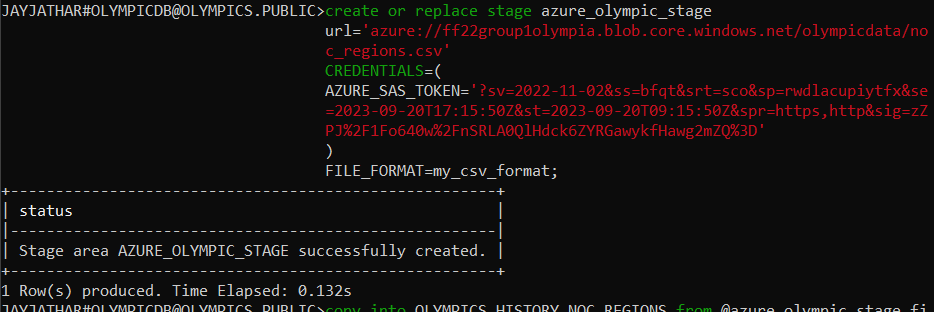












Step 2:- Transformation or Analysis of data in HIVE, HBASE, SNOWFLAKE according to the client requirements.

Analysis 1 – Snowfake

-- 1. How many Olympics games have been held?

SELECT COUNT(DISTINCT games) AS Total\_Games FROM olympics\_history;

-- 2. List down all Olympics games held so far.

SELECT DISTINCT games FROM olympics\_history;

-- 3. Mention the total number of nations who participated in each Olympics game?

SELECT games, COUNT(DISTINCT n.region) AS Total\_Nations

FROM olympics\_history o

JOIN olympics\_history\_noc\_regions n ON o.noc = n.noc

GROUP BY games;

-- 4. Which year saw the highest and lowest number of countries participating in Olympics?

WITH cte AS (

SELECT year, COUNT(DISTINCT n.region) AS Countries

FROM olympics\_history o

JOIN olympics\_history\_noc\_regions n ON o.noc = n.noc

GROUP BY year

)

SELECT year, countries

FROM (

SELECT year, countries, DENSE\_RANK() OVER (ORDER BY countries) AS l, DENSE\_RANK() OVER (ORDER BY countries DESC) AS h

FROM cte

) WHERE l = 1 OR h = 1;

-- 5. Which nation has participated in all of the Olympic games?

WITH cte AS (

SELECT n.region AS nations, COUNT(DISTINCT games) AS game\_count

FROM olympics\_history o

JOIN olympics\_history\_noc\_regions n ON o.noc = n.noc

GROUP BY nations

)

SELECT nations FROM cte WHERE game\_count = (SELECT COUNT(DISTINCT games) FROM olympics\_history);

-- 6. Identify the sport which was played in all summer Olympics.

WITH cte AS (

SELECT sport, COUNT(DISTINCT games) AS game\_count

FROM olympics\_history

WHERE season = 'Summer'

GROUP BY sport

)

SELECT sport FROM cte WHERE game\_count = (SELECT COUNT(DISTINCT games) FROM olympics\_history WHERE season = 'Summer');

-- 7. Which sports were just played only once in the Olympics?

SELECT sport, COUNT(DISTINCT games) AS game\_count

FROM olympics\_history

GROUP BY sport

HAVING game\_count = 1;

-- 8. Fetch the total number of sports played in each Olympic game.

SELECT games, COUNT(DISTINCT sport) AS Total\_Sports FROM olympics\_history GROUP BY games;

-- 9. Fetch details of the oldest athletes to win a gold medal.

SELECT name, age, games, sport, medal

FROM olympics\_history

WHERE medal = 'Gold' AND age = (SELECT MAX(age) FROM olympics\_history WHERE medal = 'Gold');

-- 10. Find the ratio of male and female athletes participated in all Olympic games.

