

Implementación de un Lakehouse en AWS

Introducción

Este proyecto se centra en la implementación de un Lakehouse en Amazon Web Services (AWS) utilizando datasets obtenidos de Kaggle, que contienen información sobre las emisiones de CO2 de diferentes países a lo largo de los años y las emisiones generadas por la producción de alimentos. El objetivo principal es diseñar e implementar un ecosistema de datos que integre un Data Lake en S3 para el almacenamiento escalable, un Data Warehouse en Redshift para consultas avanzadas, y el procesamiento de grandes volúmenes de datos mediante Hadoop y Spark en AWS EMR. Este ecosistema permitirá la ejecución de consultas SQL optimizadas tanto en Amazon Athena como en Redshift Spectrum, facilitando el análisis detallado y la obtención de insights a partir de los datos.

Descripción de los Datasets

1. CO2 emissions by country over time

Este archivo contiene datos históricos sobre las emisiones de dióxido de carbono (CO2) de diferentes países a lo largo del tiempo. La información está organizada por país y por año, permitiendo analizar las tendencias de emisiones de CO2 a nivel global y compararlas entre regiones. Los campos principales incluyen:

- **Países:** Listado de los diferentes países que reportan sus emisiones de CO2.
- **Años:** Años en los que se registraron los niveles de emisiones.
- **Emisiones de CO2 (Toneladas métricas):** La cantidad de dióxido de carbono emitida, medida en toneladas métricas, para cada país en un año específico.

Este dataset es crucial para estudiar cómo las emisiones de CO2 han cambiado a lo largo del tiempo, identificar patrones de crecimiento o disminución.

CO2 emissions from food production

Este archivo contiene datos sobre las emisiones de CO2 generadas específicamente por la producción de alimentos. Incluye varios sectores relacionados con la agricultura y la ganadería, desglosando cómo cada tipo de producción contribuye a las emisiones globales de gases de efecto invernadero. Los campos principales incluyen:

- **Tipos de producción alimentaria:** Desglose por categorías, como producción agrícola, ganadería, y procesos industriales relacionados con alimentos.
- **Países o regiones:** Países o regiones donde se recopilan los datos.
- **Años:** Períodos en los que se registran las emisiones.
- **Emisiones de CO2 (Toneladas métricas):** La cantidad de CO2 emitida por cada tipo de producción alimentaria, lo que permite medir su impacto ambiental.

Este dataset es relevante para entender el papel de la industria alimentaria en el cambio climático y para identificar qué sectores alimentarios contribuyen más a las emisiones, proporcionando una base para estrategias de mitigación en el futuro.

Diseño del Data Lake

El Data Lake se ha implementado utilizando Amazon S3, estructurando los datos en diferentes zonas:

- **Zona Raw:** Aquí se almacenan los datasets originales en formato CSV, tal como se obtuvieron.
- **Zona Trusted:** Después de aplicar procesos de transformación y limpieza, los datos se almacenan en esta zona para su análisis y consulta.

La estructura de directorios en S3 se organizó de la siguiente manera:

/trabajo-1-almac-recup-info /

/raw/

/country/

/food/

/trusted/

/country/

/food/

The screenshot shows the Amazon S3 console interface for the bucket 'trabajo-1-almac-recup-info'. The breadcrumb navigation at the top indicates the path: Amazon S3 > Buckets > trabajo-1-almac-recup-info. Below the bucket name, there are tabs for 'Objetos', 'Propiedades', 'Permisos', 'Métricas', 'Administración', and 'Puntos de acceso'. The 'Objetos' tab is selected, displaying a list of objects. Above the list, there are buttons for 'Copiar URI de S3', 'Copiar URL', 'Descargar', 'Abrir', 'Eliminar', 'Acciones', 'Crear carpeta', and 'Cargar'. A search bar is present with the placeholder text 'Buscar objetos por prefijo'. The object list has columns for 'Nombre', 'Tipo', 'Última modificación', 'Tamaño', and 'Clase de almacenamiento'. Two objects are listed: 'raw/' and 'trusted/', both of type 'Carpeta' (Folder).

	Nombre	Tipo	Última modificación	Tamaño	Clase de almacenamiento
<input type="checkbox"/>	raw/	Carpeta	-	-	-
<input type="checkbox"/>	trusted/	Carpeta	-	-	-

Amazon S3 > Buckets > trabajo-1-almac-recup-info > raw/

raw/ Copiar URI de S3

Objetos | Propiedades

Objetos (2) Información 🔄 Copiar URI de S3 Copiar URL Descargar Abrir Eliminar Acciones Crear carpeta Cargar

Los objetos son las entidades fundamentales que se almacenan en Amazon S3. Puede utilizar el [inventario de Amazon S3](#) para obtener una lista de todos los objetos de su bucket. Para que otras personas obtengan acceso a sus objetos, tendrá que concederles permisos de forma explícita. [Más información](#)

🔍 Buscar objetos por prefijo

<input type="checkbox"/>	Nombre	Tipo	Última modificación	Tamaño	Clase de almacenamiento
<input type="checkbox"/>	country/	Carpeta	-	-	-
<input type="checkbox"/>	food/	Carpeta	-	-	-

Proceso catalogación y ETL con AWS Glue

Para este proceso, se crearon dos crawlers en AWS Glue: uno encargado de extraer la información de la carpeta **country** y otro de la carpeta **food**. Cada crawler genera una tabla separada que almacena todos los datos correspondientes de los archivos CSV en su respectiva carpeta.

Crawlers.

AWS Glue > Crawlers > catalogCO2byCountry

catalogCO2byCountry Last updated (UTC) September 7, 2024 at 18:47:06 🔄 Run crawler Edit Delete

Crawler properties

Name catalogCO2byCountry	IAM role LabRole	Database labone	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](#) 🔄 Edit Remove Add a data source

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

	Type	Data source	Parameters
<input type="radio"/>	S3	s3://trabajo-1-almac-recup-info/raw/country/	Recrawl all

AWS Glue > Crawlers > catalogCO2byFood

catalogCO2byFood

Last updated (UTC)
September 7, 2024 at 18:48:47

Run crawler

Edit

Delete

Crawler properties

Name

catalogCO2byFood

IAM role

LabRole [↗](#)

Database

labone

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](#)

Edit

Remove

Add a data source

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

	Type	Data source	Parameters
<input type="radio"/>	S3	s3://trabajo-1-almac-recup-info/raw/food/	Recrawl all

Ejecutamos los Crawlers.

Starting crawler

Attempting to start run crawler "catalogCO2byFood"

AWS Glue > 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.

Crawlers (2) [Info](#)

Last updated (UTC)
September 7, 2024 at 18:49:22

Action ▼

Run

Create crawler

View and manage all available crawlers.

