IST769 Lab K

# Search Model: Kafka and KSQLdb

### In this lab, we will explore the streaming model using Apache Kafka and Confluent’s KSQLDb which allows you to use SQL for stream processing. We will also integrate Kafka into Drill.

### Learning Outcomes

At the end of this lab you should be able to:

* Query Kafka topics with KSQL, Drill and Python.
* Use KSQL to manipulate a data stream.
* Write data streams to other databases or files.

### Pre-Requisites

Before you begin:

* Open a terminal window in the lab environment
* Set the current working directory to **advanced-databases**
* Start the following services required by the lab:   
  **jupyter drill zookeeper broker ksqldb-server ksqldb-cli schema-registry connect**

### Tools Used In this Lab

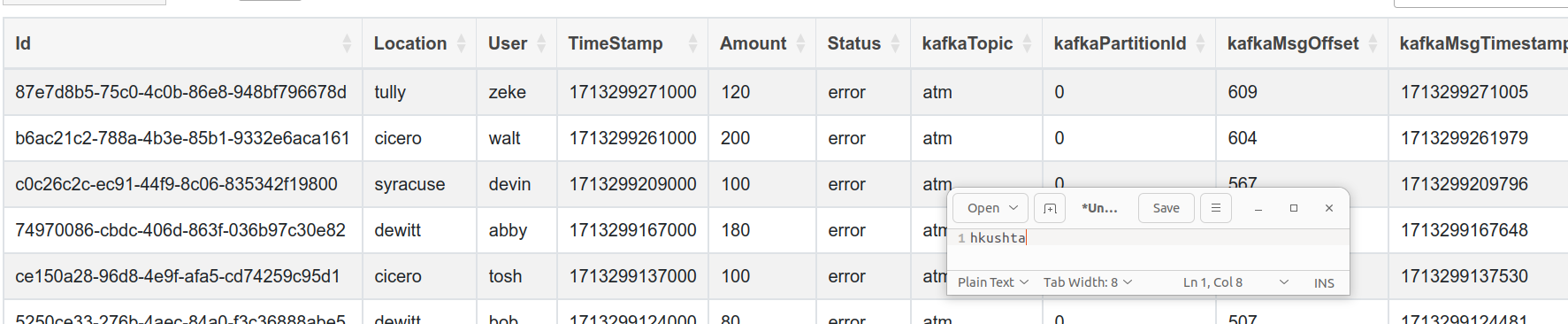
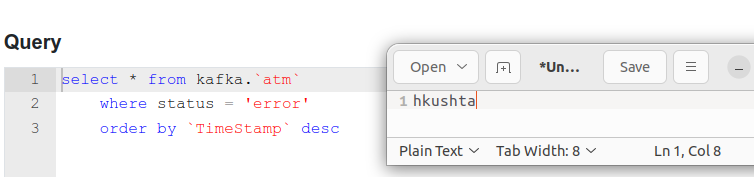
The following tools will be used in this lab:

1. To access Jupyter Lab from your Windows host:  
   [http://localhost:8888](http://localhost:8888/)   
   The password is **SU2orange!**
2. To access the Drill Use  
   [http://localhost](http://localhost:/):8047
3. To access KSQL Db Client:   
   docker-compose exec ksqldb-cli ksql [http://ksqldb-server:8088](http://ksqldb-server:8088/)
4. Start the ATM producer example from Jupyter, located at:  
   **/work/examples/Kafka-Producer.ipynb**

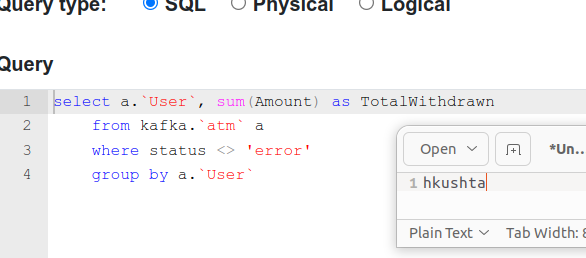
# Lab Problem Set

**QUESTIONS:   
  
For each question, include a copy of the code required to complete the question along with a screenshot of the code and a screenshot of the output.**

1. Write a drill query to display only the atm transactions that ended in an Error status. Show all columns and sort output so the newest errors are first.   
   NOTE: It is strongly suggested you use `backticks` and table aliases when working in drill

