

MILESTONE 5

PROJECT 2

GROUP – 9

TOPIC:

**SPOTIFY MUSIC STREAMING TREND
ANALYSIS**

MEMBERS:

KAVYA PATI BANDLA: Patibandla.ka@northeastern.edu

LIKHITHA GUTTAPALLI: guttapalli.l@northeastern.edu

SPOTIFY MUSIC TREND ANALYTICS

The end-to-end ETL pipeline built for the Spotify music trend analytics is explained in this report. The pipeline processes raw data from the S3 bucket, transforms it into dimension and fact tables using Amazon glue, and then Amazon Athena is used for analytics

Dataset Information:

- Source: Kaggle
- Dataset link: [Spotify Popularity Prediction-ML Practice](#)
- Format:
- Song_data: contains detailed information regarding the audio features of the song
 - Contains 18836 rows and 10 columns of different data types
- Song_info: the information regarding the song characteristics such as the artist, album, playlist etc.
 - Contains 11836 rows and 4 columns of different data types
- Generated the song_history.json file for details regarding the number of streams and chart positions.

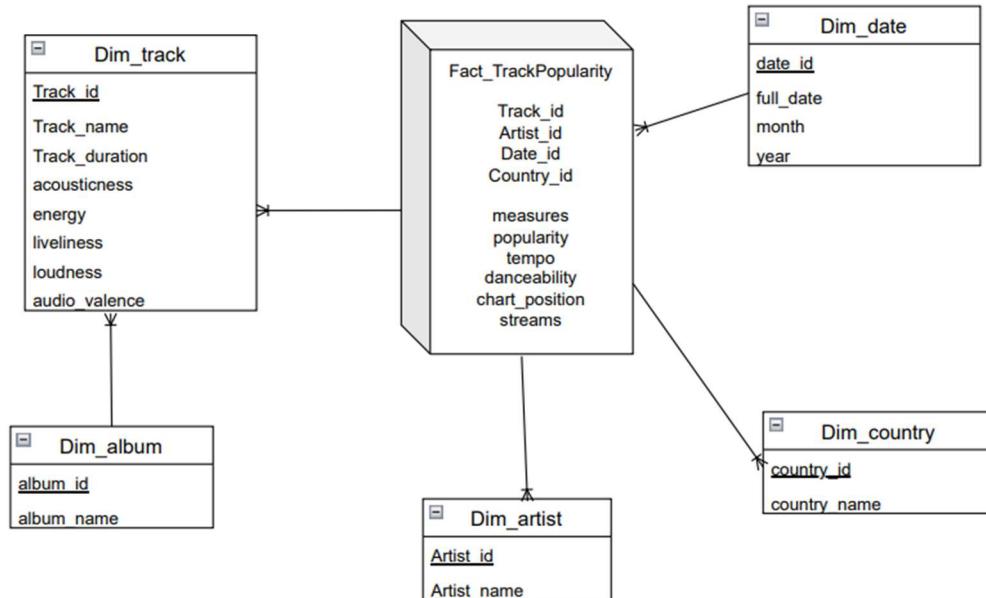
Dimension Tables:

- Dim_track: Contains information regarding the track such as track_id, track name, duration, energy, danceability, liveliness..
- Dim_artist: Contains the artist information such as the artist_id and artist name
- Dim_album: Contains the information regarding the album id and album name
- Dim_date: Contains the fulldate, month and year for further analysis
- Dim_country: contains the country_id and country name

