**Exp No: 4**

## Create UDF in PIG

## Step-by-step installation of Apache Pig on Hadoop cluster on Ubuntu

**Pre-requisite**:

· Ubuntu 16.04 or higher version running (I have installed Ubuntu on Oracle VM (Virtual Machine) VirtualBox),

· Run Hadoop on ubuntu (I have installed Hadoop 3.2.1 on Ubuntu 16.04). You may refer to my blog "How to install Hadoop installation" click here for Hadoop installation).

**Pig installation steps**

**Step 1:** Login into Ubuntu

**Step 2:** Go to https://pig.apache.org/releases.html and copy the path of the latest version of pig that you want to install. Run the following comment to download Apache Pig in Ubuntu:

```
$ wget https://dlcdn.apache.org/pig/pig-0.16.0/pig-0.16.0.tar.gz
```

**Step 3:** To untar pig-0.16.0.tar.gz file run the following command:

```
$ tar xvzf pig-0.16.0.tar.gz
```

**Step 4:** To create a pig folder and move pig-0.16.0 to the pig folder, execute the following command:

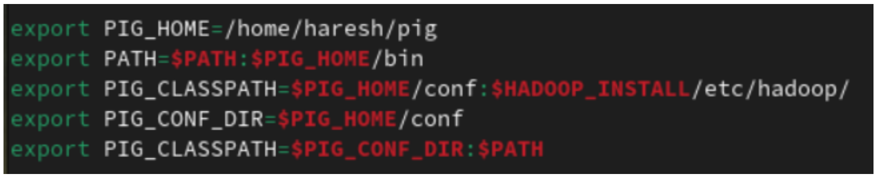
```
$ sudo mv /home/hdoop/pig-0.16.0 /home/hdoop/pig
```

**Step 5:** Now open the .bashrc file to edit the path and variables/settings for pig. Run the following command:

```
$ sudo nano .bashrc
```

Add the below given to .bashrc file at the end and save the file.

```
#PIG settingsexport PIG_HOME=/home/hdoop/pigexport
PATH=$PATH:$PIG_HOME/binexport
PIG_CLASSPATH=$PIG_HOME/conf:$HADOOP_INSTALL/etc/hadoop/export
PIG_CONF_DIR=$PIG_HOME/confexport JAVA_HOME=/usr/lib/jvm/java-8-openjdkamd64export
PIG_CLASSPATH=$PIG_CONF_DIR:$PATH#PIG setting ends
```



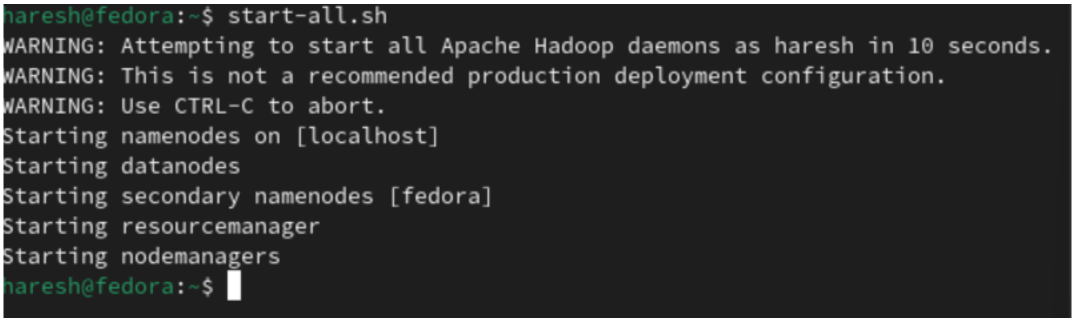
```
export PIG_HOME=/home/haresh/pig
export PATH=$PATH:$PIG_HOME/bin
export PIG_CLASSPATH=$PIG_HOME/conf:$HADOOP_INSTALL/etc/hadoop/
export PIG_CONF_DIR=$PIG_HOME/conf
export PIG_CLASSPATH=$PIG_CONF_DIR:$PATH
```

