Music Insights

TEAM CLOUD CODERS

Team Members

- Megha Singha Roy
- Pavan Shetti
- Sanketh Shinde
- Thrishala NP
- Sudeep S

Project Overview

- Builds a distributed ETL pipeline to process users music listening data.
- Uses AWS Glue to clean and transform raw data.
- Final metrics are queried through Athena for insights.

Use Cases

1. **Measure User Engagement:**
    - Measure total listens, users, and duration.

2. **Top Songs**
    - Find top 3 songs in each genre daily.

3. **Top Genres**
    - Identify 5 most popular genres overall.

Input Data Overview

- Dataset source - S3
- Data format - csv
- Sample schemas

Users schema

```
root
 |-- user_id: integer (nullable = true)
 |-- user_name: string (nullable = true)
 |-- user_age: integer (nullable = true)
 |-- user_country: string (nullable = true)
 |-- created_at: timestamp (nullable = true)
```

Input Data Overview

Songs Schema

```
root
 |-- id: integer (nullable = true)
 |-- track_id: string (nullable = true)
 |-- artists: string (nullable = true)
 |-- album_name: string (nullable = true)
 |-- track_name: string (nullable = true)
 |-- popularity: integer (nullable = true)
 |-- duration_ms: integer (nullable = true)
 |-- explicit: boolean (nullable = true)
 |-- danceability: double (nullable = true)
 |-- energy: double (nullable = true)
 |-- key: integer (nullable = true)
 |-- loudness: double (nullable = true)
 |-- mode: integer (nullable = true)
 |-- speechiness: double (nullable = true)
 |-- acousticness: double (nullable = true)
 |-- instrumentalness: double (nullable = true)
 |-- liveness: double (nullable = true)
 |-- valence: double (nullable = true)
 |-- tempo: double (nullable = true)
 |-- time_signature: integer (nullable = true)
 |-- track_genre: string (nullable = true)
```

