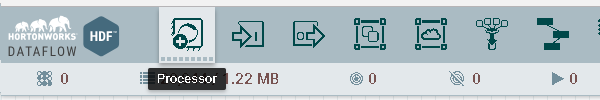
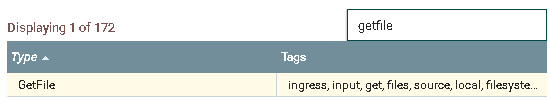
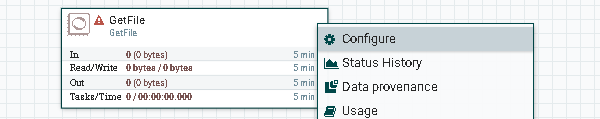
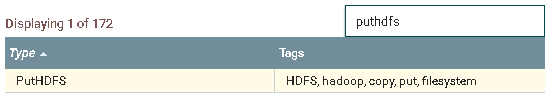
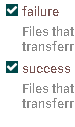
**Installing Nifi on HDP**: Sign into Ambari as ‘admin’/’admin’ > ‘Actions’ > ‘Add Service’ > check ‘Nifi’ > ‘Install

**Opening Nifi**: Go to Nifi UI interface (either using ‘<Host IP Address>:9090/nifi/’ or through ‘Ambari Dashboard’ > ‘Nifi’ > ‘Quick Links’ > ‘Nifi UI’)

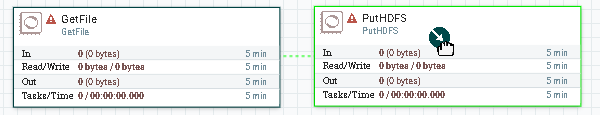
**Exercise One**: move a file from local directory to HDFS directory.

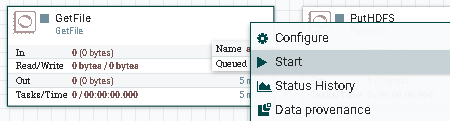
Note: default nifi directories (in HDP 2.5): local - /opt/HDF-2.0.0.0-579/, hdfs - /user/nifi/

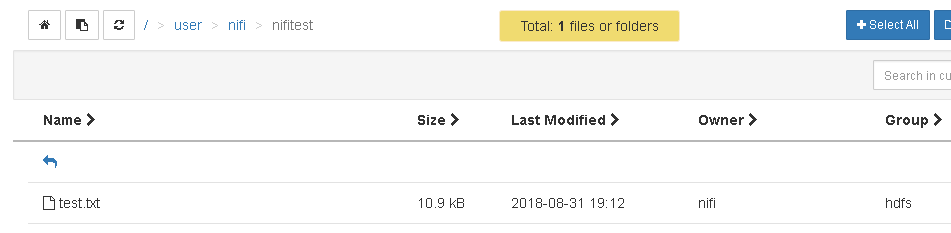
Note: Nifi, by default, does not have write access outside of its directory.

1. **Drag** ‘Processor’ component to anywhere on the grid to open a list of processes.
2. Select ‘GetFile’ and press **add** the specified processor into your dataflow. 
3. **Right-click** the new processor and press **configure** .
4. **Edit** all values according to your specifications. Note: directory needs to already exist
5. **Add** a second processor of ‘PutHDFS’ type . 
6. **Edit** the properties of the processor, and **set** the ‘Auto Terminate Relationships’. 

Note: You may need to edit ‘Hadoop Configuration Resources’ property with “/etc/hadoop/conf/core-site.xml”. If you receive a “LzoCodec not found” error, go to the end of this exercise for a possible solution.

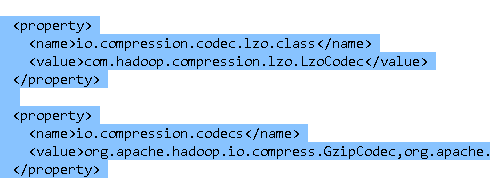
1. To **create** a connection , drag the arrow icon from the ‘GetFile’ processor to the ‘PutHDFS’ processor. 