WITH cte AS (

SELECT DISTINCT name, sex FROM olympics\_history

)

SELECT

SUM(CASE WHEN sex = 'M' THEN 1 ELSE 0 END) / (SELECT COUNT(DISTINCT name) FROM olympics\_history) \* 100 AS male\_ratio,

SUM(CASE WHEN sex = 'F' THEN 1 ELSE 0 END) / (SELECT COUNT(DISTINCT name) FROM olympics\_history) \* 100 AS female\_ratio

FROM cte;

-- 11. Fetch the top 5 athletes who have won the most gold medals.

SELECT name, COUNT(medal) AS medalsWon

FROM olympics\_history

WHERE medal = 'Gold'

GROUP BY name

ORDER BY medalsWon DESC

LIMIT 5;

-- 12. Fetch the top 5 athletes who have won the most medals (gold/silver/bronze).

SELECT name, COUNT(medal) AS medalsWon

FROM olympics\_history

GROUP BY name

ORDER BY medalsWon DESC

LIMIT 5;

-- 13. Fetch the top 5 most successful countries in Olympics. Success is defined by the number of medals won.

SELECT nr.region, COUNT(oh.medal) AS medalsWon

FROM olympics\_history oh

JOIN olympics\_history\_noc\_regions nr ON oh.noc = nr.noc

GROUP BY nr.region

ORDER BY medalsWon DESC

LIMIT 5;

-- 14. List down the total gold, silver, and bronze medals won by each country.

WITH MedalCounts AS (

SELECT

noc,

SUM(CASE WHEN medal = 'Gold' THEN 1 ELSE 0 END) AS GoldCount,

SUM(CASE WHEN medal = 'Silver' THEN 1 ELSE 0 END) AS SilverCount,

SUM(CASE WHEN medal = 'Bronze' THEN 1 ELSE 0 END) AS BronzeCount

FROM olympics\_history

GROUP BY noc

)

SELECT

nr.region,

SUM(mc.GoldCount) AS GoldWon,

SUM(mc.SilverCount) AS SilverWon,

SUM(mc.BronzeCount) AS BronzeWon

FROM MedalCounts mc

JOIN olympics\_history\_noc\_regions nr ON mc.noc = nr.noc

GROUP BY nr.region

ORDER BY GoldWon DESC, SilverWon DESC, BronzeWon DESC;

-- 15. List down the total gold, silver, and bronze medals won by each country corresponding to each Olympic game.

SELECT n.region, games, SUM(CASE WHEN medal='Gold' THEN 1 ELSE 0 END) AS gold,

SUM(CASE WHEN medal='Silver' THEN 1 ELSE 0 END) AS silver,

SUM(CASE WHEN medal='Bronze' THEN 1 ELSE 0 END) AS bronze

FROM olympics\_history o

JOIN olympics\_history\_noc\_regions n ON o.noc = n.noc

GROUP BY n.region, games;

-- 16. Identify which country won the most gold, most silver, and most bronze medals in each Olympic game.

WITH gold\_cte AS (

SELECT noc, games, region,

SUM(CASE WHEN medal='Gold' THEN 1 ELSE 0 END) AS gold\_medals,

RANK() OVER(PARTITION BY games ORDER BY gold\_medals DESC) AS gold\_rank

FROM olympics\_history

JOIN olympics\_history\_noc\_regions USING(noc)

GROUP BY noc, games, region

),

silver\_cte AS (

SELECT noc, games, region,

SUM(CASE WHEN medal='Silver' THEN 1 ELSE 0 END) AS silver\_medals,

RANK() OVER(PARTITION BY games ORDER BY silver\_medals DESC) AS silver\_rank

FROM olympics\_history

JOIN olympics\_history\_noc\_regions USING(noc)

GROUP BY noc, games, region

),

bronze\_cte AS (

SELECT noc, games, region,

SUM(CASE WHEN medal='Bronze' THEN 1 ELSE 0 END) AS bronze\_medals,

RANK() OVER(PARTITION BY games ORDER BY bronze\_medals DESC) AS bronze\_rank

FROM olympics\_history

JOIN olympics\_history\_noc\_regions USING(noc)

GROUP BY noc, games, region

)

SELECT games, g.region AS g\_region, s.region AS s\_region, b.region AS b\_region

FROM gold\_cte g

JOIN silver\_cte s USING(games)

JOIN bronze\_cte b USING(games)

WHERE gold\_rank = 1 AND silver\_rank = 1 AND bronze\_rank = 1

ORDER BY games;

-- 17. Identify which country won the most gold, most silver, most bronze medals, and the most medals in each Olympic game.