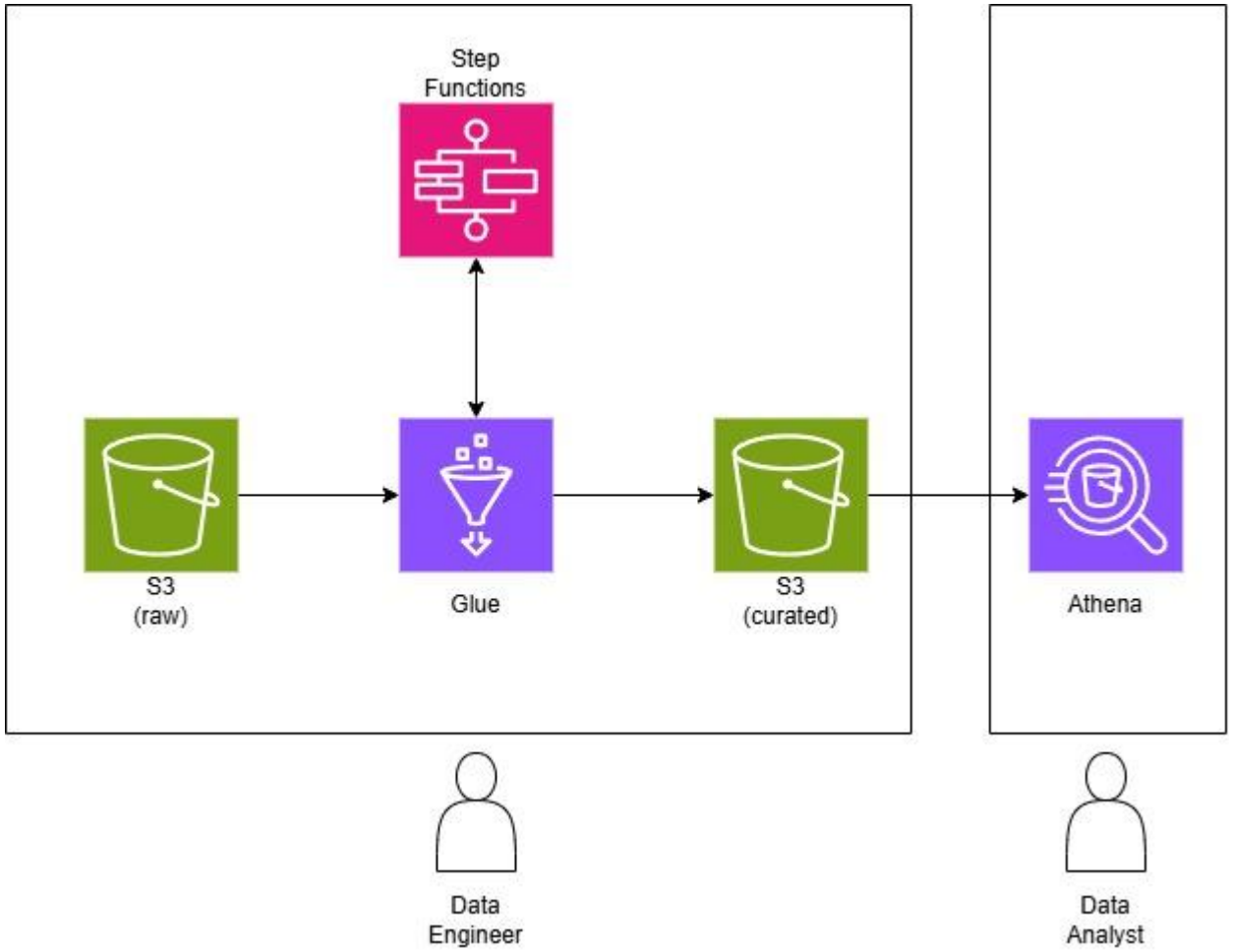
Input Data Overview

Streams Schema

```
root
 |-- user_id: integer (nullable = true)
 |-- track_id: string (nullable = true)
 |-- listen_time: timestamp (nullable = true)
```