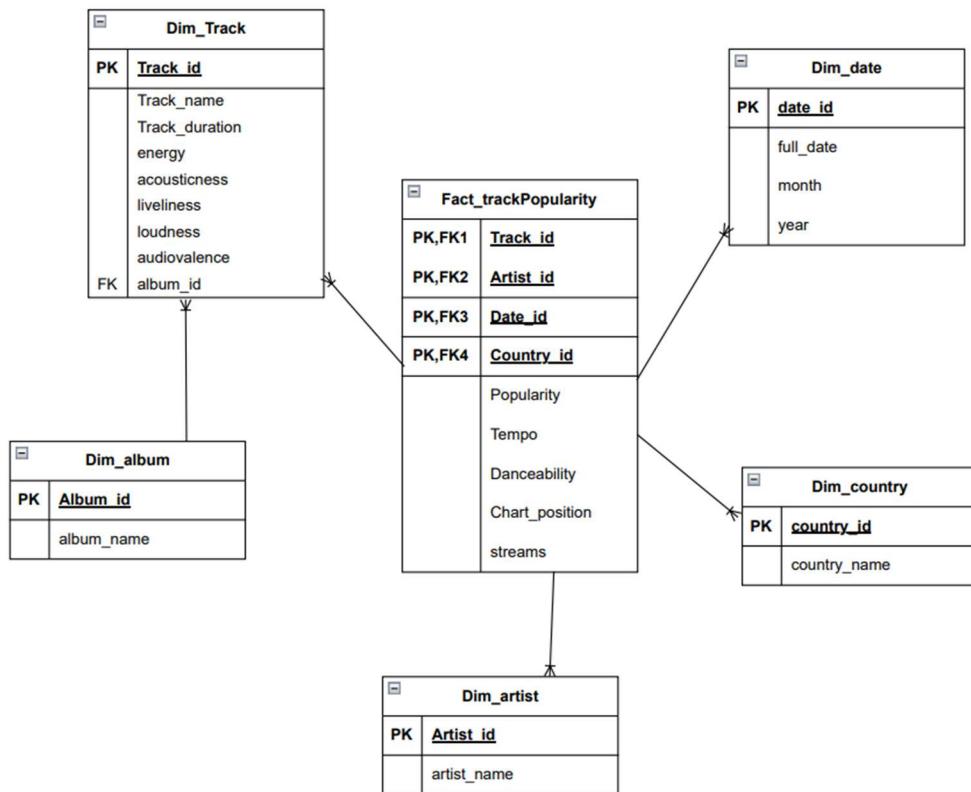
Fact Table:

- Fact_TrackPopularity: Contains the keys of the dimension table along with measures such as chart position, popularity and number of streams

Conceptual model



Logical Model



ETL PIPELINE

- **Storage layer**
 - Amazon S3 bucket:
 - Raw data path:
 - Processed data path:
- **Processing layer**
 - AWS Lambda Trigger: Activates Glue ETL jobs when all required files are present. Sequential job execution (dimension tables first, fact tables second)
 - AWS Glue ETL Jobs: Creates dimension tables from raw data and creates fact tables referencing dimension tables
- **Data Crawling**
 - Processed-data crawler: Scans the processed data files to create catalog
 - Raw-data crawler: Scans the raw data files to create catalog tables
- **Serving layer**
 - AWS Athena is used as a SQL query interface for the analytics. It enables direct querying of the processed data

Data Extraction:

- An S3 bucket is created for storing data.
- Manually uploaded the csv and the json files to the raw data folder created in the bucket
- A lambda function will be enabled on the bucket, once the files are uploaded it will trigger the ETL jobs

AWS Search [Alt+S] United States (N. Virginia) Account ID: 5699-4215-8796 vclabs/user4576769=Kavya_Pati_Bandla

Amazon S3 > Buckets

General purpose buckets All AWS Regions Directory buckets

General purpose buckets (1) Info
Buckets are containers for data stored in S3.

Find buckets by name

Name	AWS Region	Creation date
spotify-popularity-analytics	US East (N. Virginia) us-east-1	November 21, 2025, 15:38:11 (UTC-05:00)

Create bucket

Account snapshot Info
Updated daily Storage Lens provides visibility into storage usage and activity trends.

External access summary - new Info
Updated daily External access findings help you identify bucket permissions that allow public access or access from other AWS accounts.

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

46°F Mostly cloudy

Search

3:39 PM 11/21/2025

AWS Search [Alt+S] United States (N. Virginia) Account ID: 5699-4215-8796 vclabs/user4576769=Kavya_Pati_Bandla

Amazon S3 > Buckets > spotify-popularity-analytics

spotify-popularity-analytics Info

Objects Metadata Properties Permissions Metrics Management Access Points

Objects (2)
Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

Find objects by prefix

Name	Type	Last modified	Size	Storage class
processed-data/	Folder	-	-	-
raw-data/	Folder	-	-	-

Actions Create folder Upload

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

46°F Mostly cloudy

Search

3:41 PM 11/21/2025

Screenshot of the AWS S3 console showing the 'raw-data/' bucket. The 'Objects' tab is selected, displaying three files: 'song_data.csv', 'song_info.csv', and 'track_history.json'. The table includes columns for Name, Type, Last modified, Size, and Storage class.