**Step 6:** Run the following command to make the changes effective in the .bashrc file:

$ source .bashrc

**Step 7:** To start all Hadoop daemons, navigate to the hadoop-3.2.1/sbin folder and run the following commands:

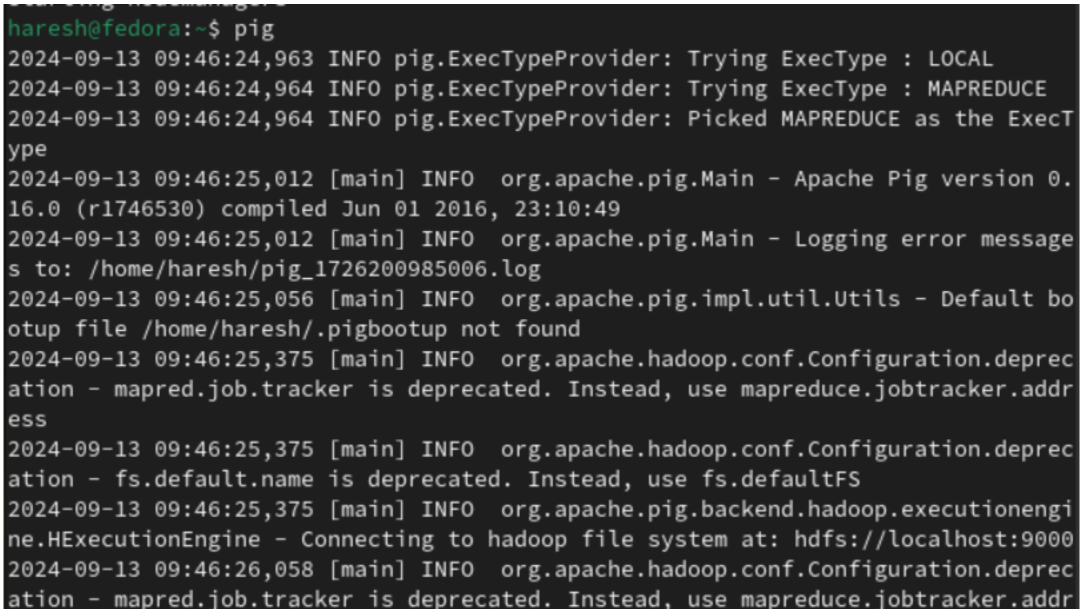
$ ./start-dfs.sh$ ./start-yarn$ jps



```
haresh@fedora:~$ start-all.sh
WARNING: Attempting to start all Apache Hadoop daemons as haresh in 10 seconds.
WARNING: This is not a recommended production deployment configuration.
WARNING: Use CTRL-C to abort.
Starting namenodes on [localhost]
Starting datanodes
Starting secondary namenodes [fedora]
Starting resourcemanager
Starting nodemanagers
haresh@fedora:~$
```

Now you can launch pig by executing the following command:

$ pig



```
haresh@fedora:~$ pig
2024-09-13 09:46:24,963 INFO pig.ExecTypeProvider: Trying ExecType : LOCAL
2024-09-13 09:46:24,964 INFO pig.ExecTypeProvider: Trying ExecType : MAPREDUCE
2024-09-13 09:46:24,964 INFO pig.ExecTypeProvider: Picked MAPREDUCE as the ExecT
ype
2024-09-13 09:46:25,012 [main] INFO  org.apache.pig.Main - Apache Pig version 0.
16.0 (r1746530) compiled Jun 01 2016, 23:10:49
2024-09-13 09:46:25,012 [main] INFO  org.apache.pig.Main - Logging error message
s to: /home/haresh/pig_1726200985006.log
2024-09-13 09:46:25,056 [main] INFO  org.apache.pig.impl.util.Utils - Default bo
otup file /home/haresh/.pigbootup not found
2024-09-13 09:46:25,375 [main] INFO  org.apache.hadoop.conf.Configuration.deprec
ation - mapred.job.tracker is deprecated. Instead, use mapreduce.jobtracker.addr
ess
2024-09-13 09:46:25,375 [main] INFO  org.apache.hadoop.conf.Configuration.deprec
ation - fs.default.name is deprecated. Instead, use fs.defaultFS
2024-09-13 09:46:25,375 [main] INFO  org.apache.pig.backend.hadoop.executionengi
ne.HExecutionEngine - Connecting to hadoop file system at: hdfs://localhost:9000
2024-09-13 09:46:26,058 [main] INFO  org.apache.hadoop.conf.Configuration.deprec
ation - mapred.job.tracker is deprecated. Instead, use mapreduce.jobtracker.addr
```