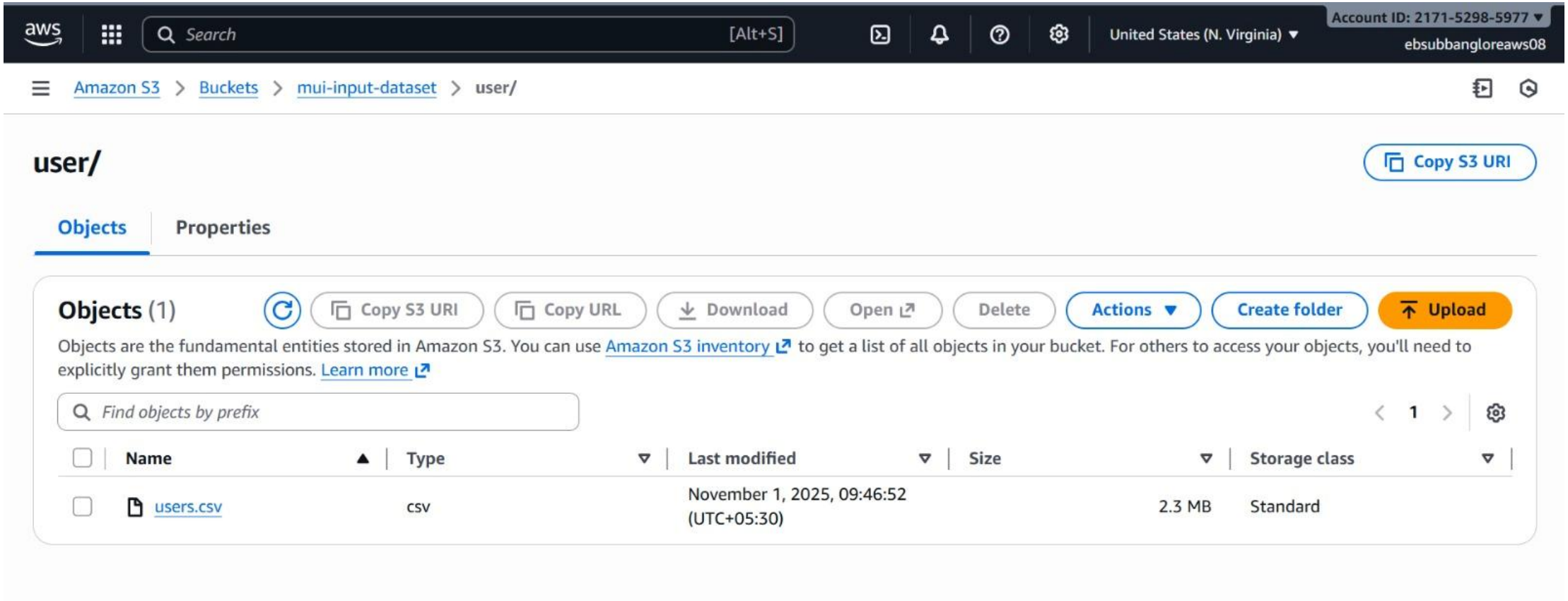
High-Level Architecture

- AWS services used: S3, Glue, Athena, Step Functions



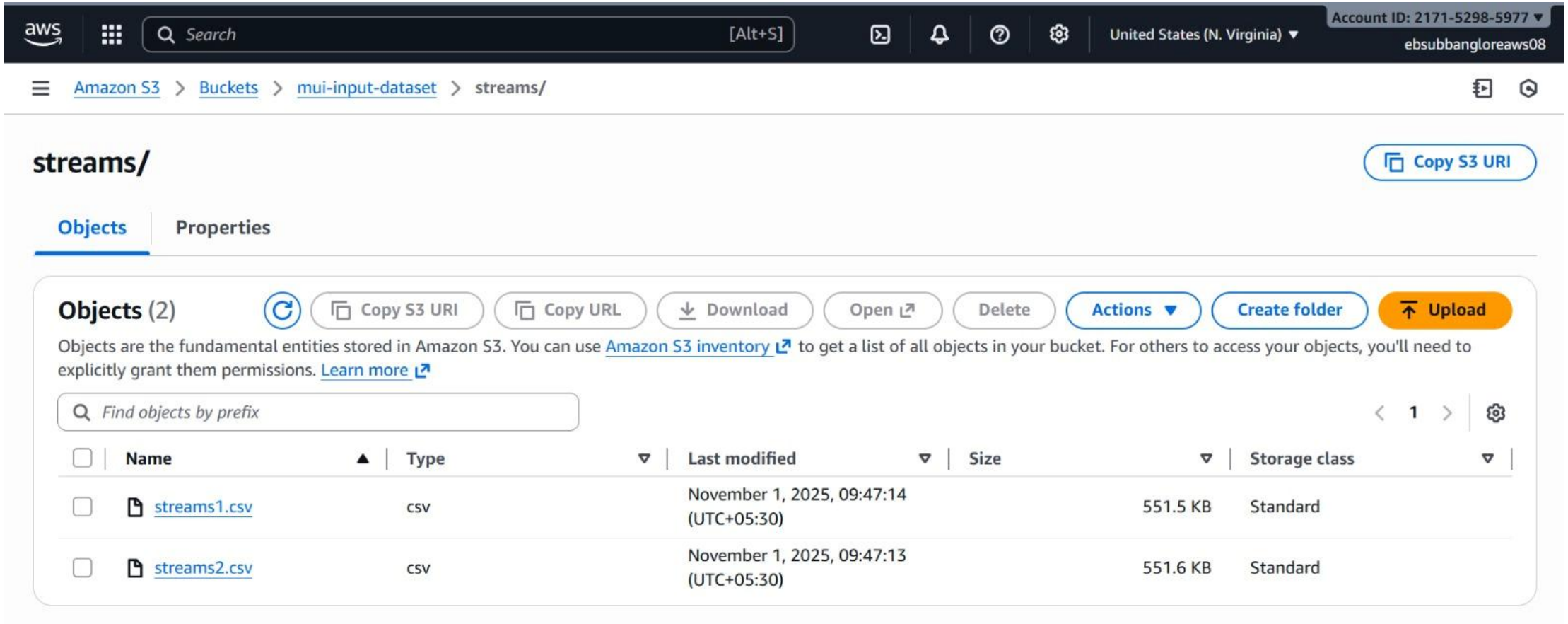
Snapshot

Loading data into S3



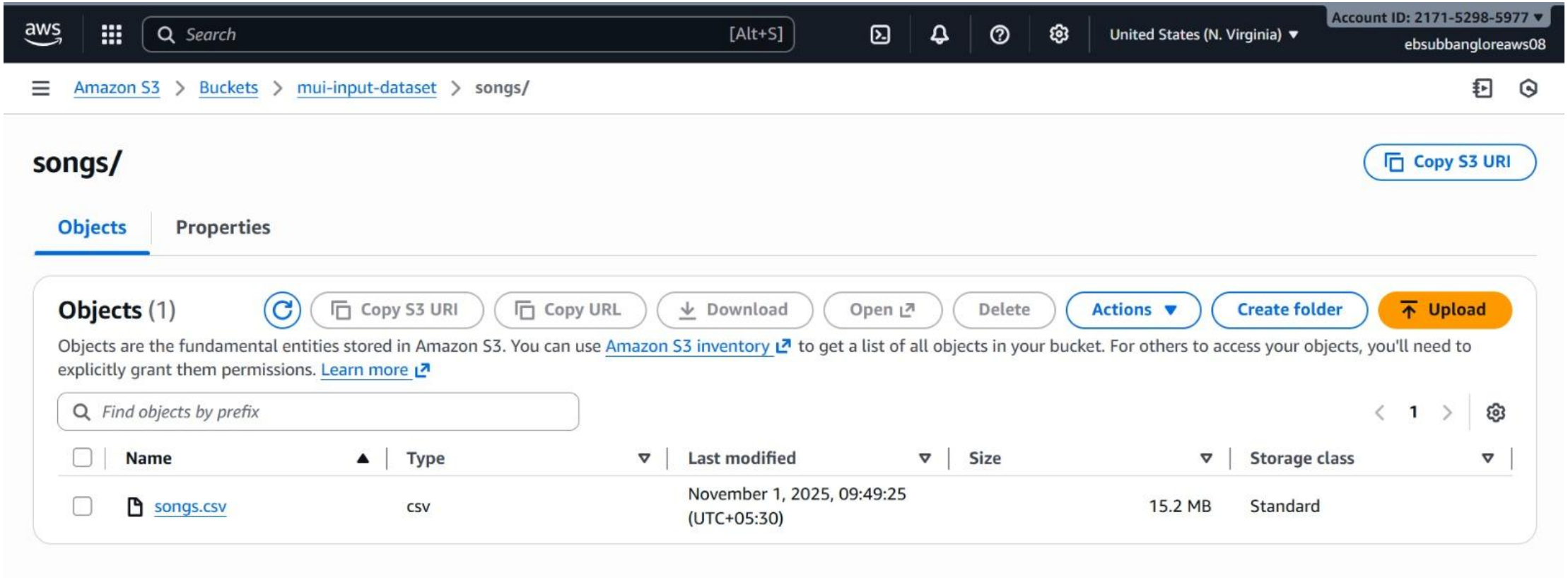
Snapshot

Loading data into S3



Snapshot

Loading data into S3



Snapshot

AWS Glue

Script Info

```
1  from pyspark.sql import SparkSession
2  spark =  SparkSession.builder.getOrCreate()
3
4  usersDf = spark.read.csv('s3://mui-input/songs/', header=True, inferSchema = True)
5  songsDf = spark.read.csv('s3://mui-input/songs/', header=True, inferSchema = True)
6  streamsDf = spark.read.csv('s3://mui-input/streams/', header=True, inferSchema = True)
7
```

Snapshot

AWS Glue

Script Info

```
def transform_data(spark, usersDf, songsDf, streamsDf):
    """drop/fill nulls, typecasting, remove duplicates"""

    #Convert listen_time to date.
    streamsDf = streamsDf.withColumn("report_date", to_date(streamsDf.listen_time))


    #filter invalid keys
    streamsDf = streamsDf.dropna(subset=['user_id', 'track_id', 'listen_time'])

    #type casting of duration to long
    songsDf = songsDf.withColumn("duration_ms", col("duration_ms").cast("long"))

    return usersDf, songsDf, streamsDf

usersDf, songsDf, streamsDf = transform_data(spark, usersDf, songsDf, streamsDf)