1. **Run** the dataflow by either: right-click and start each processor,or start while unselected

Check that the file(s) from the local directory as moved to the HDFS directory:

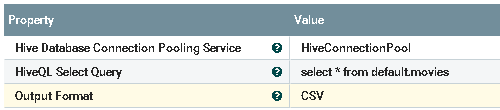
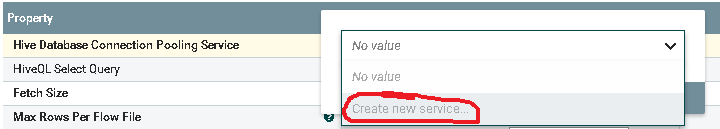
1. **Save** your dataflow as a template

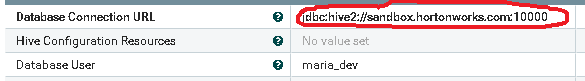


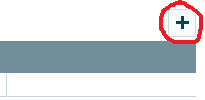
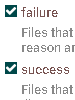
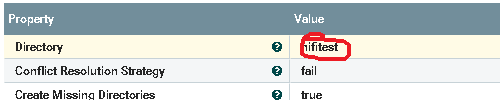
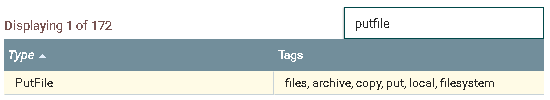
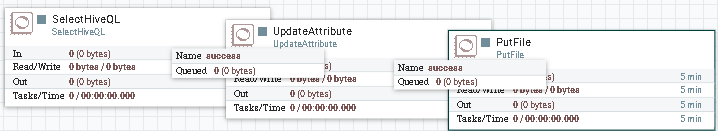
Possible Solution to “LzoCodec not found” error: edit a copy of the ‘/etc/hadoop/conf/core-site.xml’ file located in the Hadoop directory by removing compression properties:.

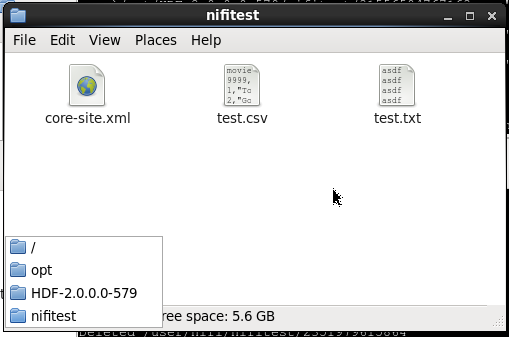
Then link the custom ‘core-site.xml’ file into your ‘PutHDFS’ processor properties: 

**Exercise Two**: Execute and save Hive query into local system.

1. **Add** a processor of type ‘SelectHiveQL’, **set** necessary ‘Auto Terminate Relationships’, then **edit** necessary properties. 
2. **Create** a ‘HiveConnectionPool’ controller service. **Edit** necessary properties, then **enable**. 

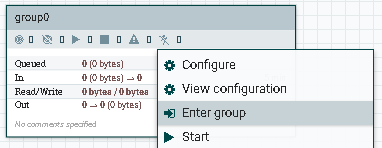
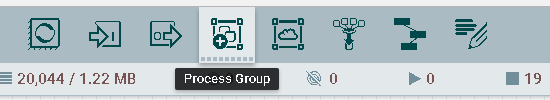
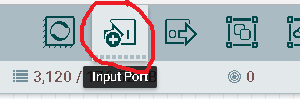
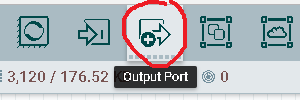
Note: “jdbc:hive2://sandbox.hortonworks.com:10000”

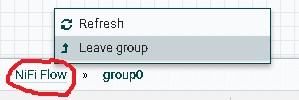
1. **Add** processor of type ‘UpdateAttribute’, **add** property named ‘filename’. **Edit** property
2. **Add** processor of type ‘PutFile’, **set** ‘Auto Terminate Relationships’, and **edit** properties.
3. To **create** a connection, drag the arrow from the ‘SelectHiveQL’ processor to the ‘UpdateAttribute’ processor and from ‘UpdateAttribute’ processor to ‘PutFile’ processor. All connections made here are using ‘success’.
4. **Run** the dataflow by right-clicking and starting the processors

Check that the file(s) are created in our local directory.

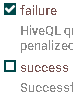
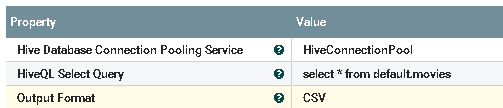
1. **Save** your dataflow.