Q Filter crawlers

< 1 > ⚙

<input type="checkbox"/>	Name ▼	State ▼	Schedule	Last run ▼	Last run time... ▼	Log	Table changes fr...
<input type="checkbox"/>	catalogCO2byCo...	Running		-	-	-	-
<input type="checkbox"/>	catalogCO2byFood	Running		-	-	-	-

Se verifica la creación de las tablas en la base de datos.

AWS Glue > Databases > labone

labone

Last updated (UTC)
September 7, 2024 at 18:50:51

Edit

Delete

Database properties

Name

labone

Description

-

Location

-

Created on (UTC)

September 7, 2024 at 18:46:50

Tables (2)

Last updated (UTC)
September 7, 2024 at 18:50:53

Delete

Add tables using crawler

Add table

View and manage all available tables.

Q Filter tables

< 1 > ⚙

<input type="checkbox"/>	Name ▲	Database ▼	Location ▼	Classification ▼	Deprecated ▼	View data	Data quality
<input type="checkbox"/>	country	labone	s3://trabajo-1-almac-	CSV	-	Table data	View data quality
<input type="checkbox"/>	food	labone	s3://trabajo-1-almac-	CSV	-	Table data	View data quality

Schema						
Partitions						
Indexes						
Column statistics - new						

Schema (9)						
View and manage the table schema.						
<input type="text" value="Filter schemas"/>						
#	Column name	Data type	Partition key	Comment		
1	country	string	-	-		
2	code	string	-	-		
3	calling code	bigint	-	-		
4	year	bigint	-	-		
5	co2 emission (tons)	bigint	-	-		
6	population(2022)	bigint	-	-		
7	area	bigint	-	-		
8	% of world	string	-	-		
9	density(km2)	string	-	-		

Schema (23)						
View and manage the table schema.						
<input type="text" value="Filter schemas"/>						
#	Column name	Data type	Partition key	Comment		
1	food product	string	-	-		
2	land use change	double	-	-		
3	animal feed	double	-	-		
4	farm	double	-	-		
5	processing	double	-	-		
6	transport	double	-	-		
7	packaging	double	-	-		
8	retail	double	-	-		
9	total_emissions	double	-	-		
10	eutrophying emissions per 1000kcal ...	double	-	-		
11	eutrophying emissions per kilogram ...	double	-	-		
12	eutrophying emissions per 100g pro...	double	-	-		
13	freshwater withdrawals per 1000kca...	double	-	-		
14	freshwater withdrawals per 100g pr...	double	-	-		
15	freshwater withdrawals per kilogra...	double	-	-		
16	greenhouse gas emissions per 1000...	double	-	-		
17	greenhouse gas emissions per 100g ...	double	-	-		
18	land use per 1000kcal (m² per 1000...	double	-	-		
19	land use per kilogram (m² per kilogr...	double	-	-		
20	land use per 100g protein (m² per 1...	double	-	-		

Consultas SQL con Athena y Hive

Con los datos ya catalogados y transformados, se procedió a realizar un análisis exploratorio utilizando **AWS Athena y Hive**. Estas herramientas permitieron ejecutar consultas SQL sobre los datasets, facilitando la extracción de información relevante para el análisis del impacto de las emisiones de CO2 a nivel global y en el sector alimentario.

1. **Consultas de Prueba:** Inicialmente, se llevaron a cabo consultas simples para verificar la integridad de los datos y asegurar que no hubiera inconsistencias, como valores nulos o duplicados.
2. **Consultas Avanzadas:** Después de las pruebas iniciales, se ejecutaron consultas más complejas que involucraban agregaciones, filtrados y cálculos sobre los datos de emisiones. Por ejemplo, se calcularon las emisiones promedio por continente y se identificaron los principales contribuyentes a las emisiones de CO2 a lo largo del tiempo.
3. **Resultados de las Consultas:** Los resultados de las consultas fueron almacenados para su posterior análisis. Estos datos fueron revisados exhaustivamente para confirmar que las transformaciones y cargas de datos se ejecutaron correctamente.

Esto permitió asegurar que los datos estuvieran listos para los siguientes pasos del análisis.

Realizamos un primer análisis exploratorio de los datos de la tabla country

The screenshot shows a data query interface with a sidebar on the left and a main panel on the right. The sidebar contains sections for 'Datos', 'Tablas y vistas', and 'Vistas'. The main panel displays a SQL query: `SELECT * FROM "labone"."country" limit 10;`. Below the query, there are buttons for 'Ejecutar de nuevo', 'Explicar', 'Cancelar', 'Borrar', and 'Crear'. The results section shows 'Resultados (10)' with a table of data. The table has columns: #, country, code, calling code, year, co2 emission (tons), population(2022), area, % of world, and density(kr). The first three rows show data for Afghanistan.

#	country	code	calling code	year	co2 emission (tons)	population(2022)	area	% of world	density(kr)
1	Afghanistan	AF	93	1750	0	41128771	652230	0.40%	63/km
2	Afghanistan	AF	93	1751	0	41128771	652230	0.40%	63/km
3	Afghanistan	AF	93	1752	0	41128771	652230	0.40%	63/km

Con la siguiente consulta podemos identificar que en el archivo se tienen en cuenta 220 países

The screenshot shows a data query interface with a sidebar on the left and a main panel on the right. The sidebar contains sections for 'Datos', 'Tablas y vistas', and 'Vistas'. The main panel displays a SQL query: `SELECT COUNT(DISTINCT "country") FROM "labone"."country";`. Below the query, there are buttons for 'Ejecutar de nuevo', 'Explicar', 'Cancelar', 'Borrar', and 'Crear'. The results section shows 'Resultados (1)' with a table of data. The table has columns: #, _col0. The first row shows the count 220.

#	_col0
1	220

Con la siguiente consulta pudimos verificar que la última entrada de los países que se tienen en el documento fue en 2020, por lo que tenemos bastante información para trabajar con estos datos.

