Cloudera Streaming Analytics 1.3.0

# Getting Started

**Date published: 2019-12-17**
**Date modified: 2021-03-25**

**https://docs.cloudera.com/**

# Legal Notice

© Cloudera Inc. 2024. All rights reserved.

The documentation is and contains Cloudera proprietary information protected by copyright and other intellectual property rights. No license under copyright or any other intellectual property right is granted herein.

Unless otherwise noted, scripts and sample code are licensed under the Apache License, Version 2.0.

Copyright information for Cloudera software may be found within the documentation accompanying each component in a particular release.

Cloudera software includes software from various open source or other third party projects, and may be released under the Apache Software License 2.0 (“ASLv2”), the Affero General Public License version 3 (AGPLv3), or other license terms. Other software included may be released under the terms of alternative open source licenses. Please review the license and notice files accompanying the software for additional licensing information.

Please visit the Cloudera software product page for more information on Cloudera software. For more information on Cloudera support services, please visit either the Support or Sales page. Feel free to contact us directly to discuss your specific needs.

Cloudera reserves the right to change any products at any time, and without notice. Cloudera assumes no responsibility nor liability arising from the use of products, except as expressly agreed to in writing by Cloudera.

Cloudera, Cloudera Altus, HUE, Impala, Cloudera Impala, and other Cloudera marks are registered or unregistered trademarks in the United States and other countries. All other trademarks are the property of their respective owners.

Disclaimer: EXCEPT AS EXPRESSLY PROVIDED IN A WRITTEN AGREEMENT WITH CLOUDERA, CLOUDERA DOES NOT MAKE NOR GIVE ANY REPRESENTATION, WARRANTY, NOR COVENANT OF ANY KIND, WHETHER EXPRESS OR IMPLIED, IN CONNECTION WITH CLOUDERA TECHNOLOGY OR RELATED SUPPORT PROVIDED IN CONNECTION THEREWITH. CLOUDERA DOES NOT WARRANT THAT CLOUDERA PRODUCTS NOR SOFTWARE WILL OPERATE UNINTERRUPTED NOR THAT IT WILL BE FREE FROM DEFECTS NOR ERRORS, THAT IT WILL PROTECT YOUR DATA FROM LOSS, CORRUPTION NOR UNAVAILABILITY, NOR THAT IT WILL MEET ALL OF CUSTOMER’S BUSINESS REQUIREMENTS. WITHOUT LIMITING THE FOREGOING, AND TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, CLOUDERA EXPRESSLY DISCLAIMS ANY AND ALL IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, QUALITY, NON-INFRINGEMENT, TITLE, AND FITNESS FOR A PARTICULAR PURPOSE AND ANY REPRESENTATION, WARRANTY, OR COVENANT BASED ON COURSE OF DEALING OR USAGE IN TRADE.

# Contents

# Running a simple SQL job

You can use this Getting Started use case to get familiar with the most simple form of running a SQL Stream job.

## About this task

The Getting Started contains the basic steps of running a SQL Stream job. When executing the job, you do not need to select a sink as the results are displayed in the browser. The SQL Stream Builder provisions a job on your cluster to run the SQL queries. You can select the Logs tab to review the status of the SQL job. As data is returned, it shows up in the Results tab.

**Note:** When adding SQL statements to the SQL window, you do not need to add the semicolon (;) at the end of the statement as SSB can execute the command without the semicolons.

## Before you begin

As the Getting Started is using the Stateful Tutorial as an example, you need to set up a Kafka topic as transaction.log.1 in Streams Messaging Manager, and submit the Kafka Data Generator job to generate data to the source topic. For more information, see the Stateful Tutorial.

## Procedure

1. Go to your cluster in Cloudera Manager.
2. Click on SQL Stream Builder from the list of Services.
3. Click on SQLStreamBuilder Console.
4. Click on Data Sources from the main menu.
5. Register a Kafka Provider.
6. Click on Console from the main menu.
7. Click on Virtual Tables tab.
   You will automatically be on the Virtual Tables Source tab.
8. Add a Kafka source.
   a) Name the Virtual Table Source to transactions
   b) Select the registered Kafka cluster.
   c) Select transaction.log.1 as the Kafka topic.
   d) Select JSON as Data Format.
   e) Click Detect Schema.
      SSB detects the schema and displays it to the Schema Definition field.
   f) Click Save Changes.
9. Click on Compose tab.
10. Provide a name for the SQL job in the SQL Job Name text box.

    **Note:** You can also use the random name button to generate a name for your job.

    **Important:** Do not add any sink to the SQL Job as the output is generated to the browser.

11. Add the following SQL statement to the SQL window:

```
SELECT * FROM transactions
```

12. Click on the Execute button.
    You can see the generated output in the Results tab.

**13.** Click on the Stop button to stop the previous query.

**14.** Add the following SQL statement to the SQL window:

```
SELECT itemId, quantity
FROM (
   SELECT itemId, quantity,
     ROW_NUMBER() OVER (
       ORDER BY '', quantity) AS rownum
   FROM transactions)
WHERE rownum <= 4
```

**Related Information**

Using Streaming SQL Console

Data Sources in SSB

Using Virtual Tables

Monitoring SQL Stream jobs