



Northeastern University

College of Engineering

Data Warehousing & Integration

IE 6750

FALL 2024

[Formula 1 Data Pipeline]

Milestone 6

Group 10

Avani Kala

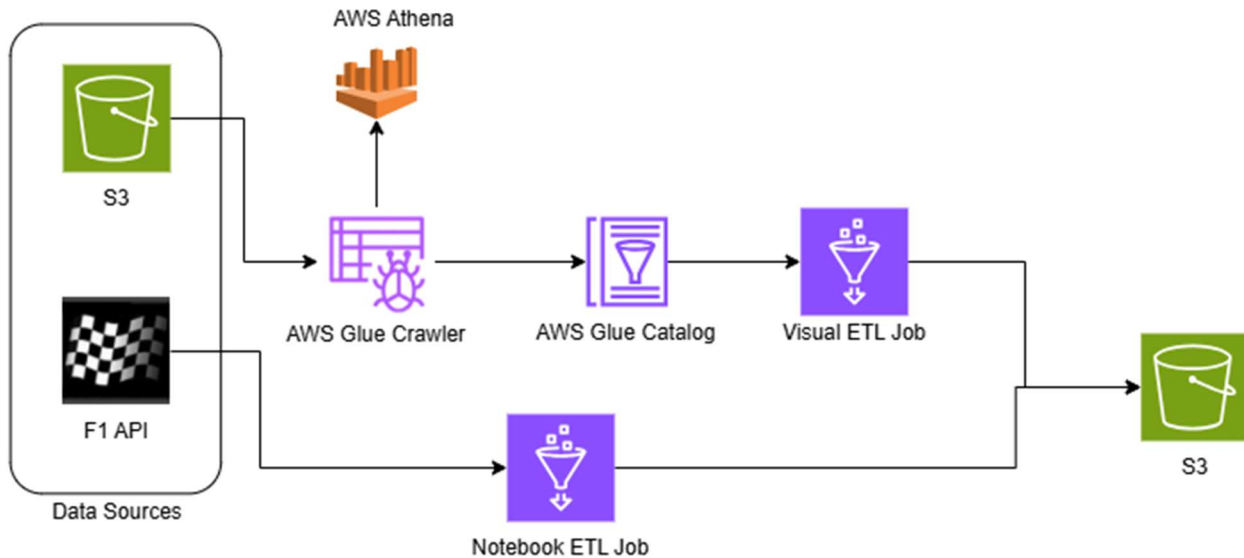
Sri Sai Prabhath Reddy Gudipalli

gudipalli.s@northeastern.edu

kala.a@northeastern.edu

Submission Date: 11/28/2024

Architecture diagram of data pipeline:



Steps involved in building data pipeline:

1. Loading csv files in AWS S3 bucket:

Loaded the csv data into S3 bucket by manual upload. Separate folders are created for each file.

[Amazon S3](#) > [Buckets](#) > [ie6750-f1-datasets](#) > [kaggle-historical-f1/](#) > [kaggle-historical-f1/](#)

kaggle-historical-f1/

[Copy S3 URI](#)

[Objects](#) | [Properties](#)

Objects (7) [Info](#)

[Copy S3 URI](#)

[Copy URL](#)

[Download](#)

[Open](#)

[Delete](#)

[Actions](#)

[Create folder](#)

[Upload](#)

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](#)

< 1 > [Settings](#)

<input type="checkbox"/>	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	circuits/	Folder	-	-	-
<input type="checkbox"/>	constructors/	Folder	-	-	-
<input type="checkbox"/>	driver_standings/	Folder	-	-	-
<input type="checkbox"/>	drivers/	Folder	-	-	-
<input type="checkbox"/>	qualifying/	Folder	-	-	-
<input type="checkbox"/>	races/	Folder	-	-	-
<input type="checkbox"/>	results/	Folder	-	-	-

2. Created AWS Glue Crawler and a new database in Glue Data Catalog. Once the crawler successfully reads the data from S3, the data was loaded in form of tables in database in Glue catalog:

TL

f1-dataset-crawler

Last updated (UTC) November 27, 2024 at 19:40:14 [Run crawler](#) [Edit](#) [Delete](#)

Crawler properties

Name f1-dataset-crawler	IAM role LabRole	Database f1-datasets	State READY
Description -	Security configuration -	Lake Formation configuration -	Table prefix -
Maximum table threshold -			

► Advanced settings

Crawler runs | Schedule | **Data sources** | Classifiers | Tags

Data sources (1) [Info](#)

The list of data sources to be scanned by the crawler.