Datos

Origen de datos: AwsDataCatalog

Base de datos: labone

Tablas y vistas: **Crear**

▼ Tablas (2) < 1 >

- country
- food

► Vistas (0) < 1 >

Consulta 5

```
1 SELECT "country", MAX("year") AS "ultima_entrada"
2 FROM "labone"."country"
3 GROUP BY "country";
```

SQL Ln 3, Col 20

Ejecutar de nuevo **Explicar** **Cancelar** **Borrar** **Crear**

☐ Volver a utilizar los resultados de la consulta hasta hace 60 minutos

Resultados de la consulta Estado de la consulta

Completado Tiempo en cola: 67 ms Tiempo de ejecución: 543 ms Datos analizados: 3.01 MB

Resultados (220) **Copiar** **Descargar resultados**

#	country	ultima_entrada
1	Austria	2020
2	Bahrain	2020
3	Bangladesh	2020

Realizamos otra consulta para evitar los datos nulos y categorizar las emisiones de CO2 por país y por año. En esta consulta encontramos que hay muchos datos que son 0 por lo cual deberían ser evitados en un futuro, debido a que no son datos relevantes

Datos

Origen de datos: AwsDataCatalog

Base de datos: labone

Tablas y vistas: **Crear**

▼ Tablas (2) < 1 >

- country
- food

► Vistas (0) < 1 >

Consulta 5

```
1 SELECT "country", "year", "co2 emission (tons)"
2 FROM "labone"."country"
3 WHERE "co2 emission (tons)" IS NOT NULL
4 ORDER BY "country", "year";
```

SQL Ln 1, Col 24

Ejecutar de nuevo **Explicar** **Cancelar** **Borrar** **Crear**

☐ Volver a utilizar los resultados de la consulta hasta hace 60 minutos

Resultados de la consulta Estado de la consulta

Completado Tiempo en cola: 117 ms Tiempo de ejecución: 836 ms Datos analizados: 3.01 MB

Resultados (59.257) **Copiar** **Descargar resultados**

#	country	year	co2 emission (tons)
1	Afghanistan	1750	0
2	Afghanistan	1751	0
3	Afghanistan	1752	0
4	Afghanistan	1753	0

Ahora realizamos una consulta para saber que países son los que han generado más CO2 a lo largo de la historia.

Datos

Origen de datos

AwsDataCatalog

Base de datos

labone

Tablas y vistas

Crear

country

food

Vistas (0)

Consulta 3

Consulta 4

Consulta 5

Consulta 6

Consulta 7

Consulta 8

Consulta 10

1 SELECT "country", SUM("co2 emission (tons)") AS "Total_CO2"

2 FROM "labone"."country"

3 GROUP BY "country"

4 ORDER BY "Total_CO2" DESC;

SQL Ln 4, Col 27

Ejecutar de nuevo

Explicar

Cancelar

Borrar

Crear

Volver a utilizar los resultados de la consulta hasta hace 60 minutos

Resultados de la consulta

Estado de la consulta

Completado

Tiempo en cola: 102 ms

Tiempo de ejecución: 634 ms

Datos analizados: 3.01 MB

Resultados (220)

Copiar

Descargar resultados

Filas de búsqueda

1 2 3 4 5 6 7 8

#	country	Total_CO2
1	United Kingdom	6162081873517
2	Germany	5300530860522
3	Russia	3000738685231
4	United States	2761767447823
5	France	2238247224410
6	Japan	2051064334740
7	China	1669896307149
8	Poland	1311553745811

Luego de observar esto se realizó una búsqueda de China, Estados Unidos y Rusia, que son los países más desarrollados del mundo, por lo tanto, sus emisiones de CO2 también deberían ser considerablemente altas y encontramos que tienen muchos años con las emisiones en CO2 en 0 o sin valores:

Datos

Origen de datos

AwsDataCatalog

Base de datos

labone

Tablas y vistas

Crear

country

food

Vistas (0)

Consulta 3

Consulta 4

Consulta 5

Consulta 6

Consulta 7

Consulta 8

Consulta 10

1 SELECT "country", SUM("co2 emission (tons)") AS "Total_CO2"

2 FROM "labone"."country"

3 GROUP BY "country"

4 ORDER BY "Total_CO2" DESC;

SQL

Ln 4, Col 27

Ejecutar de nuevo

Explicar

Cancelar

Borrar

Crear

Volver a utilizar los resultados de la consulta

hasta hace 60 minutos

Resultados de la consulta

Estado de la consulta

Completado

Tiempo en cola: 102 ms

Tiempo de ejecución: 634 ms

Datos analizados: 3.01 MB

Resultados (220)

Copiar

Descargar resultados

Filas de búsqueda

1

2

3

4

5

6

7

8

country

Total_CO2

1 United Kingdom

6162081873517

2 Germany

5300530860522

3 Russia

3000738685231

4 United States

2761767447823

5 France

2238247224410

6 Japan

2051064334740

7 China

1669896307149

8 Poland

1311553745811

Debido a esto se detectó que algunos datos correspondientes a Estados Unidos estaban ausentes en varios años, aunque deberían estar presentes según las expectativas del dataset. Este error en la carga inicial de los datos sugiere que hubo un problema durante el proceso de ingestión.

2 from "labone"."country"

3 where "country" = 'United States'

4 and "co2 emission (tons)" is null

5 ORDER BY "year" DESC;

SQLLn 5, Col 21

Ejecutar de nuevo

Explicar

Cancelar

Borrar

Crear

Volver a utilizar los resultados de la consulta

hasta hace 60 minutos

Resultados de la consulta

Estado de la consulta

Completado

Tiempo en cola: 101 ms

Tiempo de ejecución: 510 ms

Datos analizados: 3.01 MB

Resultados (67)

Copiar

Descargar resultados

Filas de búsqueda

< 1 >

#	country	code	calling code	year	co2 emission (tons)	population(2022)	area	% of world	density(km2)
1	United States	US	1	2020		338289857	9372610	6.10%	36/km
2	United States	US	1	2019		338289857	9372610	6.10%	36/km
3	United States	US	1	2018		338289857	9372610	6.10%	36/km
4	United States	US	1	2017		338289857	9372610	6.10%	36/km
5	United States	US	1	2016		338289857	9372610	6.10%	36/km
6	United States	US	1	2015		338289857	9372610	6.10%	36/km
7	United States	US	1	2014		338289857	9372610	6.10%	36/km
8	United States	US	1	2013		338289857	9372610	6.10%	36/km