WITH gold\_cte AS (

SELECT noc, games, region,

SUM(CASE WHEN medal='Gold' THEN 1 ELSE 0 END) AS gold\_medals,

RANK() OVER(PARTITION BY games ORDER BY gold\_medals DESC) AS gold\_rank

FROM olympics\_history

JOIN olympics\_history\_noc\_regions USING(noc)

GROUP BY noc, games, region

),

silver\_cte AS (

SELECT noc, games, region,

SUM(CASE WHEN medal='Silver' THEN 1 ELSE 0 END) AS silver\_medals,

RANK() OVER(PARTITION BY games ORDER BY silver\_medals DESC) AS silver\_rank

FROM olympics\_history

JOIN olympics\_history\_noc\_regions USING(noc)

GROUP BY noc, games, region

),

bronze\_cte AS (

SELECT noc, games, region,

SUM(CASE WHEN medal='Bronze' THEN 1 ELSE 0 END) AS bronze\_medals,

RANK() OVER(PARTITION BY games ORDER BY bronze\_medals DESC) AS bronze\_rank

FROM olympics\_history

JOIN olympics\_history\_noc\_regions USING(noc)

GROUP BY noc, games, region

),

total\_cte AS (

SELECT noc, games, region,

COUNT(medal) AS total\_medals,

RANK() OVER(PARTITION BY games ORDER BY total\_medals DESC) AS total\_rank

FROM olympics\_history

JOIN olympics\_history\_noc\_regions USING(noc)

GROUP BY noc, games, region

)

SELECT games, g.region AS g\_region, s.region AS s\_region, b.region AS b\_region, t.region AS total\_region

FROM gold\_cte g

JOIN silver\_cte s USING(games)

JOIN bronze\_cte b USING(games)

JOIN total\_cte t USING(games)

WHERE gold\_rank = 1 AND silver\_rank = 1 AND bronze\_rank = 1 AND total\_rank = 1

ORDER BY games;

-- 18. Which countries have never won a gold medal but have won silver/bronze medals?

WITH cte AS (

SELECT noc,

SUM(CASE WHEN medal='Gold' THEN 1 ELSE 0 END) AS gold\_medals,

SUM(CASE WHEN medal='Bronze' OR medal='Silver' THEN 1 ELSE 0 END) AS other\_medals

FROM olympics\_history

GROUP BY noc

)

SELECT region FROM cte

JOIN olympics\_history\_noc\_regions USING(noc)

WHERE gold\_medals = 0 AND other\_medals <> 0;

-- 19. In which Sport/event, India has won the highest number of medals.

SELECT sport, COUNT(medal) AS medals

FROM olympics\_history

JOIN olympics\_history\_noc\_regions USING(noc)

WHERE region='India'

GROUP BY sport

ORDER BY medals DESC

LIMIT 1;

SELECT event, COUNT(medal) AS medals

FROM olympics\_history

JOIN olympics\_history\_noc\_regions USING(noc)

WHERE region='India'

GROUP BY event

ORDER BY medals DESC

LIMIT 1;

-- 20. Break down all Olympic games where India won a medal for Hockey and how many medals in each Olympic game.

SELECT games, COUNT(medal) AS medals

FROM olympics\_history

JOIN olympics\_history\_noc\_regions USING(noc)

WHERE region='India' AND sport LIKE '%Hockey%'

GROUP BY games

HAVING medals <> 0

ORDER BY medals DESC;

Analysis 2 – Hive

-- 1. How many Olympics games have been held?

SELECT COUNT(DISTINCT games) AS total\_olympic\_games FROM olympics\_history;

-- 2. List down all Olympics games held so far.

