# **Tutorial 6:** Test Kafka and Spark Structure Streaming on Local

**Bước 1:** Khởi chạy Kafka cluster và Zookeeprt sử dụng terminal

|  |
| --- |
| Khởi chạy zookepper:  bin\windows\zookeeper-server-start.bat config\zookeeper.properties |

|  |
| --- |
| Khởi chạy Kafka cluster:  bin\windows\kafka-server-start.bat config\server.properties |

**Bước 2:** Chạy KafkaProducer in Jupyter Notebook

|  |
| --- |
| from kafka import KafkaProducer  from json import dumps # to convert the data to json format  from time import sleep # to introduce delay  topicName = 'RandomNumber' # topic name  kafkaServer = 'localhost:9092' # kafka server address  #Create a producer object  producer = KafkaProducer(bootstrap\_servers=[kafkaServer], # connect to the kafka server                          value\_serializer=lambda x: dumps(x).encode('utf-8'))                          # dumps() convert the data to JSON format                          # encode() convert the JSON data to bytes beacuse Kafka only accepts bytes data  for e in range(1000):      data = {'number' : e}      producer.send(topicName, value=data) # send the data to the topic      print(str(data) + " sent")      sleep(5)    producer.flush() # remove all the messages from the buffer |

A black background with white text

Description automatically generated

**Bước 3:** Mở thêm 1 file Jupyter Notebook khác và thực hiện code sau:

|  |
| --- |
| import findspark  findspark.init()  import pyspark  from pyspark.sql import SparkSession  scala\_version = "2.12"  spark\_version = "3.5.3"  package= [f'org.apache.spark:spark-sql-kafka-0-10\_{scala\_version}:{spark\_version}',              'org.apache.kafka:kafka-clients:2.8.0' ]  spark = SparkSession.builder.master("local[\*]").appName("kafka-example").config("spark.jars.packages", ",".join(package)).getOrCreate()  spark |

Kết quả:

A screen shot of a computer

Description automatically generated

## CÁCH 1: ĐỌC DATA TỪ KAFKA BẰNG BATCH QUERIES

* Tạo 1 dataframe từ kafka data

|  |
| --- |
| #Create dataframe from Kafka data  topic\_name = 'RandomNumber'  kafka\_server = 'localhost:9092'  kafkaDf = spark.read.format("kafka").option("kafka.bootstrap.servers", kafka\_server).option("subscribe", topic\_name).option("startiingOffsets", "earliest").load() |

* Show data (chuyển từ dataframe sang pandas để xem dễ hơn)

|  |
| --- |
| #Show data (converting dataframe to pandas for cleaner view of data)  kafkaDf.toPandas() |

**A screenshot of a computer

Description automatically generated**

* Hiển thị dữ liệu trực tuyến bằng for loop (Show streaming data using for loop)

|  |
| --- |
| # Show streaming data using for loop  from pyspark.sql.functions import col  batchDF = kafkaDf.select(col('topic'),col('offset'),col('value').cast('string').substr(12,1).alias('rand\_number'))  from time import sleep  from IPython.display import display, clear\_output  for x in range(0, 2000):      try:          print("Showing live view refreshed every 5 seconds")          print(f"Seconds passed: {x\*5}")          display(batchDF.toPandas())          sleep(5)          clear\_output(wait=True)      except KeyboardInterrupt:          print("break")          break  print("Live view ended...") |

**A screenshot of a computer program

Description automatically generated**

* Thực hiện một số phép tính và hiển thị kết quả trực tiếp (Perform some data aggregation and show live results)

|  |
| --- |
| #Perform some data aggregation and show live results  batchCountDF = batchDF.groupBy('rand\_number').count()  for x in range(0, 2000):      try:          print("Showing live view refreshed every 5 seconds")          print(f"Seconds passed: {x\*5}")          display(batchCountDF.toPandas())          sleep(5)          clear\_output(wait=True)      except KeyboardInterrupt:          print("break")          break  print("Live view ended...") |

A screenshot of a computer program

Description automatically generated

## CÁCH 2: ĐỌC DATA TỪ KAFKA BẰNG STREAMING QUERIES

* Tạo streaming dataframe từ kafka

|  |
| --- |
| # Create Streaming dataframe from Kafka  streamRawDf = spark.readStream.format("kafka").option("kafka.bootstrap.servers", kafka\_server).option("subscribe", topic\_name).load()  streamDF = streamRawDf.select(col('topic'),col('offset'),col('value').cast('string').substr(12,1).alias('rand\_number'))  checkEvenDF = streamDF.withColumn('Is\_Even',col('rand\_number').cast('int') % 2 == 0 ) |

# Tutorial 7: Kafka and MongoDB on Window

* Producer

A computer screen shot of a black background

Description automatically generated

* Consumer

A screenshot of a computer

Description automatically generated

* Kết quả

A screenshot of a computer

Description automatically generated