E57182

41700000000000

Country	Code	Calling Code	Year	CO2 emission (Tons)	Population(2022)	Area	% of World	Density(km2)
United States	US		1 1982	2,11E+13	338289857	9372610	6,1 36/km²	
United States	US		1 1983	2,15E+13	338289857	9372610	6,1 36/km²	
United States	US		1 1984	2,2E+13	338289857	9372610	6,1 36/km²	
United States	US		1 1985	2,24E+13	338289857	9372610	6,1 36/km²	
United States	US		1 1986	2,29E+13	338289857	9372610	6,1 36/km²	
United States	US		1 1987	2,34E+13	338289857	9372610	6,1 36/km²	
United States	US		1 1988	2,39E+13	338289857	9372610	6,1 36/km²	
United States	US		1 1989	2,44E+13	338289857	9372610	6,1 36/km²	
United States	US		1 1990	2,49E+13	338289857	9372610	6,1 36/km²	
United States	US		1 1991	2,54E+13	338289857	9372610	6,1 36/km²	
United States	US		1 1992	2,59E+13	338289857	9372610	6,1 36/km²	
United States	US		1 1993	2,65E+13	338289857	9372610	6,1 36/km²	
United States	US		1 1994	2,7E+13	338289857	9372610	6,1 36/km²	
United States	US		1 1995	2,75E+13	338289857	9372610	6,1 36/km²	
United States	US		1 1996	2,81E+13	338289857	9372610	6,1 36/km²	
United States	US		1 1997	2,87E+13	338289857	9372610	6,1 36/km²	
United States	US		1 1998	2,92E+13	338289857	9372610	6,1 36/km²	
United States	US		1 1999	2,98E+13	338289857	9372610	6,1 36/km²	
United States	US		1 2000	3,04E+13	338289857	9372610	6,1 36/km²	
United States	US		1 2001	3,1E+13	338289857	9372610	6,1 36/km²	
United States	US		1 2002	3,16E+13	338289857	9372610	6,1 36/km²	
United States	US		1 2003	3,22E+13	338289857	9372610	6,1 36/km²	
United States	US		1 2004	3,28E+13	338289857	9372610	6,1 36/km²	
United States	US		1 2005	3,34E+13	338289857	9372610	6,1 36/km²	
United States	US		1 2006	3,4E+13	338289857	9372610	6,1 36/km²	
United States	US		1 2007	3,47E+13	338289857	9372610	6,1 36/km²	
United States	US		1 2008	3,53E+13	338289857	9372610	6,1 36/km²	
United States	US		1 2009	3,58E+13	338289857	9372610	6,1 36/km²	
United States	US		1 2010	3,64E+13	338289857	9372610	6,1 36/km²	
United States	US		1 2011	3,69E+13	338289857	9372610	6,1 36/km²	
United States	US		1 2012	3,75E+13	338289857	9372610	6,1 36/km²	
United States	US		1 2013	3,8E+13	338289857	9372610	6,1 36/km²	
United States	US		1 2014	3,86E+13	338289857	9372610	6,1 36/km²	
United States	US		1 2015	3,91E+13	338289857	9372610	6,1 36/km²	
United States	US		1 2016	3,96E+13	338289857	9372610	6,1 36/km²	
United States	US		1 2017	4,01E+13	338289857	9372610	6,1 36/km²	
United States	US		1 2018	4,07E+13	338289857	9372610	6,1 36/km²	
United States	US		1 2019	4,12E+13	338289857	9372610	6,1 36/km²	
United States	US		1 2020	4,17E+13	338289857	9372610	6,1 36/km²	

Ahora pasamos al análisis exploratorio de la tabla food.

1SELECT * FROM "labone"."food" limit 10;

SQLLn 1, Col 1

Ejecutar de nuevo

Explicar

Cancelar

Borrar

Crear

Volver a utilizar los resultados de la consulta

hasta hace 60 minutos

Resultados de la consulta

Estado de la consulta

Completado

Tiempo en cola: 59 ms

Tiempo de ejecución: 511 ms

Datos analizados: 7.58 KB

Resultados (10)

Copiar

Descargar resultados

QFilas de búsqueda

<1>

#	food product	land use change	animal feed	farm	processing	transport	packging	retail	total_emissions	eutrophying emissions per 1000kcal (gpo.eq per 1000kcal)	eutrophying emissions per 1000kcal (gpo.eq per 1000kcal)
1	Wheat & Rye (Bread)	0.1	0.0	0.8	0.2	0.1	0.1	0.1	1.4000000000000004		
2	Maize (Meal)	0.3	0.0	0.5	0.1	0.1	0.1	0.0	1.1		
3	Barley (Beer)	0.0	0.0	0.2	0.1	0.0	0.5	0.3	1.1		
4	Oatmeal	0.0	0.0	1.4	0.0	0.1	0.1	0.0	1.6	4.281357225	11.1
5	Rice	0.0	0.0	3.6	0.1	0.1	0.1	0.1	4.0	9.51437873	35.1
6	Potatoes	0.0	0.0	0.2	0.0	0.1	0.0	0.0	0.30000000000000004	4.7540983610000005	3.4
7	Cassava	0.6	0.0	0.2	0.0	0.1	0.0	0.0	0.9	0.708418891	0.6

Total de emisiones por producto alimenticio

1 SELECT "food product",

2 SUM("land use change" + "animal feed" + "farm" + "processing" + "transport" + "packging" + "retail") AS "total"

3 FROM "labone"."food"

4 GROUP BY "food product"

5 ORDER BY "total emissions" DESC;

SQLLn 2, Col 7

Ejecutar de nuevo

Explicar

Cancelar

Borrar

Crear

Resultados de la consulta

Estado de la consulta

Completado

Tiempo en cola: 68 ms

Resultados (43)

QFilas de búsqueda

#	food product	total emissions
1	Beef (beef herd)	59.599999999999994
2	Lamb & Mutton	24.5
3	Cheese	21.2
4	Beef (dairy herd)	21.099999999999998
5	Dark Chocolate	18.7
6	Coffee	16.500000000000004
7	Shrimps (farmed)	11.8
8	Palm Oil	7.6000000000000005
9	Pig Meat	7.2

Top 5 productos con mayor uso de tierra por kilogramo

1	SELECT food_product, land_use_per_kilogram	
2	FROM environment_db.food_emissions	
3	ORDER BY land_use_per_kilogram DESC	
4	LIMIT 5;	

SQL Ln 4, Col 9

Ejecutar de nuevo

Explicar

Cancelar

Borrar

Crear ▼

Resultados de la consulta

Estado de la consulta

Completado

Resultados (5)

Filas de búsqueda

# ▼	food_product ▼	land_use_per_kilogram
1	Lamb & Mutton	369.81
2	Beef (beef herd)	326.21
3	Cheese	87.79
4	Dark Chocolate	68.96
5	Beef (dairy herd)	43.24

Promedio de emisiones de gases de efecto invernadero por cada etapa de producción

1	SELECT AVG(land_use_change) AS avg_land_use_change,	
2	AVG(animal_feed) AS avg_animal_feed,	
3	AVG(farm) AS avg_farm,	
4	AVG(packaging) AS avg_packaging,	
5	AVG(processing) AS avg_processing,	
6	AVG(transport) AS avg_transport,	
7	AVG(retail) AS avg_retail	
8	FROM environment_db.food_emissions;	

SQL Ln 6, Col 33

Ejecutar de nuevo

Explicar

Cancelar

Borrar

Crear ▼

☐ Volver a utilizar los resultados de la consulta
 hasta hace 60 minutos

Resultados de la consulta

Estado de la consulta

Completado
 Tiempo en cola: 106 ms
 Tiempo de ejecución: 470 ms
 Datos analizados: 7.58 KB

Resultados (1)