SELECT DISTINCT games AS total\_olympic\_games FROM olympics\_history;

-- 3. Mention the total number of nations who participated in each Olympics game?

SELECT games, COUNT(DISTINCT noc) FROM OLYMPICS\_HISTORY GROUP BY games;

-- 4. Which year saw the highest and lowest number of countries participating in Olympics?

SELECT year, nation\_part

FROM (

SELECT year, COUNT(DISTINCT noc) AS nation\_part

FROM olympics\_history

GROUP BY year

ORDER BY nation\_part DESC

LIMIT 1

) highest\_participation

UNION ALL

SELECT year, nation\_part

FROM (

SELECT year, COUNT(DISTINCT noc) AS nation\_part

FROM olympics\_history

GROUP BY year

ORDER BY nation\_part ASC

LIMIT 1

) lowest\_participation;

-- 5. Which nation has participated in all of the Olympic games?

SELECT noc

FROM (SELECT COUNT(DISTINCT games) AS cnt\_games FROM olympics\_history) AS cte, (SELECT noc, COUNT(DISTINCT games) AS cnt

FROM olympics\_history

GROUP BY noc

ORDER BY cnt DESC) AS cte2

WHERE cnt\_games = cnt;

-- 6. Identify the sport which was played in all summer Olympics.

SELECT sport

FROM (SELECT COUNT(DISTINCT games) AS cnt\_games FROM olympics\_history WHERE season='Summer') AS cte, (SELECT sport, COUNT(DISTINCT games) AS cnt

FROM olympics\_history

WHERE season='Summer'

GROUP BY sport

ORDER BY cnt DESC) AS cte2

WHERE cnt\_games = cnt;

-- 7. Which Sports were just played only once in the Olympics?

SELECT sport

FROM (

SELECT sport, COUNT(DISTINCT games) AS total\_games

FROM OLYMPICS\_HISTORY

GROUP BY sport

HAVING total\_games = 1

) AS tt;

-- 8. Fetch the total number of sports played in each Olympic game.

SELECT games, COUNT(DISTINCT sport) AS total\_sports\_played

FROM olympics\_history\_optimized

GROUP BY games;

-- 9. Fetch details of the oldest athletes to win a gold medal.

SELECT name, age, games, sport, event

FROM olympics\_history\_optimized

WHERE medal = 'Gold'

ORDER BY age ASC

LIMIT 1;

-- 10. Find the Ratio of male and female athletes participated in all Olympic games.

WITH athlete\_counts AS (

SELECT games, sex, COUNT(DISTINCT name) AS athlete\_count

FROM olympics\_history\_optimized

GROUP BY games, sex

),

total\_athletes AS (

SELECT games, SUM(athlete\_count) AS total\_count

FROM athlete\_counts

GROUP BY games

)

SELECT t.games,

m.athlete\_count AS male\_athletes,

f.athlete\_count AS female\_athletes,

ROUND(m.athlete\_count / f.athlete\_count, 2) AS gender\_ratio

FROM total\_athletes t

JOIN athlete\_counts m ON t.games = m.games AND m.sex = 'M'

JOIN athlete\_counts f ON t.games = f.games AND f.sex = 'F';

Partitioning table for optimization

CREATE TABLE olympics\_history\_optimized

(

id INT,

name STRING,

sex STRING,

age INT,

height STRING,

weight STRING,

team STRING,

noc STRING,

games STRING,

year INT,

season STRING,

city STRING,

sport STRING,

event STRING

)

PARTITIONED BY (medal STRING)

CLUSTERED BY (noc)

INTO 5 BUCKETS row format delimited fields terminated by ',' lines terminated by '\n' stored as orc;

INSERT OVERWRITE TABLE olympics\_history\_optimized PARTITION(medal) SELECT id, name, sex, age, height, weight, team, noc, games, year, season, city, sport, event, medal FROM OLYMPICS\_HISTORY;

-- 11. Fetch the top 5 athletes who have won the most gold medals.

