

# Lab 02\_03 - EMR

## Lab 02\_03 - EMR

The following resources were created with Terraform. The code can be found in the infrastructure folder.

### Resources

#### S3 Buckets

Buckets (4) [Info](#)

Copy ARN

Buckets are containers for data stored in S3. [Learn more](#)

Find buckets by name

	Name ▲	AWS Region ▼	Access ▼
<input type="radio"/>	<a href="#">st1612.athena.results</a>	US East (N. Virginia) us-east-1	<div><div></div>Public</div>
<input type="radio"/>	<a href="#">st1612.jupyter.notebooks</a>	US East (N. Virginia) us-east-1	<div><div></div>Public</div>
<input type="radio"/>	<a href="#">st1612.orderlogs.lab05</a>	US East (N. Virginia) us-east-1	Bucket and objects not public
<input type="radio"/>	<a href="#">st1612.web</a>	US East (N. Virginia) us-east-1	<div><div></div>Public</div>

#### Datalake

Taken from: <https://www.datos.gov.co/Salud-y-Protecci-n-Social/Casos-positivos-de-COVID-19-en-Colombia/gt2j-8ykr/data>

- Bucket with Colombia Covid-19 data.

Amazon S3 > st1612.web > covid\_colombia/

# covid\_colombia/

**Objects** | Properties

---

## Objects (1)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to generate reports about the objects in your bucket and manage their permissions. [Learn more](#)

Copy S3 URI Copy URL Download Open

<input type="checkbox"/>	Name	Type
<input type="checkbox"/>	<a href="#">Casos_positivos_de_COVID-19_en_Colombia.csv</a>	csv

## Glue

- Crawler

AWS Glue

Data catalog

Databases

Tables

Connections

Crawlers

Crawlers

A crawler connects to a data store, progresses through a prioritized list of classifiers to determine the schema for your data, and then creates metadata tables in your data catalog.

[User preferences](#)

Add crawler

Run crawler

Action

Showing: 1 - 1

<input type="checkbox"/>	Name	Schedule	Status	Logs	Last runtime	Median runtime	Tables updated	Tables added
<input type="checkbox"/>	<a href="#">read-s3-datalake-bucket</a>		Ready	<a href="#">Logs</a>	1 min	1 min	0	1

- Schema/Database

Table properties

skip.header.line.count1sizeKey863789099objectCount1UPDATED\_BY\_CRAWLERread-s3-datalake-bucket

CrawlerSchemaSerializerVersion1.0recordCount3455156averageRecordSize250

CrawlerSchemaDeserializerVersion1.0compressionTypenonecolumnsOrderedtrueareColumnsQuotedfalse

delimiter, typeOfDatafile

Schema

Showing: 1 - 23 of 23 < >

	Column name	Data type	Partition key	Comment
1	fecha reporte web	string		
2	id de caso	bigint		
3	fecha de notificación	string		
4	código divipola departamento	bigint		
5	nombre departamento	string		
6	código divipola municipio	bigint		
7	nombre municipio	string		
8	edad	bigint		
9	unidad de medida de edad	bigint		
10	sexo	string		
11	tipo de contagio	string		

## Athena

- Query Editor

New query 1New query 2+

<

1 SELECT \* FROM "st1612"."covid\_colombia" limit 10;

Run querySave asCreate

(Run time: 0.73 seconds, Data scanned: 1.24 MB)