Filas de búsqueda

Copiar

Descargar resultados

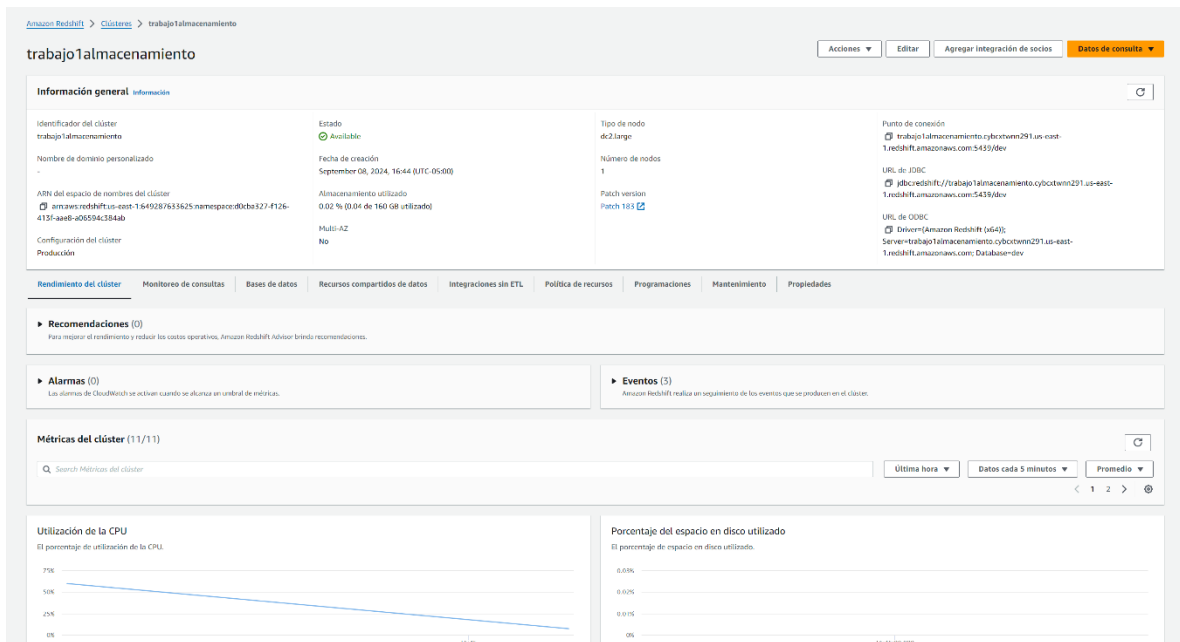
# ▼	avg_land_use_change ▼	avg_animal_feed ▼	avg_farm ▼	avg_processing ▼	avg_transport ▼	avg_packaging ▼	avg_retail ▼
1	1.26	0.45	3.47	0.25	0.20	0.27	0.07

Modelado y Análisis de Datos con Redshift

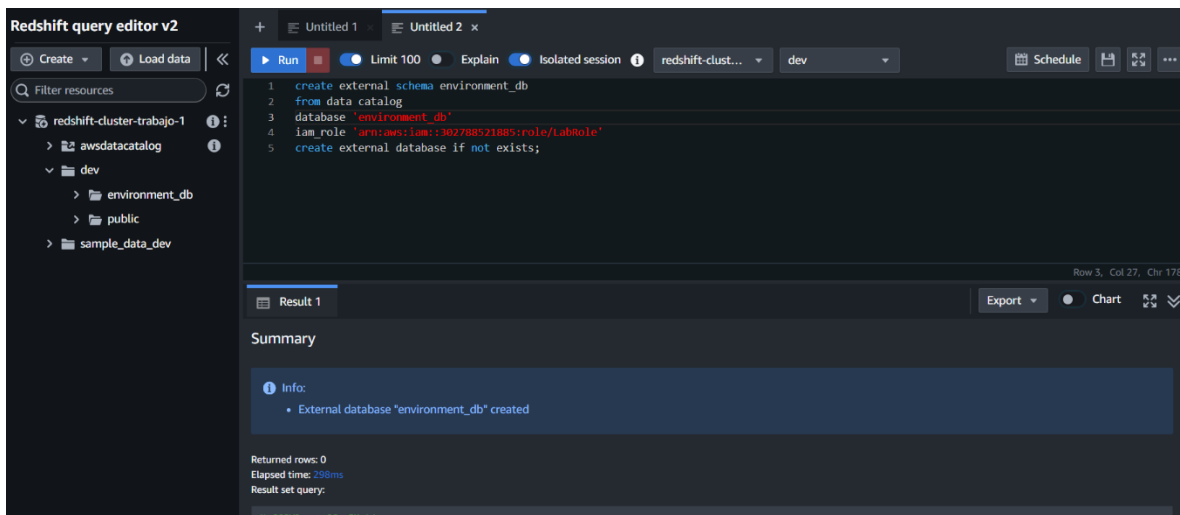
Para un análisis más avanzado, los datos fueron modelados y cargados en un Data Warehouse en Amazon Redshift:

1. **Modelado Multidimensional:** Se diseñaron tablas de hechos y dimensiones para permitir consultas complejas y análisis de tendencias en los datos de contaminación.
2. **Consultas en Redshift:** Se realizaron consultas sobre las tablas en Redshift, integrando datos desde S3 mediante Redshift Spectrum.

Primero se crea el cluster de RedShift



Creamos una nueva base de datos para poder realizar la subida de la información



Creamos la tabla para las emisiones de CO2 por país y año de la siguiente manera:

The screenshot shows a SQL IDE interface with a dark theme. At the top, there are tabs for 'Untitled 1', 'Untitled 2', 'CO2 creation', and 'Untitled 3'. Below the tabs is a toolbar with buttons for 'Run', 'Limit 100', 'Explain', 'Isolated session', and a dropdown menu showing 'redshift-clust...' and 'dev'. The main editor area contains SQL code for creating an external table named 'environment_db.co2_emissions'. The code includes a 'DROP TABLE IF EXISTS' statement, followed by a 'CREATE EXTERNAL TABLE' statement with various data types and a 'LOCATION' path. Below the code, there are tabs for 'Result 1' and 'Result 2'. The 'Summary' section shows 'Returned rows: 0', 'Elapsed time: 357ms', and 'Result set query:'. Below the summary, there is a code block with a comment and the 'DROP TABLE IF EXISTS' statement.

```
1 DROP TABLE IF EXISTS environment_db.co2_emissions;
2
3 CREATE EXTERNAL TABLE environment_db.co2_emissions (
4     country VARCHAR(255),
5     code VARCHAR(10),
6     calling_code BIGINT,
7     co2_emission_tons BIGINT,
8     population_2022 BIGINT,
9     area BIGINT,
10    percent_of_world VARCHAR(10),
11    density_km2 VARCHAR(50)
12 )
13 ROW FORMAT DELIMITED
14 FIELDS TERMINATED BY ','
15 STORED AS TEXTFILE
16 LOCATION 's3://trabajo-1-almac-recup-info/raw/country/'
17 TABLE PROPERTIES (
18     'numRows'='59620',
19     'skip.header.line.count'='1'
20 );
```

Result 1

Result 2

Summary

Returned rows: 0
Elapsed time: 357ms
Result set query:

```
/* RQEV2-DLPLYQfwsH */
DROP TABLE IF EXISTS environment_db.co2_emissions
```

Luego comprobamos que se haya subido correctamente los datos:

The screenshot shows the same SQL IDE interface. The 'Run' button is highlighted. The query in the editor is 'SELECT * FROM environment_db.co2_emissions limit 10;'. Below the query, there is a table with 10 rows and 7 columns. The columns are 'country', 'code', 'calling_code', 'co2_emission_tons', 'population_2022', and 'area'. The table shows data for Afghanistan with various codes and values.

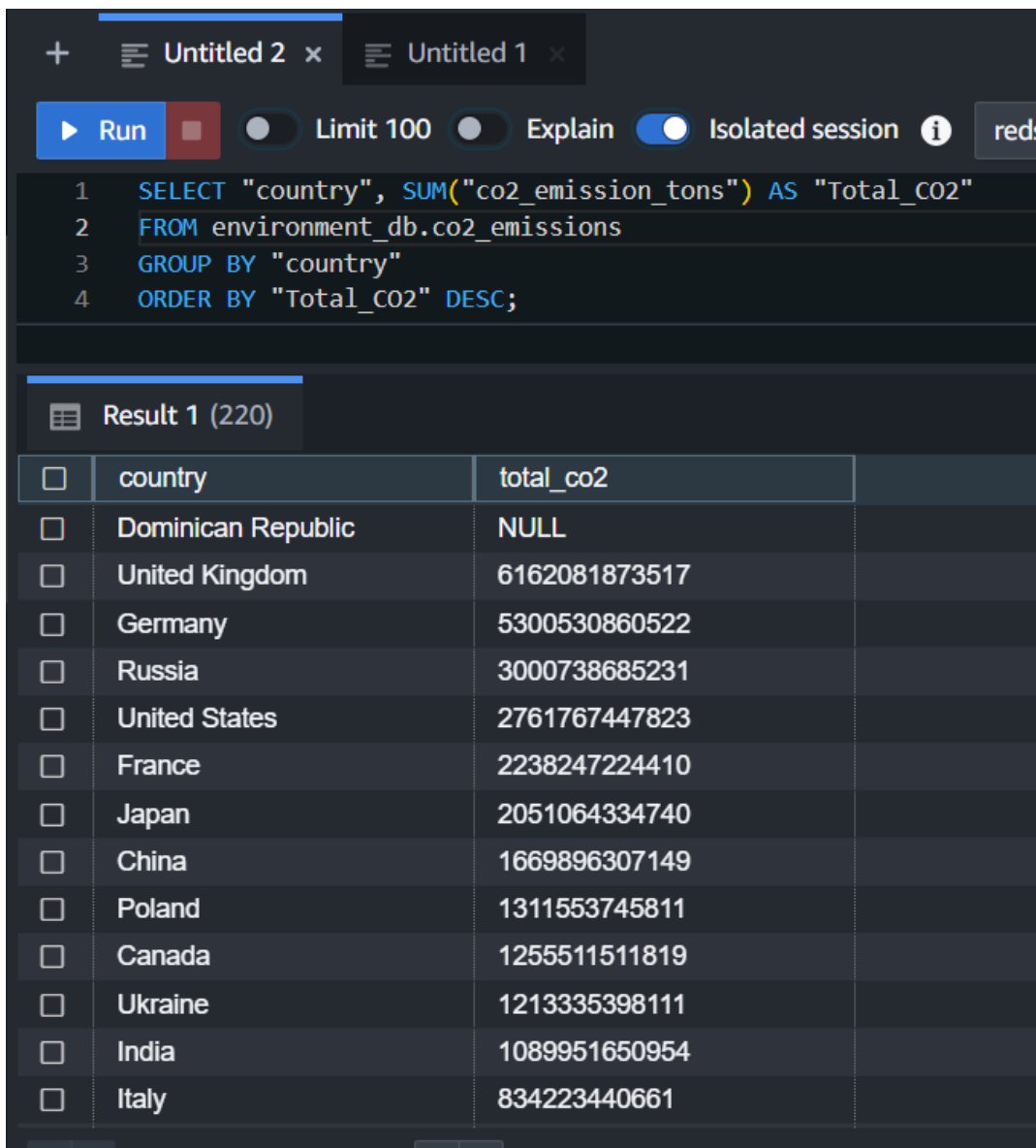
```
1 SELECT * FROM environment_db.co2_emissions limit 10;
```

Row 1, Col 53, Chr 52

Result 1 (10)

	country	code	calling_code	co2_emission_tons	population_2022	area
<input type="checkbox"/>	Afghanistan	AF	93	1750	0	41128771
<input type="checkbox"/>	Afghanistan	AF	93	1751	0	41128771
<input type="checkbox"/>	Afghanistan	AF	93	1752	0	41128771
<input type="checkbox"/>	Afghanistan	AF	93	1753	0	41128771
<input type="checkbox"/>	Afghanistan	AF	93	1754	0	41128771
<input type="checkbox"/>	Afghanistan	AF	93	1755	0	41128771
<input type="checkbox"/>	Afghanistan	AF	93	1756	0	41128771
<input type="checkbox"/>	Afghanistan	AF	93	1757	0	41128771
<input type="checkbox"/>	Afghanistan	AF	93	1758	0	41128771
<input type="checkbox"/>	Afghanistan	AF	93	1759	0	41128771

Además, podemos hacer consultas más complejas como lo hacíamos con Glue para hacer un análisis de los datos.



The screenshot shows a SQL query editor with two tabs: 'Untitled 2' and 'Untitled 1'. The query in 'Untitled 1' is as follows:

```
1 SELECT "country", SUM("co2_emission_tons") AS "Total_CO2"
2 FROM environment_db.co2_emissions
3 GROUP BY "country"
4 ORDER BY "Total_CO2" DESC;
```

Below the query, the 'Run' button is highlighted. To its right are toggle switches for 'Limit 100' (disabled), 'Explain' (disabled), and 'Isolated session' (enabled). A 'redshift' button is also visible.

The result of the query is displayed in a table titled 'Result 1 (220)'. The table has two columns: 'country' and 'total_co2'. The data is sorted in descending order of total CO2 emissions.

country	total_co2
Dominican Republic	NULL
United Kingdom	6162081873517
Germany	5300530860522
Russia	3000738685231
United States	2761767447823
France	2238247224410
Japan	2051064334740
China	1669896307149
Poland	1311553745811
Canada	1255511511819
Ukraine	1213335398111
India	1089951650954
Italy	834223440661

Ahora realizamos la creación y subida de los datos de CO2 por comida

```

DROP TABLE IF EXISTS environment_db.food_emissions;

CREATE EXTERNAL TABLE environment_db.food_emissions (
  food_product VARCHAR(255),
  land_use_change DECIMAL(10,2),
  animal_feed DECIMAL(10,2),
  farm DECIMAL(10,2),
  processing DECIMAL(10,2),
  transport DECIMAL(10,2),
  packaging DECIMAL(10,2),
  retail DECIMAL(10,2),
  total_emissions DECIMAL(10,2),
  eutrophying_emissions_per_1000kcal DECIMAL(10,2),
  eutrophying_emissions_per_kilogram DECIMAL(10,2),
  eutrophying_emissions_per_100g_protein DECIMAL(10,2),
  freshwater_withdrawals_per_1000kcal DECIMAL(10,2),
  freshwater_withdrawals_per_100g_protein DECIMAL(10,2),
  freshwater_withdrawals_per_kilogram DECIMAL(10,2),
  greenhouse_gas_emissions_per_1000kcal DECIMAL(10,2),
  greenhouse_gas_emissions_per_100g_protein DECIMAL(10,2),
  land_use_per_1000kcal DECIMAL(10,2),
  land_use_per_kilogram DECIMAL(10,2),
  land_use_per_100g_protein DECIMAL(10,2)
)
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
STORED AS TEXTFILE
LOCATION 's3://trabajo-1-almac-recup-info/raw/food/'
TABLE PROPERTIES (
  'numRows'='43',
  'skip.header.line.count'='1'
);

```

Validamos los datos.

1 `SELECT * FROM environment_db.food_emissions;`

Row 1, Col 45, Chr 44

Result 1 (43)

food_product	land_use_change	animal_feed	farm	processing	transport	packaging	retail
Wheat & Rye (Bread)	0.1	0	0.8	0.2	0.1	0.1	0.1
Maize (Meal)	0.3	0	0.5	0.1	0.1	0.1	0
Barley (Beer)	0	0	0.2	0.1	0	0.5	0.3
Oatmeal	0	0	1.4	0	0.1	0.1	0
Rice	0	0	3.6	0.1	0.1	0.1	0.1
Potatoes	0	0	0.2	0	0.1	0	0
Cassava	0.6	0	0.2	0	0.1	0	0
Cane Sugar	1.2	0	0.5	0	0.8	0.1	0
Beet Sugar	0	0	0.5	0.2	0.6	0.1	0
Other Pulses	0	0	1.1	0	0.1	0.4	0
Peas	0	0	0.7	0	0.1	0	0
Nuts	-2.1	0	2.1	0	0.1	0.1	0
Groundnuts	0.4	0	1.4	0.4	0.1	0.1	0
Soymilk	0.2	0	0.1	0.2	0.1	0.1	0.3
Tofu	1	0	0.5	0.8	0.2	0.2	0.3
Soybean Oil	3.1	0	1.5	0.3	0.3	0.8	0
Palm Oil	3.1	0	2.1	1.3	0.2	0.9	0
Sunflower Oil	0.1	0	2.1	0.2	0.2	0.9	0
Rapeseed Oil	0.2	0	2.3	0.2	0.2	0.8	0
Olive Oil	-0.4	0	4.3	0.7	0.5	0.9	0
Tomatoes	0.4	0	0.7	0	0.2	0.1	0
Onions & Leeks	0	0	0.2	0	0.1	0	0
Root Vegetables	0	0	0.2	0	0.1	0	0
Brassicas	0	0	0.3	0	0.1	0	0

Query ID 203612 Elapsed time: 6316 ms Total rows: 43

También podemos realizar consultas como lo hicimos con Glue

+ Untitled 2 x Untitled 1 x

Run Limit 100 Explain Isolated session redshift-clust... dev

```

1 SELECT food_product,
2 SUM(land_use_change + animal_feed + farm + processing + transport + packaging + retail) AS total_emissions
3 FROM environment_db.food_emissions
4 GROUP BY food_product
5 ORDER BY total_emissions DESC;

```

Result 1 (43)

food_product	total_emissions
Beef (beef herd)	59.6
Lamb & Mutton	24.5
Cheese	21.2
Beef (dairy herd)	21.1
Dark Chocolate	18.7
Coffee	16.5
Shrimps (farmed)	11.8
Palm Oil	7.6
Pig Meat	7.2
Poultry Meat	6.1
Olive Oil	6
Soybean Oil	6
Fish (farmed)	5.1
Eggs	4.5
Rice	4
Rapeseed Oil	3.7
Sunflower Oil	3.5
Tofu	3
Milk	2.8
Cane Sugar	2.6
Groundnuts	2.4

Implementación de Clúster EMR para Procesamiento con Spark

Finalmente, se desplegó un clúster de EMR en AWS para realizar análisis exploratorios utilizando PySpark:

1. **Configuración del Clúster EMR:** Se configuró un clúster EMR con soporte para PySpark y Jupyter Notebooks, permitiendo el análisis interactivo de los datos.
2. **Análisis Exploratorio:** Se ejecutaron notebooks de Jupyter que procesaron y analizaron los datos, generando visualizaciones y modelos preliminares.

Primero se crea el Cluster en EMR que venga con entre otras cosas, Spark y JupyterHub para soportar PySpark y Jupyter Notebooks

The screenshot displays the AWS EMR console interface for a cluster named "ClusterTrabajo1AlmacenamientoRecuperacionInformacion". The cluster is in the "Comenzando" (Starting) state. The console shows various configuration details, including the cluster ID, applications installed (HBase, Hive, Hue, Jupyter, Mahout, Pig, Tez, ZooKeeper), and the state of the cluster. The "Terminación del clúster y reemplazo de nodos" section shows the cluster is not yet terminated. The "Red y seguridad" section shows the cluster is using the default VPC and security groups.

Propiedades	Acciones de arranque	Instancias (hardware)	Paras	Aplicaciones	Configuraciones	Monitorización	Eventos	Etiquetas (0)
Resumen								
Información del clúster								
ID del clúster j-1H06G5V7TTX9								
Configuración del clúster								
Grupos de instancias								
Capacidad								
1 Primary (Principal)								
1 Tera								
Aplicaciones								
Versión de Amazon EMR emr-6.15.0								
Aplicaciones instaladas								
HBase 2.4.17, HCatalog 3.1.3, Hive 3.13.0, Hue 4.11.0, JupyterEnterpriseGateway 2.6.0, Mahout 0.12.3, Pig 0.16.0, Tez 0.8.4, ZooKeeper 3.5.10								
Administración de clústeres								
Destino del registro en Amazon S3 aws-logs-643287633625-us-east-1:elasticmapreduce								
DNS público del nodo principal -								
Estado y hora								
Estado Comenzando								
Hora de creación 8 de septiembre de 2024 23:02 (UTC-05:00)								
Tiempo transcurrido 0 segundos								
Registros de clúster								
Archivar los archivos de registro en Amazon S3 Activado								
Ubicación de Amazon S3 s3://aws-logs-643287633625-us-east-1:elasticmapreduce/								
Terminación del clúster y reemplazo de nodos								
Opción de terminación Terminar manualmente el clúster								
Protección contra la terminación Desactivado								
Tiempo de inactividad -								
Reemplazo de nodos en mal estado Activado								
Red y seguridad								
Red Virtual Private Cloud (VPC) vpc-0b3a70059a6a3a54								
Subredes y zonas de disponibilidad (AZ) subnet-00b6705105a6f132 us-east-1d								
Grupos de seguridad de EC2 (firewall)								
Configuración de seguridad								
Configuración de seguridad Ninguna								
Par de claves de EC2 us-east-1d								
Permisos								
Pol de servicio para Amazon EMR EMR_DefaultRole								
Pol de instancia EC2 EMR_EC2_DefaultRole								
Pol de escalamiento automático personalizada AutoScalingRole								

Luego desde la aplicación de JupyterHub, iniciamos sección, con las credenciales por defecto en AWS y empezamos a crear nuestro notebook

In [2]: # 1. Cargar Datos desde un Bucket de S3 (CSV a PySpark DataFrame)

```
In [1]: from pyspark.sql import SparkSession

# Crear la sesión de Spark
spark = SparkSession.builder.appName("CO2 Emissions from S3").getOrCreate()

# Cargar el archivo CSV desde el bucket de S3
df = spark.read.csv("s3://tduquegtrabajo1-almac/raw/country/CO2 emission by countries.csv", header=True, inferSchema=True)

# Mostrar las primeras filas
df.show(5)

# Mostrar el esquema
df.printSchema()
```

Starting Spark application

ID	YARN Application ID	Kind	State	Spark UI	Driver log	User	Current session?
0	application_1725855336873_0001	pyspark	idle	Link	Link	None	✓

SparkSession available as 'spark'.

Country	Code	Calling Code	Year	CO2 emission (Tons)	Population(2022)	Area	% of World	Density(km2)
Afghanistan	AF	93	1750	0.0	41128771	652230	0.40%	63/km2
Afghanistan	AF	93	1751	0.0	41128771	652230	0.40%	63/km2
Afghanistan	AF	93	1752	0.0	41128771	652230	0.40%	63/km2
Afghanistan	AF	93	1753	0.0	41128771	652230	0.40%	63/km2
Afghanistan	AF	93	1754	0.0	41128771	652230	0.40%	63/km2

only showing top 5 rows

```
root
|-- Country: string (nullable = true)
|-- Code: string (nullable = true)
|-- Calling Code: string (nullable = true)
|-- Year: integer (nullable = true)
|-- CO2 emission (Tons): double (nullable = true)
|-- Population(2022): integer (nullable = true)
|-- Area: integer (nullable = true)
|-- % of World: string (nullable = true)
|-- Density(km2): string (nullable = true)
```

Luego de realizar los diferentes análisis a la base de datos estos se guardan automáticamente en s3 como un archivo parquet, el notebook se encuentra en el GitHub con más detalle: <https://github.com/tduqueg/Trabajo1>

```
|United States| US| 1|2020| 4.1/11| 33828985|/95/2010| 6.10%| 36/Km|
only showing top 1 row
```

8. Agregar una Columna para Emisiones de CO2 en Millones de Toneladas

```
In [18]: # Crear una nueva columna que convierta Las emisiones de kilotoneladas a millones de toneladas
df = df.withColumn("co2_emission_mt", df.co2_emission_kt / 1000)

# Mostrar Las primeras filas con La nueva columna
df.show(5)
```

```
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|country_name|Code|Calling Code|year|co2_emission_kt|Population(2022)|Area|% of World|Density(km2)|co2_emission_mt|
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|Afghanistan|AF|93|1750|0.0|41128771|652230|0.40%|63/km|0.0|
|Afghanistan|AF|93|1751|0.0|41128771|652230|0.40%|63/km|0.0|
|Afghanistan|AF|93|1752|0.0|41128771|652230|0.40%|63/km|0.0|
|Afghanistan|AF|93|1753|0.0|41128771|652230|0.40%|63/km|0.0|
|Afghanistan|AF|93|1754|0.0|41128771|652230|0.40%|63/km|0.0|
only showing top 5 rows
```

9. Calcular el Total de Emisiones de CO2 por País Entre 1990 y 2020

```
In [17]: # Filtrar Los datos entre 1990 y 2020
df_filtered = df.filter((df.year >= 1990) & (df.year <= 2020))

# Calcular el total de emisiones por país en ese rango de años
total_co2_by_country = df_filtered.groupBy("country_name").sum("co2_emission_mt")

# Mostrar el total de emisiones de CO2 por país
total_co2_by_country.show(10)
```

```
+-----+-----+
|country_name|sum(co2_emission_mt)|
+-----+-----+
|Guyana|2195725.491|
|Turkey|1.83696123205E8|
|Saint Helena|6615.16|
|Argentina|1.8167600238599998E8|
|Angola|1.0040569527999999E7|
|Albania|6855301.317|
|Nicaragua|3342924.3349999995|
|Peru|3.9960106194999985E7|
|China|3.571127691988E9|
|Somalia|738506.1030000001|
only showing top 10 rows
```

10. Guardar los Resultados en S3 en Formato Parquet

```
In [19]: # Guardar Los resultados en formato Parquet en el bucket de S3
total_co2_by_country.write.mode("overwrite").parquet("s3://tduquegtrabajo1-almac/trusted/country/co2_emissions_parquet")
```

Amazon S3 > Buckets > tduquegtrabajo1-almac > trusted > country > co2_emissions_parquet/

co2_emissions_parquet/

Copiar URL de S3

Objetos

Propiedades

Objetos (2) [información](#)

Los objetos son las entidades fundamentales que se almacenan en Amazon S3. Puede utilizar el [inventario de Amazon S3](#) para obtener una lista de todos los objetos de su bucket. Para que otras personas obtengan acceso a sus objetos, tendrá que concederles permisos de forma explícita. [Más información](#)

Buscar objetos por prefijo

<input type="checkbox"/>	Nombre	Tipo	Última modificación	Tamaño	Clase de almacenamiento
<input checked="" type="checkbox"/>	success	-	8 Sep 2024 11:31:16 PM -05		0 B Estándar
<input checked="" type="checkbox"/>	part-00000-1665c766-83c6-4f69-8976-ar85df6a23cb-0000.parquet	parquet	8 Sep 2024 11:31:16 PM -05	4.7 KB	Estándar