Name	Type	Last modified	Size	Storage class
song_data.csv	csv	November 21, 2025, 15:39:45 (UTC-05:00)	2.1 MB	Standard
song_info.csv	csv	November 21, 2025, 15:39:46 (UTC-05:00)	1.2 MB	Standard
track_history.json	json	November 21, 2025, 15:39:44 (UTC-05:00)	2.8 MB	Standard

Screenshot of a Windows taskbar showing various pinned icons and system status information (46°F, Mostly cloudy).

Data Crawling

- The datasets in the raw data are made available in the data catalogue using the AWS Glue Crawler. Which crawls all the sub folders present in the raw data for data.

Screenshot of the AWS Glue console showing the 'rawdata' crawler configuration. The crawler is successfully starting. The 'Crawler runs' section shows one run completed on November 21, 2025, at 20:49:04, with 3 table changes and 0 partition changes.

Start time (UTC)	End time (UTC)	Current/last duration	Status	DPU hours	Table changes
November 21, 2025 at 20:49:04	November 21, 2025 at 20:49:46	42 s	Completed	0.121	3 table changes, 0 partition changes

The screenshot shows the AWS Glue Data Catalog interface. On the left, there's a sidebar with navigation links for AWS Glue, Data Catalog, Data Integration and ETL, and Legacy pages. The main content area is titled 'spotify' and displays 'Database properties' with columns for Name (spotify), Description (-), Location (-), and Created on (UTC) (November 21, 2025 at 20:48:00). Below this is a table titled 'Tables (3)' showing three entries: 'song_data_csv', 'song_info_csv', and 'track_history_json'. Each row includes columns for Name, Database (spotify), Location (s3://spotify-popularity-ana), Classification (CSV or JSON), Deprecated (-), View data (Table data), Data quality (View data quality), and Column statistics (View statistics). At the bottom of the page, there are links for CloudShell, Feedback, and various AWS services like Lambda, S3, and CloudWatch.

Data Transformation

- The AWS Glue python script is used to transform the raw data into processed data, that is the dimension tables and fact table
 - Dimsong
 - Dimartist
 - Dimalbum
 - Dimdate
 - Dimcountry
 - FactSongPopularity
- Created two scripts for the data transformation. Create_dim_tables and create_fact_table for the transformation of raw data to processed data.

AWS Glue > Jobs

AWS Glue Studio Info

Create job Info

Author in a visual interface focused on data flow.

Visual ETL Notebook

Author using an interactive code notebook.

Notebook

Author code with a script editor.

Script editor

Example jobs Info

Your jobs (2) Info

Filter jobs by property

Job name	Type	Created by	Last modified	AWS Glue version	Action
create_fact_table	Glue ETL	Script	11/21/2025, 5:08:02 PM	5.0	-
create_dim_tables	Glue ETL	Script	11/21/2025, 5:04:37 PM	5.0	-

Data Catalog

Databases
Tables
Stream schema registries
Schemas
Connections
Crawlers
Classifiers
Catalog settings

Data Integration and ETL

Legacy pages

What's New Feedback
Documentation Feedback
AWS Marketplace

Enable compact mode
Enable new navigation

CloudShell Feedback

43°F Mostly cloudy

Search

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

5:08 PM 11/21/2025

AWS Glue > create_dim_tables

create_dim_tables

Last modified on 11/21/2025, 5:04:37 PM

Actions Save Run

Runs

Job runs (1/15) Info

Last updated (UTC) November 21, 2025 at 22:06:39

Run status	Retries	Start time (Local)	End time (Local)	Duration	Capacity (DPUs)	Worker type	Glue version
Succeeded	0	11/21/2025 17:04:40	11/21/2025 17:06:09	1 m 13 s	10 DPUs	G.1X	5.0
Failed	0	11/21/2025 17:02:27	11/21/2025 17:03:54	1 m 3 s	10 DPUs	G.1X	5.0
Failed	0	11/21/2025 16:59:55	11/21/2025 17:01:09	1 m 7 s	10 DPUs	G.1X	5.0
Succeeded	0	11/21/2025 16:54:27	11/21/2025 16:55:57	1 m 22 s	10 DPUs	G.1X	5.0