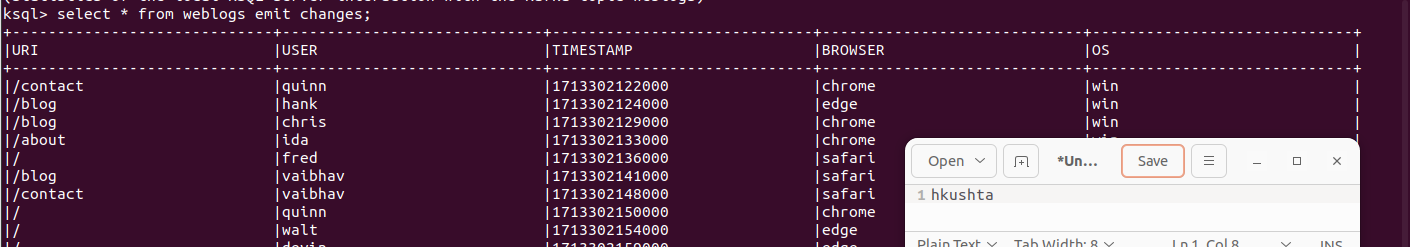
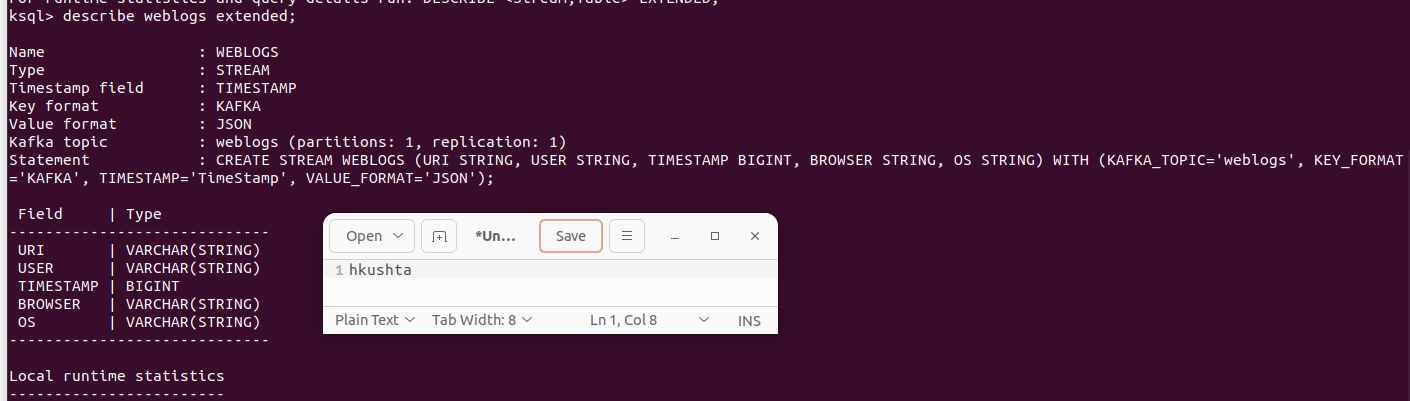
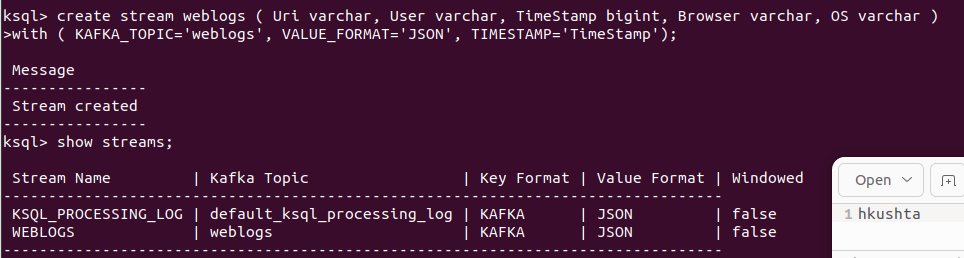


1. Write a drill query to display the total amount withdrawn by user and do not include error transactions in the totals.