**Exercise Three**: Create Process Group and Using Expression Language to Split Table Data

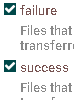
1. **Drag** the ‘Process Group’ component to anywhere on the grid, and **Enter** the group
2. **Drag** the ‘Input Port’ and ‘Output Port’ component onto the grid 
3. **Leave** the group to by clicking ‘Nifi Flow’ or by right-clicking and pressing ‘Leave Group’

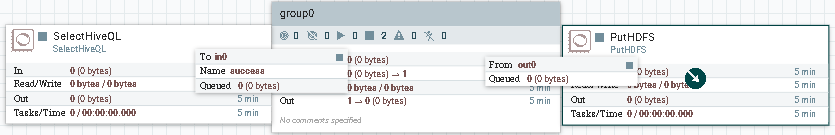


1. **Add** a processor of type ‘SelectHiveQL’ using CSV format, and relevant Properties and Auto Terminate Relationships.

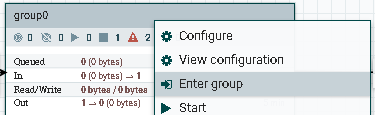


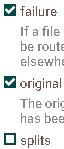
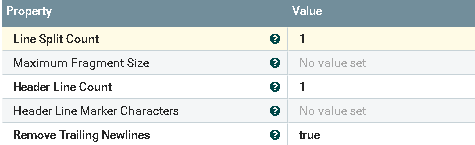
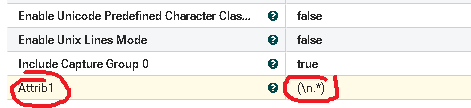
Create a ‘HiveConnectionPool’ service if you don’t already have one.

1. **Add** a processor of type ‘PutHDFS’ using relevant Properties and Relationships. 
2. **Create** connections from your ‘SelectHiveQL’ processor to your group and from your group to your ‘PutHDFS’ processor.

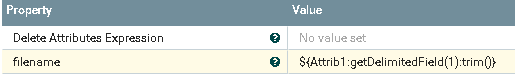


1. **Enter** your group

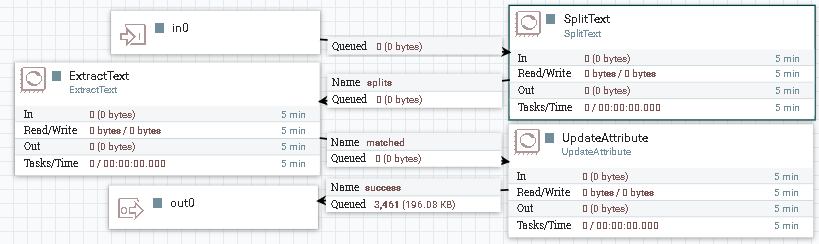
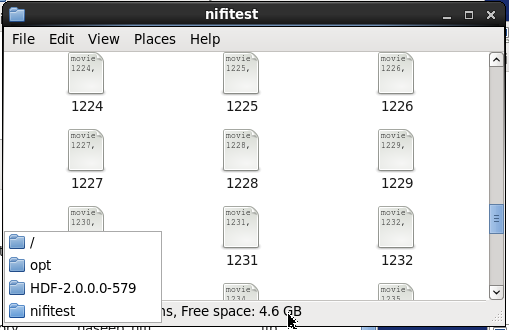


1. **Add** a processor of type ‘SplitText’ with the following Properties and Relationships:
2. **Add** a processor of type ‘ExtractText’ with the following Properties and Relationships:

Note: This property uses regular expressions, see end of exercise for more information.

1. **Add** a processor of type ‘UpdateAttribute’ with the following Properties:

Note: the value used returns the first field delimited (by comma) and trims the new line.

1. **Create** connections from your ‘Input Port’ to ‘SplitText’, from ‘SplitText’ to ‘ExtractText’ using ‘splits’, from ‘ExtractText’ to ‘UpdateAttribute’ using ‘matched’, and from ‘UpdateAttribute’ to ‘Output Port’.
2. **Leave** the group and **start** your dataflow. Your files should be named after their id

Regular Expressions: <https://docs.oracle.com/javase/7/docs/api/java/util/regex/Pattern.html>

Expression Language: <https://nifi.apache.org/docs/nifi-docs/html/expression-language-guide.html>

For a more in-depth guide: <https://nifi.apache.org/docs.html>