Type	Data source	Parameters
<input type="radio"/> S3	s3://ie6750-f1-datasets	Recrawl all

AWS Glue > Databases > f1-datasets

f1-datasets

Last updated (UTC) November 27, 2024 at 19:53:45 [Edit](#) [Delete](#)

Database properties

Name f1-datasets	Description -	Location -	Created on (UTC) November 26, 2024 at 04:52:41
---------------------	------------------	---------------	---

Tables (7)

View and manage all available tables.

Filter tables

<input type="checkbox"/>	Name	Database	Location	Classification	Deprecated	View data	Data quality	Column statistics
<input type="checkbox"/>	circuits	f1-datasets	s3://ie6750-f1-datasets/ka	CSV	-	Table data	View data quality	View statistics
<input type="checkbox"/>	constructors	f1-datasets	s3://ie6750-f1-datasets/ka	CSV	-	Table data	View data quality	View statistics
<input type="checkbox"/>	driver_standings	f1-datasets	s3://ie6750-f1-datasets/ka	CSV	-	Table data	View data quality	View statistics
<input type="checkbox"/>	drivers	f1-datasets	s3://ie6750-f1-datasets/ka	CSV	-	Table data	View data quality	View statistics
<input type="checkbox"/>	qualifying	f1-datasets	s3://ie6750-f1-datasets/ka	CSV	-	Table data	View data quality	View statistics
<input type="checkbox"/>	races	f1-datasets	s3://ie6750-f1-datasets/ka	CSV	-	Table data	View data quality	View statistics
<input type="checkbox"/>	results	f1-datasets	s3://ie6750-f1-datasets/ka	CSV	-	Table data	View data quality	View statistics

3. To read the data from Catalog database tables, used AWS Athena to run the SQL queries:

Amazon Athena > Query editor

Editor | Recent queries | Saved queries | Settings

Workgroup: primary

Athena now supports typeahead code suggestions to speed up SQL query development. Typeahead suggestions are turned on by default. You can change this setting in query editor preferences. [Edit preferences](#)

Data

Data source: AwsDataCatalog

Database: f1-datasets

Tables and views: [Create](#)

Filter tables and views

Tables (7)

- circuits
- constructors
- driver_standings
- drivers
- qualifying
- races
- results

Views (0)

Query 14

```
select * from circuits limit 10;
```

SQL Ln 1, Col 33

[Run again](#) [Explain](#) [Cancel](#) [Clear](#) [Create](#)

[Reuse query results](#) up to 60 minutes ago

Query results | Query stats

Completed Time in queue: 112 ms Run time: 531 ms Data scanned: 9.87 KB

[Copy](#) [Download results](#)

Results (10)