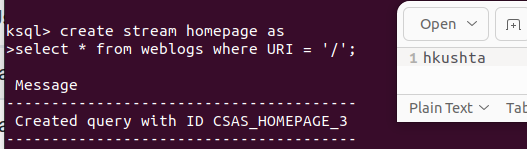


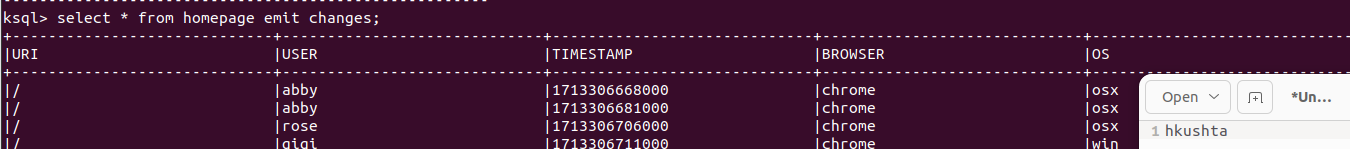


1. Write KSQL to create a stream named **weblogs** from the JSON keys in the weblogs Kafka topic. Make sure to set the TIMESTAMP property to the timestamp from the stream.

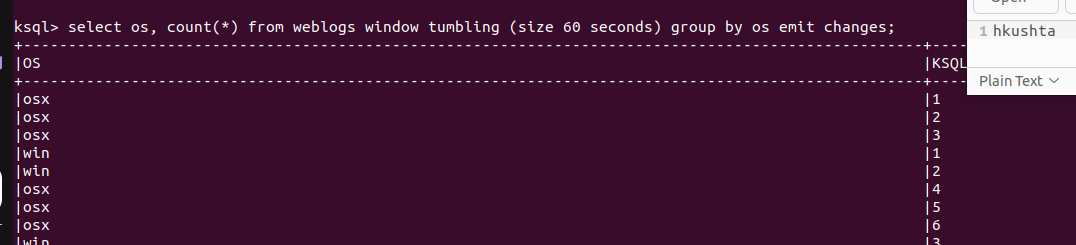


1. Write a KSQL statement create a persistent stream/table called **homepage** which only displays visitors to the root of the website (/). It should display all columns from the weblogs stream.

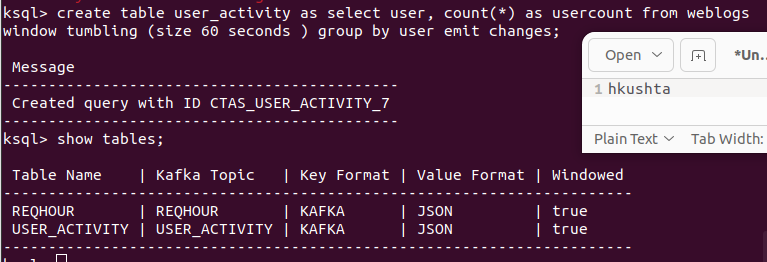
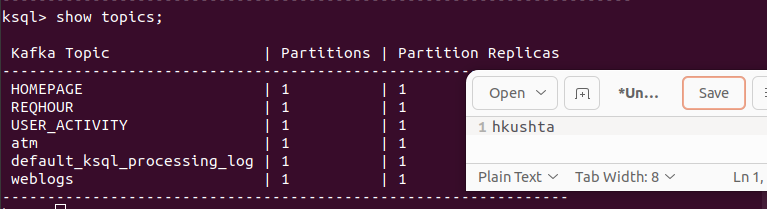


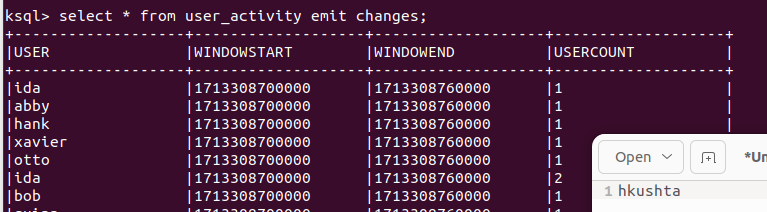


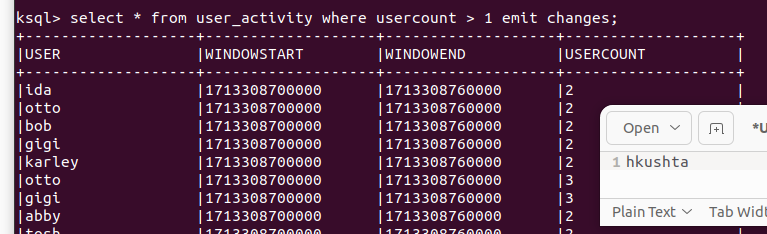
1. Write a KSQL statement to count operating systems users (os) in 60 second windows. After 60 seconds, the counter should reset, and counts should begin again.