**Step 9:** Now you are in pig and can perform your desired tasks on pig. You can come out of the pig by the quit command:

> quit

## CREATE USER DEFINED FUNCTION(UDF)

**Aim :**

To create User Define Function in Apache Pig and execute it on map reduce.

**Procedure:**

Create a sample text file

```
hadoop@Ubuntu:~/Documents$ nano sample.txt
```

Paste the below content to sample.txt

```
1,John
2,Jane
3,Joe
4,Emma
```

```
hadoop@Ubuntu:~/Documents$ hadoop fs -put sample.txt /home/hadoop/piginput/
```

Create PIG File

```
hadoop@Ubuntu:~/Documents$ nano demo_pig.pig
```

paste the below the content to demo_pig.pig

```
-- Load the data from HDFS
data = LOAD '/home/hadoop/piginput/sample.txt' USING PigStorage(',') AS (id:int>
-- Dump the data to check if it was loaded correctly
DUMP data;
--------------------------------------------------------------------------------
```

**Run the above file**

```
hadoop@Ubuntu:~/Documents$ pig demo_pig.pig
2024-08-07 12:13:08,791 [main] INFO
org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil
- Total input paths to process : 1
(1,John)
(2,Jane)
(3,Joe)
(4,Emma)
--------------------------------------------------------------------------------------------------
```

**Create udf file an save as uppercase_udf.py**

uppercase_udf.py

```
------------------------------------------------------------------------------------------
def uppercase(text):
return text.upper()
if __name__ == "__main__":
import sys
for line in sys.stdin:
line = line.strip()
result = uppercase(line)
print(result)
-----------------------------------------------------------------------------------------
```

**Create the udfs folder on hadoop**

```
hadoop@Ubuntu:~/Documents$ hadoop fs -mkdir /home/hadoop/udfs
```

put the upppercase_udf.py in to the abv folder

```
hadoop@Ubuntu:~/Documents$ hdfs dfs -put uppercase_udf.py /home/hadoop/udfs/
-------------------------------------------------------------------------------
hadoop@Ubuntu:~/Documents$ nano udf_example.pig
```

copy and paste the below content on udf_example.pig

```
-- Register the Python UDF script
REGISTER 'hdfs:///home/hadoop/udfs/uppercase_udf.py' USING jython AS udf;
-- Load some data
data = LOAD 'hdfs:///home/hadoop/sample.txt' AS (text:chararray);
-- Use the Python UDF
uppercased_data = FOREACH data GENERATE udf.uppercase(text) AS uppercase_text;
-- Store the result
STORE uppercased_data INTO 'hdfs:///home/hadoop/pig_output_data';
-----------------------------------------------------------------------------------------------------------
```

