

# **PR\_07.3 Dani Gayol Rodríguez**

PR_07.3 Dani Gayol Rodríguez.....	1
Apartado A.....	1
1. Crear tres buckets en la región por defecto (con nombres similares a: primero, copia, sincro).....	2
2.) Listar todos los cubos.....	2
3.) Ponte en una carpeta donde tengas varios archivos, mejor CSV's o JSON, Sube al menos dos archivos de esa carpeta al bucket primero. ....	2
4.) Copia los archivos del bucket primero a copia de uno en uno. .....	3
5.) Mover todos los objetos de primero a copia dentro de la carpeta datos de forma recursiva. ....	3
6.) Listar los objetos de un copia/datos.....	3
7.) Listar recursivamente todos los objetos el cubo copia. ....	3
8.) Lista recursivamente los datos del cubo en formato legible por humanos. ¿Qué significa eso?.....	3
9.) Mostrar la información resumida, incluido el número de objetos y el tamaño total del cubo copia. ....	3
10.) Muévete a otra carpeta en local y descarga todos los objetos de copia a ella. Borra en local los archivos descargados. ....	4
11.) Sincronizar el contenido del un cubo copia con el cubo sincro. ....	4
12.) Sincronizar el contenido del un cubo copia con la carpeta local. ....	4
13.) Haz al menos tres consultas SQL cualesquiera sobre algunos de los archivos CSV o JSON que hayas subido.....	4
14.) Elimina los objetos de primero de uno en uno. ....	4
15.) Elimina todos los objetos de copia de forma recursiva.....	4
16.) Elimina los cubos primero y copia.....	4
17.) Fuerza la eliminación del cubo sincro sin vaciarlo previamente. ....	5
Apartado B.....	5
1.) Crea un nuevo bucket en S3 y utilizando el comando anterior, copia el archivo ventas.csv que tenemos en HDFS y obtuviste en la práctica 7.1 a él.....	6

2.) ¿Cómo podrás desde HIVE lanzar las mismas consultas que hicimos en los apartados D y E de la práctica 7.1, pero en esta ocasión sobre ficheros en S3? ..... 6

# Apartado A

## 1. Crear tres buckets en la región por defecto (con nombres similares a: primero, copia, sincro).

Los nombres de los buckets deben ser únicos así que voy a añadirles mi nombre

```
C:\Users\Mañana\.aws>aws s3 mb s3://primero-danigayol  
make_bucket: primero-danigayol
```

```
C:\Users\Mañana\.aws>aws s3 mb s3://copia-danigayol  
make_bucket: copia-danigayol
```

```
C:\Users\Mañana\.aws>aws s3 mb s3://sincro-danigayol  
make_bucket: sincro-danigayol
```

## 2.) Listar todos los cubos.

```
C:\Users\Mañana\.aws>aws s3 ls  
2026-01-08 10:10:38 copia-danigayol  
2025-12-16 09:17:56 hadoop-emr-dani-gayol  
2026-01-08 10:10:26 primero-danigayol  
2026-01-08 10:10:50 sincro-danigayol
```

## 3.) Ponte en una carpeta donde tengas varios archivos, mejor CSV's o JSON, Sube al menos dos archivos de esa carpeta al bucket primero.

Primero creo los archivos “datos1.csv” y “datos2.json” y después los pego en la carpeta

```
C:\Users\Mañana\.aws>cd \Users\Mañana\Documents
```

```
C:\Users\Mañana\Documents>aws s3 cp datos1.csv s3://primero-danigayol/  
upload: .\datos1.csv to s3://primero-danigayol/datos1.csv
```

```
C:\Users\Mañana\Documents>aws s3 cp datos2.json s3://primero-danigayol/  
upload: .\datos2.json to s3://primero-danigayol/datos2.json
```

```
C:\Users\Mañana\Documents>aws s3 ls s3://primero-danigayol/
2026-01-08 10:17:06      0 datos1.csv
2026-01-08 10:17:18      0 datos2.json
```

#### 4.) Copia los archivos del bucket primero a copia de uno en uno.

```
C:\Users\Mañana\Documents>aws s3 cp s3://primero-danigayol/datos1.csv s3://copia-danigayol/datos1.csv
copy: s3://primero-danigayol/datos1.csv to s3://copia-danigayol/datos1.csv

C:\Users\Mañana\Documents>aws s3 cp s3://primero-danigayol/datos2.json s3://copia-danigayol/datos2.json
copy: s3://primero-danigayol/datos2.json to s3://copia-danigayol/datos2.json
```

```
C:\Users\Mañana\Documents>aws s3 ls s3://copia-danigayol/
2026-01-08 10:19:01      0 datos1.csv
2026-01-08 10:19:22      0 datos2.json
```