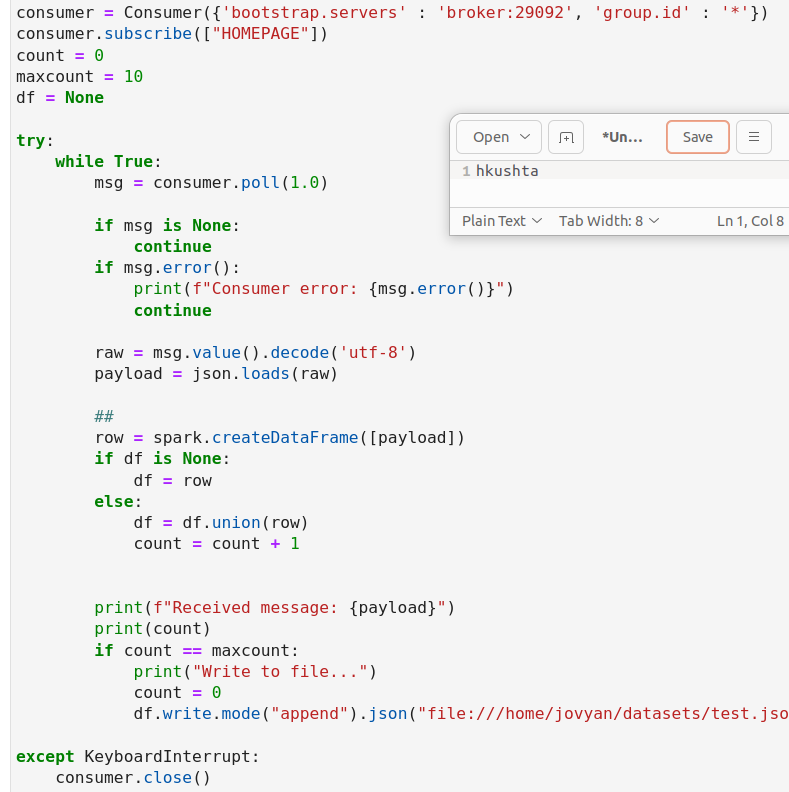
1. Write a KSQL persistent stream/table called **user\_activity** which will display a count of user activity on the website within 1-minute sessions.

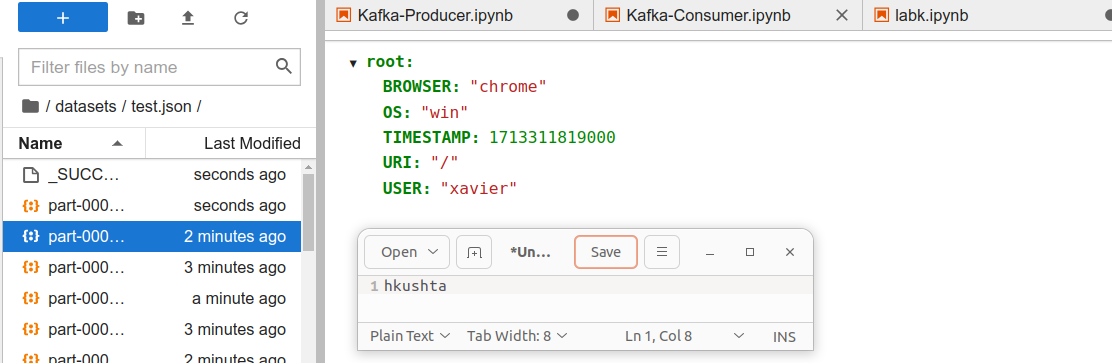
  


1. Write a KSQL statement to display users who have more than 1 pages of activity in a 1-minute window.



1. In Jupyter, write a program to subscribe to the homepage topic generated by the stream/table in Question 4 and display the messages to the console.   
   NOTE: We could easily then write these to elasticsearch, but we will not do that in this lab.



**** **IMPORTANT:** When you are finished with the lab, execute:

PS:> docker-compose stop

To turn off all running services, then shut down your Azure Lab instance.