**place sample.txt file on hadoop**

```
hadoop@Ubuntu:~/Documents$ hadoop fs -put sample.txt /home/hadoop/
```

**To Run the pig file**

hadoop@Ubuntu:~/Documents$ pig -f udf_example.pig

**finally u get**

**Success!**

**Job Stats (time in seconds):**

JobId Maps Reduces MaxMapTimeMinMapTime AvgMapTime MedianMapTime

MaxReduceTime MinReduceTime AvgReduceTime MedianReducetime

Alias Feature Outputs

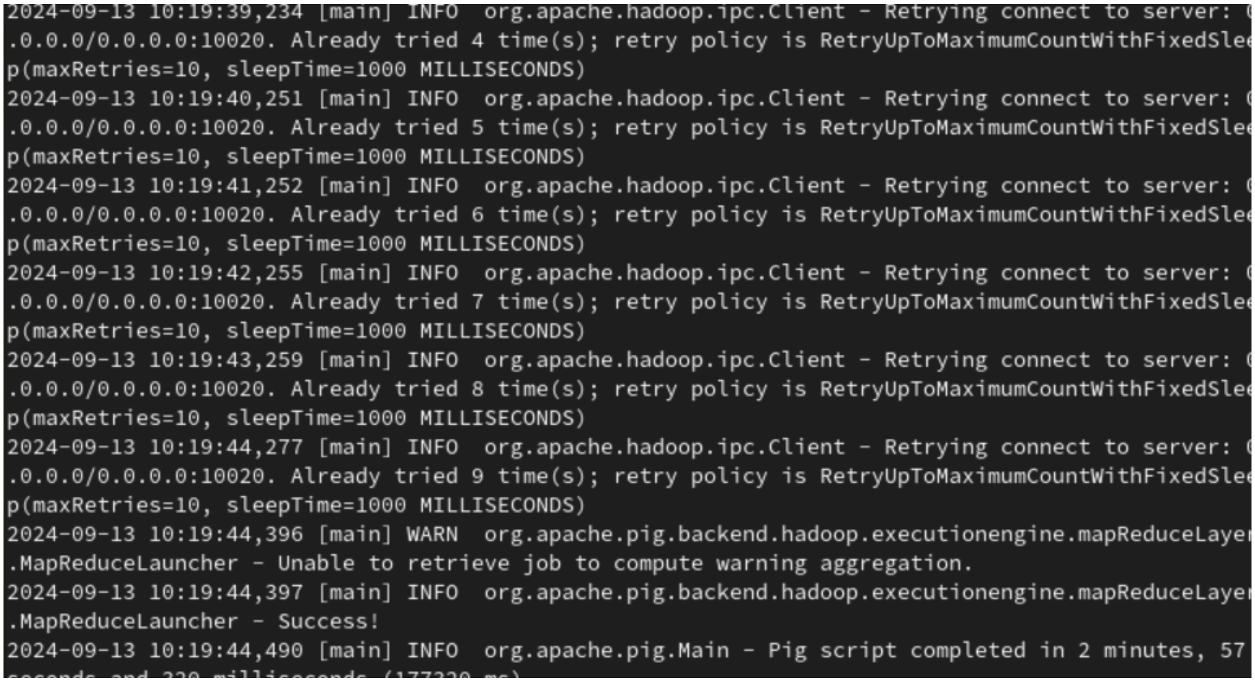
job_local1786848041_0001 1 0 n/a n/a n/a n/a 00 0 0

data,uppercased_data MAP_ONLY hdfs:///home/hadoop/pig_output_data,

Input(s):

Successfully read 4 records (42778068 bytes) from: "hdfs:///home/hadoop/sample.txt"