#### 5.) Mover todos los objetos de primero a copia dentro de la carpeta datos de forma recursiva.

```
C:\Users\Mañana\Documents>aws s3 mv s3://primero-danigayol/ s3://copia-danigayol/datos/ --recursive
move: s3://primero-danigayol/datos1.csv to s3://copia-danigayol/datos/datos1.csv
move: s3://primero-danigayol/datos2.json to s3://copia-danigayol/datos/datos2.json
```

#### 6.) Listar los objetos de un copia/datos.

```
C:\Users\Mañana\Documents>aws s3 ls s3://copia-danigayol/datos/
2026-01-08 10:20:29      0 datos1.csv
2026-01-08 10:20:29      0 datos2.json
```

#### 7.) Listar recursivamente todos los objetos el cubo copia.

```
C:\Users\Mañana\Documents>aws s3 ls s3://copia-danigayol/ --recursive
2026-01-08 10:20:29      0 datos/datos1.csv
2026-01-08 10:20:29      0 datos/datos2.json
2026-01-08 10:19:01      0 datos1.csv
2026-01-08 10:19:22      0 datos2.json
```

#### 8.) Lista recursivamente los datos del cubo en formato legible por humanos. ¿Qué significa eso?

**Significa que los tamaños aparecen en “KB, MB, GB” en lugar de bytes**

```
C:\Users\Mañana\Documents>aws s3 ls s3://copia-danigayol/ --recursive --human-readable
2026-01-08 10:20:29    0 Bytes datos/datos1.csv
2026-01-08 10:20:29    0 Bytes datos/datos2.json
2026-01-08 10:19:01    0 Bytes datos1.csv
2026-01-08 10:19:22    0 Bytes datos2.json
```

## **9.) Mostrar la información resumida, incluido el número de objetos y el tamaño total del cubo copia.**

```
C:\Users\Mañana\Documents>aws s3 ls s3://copia-danigayol/ --recursive --summarize
2026-01-08 10:20:29      0 datos/datos1.csv
2026-01-08 10:20:29      0 datos/datos2.json
2026-01-08 10:19:01      0 datos1.csv
2026-01-08 10:19:22      0 datos2.json

Total Objects: 4
Total Size: 0
```

## **10.) Muévete a otra carpeta en local y descarga todos los objetos de copia a ella. Borra en local los archivos descargados.**

Primero, creo la carpeta

```
C:\Users\Mañana\Documents>mkdir carpeta_s3
C:\Users\Mañana\Documents>cd carpeta_s3
```

Luego, descargo todos los archivos en la carpeta

```
C:\Users\Mañana\Documents\carpeta_s3>aws s3 cp s3://copia-danigayol/ . --recursive
download: s3://copia-danigayol/datos1.csv to .\datos1.csv
download: s3://copia-danigayol/datos/datos2.json to datos\datos2.json
download: s3://copia-danigayol/datos/datos1.csv to datos\datos1.csv
download: s3://copia-danigayol/datos2.json to .\datos2.json
```

Finalmente, borro los archivos (estoy desde windows por eso uso ese comando)

```
C:\Users\Mañana\Documents\carpeta_s3>del /Q *
```

## **11.) Sincronizar el contenido del un cubo copia con el cubo sincro.**

```
C:\Users\Mañana\Documents\carpeta_s3>aws s3 sync s3://copia-danigayol/ s3://sincro-danigayol/
copy: s3://copia-danigayol/datos/datos1.csv to s3://sincro-danigayol/datos/datos1.csv
copy: s3://copia-danigayol/datos/datos2.json to s3://sincro-danigayol/datos/datos2.json
copy: s3://copia-danigayol/datos2.json to s3://sincro-danigayol/datos2.json
copy: s3://copia-danigayol/datos1.csv to s3://sincro-danigayol/datos1.csv
```

## 12.) Sincronizar el contenido del un cubo copia con la carpeta local.

```
C:\Users\Mañana\Documents\carpeta_s3>aws s3 sync s3://copia-danigayol/ \Users\Mañana\Documents\carpeta_s3
download: s3://copia-danigayol/datos2.json to .\datos2.json
download: s3://copia-danigayol/datos1.csv to .\datos1.csv
```