SELECT name, COUNT(\*) AS gold\_medals\_count

FROM olympics\_history\_optimized

WHERE medal = 'Gold'

GROUP BY name

ORDER BY gold\_medals\_count DESC

LIMIT 5;

-- 12. Fetch the top 5 athletes who have won the most medals (gold/silver/bronze).

SELECT name, COUNT(medal) AS medalsWon

FROM olympics\_history\_optimized

GROUP BY name

ORDER BY medalsWon DESC

LIMIT 5;

-- 13. Fetch the top 5 most successful countries in Olympics. Success is defined by the number of medals won.

SELECT nr.region, COUNT(oh.medal) AS medalsWon

FROM olympics\_history\_optimized oh

JOIN olympics\_history\_noc\_regions nr ON oh.noc = nr.noc

GROUP BY nr.region

ORDER BY medalsWon DESC

LIMIT 5;

-- 14. List down the total number of gold, silver, and bronze medals won by each country.

SELECT nr.region,

SUM(mc.GoldCount) AS GoldWon,

SUM(mc.SilverCount) AS SilverWon,

SUM(mc.BronzeCount) AS BronzeWon

FROM (

SELECT noc,

SUM(CASE WHEN medal = 'Gold' THEN 1 ELSE 0 END) AS GoldCount,

SUM(CASE WHEN medal = 'Silver' THEN 1 ELSE 0 END) AS SilverCount,

SUM(CASE WHEN medal = 'Bronze' THEN 1 ELSE 0 END) AS BronzeCount

FROM olympics\_history\_optimized

GROUP BY noc

) mc

JOIN olympics\_history\_noc\_regions nr ON mc.noc = nr.noc

GROUP BY nr.region

ORDER BY GoldWon DESC, SilverWon DESC, BronzeWon DESC;

-- 15. List down the total number of gold, silver, and bronze medals won by each country corresponding to each Olympic game.

SELECT games, n.region,

SUM(CASE WHEN medal = 'Gold' THEN 1 ELSE 0 END) AS gold,

SUM(CASE WHEN medal = 'Silver' THEN 1 ELSE 0 END) AS silver,

SUM(CASE WHEN medal = 'Bronze' THEN 1 ELSE 0 END) AS bronze

FROM olympics\_history\_optimized o

JOIN olympics\_history\_noc\_regions n ON o.noc = n.noc

GROUP BY games, n.region;

-- 16. Identify which country won the most gold, most silver, and most bronze medals in each Olympic game.

SELECT games,

MAX(CASE WHEN medal = 'Gold' THEN region END) AS most\_gold\_country,

MAX(CASE WHEN medal = 'Silver' THEN region END) AS most\_silver\_country,

MAX(CASE WHEN medal = 'Bronze' THEN region END) AS most\_bronze\_country

FROM (

SELECT games, region, medal,

RANK() OVER (PARTITION BY games, medal ORDER BY COUNT(\*) DESC) AS r

FROM olympics\_history\_optimized oh

JOIN olympics\_history\_noc\_regions nr ON oh.noc = nr.noc

GROUP BY games, region, medal

) ranked

WHERE r = 1

GROUP BY games;

-- 17. Identify which country won the most gold, most silver, most bronze medals, and the most medals in each Olympic game.

WITH medal\_counts AS (

SELECT games, region, medal, COUNT(\*) AS medal\_count

FROM olympics\_history\_optimized oh

JOIN olympics\_history\_noc\_regions nr ON oh.noc = nr.noc

GROUP BY games, region, medal

),

ranked\_medals AS (

SELECT games, region, medal,

RANK() OVER (PARTITION BY games, medal ORDER BY medal\_count DESC) AS r

FROM medal\_counts

)

SELECT games,

MAX(CASE WHEN medal = 'Gold' THEN region END) AS most\_gold\_country,

MAX(CASE WHEN medal = 'Silver' THEN region END) AS most\_silver\_country,

MAX(CASE WHEN medal = 'Bronze' THEN region END) AS most\_bronze\_country,

MAX(CASE WHEN r = 1 THEN region END) AS most\_medal\_country

FROM ranked\_medals

GROUP BY games;

Analysis 3 – Hbase

First, in HBase shell, create 'olympics\_history', 'person\_details', 'games\_details'.