Output(s):



```
2024-09-13 10:19:39,234 [main] INFO  org.apache.hadoop.ipc.Client - Retrying connect to server: 0
.0.0.0/0.0.0.0:10020. Already tried 4 time(s); retry policy is RetryUpToMaximumCountWithFixedSlee
p(maxRetries=10, sleepTime=1000 MILLISECONDS)
2024-09-13 10:19:40,251 [main] INFO  org.apache.hadoop.ipc.Client - Retrying connect to server: 0
.0.0.0/0.0.0.0:10020. Already tried 5 time(s); retry policy is RetryUpToMaximumCountWithFixedSlee
p(maxRetries=10, sleepTime=1000 MILLISECONDS)
2024-09-13 10:19:41,252 [main] INFO  org.apache.hadoop.ipc.Client - Retrying connect to server: 0
.0.0.0/0.0.0.0:10020. Already tried 6 time(s); retry policy is RetryUpToMaximumCountWithFixedSlee
p(maxRetries=10, sleepTime=1000 MILLISECONDS)
2024-09-13 10:19:42,255 [main] INFO  org.apache.hadoop.ipc.Client - Retrying connect to server: 0
.0.0.0/0.0.0.0:10020. Already tried 7 time(s); retry policy is RetryUpToMaximumCountWithFixedSlee
p(maxRetries=10, sleepTime=1000 MILLISECONDS)
2024-09-13 10:19:43,259 [main] INFO  org.apache.hadoop.ipc.Client - Retrying connect to server: 0
.0.0.0/0.0.0.0:10020. Already tried 8 time(s); retry policy is RetryUpToMaximumCountWithFixedSlee
p(maxRetries=10, sleepTime=1000 MILLISECONDS)
2024-09-13 10:19:44,277 [main] INFO  org.apache.hadoop.ipc.Client - Retrying connect to server: 0
.0.0.0/0.0.0.0:10020. Already tried 9 time(s); retry policy is RetryUpToMaximumCountWithFixedSlee
p(maxRetries=10, sleepTime=1000 MILLISECONDS)
2024-09-13 10:19:44,396 [main] WARN  org.apache.pig.backend.hadoop.executionengine.mapReduceLayer
.MapReduceLauncher - Unable to retrieve job to compute warning aggregation.
2024-09-13 10:19:44,397 [main] INFO  org.apache.pig.backend.hadoop.executionengine.mapReduceLayer
.MapReduceLauncher - Success!
2024-09-13 10:19:44,490 [main] INFO  org.apache.pig.Main - Pig script completed in 2 minutes, 57
seconds and 320 milliseconds (177320 ms)
```

Successfully stored 4 records (42777870 bytes) in: "hdfs:///home/hadoop/pig_output_data"

Counters:

Total records written : 4

Total bytes written : 42777870

Spillable Memory Manager spill count : 0

```
Total bags proactively spilled: 0
Total records proactively spilled: 0
Job DAG:
job_local1786848041_0001
2024-08-07 13:33:04,631 [main] WARN
org.apache.hadoop.metrics2.impl.MetricsSystemImpl -
JobTracker metrics system already initialized!
2024-08-07 13:33:04,639 [main] WARN
org.apache.hadoop.metrics2.impl.MetricsSystemImpl -
JobTracker metrics system already initialized!
2024-08-07 13:33:04,644 [main] WARN
org.apache.hadoop.metrics2.impl.MetricsSystemImpl -
JobTracker metrics system already initialized!
2024-08-07 13:33:04,667 [main] INFO
org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher -
Success!
```

**Note :**

**If any error check jython package is installed and check the path specified on the above steps are give correctly**

----------------------------------------------------------------------------------------------------------------

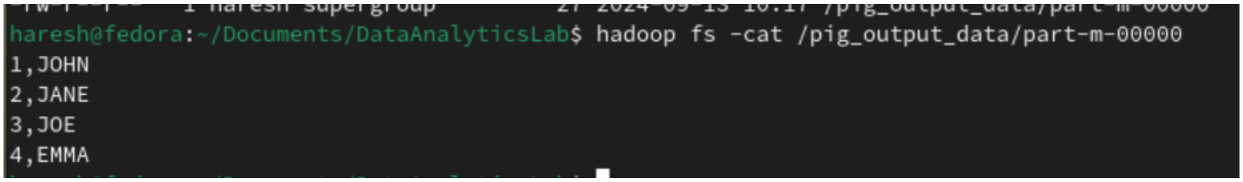
**To check the output file is created**

```
hadoop@Ubuntu:~/Documents$ hdfs dfs -ls /home/hadoop/pig_output_data
Found 2 items
```

If you need to examine the files in the output folder, use:

**To view the output**

```
hadoop@Ubuntu:~/Documents$ hdfs dfs -cat /home/hadoop/pig_output_data/part-m00000
1,JOHN
2,JANE
3,JOE
4,EMMA
```



```
-rw-r--r--   1 haresh supergroup         27 2024-09-13 10:17 /pig_output_data/part-m-00000
haresh@fedora:~/Documents/DataAnalyticsLab$ hadoop fs -cat /pig_output_data/part-m-00000
1,JOHN
2,JANE
3,JOE
4,EMMA
haresh@fedora:~/Documents/DataAnalyticsLab$
```

**Result:**

Thus, the program is executed successfully