Run details

Job name	Start time (Local)	Glue version	Last modified on (Local)
create_dim_tables	11/21/2025 17:04:40	5.0	11/21/2025 17:06:09
Id	End time (Local)	Worker type	Log group name
jr_c2ae95f290eac899eb16949d2069f55488c6956e91099b	11/21/2025 17:06:09	G.1X	/aws-glue/jobs
0413e7c79385b77c42	Run status	Max capacity	Number of workers
Succeeded	15 seconds	10 DPUs	10
Retry attempt number	Execution time	Execution class	Timeout
Initial run	1 minute 13 seconds	Standard	480 minutes
Trigger name	Security configuration	Cloudwatch logs	Usage profile
-	-	• Output logs	-
-	-	-	-

CloudShell Feedback

43°F Mostly cloudy

Search

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

5:06 PM 11/21/2025

AWS Glue ●

- Getting started
- ETL jobs
- Visual ETL
- Notebooks**
- Job run monitoring
- Data Catalog tables
- Data connections
- Workflows (orchestration)
- Zero-ETL integrations New

▼ Data Catalog

- Databases
- Tables
- Stream schema registries
- Schemas
- Connections
- Crawlers
- Classifiers
- Catalog settings

► Data Integration and ETL

► Legacy pages

What's New ●

Documentation ●

AWS Marketplace

Enable compact mode

Enable new navigation

create_fact_table

Last modified on 11/21/2025, 6:00:20 PM Actions Save Run

Script Job details Runs Data quality Schedules Version Control

Job runs (1/8) Info Last updated (UTC) November 21, 2025 at 23:02:27 View details Stop job run Troubleshoot with AI Table View Card View

Run status	Retries	Start time (Local)	End time (Local)	Duration	Capacity (DPUs)	Worker type	Glue version
● Succeeded	0	11/21/2025 18:00:23	11/21/2025 18:02:16	1 m 41 s	10 DPUs	G.1X	5.0
● Failed	0	11/21/2025 17:57:45	11/21/2025 17:59:27	1 m 35 s	10 DPUs	G.1X	5.0
● Failed	0	11/21/2025 17:54:47	11/21/2025 17:56:21	1 m 27 s	10 DPUs	G.1X	5.0
● Failed	0	11/21/2025 17:50:28	11/21/2025 17:52:01	1 m 27 s	10 DPUs	G.1X	5.0

Run details Input arguments (9) Logs Run insights Metrics Troubleshooting analysis - preview Spark UI

Job name: `create_fact_table` Start time (Local): `11/21/2025 18:00:23` Glue version: `5.0` Last modified on (Local): `11/21/2025 18:02:16`
 Id: `jr_e20d34e90987f86e0a0e33ef7c4faa45d4bd59e176fe11b11/21/2025 18:02:16 8c7b901640d814d42` Worker type: `G.1X` Log group name: `/aws-glue/jobs`
 Run status: ● Succeeded Start-up time: `12 seconds` Max capacity: `10 DPUs` Number of workers: `10`
 Retry attempt number: `-` Execution time: `-` Execution class: `Standard` Timeout: `480 minutes`
 Initial run: Trigger name: Security configuration: Output logs: ● Cloudwatch logs ● Error logs ● External logs ●

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

43°F Mostly cloudy ● Search ● 6:02 PM 11/21/2025

AWS Glue ●

- Getting started
- ETL jobs
- Visual ETL
- Notebooks**
- Job run monitoring
- Data Catalog tables
- Data connections
- Workflows (orchestration)
- Zero-ETL integrations New

▼ Data Catalog

- Databases
- Tables
- Stream schema registries
- Schemas
- Connections
- Crawlers
- Classifiers
- Catalog settings