Format queryClear

Use Ctrl + Enter to run query, Ctrl + Space to autocomplete

Athena engine version 2Release versions

Results

	fecha reporte web	id de caso	fecha de notificación	código divipola departamento	nombre departamento	código divipola municipio	nombre municip
1	7/1/2021 0:00:00	1726079	26/12/2020 0:00:00	11	BOGOTA	11001	BOGOTA
2	7/1/2021 0:00:00	1726080	26/12/2020 0:00:00	11	BOGOTA	11001	BOGOTA
3	7/1/2021 0:00:00	1726081	26/12/2020 0:00:00	11	BOGOTA	11001	BOGOTA
4	7/1/2021 0:00:00	1726082	26/12/2020 0:00:00	13001	CARTAGENA	13001	CARTAGENA
5	7/1/2021 0:00:00	1726083	25/12/2020 0:00:00	8001	BARRANQUILLA	8001	BARRANQUILLA
6	7/1/2021 0:00:00	1726084	25/12/2020 0:00:00	41	HUILA	41001	NEIVA
7	7/1/2021 0:00:00	1726085	25/12/2020 0:00:00	25	CUNDINAMARCA	25214	COTA
8	7/1/2021 0:00:00	1726086	26/12/2020 0:00:00	25	CUNDINAMARCA	25899	ZIPAQUIRA
9	7/1/2021 0:00:00	1726087	25/12/2020 0:00:00	68	SANTANDER	68001	BUCARAMANGA
10	7/1/2021 0:00:00	1726088	25/12/2020 0:00:00	47	MAGDALENA	47288	FUNDACION

## EMR

- Cluster

Amazon EMR

EMR Studio

EMR on EC2

Clusters

Notebooks

Git repositories

Security configurations

Block public access

VPC subnets

Events

EMR on EKS

Virtual clusters

Help

What's new

SummaryApplication user interfacesMonitoringHardwareConfigurationsEventsStepsBootstrap actions

Summary

ID: j-3VMGHE6WLJVPJ

Creation date: 2021-11-08 20:22 (UTC-5)

Elapsed time: 3 hours, 12 minutes

After last step completes: Cluster waits

Termination protection: Off [Change](#)

Tags: -- [View All](#) / [Edit](#)

Master public DNS: ec2-18-232-96-10.compute-1.amazonaws.com [Connect to the Master Node Using SSH](#)

Configuration details

Release label: emr-6.4.0

Hadoop distribution: Amazon 3.2.1

Applications: Tez 0.9.2, Spark 3.1.2, Hive 3.1.2, JupyterHub 1.4.1, HCatalog 3.1.2, Zeppelin 0.9.0, Hue 4.9.0

Log URI: --

EMRFS consistent view: Disabled

Custom AMI ID: --

Application user interfaces

Persistent user interfaces [\[?\]](#): [Spark history server](#), [YARN timeline server](#), [Tez UI](#)

On-cluster user interfaces [\[?\]](#): Not Enabled [Enable an SSH Connection](#)

Network and hardware

Availability zone: us-east-1a

Subnet ID: [subnet-0c9db81b9eb7b5386](#) [\[?\]](#)

Master: Running 1 m4.xlarge

Core: Running 1 m4.xlarge

Task: --

Cluster scaling: Not enabled

Auto-termination: Not enabled

Security and access

Key name: ssh\_ec2\_instance\_lab02

EC2 instance profile: [arn:aws:iam::368396404142:instance-profile/emr\\_profile](#)

EMR role: [arn:aws:iam::368396404142:role/iam\\_emr\\_service\\_role](#)

Visible to all users: All [Change](#)

Security groups for Master: [sg-05059f3ea78a7d67e](#) [\[?\]](#)  
([emr\\_apps\\_allow\\_public\\_access](#)) [More](#)

Security groups for Core & [sg-05059f3ea78a7d67e](#) [\[?\]](#)

- Deployed applications

Amazon EMR

EMR Studio

EMR on EC2

Clusters

Notebooks

Git repositories

Security configurations

Block public access

VPC subnets

Events

EMR on EKS

Virtual clusters

Help

What's new

SummaryApplication user interfacesMonitoringHardwareConfigurationsEventsSteps

Persistent application user interfaces

Applications installed on the Amazon EMR cluster publish user interfaces (UI) as web sites to monitor cluster activity. Persister don't required SSH tunneling. They are hosted off of the cluster.