```

Snapshot

AWS Glue

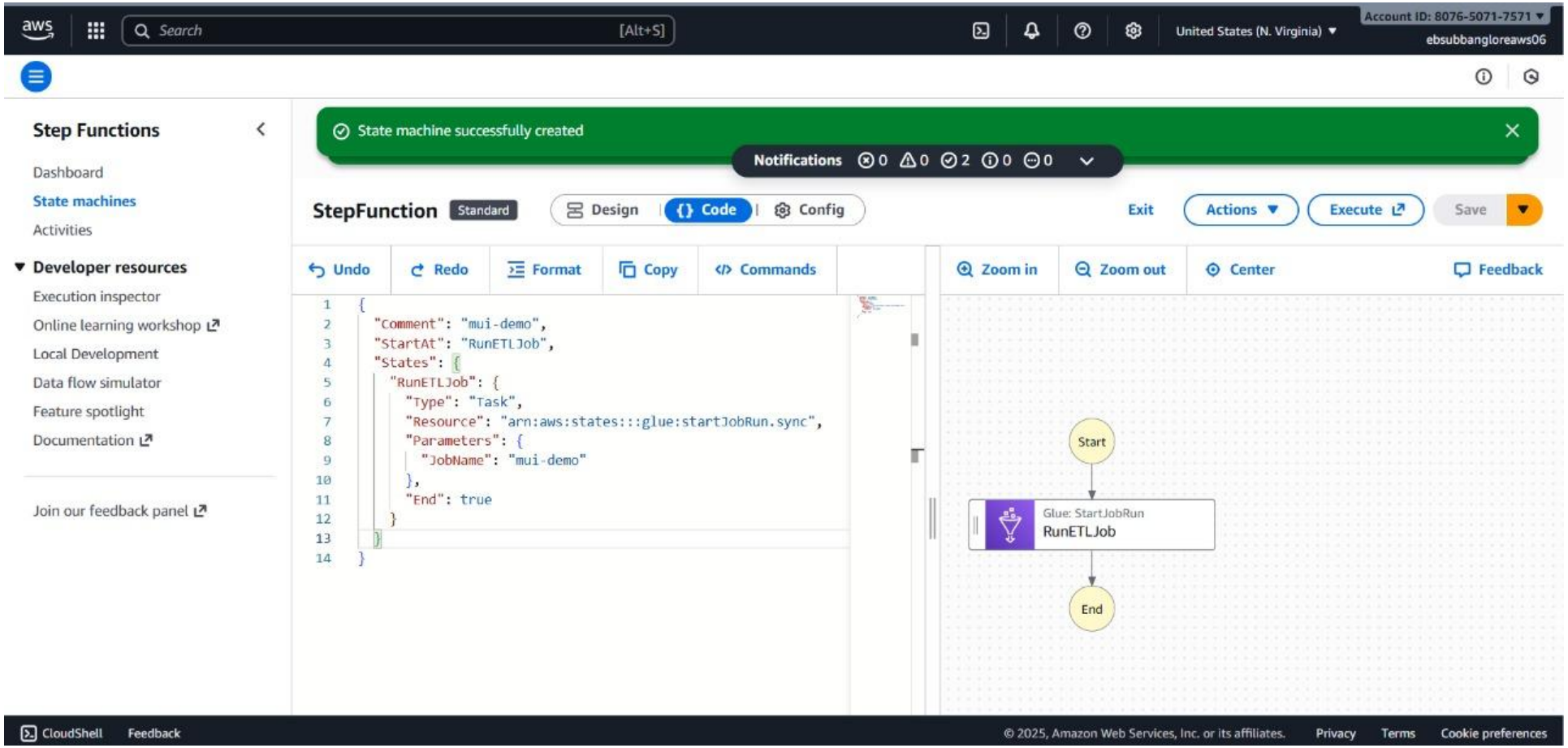
Script Info

```
29
30 def load_data(spark, usersDf, songsDf, streamsDf):
31     db_name = "music_insights"
32     s3_output_path = "s3://mui-output/"
33
34      #Creating database
35     spark.sql(f"CREATE DATABASE IF NOT EXISTS {db_name} LOCATION '{s3_output_path}';")
36
37     #Save dataframe as glue table
38     def save_as_table(df, table_name):
39         table_path = f"{s3_output_path}{table_name}/"
40         full_name = f"{db_name}.{table_name}"
41
42         df.write.format("parquet").mode("overwrite") \
43             .option("path", table_path) \
44             .saveAsTable(full_name)
45         print(f"Created table: {full_name}")
46
47     save_as_table(songsDf, "songs")
48     save_as_table(usersDf, "users")
49     save_as_table(streamsDf, "streams")
50
51 load_data(spark, usersDf, songsDf, streamsDf)
52
```

Python Ln 42, Col 35 Errors: 0 Warnings: 0

Snapshot

Step Functions



Snapshot

Track metrics

Snapshot

Result of Track Metrics

Query results | Query stats

Completed — Time in queue: 102 ms — Run time: 976 ms — Data scanned: 3.12 MB

Results (10)

