You have been given three files containing Youtube videos related data from different regions. You need to put this data into a folder in Azure Blob Storage under a new container. Mount this container into Databricks using Spark. Then, implement the spark code that will stream the data from these files and:

1. Remove the “trending_date” column
2. Change the “thumbnail_link” column name to “url”

Here is the schema that you can use for the different columns:

Column	**Data type**
video_id	String
trending_date	String
title	String
channel_title	String
category_id	Integer
publish_time	String
tags	String
views	Long
likes	Long
dislikes	Long
comment_count	Long
thumbnail_link	String
comments_disabled	String
ratings_disabled	String
video_error_or_removed	String

Print the output of this data. Also, create an event hub, and stream this data into the event hub.

Sol)

Since Data Bricks is of cost I tried to do a POC on local and came up with the below steps. I will explain with respect to Azure terms and its services.

Step1: Set up Azure Blob Storage

- Create a new container in Azure Blob Storage to store the data files.
- Upload the three YouTube video data files to the container.

Step2: Mount Azure Blob Storage in Databricks

Step3: Implement the Spark code to stream and transform the data:

Step4: Execute the Spark code and observe the transformed data:

Step5: Write Data from the Data bricks to Event hub:

In this process

- Data Bricks = Producer
- EventHub = Consumer

CODE for all the above Steps

```
from azure.eventhub import EventHubProducerClient, EventData
from pyspark.sql import SparkSession

# Azure Event Hub configurations
event_hub_connection_string = "<event_hub_connection_string>"
event_hub_name = "<event_hub_name>"

# Create a SparkSession
spark = SparkSession.builder.appName("EventHubStreaming").getOrCreate()

# Read the data files and perform necessary transformations
data_path = "/mnt/<mount_point>/<data_folder>/"
df = spark.read.format("csv").options(header=True, inferSchema=True).load(data_path)
df = df.drop("trending_date").withColumnRenamed("thumbnail_link", "url")

# Configure Event Hub connection
producer = EventHubProducerClient.from_connection_string(event_hub_connection_string,
eventhub_name=event_hub_name)

# Define the streaming query to send data to Event Hub
query = df \
    .writeStream \
    .format("eventhubs") \
    .outputMode("append") \
    .option("checkpointLocation", "/mnt/<mount_point>/checkpoint") \
    .option("eh.streaming.connectionString", event_hub_connection_string) \
    .option("eh.streaming.entityPath", event_hub_name) \
    .start()
```

Title: Streaming YouTube Video Data with Spark and Azure Event Hub

Objective: The objective of this exercise is to stream YouTube video data from multiple files using Spark in Databricks. The data will be sourced from Azure Blob Storage and processed using Spark to remove a column, rename another column, and print the output. Additionally, an Azure Event Hub will be created, and the processed data will be streamed into the Event Hub.

Tasks:

1. Set up Azure Blob Storage: Create a new container in Azure Blob Storage and upload the three files containing YouTube video data to a folder within the container.
2. Mount Azure Blob Storage in Databricks: Mount the Azure Blob Storage container in Databricks to access the data files. Configure the necessary credentials and permissions for the mount.
3. Develop Spark code: Write Spark code in Databricks to stream the data from the files. Remove the "trending_date" column and rename the "thumbnail_link" column to "url". Process the data using Spark transformations and actions.
4. Print the output: Display the processed data in the output console to verify the column removal and renaming operations.
5. Create an Azure Event Hub: Set up an Azure Event Hub to receive the streamed data from Spark. Configure the necessary Event Hub settings and obtain the connection string.
6. Stream data into Event Hub: Modify the Spark code to include the streaming functionality and send the processed data to the Azure Event Hub. Configure the connection details and ensure successful data streaming.

Learning Outcomes:

- Setting up Azure Blob Storage and uploading files.
- Mounting Azure Blob Storage in Databricks for data access.
- Processing streaming data using Spark in Databricks.
- Removing and renaming columns in Spark DataFrames.
- Printing the output of processed data.
- Creating and configuring Azure Event Hub.
- Streaming data into Azure Event Hub using Spark.

Column	**Data type**
video_id	String
trending_date	String
title	String
channel_title	String
category_id	Integer
publish_time	String
tags	String
views	Long
likes	Long
dislikes	Long
comment_count	Long
thumbnail_link	String
comments_disabled	String
ratings_disabled	String
video_error_or_removed	String