Then, in Hive shell, create an external table olympics\_history\_integrated and import data from HBase.

In HBase shell:

Create 'olympics\_history', 'person\_details', 'games\_details' tables.

In Hive shell:

CREATE EXTERNAL TABLE olympics\_history\_integrated

(

num INT,

id INT,

name STRING,

sex STRING,

age INT,

height INT,

weight INT,

team STRING,

noc STRING,

games STRING,

year INT,

season STRING,

city STRING,

sport STRING,

event STRING,

medal STRING

)

ROW FORMAT DELIMITED FIELDS TERMINATED BY ','

STORED BY 'org.apache.hadoop.hive.hbase.HBaseStorageHandler'

WITH SERDEPROPERTIES (

"hbase.columns.mapping" = ":key,person\_details:id,person\_details:name,person\_details:sex,person\_details:age,person\_details:height,person\_details:weight,person\_details:team,person\_details:noc,games\_details:games,games\_details:year,games\_details:season,games\_details:city,games\_details:sport,games\_details:event,games\_details:medal"

)

TBLPROPERTIES (

"hbase.table.name" = "olympics\_history"

);

Import data from HBase into the external table.

Make sure the HBase table has the necessary data populated. Now you can run queries on the olympics\_history\_integrated table in Hive.

-- 18. Which countries have never won a gold medal but have won silver/bronze medals?

WITH cte AS (

SELECT noc,

SUM(CASE WHEN medal='Gold' THEN 1 ELSE 0 END) AS gold\_medals,

SUM(CASE WHEN medal='Bronze' OR medal='Silver' THEN 1 ELSE 0 END) AS other\_medals

FROM olympics\_history\_integrated

GROUP BY noc

)

SELECT region FROM cte

JOIN olympics\_history\_noc\_regions USING(noc)

WHERE gold\_medals = 0 AND other\_medals <> 0;

-- 19. In which Sport/event, India has won the highest number of medals.

SELECT sport, COUNT(medal) AS medals

FROM olympics\_history\_integrated

JOIN olympics\_history\_noc\_regions USING(noc)

WHERE region='India'

GROUP BY sport

ORDER BY medals DESC

LIMIT 1;

SELECT event, COUNT(medal) AS medals

FROM olympics\_history\_integrated

JOIN olympics\_history\_noc\_regions USING(noc)

WHERE region='India'

GROUP BY event

ORDER BY medals DESC

LIMIT 1;

-- 20. Break down all Olympic games where India won a medal for Hockey and how many medals in each Olympic game.

SELECT games, COUNT(medal) AS medals

FROM olympics\_history\_integrated

JOIN olympics\_history\_noc\_regions USING(noc)

WHERE region='India' AND sport LIKE '%Hockey%'

GROUP BY games

HAVING medals <> 0

ORDER BY medals DESC;

Step 3:- Loading Results to client database or snowflake data warehouse.

In Snowflake, we create result table in data warehouse

--How many olympics games have been held?

create table result1 as select count(distinct games) as Total\_Games from olympics\_history;

--List down all Olympics games held so far.

create table result2 as select distinct games from olympics\_history;

In Hive/Hbase, We create external table and then import them to client database

--List down all Olympics games held so far.

create external table op\_ol\_2(distinct\_olympics\_game string) row format delimited fields terminated by ‘,’ location '/user/cloudera/olympic/op2';

INSERT OVERWRITE TABLE op\_ol\_2 SELECT DISTINCT games AS total\_olympic\_games FROM olympics\_history;

--Mention the total no of nations who participated in each olympics game?

Create external table op\_ol\_3(games string, total\_participating\_nations int) row format delimited fields terminated by ‘,’ location '/user/cloudera/olympic/op3';

INSERT OVERWRITE TABLE op\_ol\_3 SELECT games, COUNT(DISTINCT noc) FROM OLYMPICS\_HISTORY GROUP BY games;