Copy | Download results CSV

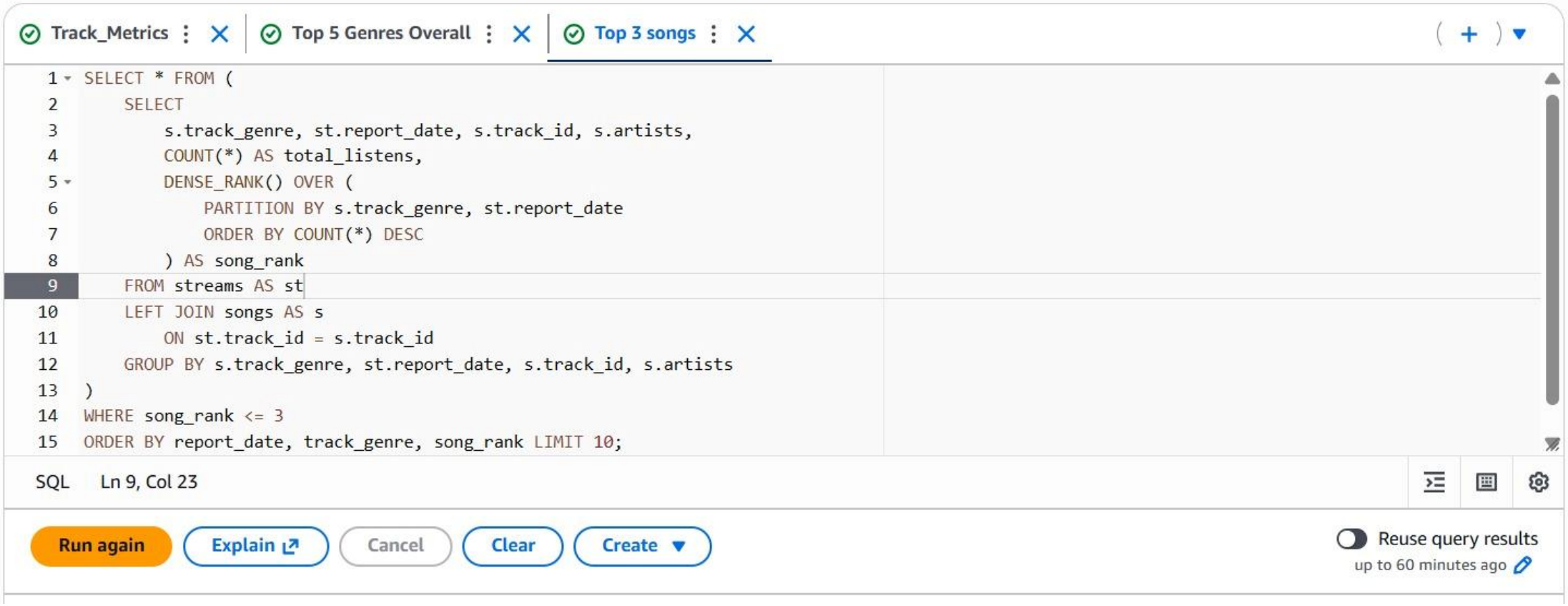
Search rows

1

#	track_id	report_date	total_listens	unique_users	total_listening_duration	avg_listening_time_per_user
1	0BtD9k8XjGliQJMS2hxwNx	2024-06-25	5	5	1377705	275541
2	1kMObCQiYe5opqybH7ZNPD	2024-06-25	4	4	956332	239083
3	4686EWn7tO5A7ABm8DgY0w	2024-06-25	4	4	582544	145636
4	19coiw9dDWBNHtQhdKL1VC	2024-06-25	4	4	640152	160038
5	2u1EtHkbpeRjkxCywhXEYp	2024-06-25	4	4	1329396	332349
6	3a2jFwnts4Cf0OwJbK61SL	2024-06-25	4	4	840108	210027
7	2vdNa2nf6ptRqSiUgGjxlS	2024-06-25	4	4	1799252	449813
8	0k5uPIy8GLdwF16eiIwzmI	2024-06-25	4	4	930024	232506
9	7HcM82GiytrUB4YrkQwY6Y	2024-06-25	4	4	489012	122253
10	3cg38isdTrBH63B4BMywsw	2024-06-25	4	4	1099292	274823