► Data Integration and ETL

► Legacy pages

What's New ●

Documentation ●

AWS Marketplace

Enable compact mode

Enable new navigation

create_fact_table

Last modified on 11/21/2025, 6:00:20 PM Actions Save Run

Script Job details Runs Data quality Schedules Version Control

Script Info

```

1 import sys
2 from awsglue.context import GlueContext
3 from pyspark.context import SparkContext
4 from pyspark.sql import functions as F
5 from pyspark.sql.window import Window
6 from pyspark.sql.functions import col
7
8
9 # Glue context
10 sc = SparkContext()
11 glueContext = GlueContext(sc)
12 spark = glueContext.spark_session
13
14 # Input paths for fact job
15 song_history_path = "s3://spotify-popularity-analytics/raw-data/track_history.json"
16 song_info_path = "s3://spotify-popularity-analytics/raw-data/song_info.csv"
17 song_data_path = "s3://spotify-popularity-analytics/raw-data/song_data.csv"
18 dim_song_path = "s3://spotify-popularity-analytics/processed-data/DimSong/"
19 dim_artist_path = "s3://spotify-popularity-analytics/processed-data/DimArtist/"
20 dim_album_path = "s3://spotify-popularity-analytics/processed-data/DimAlbum/"
21 dim_date_path = "s3://spotify-popularity-analytics/processed-data/DimDate/"
22 dim_country_path = "s3://spotify-popularity-analytics/processed-data/DimCountry/"
23
24
25 # Load dimension tables
  
```

Python Ln 67, Col 45 Errors: 0 Warnings: 0

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

5 43°F Mostly cloudy ● Search ● 6:03 PM 11/21/2025

AWS Glue

Getting started
ETL jobs
Visual ETL
Notebooks
Job run monitoring
Data Catalog tables
Data connections
Workflows (orchestration)
Zero-ETL integrations [New](#)

▼ Data Catalog
Databases
Tables
Stream schema registries
Schemas
Connections
Crawlers
Classifiers
Catalog settings

► Data Integration and ETL
► Legacy pages

What's New [New](#)
Documentation [New](#)
AWS Marketplace

Enable compact mode
Enable new navigation

CloudShell Feedback

43°F Mostly cloudy

Search [Alt+S]

Last modified on 11/21/2025, 5:04:57 PM [Actions](#) [Save](#) [Run](#)

create_dim_tables

Script Job details Runs Data quality Schedules Version Control

Script Info

```

1 import sys
2 from awsglue.context import GlueContext
3 from pyspark.context import SparkContext
4 from pyspark.sql import functions as F
5 from pyspark.sql.window import Window
6 from pyspark.sql.types import StructType, StructField, StringType, IntegerType, LongType
7
8 # Glue context
9 sc = SparkContext()
10 glueContext = GlueContext(sc)
11 spark = glueContext.spark_session
12
13 # Input paths
14 song_info_path = "s3://spotify-popularity-analytics/raw-data/song_info.csv"
15 song_data_path = "s3://spotify-popularity-analytics/raw-data/song_data.csv"
16 song_history_path = "s3://spotify-popularity-analytics/raw-data/track_history.json"
17
18 # Read raw data
19 song_info_df = spark.read.option("header", True).csv(song_info_path)
20 song_data_df = spark.read.option("header", True).csv(song_data_path)
21 song_history_df = spark.read.json(song_history_path)
22
23 # DimAlbum
24 # -----
25 album_dim = song_info_df.select("album_names").dropDuplicates() \

```

Python Ln 21, Col 53 Errors: 0 Warnings: 0

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

6:06 PM 11/21/2025

Amazon S3 > Buckets > [spotify-popularity-analytics](#) > processed-data/

Amazon S3

General purpose buckets
Directory buckets
Table buckets
Vector buckets
Access Grants
Access Points (General Purpose Buckets, Fsx file systems)
Access Points (Directory Buckets)
Object Lambda Access Points
Multi-Region Access Points
Batch Operations
IAM Access Analyzer for S3

Block Public Access settings for this account

Storage Lens
Dashboards
Storage Lens groups
AWS Organizations settings

Feature spotlight [11](#)

AWS Marketplace for S3

CloudShell Feedback

43°F Mostly cloudy

Search [Alt+S]

