

## Guía de ejecución del consumidor Spark Streaming

### Instalación

Necesitas Java 17.

Puedes comprobar tu versión con:

```
java --version
```

Si no lo tienes, puedes instalarlo con:

```
sudo apt install openjdk-17-jdk
```

### Ejecución

#### 1. Levantar Kafka (Docker)

- Activa el entorno:  
conda activate arqesp
- Ve a la carpeta del docker:  
cd ~/AE/proyecto/AE\_spark-streaming/docker
- Levanta los contenedores:  
sudo docker-compose up -d

#### 2. Probar conexión rápida con el tester

- Vuelve a la raíz del proyecto:  
cd ~/AE/proyecto/AE\_spark-streaming
- Ejecuta el tester:  
python tests/tester.py

Si al final ves “¡TODO FUNCIONA!”, Kafka y el topic están bien.

#### 3. Lanzar el productor

- Activa el entorno (si hiciera falta):  
conda activate arqesp
- Entra en src:  
cd ~/AE/proyecto/AE\_spark-streaming/src
- Lanza el productor:  
python producer.py

#### 4. Lanzar el consumer de Spark en otra terminal

En una nueva terminal:

- Activa el entorno:  
conda activate arqesp
- Entra en src:  
cd ~/AE/proyecto/AE\_spark-streaming/src
- Lanza el consumer:  
python spark\_consumer.py

Al principio aparecerán muchos mensajes WARN de Spark, Ivy, Hadoop, etc. Son mensajes de carga y resolución de dependencias, no errores. Lo importante es cuando empiezan a salir los bloques de “Batch ... – Snapshot en ...”.

## Salida

El script `spark_consumer.py` hace lo siguiente:

- Lee el topic “`tweets_topic`” de Kafka.
- Cada mensaje tiene un campo “`hashtag_principal`”.
- Cuenta cuántas veces aparece cada hashtag en una ventana de 1 minuto.
- Cada 10 segundos imprime un snapshot del ranking de hashtags de ese último minuto.

Ejemplo de bloque de salida:

```
=====
Batch 2 - Snapshot en 2025-12-11 21:21:32
Ventana del minuto [21:21:00 , 21:22:00]
Top hashtags en este minuto:
+-----+-----+-----+
|window|hashtag_principal|num_ocurrencias|
+-----+-----+-----+
|{2025-12-11 21:21:00, 2025-12-11 21:22:00}|#Examen|12|
|{2025-12-11 21:21:00, 2025-12-11 21:22:00}|#Python|9|
...
```

- “Batch N” es el número de actualización desde que se arrancó el consumer.
- “Ventana del minuto [.. , ..]” es el intervalo de 60 segundos que se está analizando.
- La tabla muestra los hashtags ordenados por “`num_ocurrencias`” en ese minuto.

## Finalización

- Parar el productor (terminal donde corre `productor.py`):  
Ctrl + C
- Parar el consumer (terminal donde corre `spark_consumer.py`):  
Ctrl + C
- Parar Kafka y Zookeeper (Docker):  
`cd ~/AE/proyecto/AE_spark-streaming/docker`  
`sudo docker-compose down`