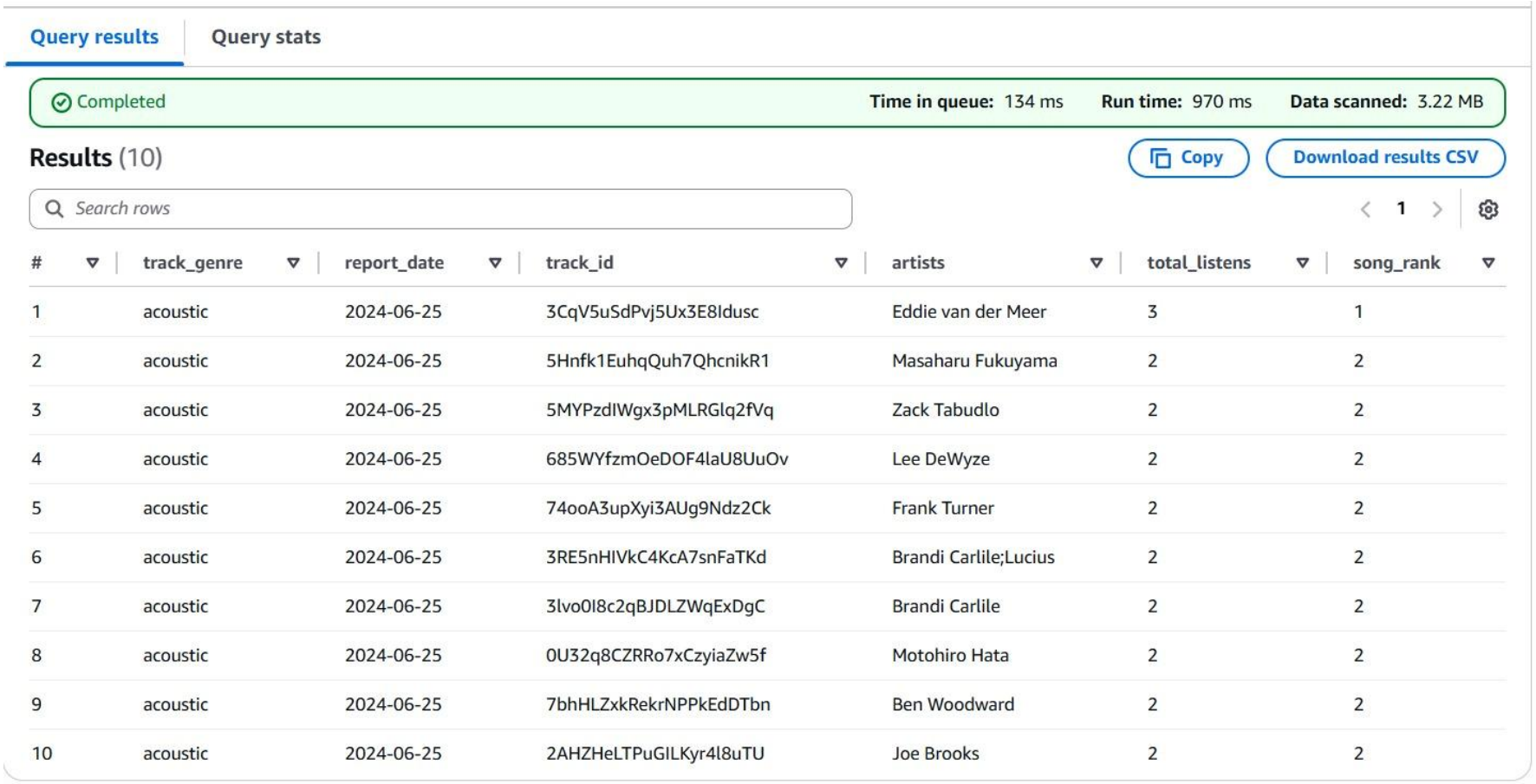
Snapshot

Top 3 songs



Snapshot

Result of Top 3 Songs



Query results | Query stats

Completed — Time in queue: 134 ms — Run time: 970 ms — Data scanned: 3.22 MB

Results (10) — Copy — Download results CSV

Search rows — 1

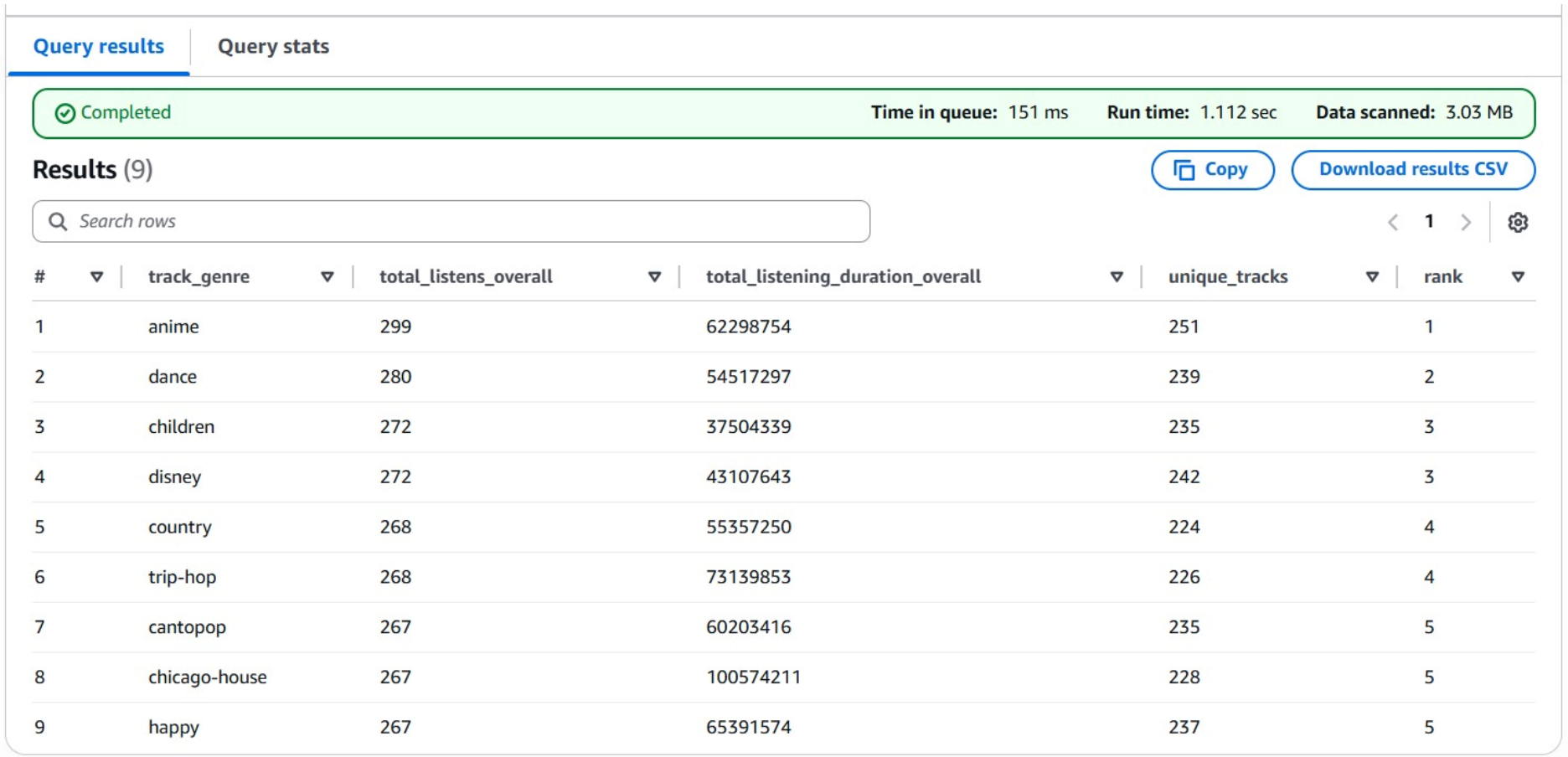
#	track_genre	report_date	track_id	artists	total_listens	song_rank
1	acoustic	2024-06-25	3CqV5uSdPvj5Ux3E8Idusc	Eddie van der Meer	3	1
2	acoustic	2024-06-25	5Hnfk1EuhqQuh7QhcnikR1	Masaharu Fukuyama	2	2
3	acoustic	2024-06-25	5MYPzdIWgx3pMLRGlq2fVq	Zack Tabudlo	2	2
4	acoustic	2024-06-25	685WYfzmOeDOF4laU8UuOv	Lee DeWyze	2	2
5	acoustic	2024-06-25	74ooA3upXyi3AUg9Ndz2Ck	Frank Turner	2	2
6	acoustic	2024-06-25	3RE5nHIVkC4KcA7snFaTKd	Brandi Carlile;Lucius	2	2
7	acoustic	2024-06-25	3lvo0I8c2qBJDLZWqExDgC	Brandi Carlile	2	2
8	acoustic	2024-06-25	0U32q8CZRRo7xCzyiaZw5f	Motohiro Hata	2	2
9	acoustic	2024-06-25	7bhHLZxkRekrNPPkEdDTbn	Ben Woodward	2	2
10	acoustic	2024-06-25	2AHZHeLTPuGILKyr4l8uTU	Joe Brooks	2	2

Snapshot

Top 5 Genres

Snapshot

Result of Top 5 Genres



Query results | Query stats

Completed — Time in queue: 151 ms — Run time: 1.112 sec — Data scanned: 3.03 MB

Results (9) — Copy — Download results CSV

Search rows — 1

#	track_genre	total_listens_overall	total_listening_duration_overall	unique_tracks	rank
1	anime	299	62298754	251	1
2	dance	280	54517297	239	2
3	children	272	37504339	235	3
4	disney	272	43107643	242	3
5	country	268	55357250	224	4
6	trip-hop	268	73139853	226	4
7	cantopop	267	60203416	235	5
8	chicago-house	267	100574211	228	5
9	happy	267	65391574	237	5

Analytical Insights

- Identified Top Songs per Genre (daily) to track music popularity.
- Highlighted Top Genres overall, showing user listening trends.
- Calculated user engagement metrics - total listens, unique users, and listening duration.

Future Scope

1. Real-Time Data Processing
2. Automated Data Ingestion and Crawling
3. Integrate Boto3 SDK


THANK YOU!