Objects (6) [Copy S3 URI](#) [Copy URL](#) [Download](#) [Open](#) [Delete](#) [Actions](#) [Create folder](#) [Upload](#)

Find objects by prefix

Name	Type	Last modified	Size	Storage class
DimAlbum/_	Folder	-	-	-
DimArtist/_	Folder	-	-	-
DimCountry/_	Folder	-	-	-
DimDate/_	Folder	-	-	-
DimSong/_	Folder	-	-	-
FactSongPopularity/_	Folder	-	-	-

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

6:10 PM 11/21/2025

Data Catalog

- A glue crawler is used to make the data in the processed S3 bucket available in the Data Catalog by scanning all the subfolders within the specified bucket.

AWS Glue > Crawlers > processeddata

Crawler successfully starting
The following crawler is now starting: "processeddata"

Last updated (UTC) November 21, 2025 at 23:08:17

Run crawler Edit Delete

Crawler properties

Name: processeddata	IAM role: LabRole	Database: spotify-popularity	State: READY
Description:	Security configuration:	Lake Formation configuration:	Table prefix:
Maximum table threshold:			
▶ Advanced settings			

Crawler runs (1)
The list of crawler runs for this crawler.

Start time (UTC)	End time (UTC)	Current/last duration	Status	DPU hours	Table changes
November 21, 2025 at 23:08:43	November 21, 2025 at 23:09:45	01 min 02 s	Completed	-	-

Filter data Filter by a date and time range

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

43°F Mostly cloudy 6:09 PM 11/21/2025

AWS Glue > Databases

Databases (2)
A database is a set of associated table definitions, organized into a logical group.

Name	Description	Location URI	Created on (UTC)
spotify	-	-	November 21, 2025 at 20:48:00
spotify-popularity	-	-	November 21, 2025 at 23:07:25

Last updated (UTC) November 21, 2025 at 23:11:07

Edit Delete Add database

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

43°F Mostly cloudy 6:11 PM 11/21/2025

Screenshot of the AWS Glue Database console showing the 'spotify-popularity' database properties and tables.

Database properties:

- Name: spotify-popularity
- Description: -
- Location: -
- Created on (UTC): November 21, 2025 at 23:07:25

Tables (6):

Name	Database	Location	Classification	Deprecated	View data	Data quality	Column statistics
dimalbum	spotify-popularity	s3://spotify-popularity-ana	Parquet	-	Table data	View data quality	View statistics
dimartist	spotify-popularity	s3://spotify-popularity-ana	Parquet	-	Table data	View data quality	View statistics
dimcountry	spotify-popularity	s3://spotify-popularity-ana	Parquet	-	Table data	View data quality	View statistics
dimdate	spotify-popularity	s3://spotify-popularity-ana	Parquet	-	Table data	View data quality	View statistics
dimsong	spotify-popularity	s3://spotify-popularity-ana	Parquet	-	Table data	View data quality	View statistics
factsongpopularity	spotify-popularity	s3://spotify-popularity-ana	Parquet	-	Table data	View data quality	View statistics

CloudShell Feedback: 43°F Mostly cloudy

System Bar: © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 6:11 PM 11/21/2025

Data Loading:

- The target tables are then loaded into Athena for querying and for popularity analytics of the Spotify music

Screenshot of the Amazon Athena Query editor showing a query against the 'factsongpopularity' table.

Data:

- Data source: AwsDataCatalog
- Catalog: None
- Database: spotify-popularity
- Tables and views: factsongpopularity

Query:

```
1 select * from factsongpopularity
```

Run again Explain Cancel Clear Create

Query results:

Completed Time in queue: 106 ms Run time: 1.109 sec Data scanned: 521.14 KB

Results (18,835):

#	song_id	arist_id	date_id	country_id	chart_position	streams	song_popularity	tempo
1	1600	2510	20160611	2	200	8642887002	73	167.06
2	5564	3829	20990115	4	179	7725307844	66	105.256
3	9649	6728	20961022	2	34	3398903753	76	123.881
4	1799	5359	20960924	3	47	7815606665	74	122.444

CloudShell Feedback: 44°F Light rain

System Bar: © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 11:49 AM 11/22/2025