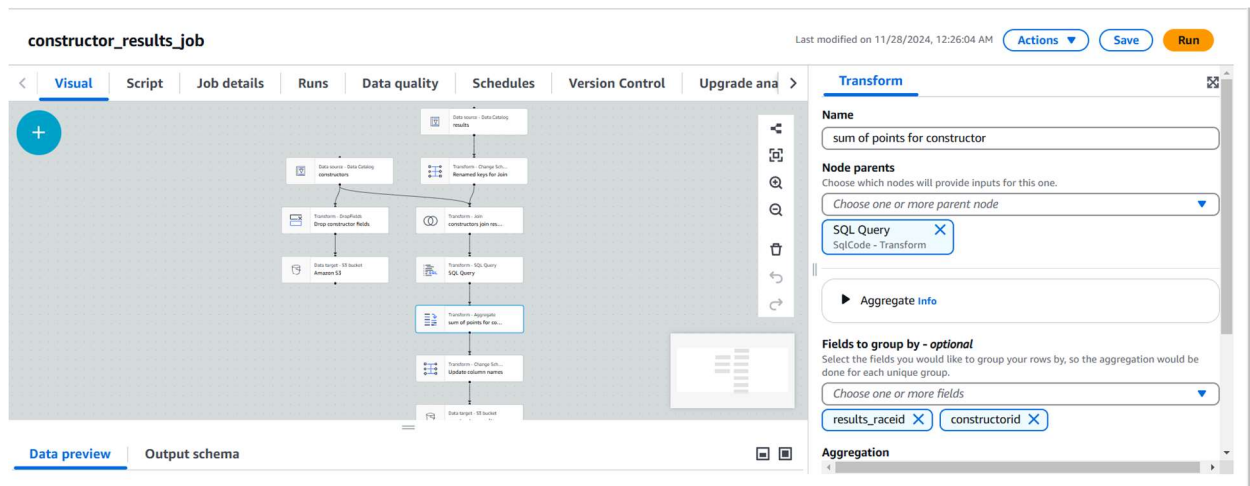
Search rows

#	circuitid	circuitref	name	location	country	lat	lng	alt	url
1	1	"albert_park"	"Albert Park Grand Prix Circuit"	"Melbourne"	"Australia"	-37.8497	144.968	10	"http://en.wikipedia.org/wiki/Melbourne_Grand_Prix_Circ"
2	2	"sepang"	"Sepang International Circuit"	"Kuala Lumpur"	"Malaysia"	2.76083	101.738	18	"http://en.wikipedia.org/wiki/Sepang_International_Circ"
3	3	"bahrain"	"Bahrain International Circuit"	"Sakhir"	"Bahrain"	26.0325	50.5106	7	"http://en.wikipedia.org/wiki/Bahrain_International_Circ"
4	4	"catalunya"	"Circuit de Barcelona-Catalunya"	"Montmeló"	"Spain"	41.57	2.26111	109	"http://en.wikipedia.org/wiki/Circuit_de_Barcelona-Catalu"
5	5	"istanbul"	"Istanbul Park"	"Istanbul"	"Turkey"	40.9517	29.405	130	"http://en.wikipedia.org/wiki/Istanbul_Park"

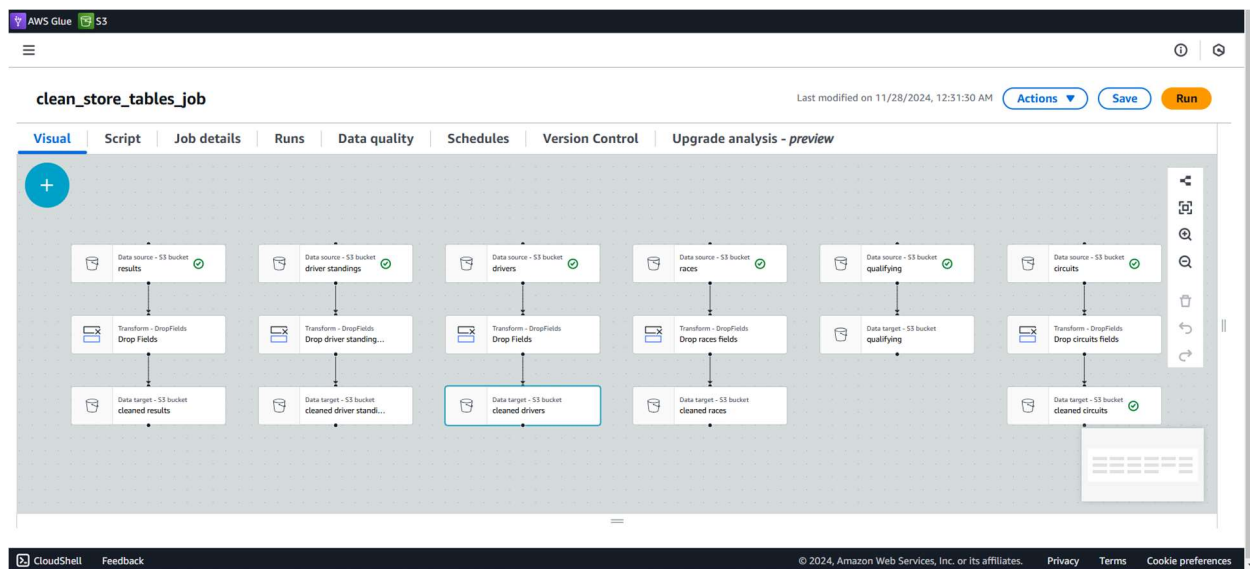
© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

- Created 2 jobs using Visual ETL to clean and transform the data and store the generated data into S3 bucket in csv format:

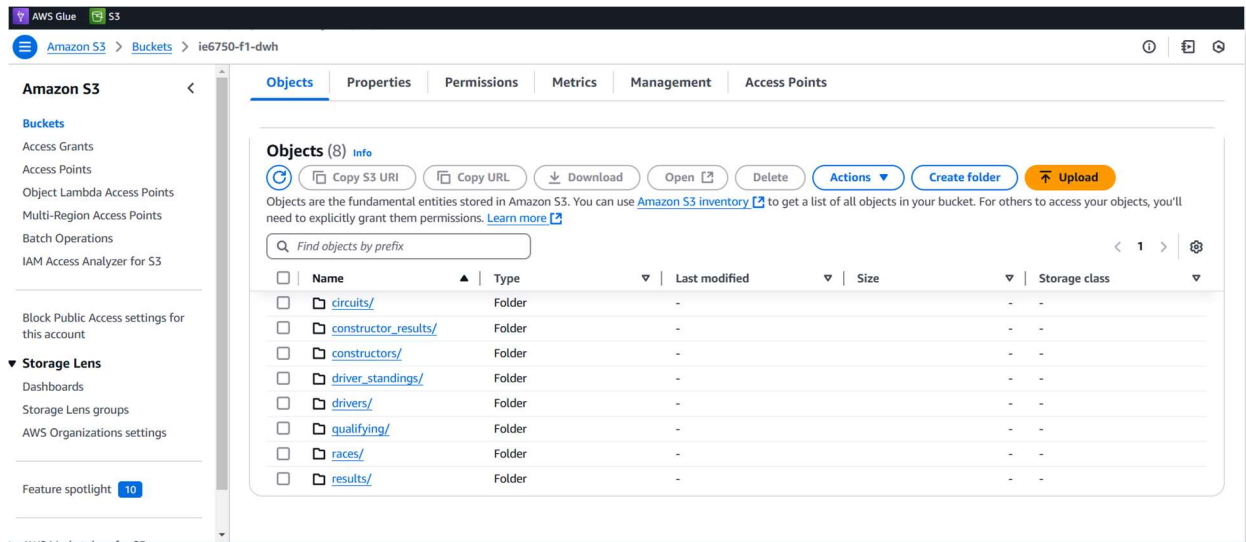
- Joined two input tables to find the sum(points) for each constructor for all of its drivers



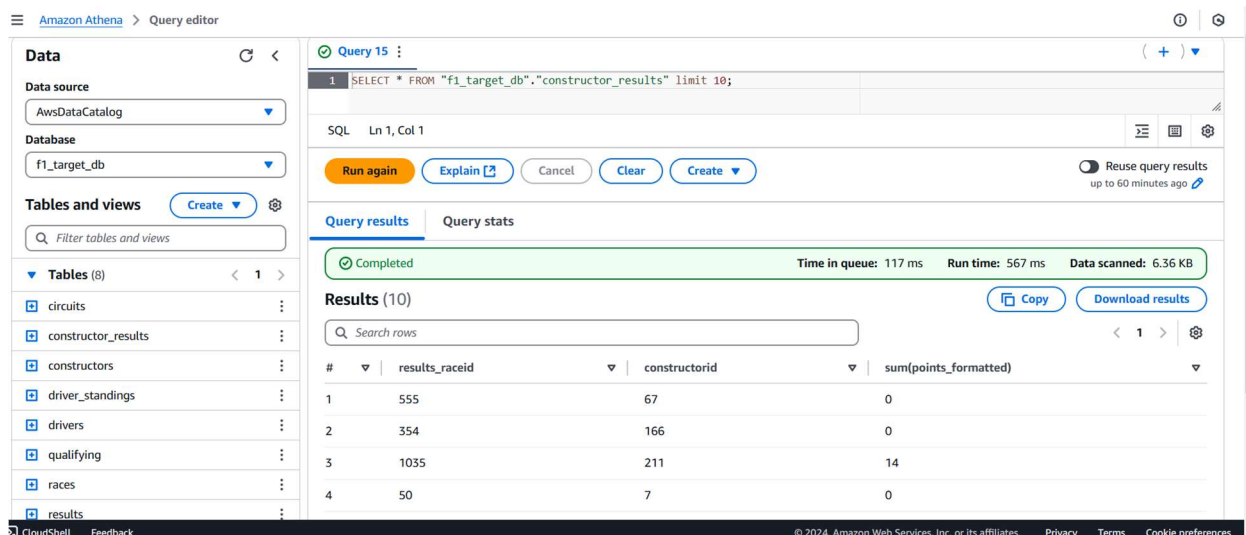
- Cleaned and kept the attributes required for analysis:



- ETL Job output loaded in S3 bucket:



6. Querying the target data in Athena:



7. Created Notebook for accessing data from second data source (Ergast F1 API) to see for new races and store the new data if any in S3:

