

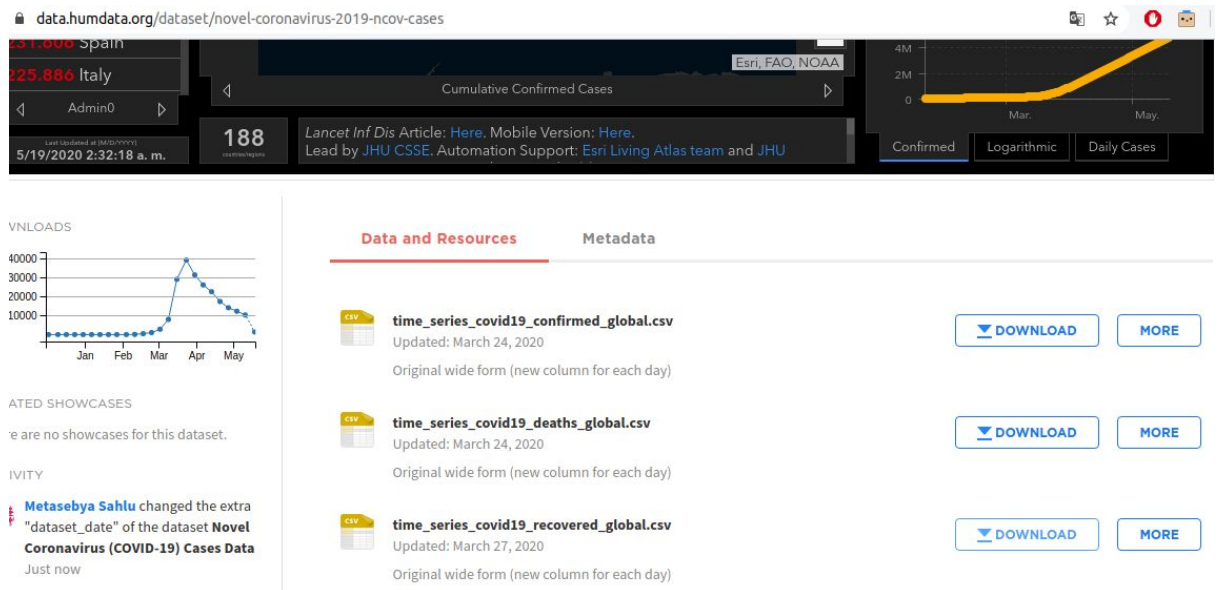
## Trabajo 3 Análisis exploratorio en datasets de COVID-19

### Paso 1->

En este punto vamos, iremos a las páginas dadas por el docente y descargamos los datos en formato .csv

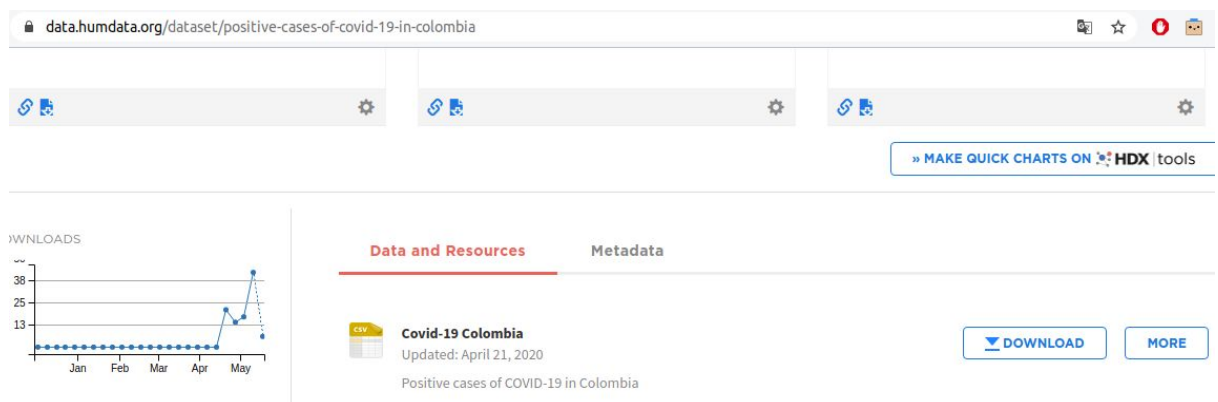
1. <https://data.humdata.org/dataset/novel-coronavirus-2019-ncov-cases>

#### Casos globales de covid19



2. <https://data.humdata.org/dataset/positive-cases-of-covid-19-in-colombia>

#### Casos de colombia



3. <https://www.ins.gov.co/Noticias/Paginas/Coronavirus.aspx>

#### Más datos de colombia

## COVID-19 Colombia | Reporte 18-05-2020 6:30pm

Descargar DataSet de casos



Distribución por edad

Distribución por sexo

¡Hola! Pregúntame tus dudas sobre el nuevo Coronavirus

Este enlace de descarga nos lleva a una página del gobierno

datos.gov.co/Salud-y-Protección-Social/Casos-positivos-de-COVID-19-en-Colombia/gt2j-8ykr/data

GOV.CO DATOS ABIERTOS

Casos positivos de COVID-19 en Colombia

FE DE ERRATAS 18-05-2020: Caso 14672, ajuste departamento: figura de Cartagena, se ajusta a Bolívar

ID de caso	Fecha d...	Código D...	Ciudad d...	Departa...	atención	Edad	Sexo	Tipo	Estado	País de ...	FIS	Fecha d...	Fecha
1	2020-03-02T...	11001	Bogotá D.C.	Bogotá D.C.	Recuperado	19	F	Importado	Leve	Italia	2020-02-27T...	--	2020-03-02T...
2	2020-03-06T...	76111	Guadalajara ...	Valle del Cauca	Recuperado	34	M	Importado	Leve	España	2020-03-04T...	--	2020-03-06T...
3	2020-03-07T...	5001	Medellín	Antioquia	Recuperado	50	F	Importado	Leve	España	2020-02-29T...	--	2020-03-07T...
4	2020-03-09T...	5001	Medellín	Antioquia	Recuperado	55	M	Relacionado	Leve	Colombia	2020-03-06T...	--	2020-03-09T...
5	2020-03-09T...	5001	Medellín	Antioquia	Recuperado	25	M	Relacionado	Leve	Colombia	2020-03-08T...	--	2020-03-09T...
6	2020-03-10T...	5360	Itagüí	Antioquia	Recuperado	27	F	Relacionado	Leve	Colombia	2020-03-06T...	--	2020-03-10T...
7	2020-03-08T...	13001	Cartagena de...	Cartagena D...	Recuperado	85	F	Importado	Leve	Estados Unid...	2020-03-02T...	--	2020-03-08T...
8	2020-03-09T...	11001	Bogotá D.C.	Bogotá D.C.	Recuperado	22	F	Importado	Leve	España	2020-03-06T...	--	2020-03-09T...
9	2020-03-08T...	11001	Bogotá D.C.	Bogotá D.C.	Recuperado	28	F	Importado	Leve	España	2020-03-07T...	--	2020-03-08T...
10	2020-03-12T...	11001	Bogotá D.C.	Bogotá D.C.	Recuperado	36	F	Importado	Leve	España	2020-03-06T...	--	2020-03-12T...
11	2020-03-11T...	11001	Bogotá D.C.	Bogotá D.C.	Recuperado	42	F	Importado	Leve	España	2020-03-06T...	--	2020-03-11T...

< Anterior Siguiente >

Mostrando filas 1-100 de 16.295

## Parte 2->

entro a mi bucket en amazon y creo la carpeta del proyecto 3, donde agregare los datasets, previamente descargados en una subcarpeta llamada datasets

# sebasawsbucket

Información general

Propiedades

Permisos

Administración



Escriba un prefijo y pulse Intro para buscar. Pulse ESC para borrar.



Cargar



Crear carpeta

Descargar

Acciones ▾



Nombre ▾



Proyecto 3|

Cuando se crea una carpeta, la consola de S3 crea un objeto con el nombre anterior seguido del sufijo "/" y ese objeto se muestra como una carpeta en la consola de S3. Elija la configuración de cifrado para el objeto:



Ninguna (Utilizar la configuración del bucket)



AES-256

Utilizar cifrado del lado del servidor con claves administradas por Amazon S3 (SSE-S3)



AWS-KMS

Utilizar cifrado del lado del servidor con claves administradas por AWS KMS (SSE-KMS)

Guardar

Cancelar

5 Archivos Tamaño: 6.1 MB Ruta de destino: sebasawsbucket/Proyecto 3/datasets/

Para cargar un archivo de más de 160 GB, utilice la interfaz de línea de comandos (CLI) de AWS, el SDK de AWS o las API de REST de Amazon S3. [Más información](#)

+ Añadir más archivos



Casos\_positivos\_de\_COVID-19\_en\_Colombia.csv

- 2.9 MB



covid19\_colombia.csv

- 2.9 MB



time\_series\_covid19\_confirmed\_global.csv

- 105.9 KB



time\_series\_covid19\_deaths\_global.csv

- 79.8 KB



time\_series\_covid19\_recovered\_global.csv

- 89.8 KB



Cargar

Siguiente

## Paso 3 y 4-> analitica de datos usando pyspark

### Create database

#### Database details

Create a database in the Data Catalog.

##### Name

Names may contain letters (A-Z), numbers (0-9), hyphens (-), or underscores (\_), and must be less than 256 characters long.

##### Location - *optional*

Choose an Amazon S3 path for this database, which eliminates the need to grant data location permissions on catalog table paths that are this location's children

Browse

##### Description - *optional*

Descriptions can be up to 2048 characters long.

##### Default permissions for newly created tables

This setting maintains existing AWS Glue data catalog behavior. You can still set individual permissions, which will take effect when you revoke the Super permission from IAMAllowedPrincipals. See [Changing Default Settings for Your Data Lake](#).

☐ Use only IAM access control for new tables in this database

### Register location

#### Amazon S3 location

Register an Amazon S3 path as the storage location for your data lake.

##### Amazon S3 path

Choose an Amazon S3 path for your data lake.

Browse

##### Review location permissions - *strongly recommended*

Registering the selected location may result in your users gaining access to data already at that location. Before registering a location, we recommend that you review existing location permissions on resources in that location.

Review location permissions

##### IAM role

To add or update data, Lake Formation needs read/write access to the chosen Amazon S3 path. Choose a role that you know has permission to do this, or choose the **AWSServiceRoleForLakeFormationDataAccess** service-linked role. When you register the first Amazon S3 path, the service-linked role and a new inline policy are created on your behalf. Lake Formation adds the first path to the inline policy and attaches it to the service-linked role. When you register subsequent paths, Lake Formation adds the path to the existing policy.

CancelRegister location

## Asigne un nombre al bloc de notas y configúrelo

Asigne un nombre al bloc de notas de Jupyter administrado por EMR, elija o cree un clúster y person desea. [Más información](#)

Nombre del bloc de notas\*

Proyecto3

Los nombres solo pueden contener letras (a-z), números (0-9), guiones (-) o caracteres de

Descripción

Analisis edescriptivo y exploratorio

256 caracteres como máximo.

Ubicación del bloc de notas\* Choose an S3 location where files for this notebook are saved.

☐ Use a location that EMR creates

☒ Choose an existing S3 location in us-east-1

s3://sebasawsbucket/

► Repositorio de Git

Enlace a un repositorio de Git para guardar el bloc de notas en un entorno con control de versiones

► Etiquetas

\* Obligatorio

Cancelar

Crear un bloc de notas

Amazon EMR

Clústeres

Configuraciones de seguridad

Bloqueo de acceso público

Subredes de la VPC

Eventos

Blocs de notas

Git repositories

Bloc de notas: Proyecto3 Listo Notebook is ready to run jobs on cluster j-1N4KJKPETCSQ7.

Abrir en JupyterLab

Abrir en Jupyter

Detener

Eliminar

Bloc de notas

ID del bloc de notas: e-C28TE585VXF73LIRFZPDTXNKZ

Descripción: Analisis descriptivo y exploratorio

Última modificación: Hace 8 segundos

Modificado por última vez por: ...assumed-role/vocstartsoft/user589291=sgomezp1@eafit.edu.co

Fecha de creación: 2020-05-19 04:17 (UTC-5)

### #Análisis

Durante los pantallazos se verá en Markdown los comentarios referentes a las etapas del desarrollo y lo que se buscaba mostrar, al final de cada paso podremos ver notas en forma de **Patrones y de Conclusiones**



# Proyecto 3

Paso 0-> Importar Librerías encesarias

```
[1]: #import SparkSession
from pyspark.sql import SparkSession
from pyspark.sql.types import StringType, DoubleType, IntegerType
# UDF
from pyspark.sql.functions import udf
#pandas udf
from pyspark.sql.functions import pandas_udf, PandasUDFType
```

## ► Spark Job Progress

Starting Spark application

ID	YARN Application ID	Kind	State	Spark UI	Driver log	Current session?
2	application_1589880021320_0003	pyspark	idle	<a href="#">Link</a>	<a href="#">Link</a>	✓

SparkSession available as 'spark'.

```
[2]: #create spar session object
spark=SparkSession.builder.appName('data_processing').getOrCreate()
```

Paso 1-> Convertir los csv en dataframes

```
[3]: # Cargando los csv
#cpcc = Casos positivos covid Colombia
cpcc=spark.read.csv('s3://sebasawsbucket/Proyecto 3/datasets/Casos_positivos_de_COVID-19_en_Colombia.csv',inferSchema=True)
#dcc = Datos covid Colombia
dcc=spark.read.csv('s3://sebasawsbucket/Proyecto 3/datasets/covid19_colombia.csv',inferSchema=True,header=True)
#cccg = Casos covid confirmados globalmente
cccg=spark.read.csv('s3://sebasawsbucket/Proyecto 3/datasets/time_series_covid19_confirmed_global.csv',inferSchema=True)
#ccmg = Casos covid muertos globalmente
ccmg=spark.read.csv('s3://sebasawsbucket/Proyecto 3/datasets/time_series_covid19_deaths_global.csv',inferSchema=True)
#ccrg = Casos covid recuperados globalmente
ccrg=spark.read.csv('s3://sebasawsbucket/Proyecto 3/datasets/time_series_covid19_recovered_global.csv',inferSchema=True)
```

## ► Spark Job Progress

Paso 2-> Visualizar los nombres de las columnas y encontrar patrones

```
[4]: #Columnas cpcc = Casos positivos covid Colombia
cpcc.columns
```

```
['ID de caso', 'Fecha de notificación', 'Codigo DIVIPOLA', 'Ciudad de ubicación', 'Departamento o Distrito ', 'atención', 'Edad', 'Sexo', 'Tipo', 'Estado', 'País de procedencia', 'FIS', 'Fecha de muerte', 'Fecha diagnostico', 'Fecha recuperado', 'Fecha de reporte web']
```

```
[5]: #Columnas dcc = Datos covid Colombia
dcc.columns
```

```
['ID de caso', 'Fecha de notificación', 'Codigo DIVIPOLA', 'Ciudad de ubicación', 'Departamento o Distrito ', 'atención', 'Edad', 'Sexo', 'Tipo', 'Estado', 'País de procedencia', 'FIS', 'Fecha de muerte', 'Fecha diagnostico', 'Fecha recuperado', 'Fecha de reporte web']
```

Patron-> Observamos que en las dos primeras tablas los datos poseen la misma estructura, esto nos habla de los bien que estan articuladas las multiples instituciones que producen estos datos ya que los datos fueron tomados de fuentes diferentes, el cambio más sustancial debe ser la fecha del reporte.

**Patron->** Observamos que en las dos primeras tablas los datos poseen la misma estructura, esto nos habla de los bien que estan articuladas las múltiples instituciones que producen estos datos ya que los datos fueron tomados de fuentes diferentes, el cambio más sustancial debe ser la fecha del reporte.

```
[6]: #Columnas cccg = Casos covid confirmados globalmente
      cccg.columns
```

```
['Province/State', 'Country/Region', 'Lat', 'Long', '1/22/20', '1/23/20', '1/24/20', '1/25/20', '1/26/20', '1/27/20', '1/28/20', '1/29/20', '1/30/20', '1/31/20', '2/1/20', '2/2/20', '2/3/20', '2/4/20', '2/5/20', '2/6/20', '2/7/20', '2/8/20', '2/9/20', '2/10/20', '2/11/20', '2/12/20', '2/13/20', '2/14/20', '2/15/20', '2/16/20', '2/17/20', '2/18/20', '2/19/20', '2/20/20', '2/21/20', '2/22/20', '2/23/20', '2/24/20', '2/25/20', '2/26/20', '2/27/20', '2/28/20', '2/29/20', '3/1/20', '3/2/20', '3/3/20', '3/4/20', '3/5/20', '3/6/20', '3/7/20', '3/8/20', '3/9/20', '3/10/20', '3/11/20', '3/12/20', '3/13/20', '3/14/20', '3/15/20', '3/16/20', '3/17/20', '3/18/20', '3/19/20', '3/20/20', '3/21/20', '3/22/20', '3/23/20', '3/24/20', '3/25/20', '3/26/20', '3/27/20', '3/28/20', '3/29/20', '3/30/20', '3/31/20', '4/1/20', '4/2/20', '4/3/20', '4/4/20', '4/5/20', '4/6/20', '4/7/20', '4/8/20', '4/9/20', '4/10/20', '4/11/20', '4/12/20', '4/13/20', '4/14/20', '4/15/20', '4/16/20', '4/17/20', '4/18/20', '4/19/20', '4/20/20', '4/21/20', '4/22/20', '4/23/20', '4/24/20', '4/25/20', '4/26/20', '4/27/20', '4/28/20', '4/29/20', '4/30/20', '5/1/20', '5/2/20', '5/3/20', '5/4/20', '5/5/20', '5/6/20', '5/7/20', '5/8/20', '5/9/20', '5/10/20', '5/11/20', '5/12/20', '5/13/20', '5/14/20', '5/15/20', '5/16/20', '5/17/20', '5/18/20']
```

```
[7]: #Columnas ccmg = Casos covid muertos globalmente
      ccmg.columns
```

```
['Province/State', 'Country/Region', 'Lat', 'Long', '1/22/20', '1/23/20', '1/24/20', '1/25/20', '1/26/20', '1/27/20', '1/28/20', '1/29/20', '1/30/20', '1/31/20', '2/1/20', '2/2/20', '2/3/20', '2/4/20', '2/5/20', '2/6/20', '2/7/20', '2/8/20', '2/9/20', '2/10/20', '2/11/20', '2/12/20', '2/13/20', '2/14/20', '2/15/20', '2/16/20', '2/17/20', '2/18/20', '2/19/20', '2/20/20', '2/21/20', '2/22/20', '2/23/20', '2/24/20', '2/25/20', '2/26/20', '2/27/20', '2/28/20', '2/29/20', '3/1/20', '3/2/20', '3/3/20', '3/4/20', '3/5/20', '3/6/20', '3/7/20', '3/8/20', '3/9/20', '3/10/20', '3/11/20', '3/12/20', '3/13/20', '3/14/20', '3/15/20', '3/16/20', '3/17/20', '3/18/20', '3/19/20', '3/20/20', '3/21/20', '3/22/20', '3/23/20', '3/24/20', '3/25/20', '3/26/20', '3/27/20', '3/28/20', '3/29/20', '3/30/20', '3/31/20', '4/1/20', '4/2/20', '4/3/20', '4/4/20', '4/5/20', '4/6/20', '4/7/20', '4/8/20', '4/9/20', '4/10/20', '4/11/20', '4/12/20', '4/13/20', '4/14/20', '4/15/20', '4/16/20', '4/17/20', '4/18/20', '4/19/20', '4/20/20', '4/21/20', '4/22/20', '4/23/20', '4/24/20', '4/25/20', '4/26/20', '4/27/20', '4/28/20', '4/29/20', '4/30/20', '5/1/20', '5/2/20', '5/3/20', '5/4/20', '5/5/20', '5/6/20', '5/7/20', '5/8/20', '5/9/20', '5/10/20', '5/11/20', '5/12/20', '5/13/20', '5/14/20', '5/15/20', '5/16/20', '5/17/20', '5/18/20']
```

```
[8]: #Columnas ccrg = Casos covid recuperados globalmente
      ccrg.columns
```

```
['Province/State', 'Country/Region', 'Lat', 'Long', '1/22/20', '1/23/20', '1/24/20', '1/25/20', '1/26/20', '1/27/20', '1/28/20', '1/29/20', '1/30/20', '1/31/20', '2/1/20', '2/2/20', '2/3/20', '2/4/20', '2/5/20', '2/6/20', '2/7/20', '2/8/20', '2/9/20', '2/10/20', '2/11/20', '2/12/20', '2/13/20', '2/14/20', '2/15/20', '2/16/20', '2/17/20', '2/18/20', '2/19/20', '2/20/20', '2/21/20', '2/22/20', '2/23/20', '2/24/20', '2/25/20', '2/26/20', '2/27/20', '2/28/20', '2/29/20', '3/1/20', '3/2/20', '3/3/20', '3/4/20', '3/5/20', '3/6/20', '3/7/20', '3/8/20', '3/9/20', '3/10/20', '3/11/20', '3/12/20', '3/13/20', '3/14/20', '3/15/20', '3/16/20', '3/17/20', '3/18/20', '3/19/20', '3/20/20', '3/21/20', '3/22/20', '3/23/20', '3/24/20', '3/25/20', '3/26/20', '3/27/20', '3/28/20', '3/29/20', '3/30/20', '3/31/20', '4/1/20', '4/2/20', '4/3/20', '4/4/20', '4/5/20', '4/6/20', '4/7/20', '4/8/20', '4/9/20', '4/10/20', '4/11/20', '4/12/20', '4/13/20', '4/14/20', '4/15/20', '4/16/20', '4/17/20', '4/18/20', '4/19/20', '4/20/20', '4/21/20', '4/22/20', '4/23/20', '4/24/20', '4/25/20', '4/26/20', '4/27/20', '4/28/20', '4/29/20', '4/30/20', '5/1/20', '5/2/20', '5/3/20', '5/4/20', '5/5/20', '5/6/20', '5/7/20', '5/8/20', '5/9/20', '5/10/20', '5/11/20', '5/12/20', '5/13/20', '5/14/20', '5/15/20', '5/16/20', '5/17/20', '5/18/20']
```

Patron-> Ya que estos datos son tomados de la organización mundial para la salud, es entendible que manejen las mismas fechas para los 3 formatos.

**Patron->** Ya que estos datos son tomados de la organización mundial para la salud, es entendible que manejen las mismas fechas para los 3 formatos.

### Paso 3-> Visualizar la Cantidad de Registros en los dataframes filas\*columnas

```
[9]: #cpcc = Casos positivos covid Colombia  
print((cpcc.count(),len(cpcc.columns)))
```

▸ Spark Job Progress

(16295, 16)

```
[10]: #dcc = Datos covid Colombia  
print((dcc.count(),len(dcc.columns)))
```

▸ Spark Job Progress

(16296, 16)

```
[11]: #cccg = Casos covid confirmados globalmente  
print((cccg.count(),len(cccg.columns)))
```

▸ Spark Job Progress

(266, 122)

```
[12]: #ccmg = Casos covid muertos globalmente  
print((ccmg.count(),len(ccmg.columns)))
```

▸ Spark Job Progress

(266, 122)

```
[13]: #ccrg = Casos covid recuperados globalmente  
print((ccrg.count(),len(ccrg.columns)))
```

▸ Spark Job Progress

(253, 122)



## Paso 4-> Visualizar la estructura de los datos

```
[14]: #cpcc = Casos positivos covid Colombia  
cpcc.printSchema()
```

```
root  
|-- ID de caso: integer (nullable = true)  
|-- Fecha de notificación: timestamp (nullable = true)  
|-- Codigo DIVIPOLA: integer (nullable = true)  
|-- Ciudad de ubicación: string (nullable = true)  
|-- Departamento o Distrito : string (nullable = true)  
|-- atención: string (nullable = true)  
|-- Edad: integer (nullable = true)  
|-- Sexo: string (nullable = true)  
|-- Tipo: string (nullable = true)  
|-- Estado: string (nullable = true)  
|-- País de procedencia: string (nullable = true)  
|-- FIS: string (nullable = true)  
|-- Fecha de muerte: string (nullable = true)  
|-- Fecha diagnostico: timestamp (nullable = true)  
|-- Fecha recuperado: string (nullable = true)  
|-- fecha reporte web: timestamp (nullable = true)
```

```
[15]: #dcc = Datos covid Colombia  
dcc.printSchema()
```

```
root  
|-- ID de caso: string (nullable = true)  
|-- Fecha de notificación: string (nullable = true)  
|-- Codigo DIVIPOLA: string (nullable = true)  
|-- Ciudad de ubicación: string (nullable = true)  
|-- Departamento o Distrito : string (nullable = true)  
|-- atención: string (nullable = true)  
|-- Edad: string (nullable = true)  
|-- Sexo: string (nullable = true)  
|-- Tipo: string (nullable = true)  
|-- Estado: string (nullable = true)  
|-- País de procedencia: string (nullable = true)  
|-- FIS: string (nullable = true)  
|-- Fecha de muerte: string (nullable = true)  
|-- Fecha diagnostico: string (nullable = true)  
|-- Fecha recuperado: string (nullable = true)  
|-- fecha reporte web: string (nullable = true)
```

El 17 y 18, son exactamente iguales en estructura, por eso no anexare esos pantallazos.

```
[16]: #cccg = Casos covid confirmados globalmente
      ccg.printSchema()
```

```
root
 |-- Province/State: string (nullable = true)
 |-- Country/Region: string (nullable = true)
 |-- Lat: double (nullable = true)
 |-- Long: double (nullable = true)
 |-- 1/22/20: integer (nullable = true)
 |-- 1/23/20: integer (nullable = true)
 |-- 1/24/20: integer (nullable = true)
 |-- 1/25/20: integer (nullable = true)
 |-- 1/26/20: integer (nullable = true)
 |-- 1/27/20: integer (nullable = true)
 |-- 1/28/20: integer (nullable = true)
 |-- 1/29/20: integer (nullable = true)
 |-- 1/30/20: integer (nullable = true)
 |-- 1/31/20: integer (nullable = true)
 |-- 2/1/20: integer (nullable = true)
 |-- 2/2/20: integer (nullable = true)
 |-- 2/3/20: integer (nullable = true)
 |-- 2/4/20: integer (nullable = true)
 |-- 2/5/20: integer (nullable = true)
 |-- 2/6/20: integer (nullable = true)
 |-- 2/7/20: integer (nullable = true)
 |-- 2/8/20: integer (nullable = true)
 |-- 2/9/20: integer (nullable = true)
```

```
|-- 2/10/20: integer (nullable = true)
|-- 2/11/20: integer (nullable = true)
|-- 2/12/20: integer (nullable = true)
|-- 2/13/20: integer (nullable = true)
|-- 2/14/20: integer (nullable = true)
|-- 2/15/20: integer (nullable = true)
|-- 2/16/20: integer (nullable = true)
|-- 2/17/20: integer (nullable = true)
|-- 2/18/20: integer (nullable = true)
|-- 2/19/20: integer (nullable = true)
|-- 2/20/20: integer (nullable = true)
|-- 2/21/20: integer (nullable = true)
|-- 2/22/20: integer (nullable = true)
|-- 2/23/20: integer (nullable = true)
|-- 2/24/20: integer (nullable = true)
|-- 2/25/20: integer (nullable = true)
|-- 2/26/20: integer (nullable = true)
|-- 2/27/20: integer (nullable = true)
|-- 2/28/20: integer (nullable = true)
|-- 2/29/20: integer (nullable = true)
|-- 3/1/20: integer (nullable = true)
|-- 3/2/20: integer (nullable = true)
|-- 3/3/20: integer (nullable = true)
|-- 3/4/20: integer (nullable = true)
|-- 3/5/20: integer (nullable = true)
|-- 3/6/20: integer (nullable = true)
|-- 3/7/20: integer (nullable = true)
|-- 3/8/20: integer (nullable = true)
|-- 3/9/20: integer (nullable = true)
```

```
|-- 3/10/20: integer (nullable = true)
|-- 3/11/20: integer (nullable = true)
|-- 3/12/20: integer (nullable = true)
|-- 3/13/20: integer (nullable = true)
|-- 3/14/20: integer (nullable = true)
|-- 3/15/20: integer (nullable = true)
|-- 3/16/20: integer (nullable = true)
|-- 3/17/20: integer (nullable = true)
|-- 3/18/20: integer (nullable = true)
|-- 3/19/20: integer (nullable = true)
|-- 3/20/20: integer (nullable = true)
|-- 3/21/20: integer (nullable = true)
|-- 3/22/20: integer (nullable = true)
|-- 3/23/20: integer (nullable = true)
|-- 3/24/20: integer (nullable = true)
|-- 3/25/20: integer (nullable = true)
|-- 3/26/20: integer (nullable = true)
|-- 3/27/20: integer (nullable = true)
|-- 3/28/20: integer (nullable = true)
|-- 3/29/20: integer (nullable = true)
|-- 3/30/20: integer (nullable = true)
|-- 3/31/20: integer (nullable = true)
|-- 4/1/20: integer (nullable = true)
|-- 4/2/20: integer (nullable = true)
|-- 4/3/20: integer (nullable = true)
|-- 4/4/20: integer (nullable = true)
|-- 4/5/20: integer (nullable = true)
|-- 4/6/20: integer (nullable = true)
|-- 4/7/20: integer (nullable = true)
|-- 4/8/20: integer (nullable = true)
|-- 4/9/20: integer (nullable = true)
```



```
|-- 4/10/20: integer (nullable = true)
|-- 4/11/20: integer (nullable = true)
|-- 4/12/20: integer (nullable = true)
|-- 4/13/20: integer (nullable = true)
|-- 4/14/20: integer (nullable = true)
|-- 4/15/20: integer (nullable = true)
|-- 4/16/20: integer (nullable = true)
|-- 4/17/20: integer (nullable = true)
|-- 4/18/20: integer (nullable = true)
|-- 4/19/20: integer (nullable = true)
|-- 4/20/20: integer (nullable = true)
|-- 4/21/20: integer (nullable = true)
|-- 4/22/20: integer (nullable = true)
|-- 4/23/20: integer (nullable = true)
|-- 4/24/20: integer (nullable = true)
|-- 4/25/20: integer (nullable = true)
|-- 4/26/20: integer (nullable = true)
|-- 4/27/20: integer (nullable = true)
|-- 4/28/20: integer (nullable = true)
|-- 4/29/20: integer (nullable = true)
|-- 4/30/20: integer (nullable = true)
|-- 5/1/20: integer (nullable = true)
|-- 5/2/20: integer (nullable = true)
|-- 5/3/20: integer (nullable = true)
|-- 5/4/20: integer (nullable = true)
|-- 5/5/20: integer (nullable = true)
|-- 5/6/20: integer (nullable = true)
|-- 5/7/20: integer (nullable = true)
|-- 5/8/20: integer (nullable = true)
|-- 5/9/20: integer (nullable = true)
|-- 5/10/20: integer (nullable = true)
|-- 5/11/20: integer (nullable = true)
|-- 5/12/20: integer (nullable = true)
|-- 5/13/20: integer (nullable = true)
|-- 5/14/20: integer (nullable = true)
|-- 5/15/20: integer (nullable = true)
|-- 5/16/20: integer (nullable = true)
|-- 5/17/20: integer (nullable = true)
|-- 5/18/20: integer (nullable = true)
```

## Paso 5-> Identificando comportamiento en Colombia, usando la muestra de medellin

```
[19]: cpcc.count()
```

► Spark Job Progress

16295

```
[20]: cpcc.filter(cpcc['Ciudad de ubicación']=='Medellín').count()
```

