

# Spark Streaming

## kafka

Vamos a crear una estructura de Kafka STREAMING

Para ello vamos a usar la estructura de código PRODUCTOR y código CONSUMIDOR

Trabajo que debe desarrollar el PRODUCTOR. → Deberá mandar el contenido de un JSON que se os facilitará para que el consumidor que lo escuche pueda tratarlo.

Para la creación de dicho productor se os pedirá que lo hagáis de dos maneras

1. **Usando la sintaxis vista en clase con el método `Kafka-console-producer.sh`**

Trabajo que debe desarrollar el CONSUMIDOR → Nuestro consumidor deberá también tener dos entornos distintos de trabajo, para poder chequear que Kafka funciona correctamente.

1. **Usando la sintaxis de `Kafka-console-consumer.sh`. En éste caso solo queremos comprobar que nuestro consumidor puede leerlo sin problema**
2. Creando un código en Scala donde vamos a ampliar nuestro trabajo. Deberá leerlo igual que en el caso anterior, pero en éste caso también deberá hacer un tratamiento especial de los datos que vayamos leyendo.  
Para cada apartado que se os va a proponer debemos crear funciones de SCALA.
  - a. Queremos que filtre (que no aparezcan) del fichero JSON dos palabras que elegiréis cada uno (así evitamos tentaciones de copia jejej). Debo ver el JSON con esas palabras filtradas

## Parte de Investigación.

Si alguien quiere obtener una mejor nota, tenéis instalado en vuestra máquina Zeppelin.  
La idea es que me expliquéis los pasos que deberíamos seguir para conseguir lo mismo que yo.

Necesito que me expliquéis los pasos más relevantes de la instalación, y la forma de configurar nuestro SPARK para que podamos trabajar en el note de zepellin.

Una vez lo tengamos todo montado:

Debemos hacer un pequeño ejercicio para repasar la parte inicial de nuestro curso.

Deberemos trabajar con el csv **“amigos.csv”** y de él mostrar datos:

- a. Calcular el número de registros que tenemos

### CONSIDERACIONES DE ENTREGA:

1. Necesito el proyecto concretamente el src con los archivos y los códigos de scala que tengáis hechos
2. Quiero un pantallazo del código de scala acompañado del pantallazo de la ejecución del mismo
3. Si hacéis la parte de zeppelin :
  - i. Para la parte de instalación pantallazos de las imágenes más relevantes
  - ii. Para el código me entregáis el código de zepellin y un pantallazo de su ejecución

Valoración:

- Claridad en el código
  - Favorecer la legibilidad por encima de un código compacto
  - Explicación (comentada en el propio código) el porqué de cada una de las Decisiones tomadas
    - Uso preferido de spark sql con Kafka aunque no obligatorio
    - Investigación y ensayo autodidacta de Scala y de Zeppelin
  - *Para sacar un buen resultado de la práctica no es estrictamente necesario que ésta funcione de forma completa mientras sí cumpla la mayoría de los Puntos anteriores de valoración.*
- 
- La práctica que se os entrega puede ser modificada si vemos que hay alguna errata o error en ella.