Application user interface [\[?\]](#)

[YARN timeline server](#)

[Tez UI](#)

[Spark history server](#)

On-cluster application user interfaces

On-cluster UI are available only while clusters are running. Because they are hosted on the master node, on-cluster UI require these application UI. [Learn more](#) [\[?\]](#)

Application	User interface URL <a href="#">[?]</a>
HDFS Name Node	<a href="#">http://ec2-18-232-96-10.compute-1.amazonaws.com:9870/</a>
Tez UI	<a href="#">http://ec2-18-232-96-10.compute-1.amazonaws.com:8080/tez-ui</a>
Spark History Server	<a href="#">http://ec2-18-232-96-10.compute-1.amazonaws.com:18080/</a>
JupyterHub	<a href="#">https://ec2-18-232-96-10.compute-1.amazonaws.com:9443/</a>
Zeppelin	<a href="#">http://ec2-18-232-96-10.compute-1.amazonaws.com:8890/</a>
Hue	<a href="#">http://ec2-18-232-96-10.compute-1.amazonaws.com:8888/</a>
Resource Manager	<a href="#">http://ec2-18-232-96-10.compute-1.amazonaws.com:8088/</a>

The following table lists web interfaces you can view on the task nodes:

Application	User interface URL
HDFS Data Node	<a href="#">http://ec2-000-000-000-000.compute-1.amazonaws.com:9864/</a>

Hue

- Query data from Glue Database

The screenshot shows the AWS Glue console interface. On the left, a sidebar lists tables under the catalog 'st1612'. The main area displays a Hive query: `select * from covid_colombia limit 50;`. Below the query, the execution status is shown as 'Completed' with a time taken of 0.884 seconds. A 'Query History' tab is active, showing a table of results with 50 rows. The table has three columns: `covid_colombia.fecha reporte web`, `covid_colombia.id de caso`, and `covid_colombia.fecha de notificación`. The results show dates from 6/3/2020 to 11/3/2020 and IDs from 1 to 8. On the right, a 'Tables' panel lists the columns and their data types for the `st1612.covid_colombia` table.

id	fecha reporte web	id de caso	fecha de notificación
1	6/3/2020 0:00:00	1	2/3/2020 0:00:00
2	9/3/2020 0:00:00	2	6/3/2020 0:00:00
3	9/3/2020 0:00:00	3	7/3/2020 0:00:00
4	11/3/2020 0:00:00	4	9/3/2020 0:00:00
5	11/3/2020 0:00:00	5	9/3/2020 0:00:00
6	11/3/2020 0:00:00	6	10/3/2020 0:00:00
7	11/3/2020 0:00:00	7	8/3/2020 0:00:00
8	11/3/2020 0:00:00	8	9/3/2020 0:00:00

## Jupyter

- Notebook

```
In [4]: sqlDF = spark.sql("SELECT * FROM raw")
df.show()

FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(height='25px', width='50%'),...

In [5]: sqlDF.columns

FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(height='25px', width='50%'),...

['fecha reporte web', 'id de caso', 'fecha de notificación', 'código divipola departamento', 'nombre departament
o', 'código divipola municipio', 'nombre municipio', 'edad', 'unidad de medida de edad', 'sexo', 'tipo de contagi
o', 'ubicación del caso', 'estado', 'código iso del país', 'nombre del país', 'recuperado', 'fecha de inicio de sí
ntomas', 'fecha de muerte', 'fecha de diagnóstico', 'fecha de recuperación', 'tipo de recuperación', 'pertenencia
étnica', 'nombre del grupo étnico', 'partition_0']

In [8]: sqlDF.select("fecha reporte web", "estado").show(5)

FloatProgress(value=0.0, bar_style='info', description='Progress:', layout=Layout(height='25px', width='50%'),...

+-----+-----+
|fecha reporte web|estado|
+-----+-----+
|fecha reporte web|Estado|
| 6/3/2020 0:00:00| Leve|
| 9/3/2020 0:00:00| Leve|
| 9/3/2020 0:00:00| Leve|
|11/3/2020 0:00:00| Leve|
+-----+-----+
only showing top 5 rows
```