## 13.) Haz al menos tres consultas SQL cualesquiera sobre algunos de los archivos CSV o JSON que hayas subido.

En los “.csv” no tengo ninguna información, pero las consultas podrían ser tal que así;

```
C:\Users\Mañana\Documents\carpeta_s3>aws s3api select-object-content --bucket copia-danigayol --key datos/datos1.csv --expression "SELECT * FROM s3object s" --expression-type SQL --input-serialization '{"CSV": {"FileHeaderInfo": "USE"}}' --output-serialization '{"CSV": {}}' output.csv
```

```
C:\Users\Mañana\Documents\carpeta_s3>aws s3api select-object-content --bucket copia-danigayol --key datos/datos1.csv --expression "SELECT s.nombre, s.edad FROM s3object s" --expression-type SQL --input-serialization '{"CSV": {"FileHeaderInfo": "USE"}}' --output-serialization '{"CSV": {}}' output.csv
```

```
C:\Users\Mañana\Documents\carpeta_s3>aws s3api select-object-content --bucket copia-danigayol --key datos/datos1.csv --expression "SELECT * FROM s3object s WHERE s.edad > 30" --expression-type SQL --input-serialization '{"CSV": {"FileHeaderInfo": "USE"}}' --output-serialization '{"CSV": {}}' output.csv
```

## 14.) Elimina los objetos de primero de uno en uno.

Primero los listo para ver que archivos contiene

```
C:\Users\Mañana\Documents>aws s3 ls s3://primero-danigayol/
2026-01-12 09:09:37          0 datos1.csv
2026-01-12 09:09:55          0 datos2.json
```

Y ahora los borro uno a uno

```
C:\Users\Mañana\Documents>aws s3 rm s3://primero-danigayol/datos1.csv
delete: s3://primero-danigayol/datos1.csv
```

```
C:\Users\Mañana\Documents>aws s3 rm s3://primero-danigayol/datos2.json
delete: s3://primero-danigayol/datos2.json
```

## 15.) Elimina todos los objetos de copia de forma recursiva.

```
C:\Users\Mañana\Documents>aws s3 rm s3://copia-danigayol/ --recursive
delete: s3://copia-danigayol/datos2.json
delete: s3://copia-danigayol/datos/datos1.csv
delete: s3://copia-danigayol/datos/datos2.json
delete: s3://copia-danigayol/datos1.csv
```

## 16.) Elimina los cubos primero y copia..

```
C:\Users\Mañana\Documents>aws s3 rb s3://primero-danigayol
remove_bucket: primero-danigayol

C:\Users\Mañana\Documents>aws s3 rb s3://copia-danigayol
remove_bucket: copia-danigayol
```

## 17.) Fuerza la eliminación del cubo sincro sin vaciarlo previamente.

```
C:\Users\Mañana\Documents>aws s3 rb s3://sincro-danigayol --force
delete: s3://sincro-danigayol/datos2.json
delete: s3://sincro-danigayol/datos1.csv
delete: s3://sincro-danigayol/datos/datos1.csv
delete: s3://sincro-danigayol/datos/datos2.json
remove_bucket: sincro-danigayol
```

# Apartado B

**1.) Crea un nuevo bucket en S3 y utilizando el comando anterior, copia el archivo ventas.csv que tenemos en HDFS y obtuviste en la práctica 7.1 a él.**

Primero, vamos a crear el bucket en S3

```
C:\Users\Mañana\Documents>aws s3 mb s3://ventas-hdfs-danigayol  
make_bucket: ventas-hdfs-danigayol
```

Una vez creado, vamos a copiar el archivo “ventas.csv” del HDFS al nuevo bucket. Pero al tenerlo en un EMR, al terminarlo, no podemos volver a iniciarla, por lo tanto, voy a copiar el archivo desde el apartado de descargas en Windows

```
C:\Users\Mañana>cd \Users\Mañana\Downloads  
  
C:\Users\Mañana\Downloads>aws s3 cp ventas.csv s3://ventas-hdfs-danigayol/  
upload: ./ventas.csv to s3://ventas-hdfs-danigayol/ventas.csv
```

```
C:\Users\Mañana\Downloads>aws s3 ls s3://ventas-hdfs-danigayol/  
2026-01-12 10:20:56      107396 ventas.csv
```

**2.) ¿Cómo podrás desde HIVE lanzar las mismas consultas que hicimos en los apartados D y E de la práctica 7.1, pero en esta ocasión sobre ficheros en S3?**

Hive permite consultar datos almacenados en Amazon S3 mediante el conector S3A, definiendo tablas externas cuya ubicación apunta directamente a un bucket S3, lo que posibilita ejecutar las mismas consultas SQL que sobre HDFS sin mover los datos.