► Spark Job Progress

356

```
[21]: cpcc.filter(cpcc['atención']=='Recuperado').filter(cpcc['Ciudad de ubicación']=='Medellín').count()
```

► Spark Job Progress

200

```
[22]: cpcc.filter(cpcc['atención']=='Casa').filter(cpcc['Ciudad de ubicación']=='Medellín').count()
```

► Spark Job Progress

141

```
[23]: cpcc.filter(cpcc['atención']=='Hospital').filter(cpcc['Ciudad de ubicación']=='Medellín').count()
```

► Spark Job Progress

8

```
[24]: cpcc.filter(cpcc['atención']=='Fallecido').filter(cpcc['Ciudad de ubicación']=='Medellín').select(
```

► Spark Job Progress

```
+---+---+-----+-----+
|Edad|Sexo|Ciudad de ubicación|    Fecha de muerte|
+---+---+-----+-----+
|  91|  F|           Medellín|2020-04-03T00:00:...|
|  67|  M|           Medellín|2020-04-18T00:00:...|
|  74|  M|           Medellín|2020-04-27T00:00:...|
+---+---+-----+-----+
```

```
[25]: ]=='Hospital UCI').filter(cpcc['Ciudad de ubicación']=='Medellín').select('atención','Edad','Sexo','Ciudad de u
```

► Spark Job Progress

```
+-----+---+---+-----+
| atención|Edad|Sexo|Ciudad de ubicación|
+-----+---+---+-----+
|Hospital UCI| 56|  M|           Medellín|
|Hospital UCI| 37|  M|           Medellín|
|Hospital UCI| 71|  F|           Medellín|
|Hospital UCI| 80|  M|           Medellín|
+-----+---+---+-----+
```

Patron-> La ciudad ha tenido un comportamiento favorable de un total de 16295 casos, la ciudad solo tiene 356: Diistribuidos d ela siguiente forma, 200 recuperados, 141 enviados a casa, 8 Hospitalizados, 4 en una unidad UCI(Unidad de Cuidados Intensivos)y 3 Fallecidos.

Patron-> La ciudad ha tenido un comportamiento favorable de un total de 16295 casos, la ciudad solo tiene 356: Diistribuidos d ela siguiente forma, 200 recuperados,

141 enviados a casa, 8 Hospitalizados, 4 en una unidad UCI(Unidad de Cuidados Intensivos)y 3 Fallecidos.

Paso 6-> Vamos a revisar la proporción de casos de covid en el mundo con respecto a Colombia

```
[26]: cccg.count()
```

► Spark Job Progress

266

```
[27]: cccg.filter(cccg['Country/Region']=='Colombia').select('Country/Region','5/18/20',).show(30)
```

► Spark Job Progress

```
+-----+-----+
|Country/Region|5/18/20|
+-----+-----+
|      Colombia| 16295|
+-----+-----+
```

```
[28]: cccg.filter(cccg['5/18/20']>='16295').select('Country/Region','5/18/20',).show(39)
```

► Spark Job Progress

```
+-----+-----+
|      Country/Region|5/18/20|
+-----+-----+
|      Bangladesh| 23870|
|      Belarus| 30572|
|      Belgium| 55559|
|      Brazil| 255368|
|      Canada| 24286|
|      Canada| 43636|
|      Chile| 46059|
|      China| 68135|
|      Colombia| 16295|
|      Ecuador| 33582|
|      France| 177554|
|      Germany| 176551|
|      India| 100328|
|      Indonesia| 18010|
|      Iran| 122492|
|      Ireland| 24200|
|      Israel| 16643|
|      Italy| 225886|
|      Japan| 16305|
|      Mexico| 51633|
```

	Mexico	51633
	Netherlands	44141
	Pakistan	42125
	Peru	94933
	Poland	18885
	Portugal	29209
	Qatar	33969
	Romania	17036
	Russia	290678
	Saudi Arabia	57345
	Singapore	28343
	South Africa	16433
	Spain	231606
	Sweden	30377
	Switzerland	30597
	Turkey	150593
	Ukraine	18616
	United Arab Emirates	24190
	United Kingdom	246406
	US	1508308
+-----+-----+		

Conclusión-> Hay 266 países con covid 19, para la última fecha del informe, osea el 5/18/20 Colombia tenía en total 16.295 y en el mundo hay 39 países con más casos que Colombia, siendo el caso más preocupante Estados Unidos(US) con 1'508.308

Conclusión-> Hay 266 países con covid 19, para la última fecha del informe, osea el 5/18/20 Colombia tenía en total 16.295 y en el mundo hay 39 países con más casos que Colombia, siendo el caso más preocupante Estados Unidos(US) con 1'508.308