EXP 4: Create UDF (User Defined Functions) in Apache Pigand execute it in MapReduce / HDFS mode

AIM:

To create UDF in Apache Pig and execute it in MapReduce/HDFS mode.

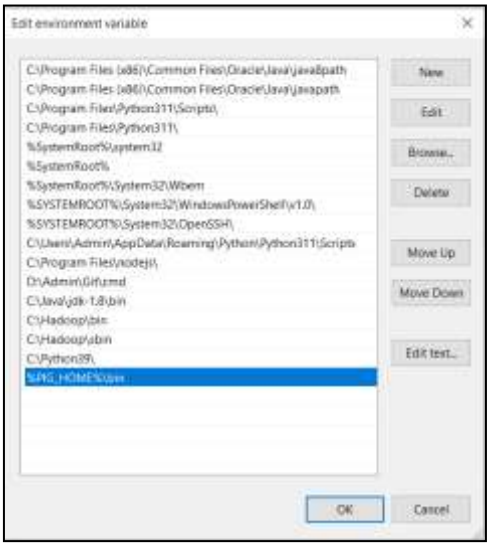
PROCEDURE:

Pig Download and installation:

1. Download Pig:

Download Pig from "https://downloads.apache.org/pig/pig-0.17.0/"

2. Add the environment variable for Pig:



3. Go to C:\pig-0.17.0\bin and open pig (Windows Command Script)

```
set HADOOP_BIN_PATH=%HADOOP_HOME%\libexec
```

4. Open Windows Powershell and type "pig –x local" and check whether pig grunt appears.

Pig is successfully installed.

Create UDF:

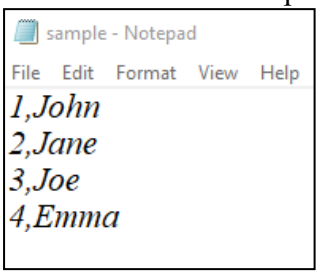
1. Start Hadoop services:

Open command prompt as an administrator

```
start-dfs.cmd
start-yarn.cmd
```

2. Open the browser and go to the URL "localhost:9870"

3. Create a text file "sample.txt":



4. Create a Directory in HDFS and copy the Input File to HDFS

```
hdfs dfs –mkdir /UDF
```

```
hadoop fs -put C:/Users/user/Documents/Pig/sample.txt /UDF
```

```
C:\hadoop\sbin>hdfs dfs -mkdir /UDF
C:\hadoop\sbin>hadoop fs -put C:/Users/user/Documents/Pig/sample.txt /UDF
```

5. Create a Python file "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)
```

6. Create a Directory in HDFS and copy the Input File to HDFS

```
hdfs dfs –mkdir /UDF/udfs
```

```
hadoop fs -put C:/Users/user/Documents/Pig/Uppercase_udf.py /UDF/udfs
```

```
C:\hadoop\sbin>hdfs dfs -mkdir /UDF/udfs
C:\hadoop\sbin>hadoop fs -put C:/Users/user/Documents/Pig /Uppercase_udf.py /UDF/udfs
```

7. Create pig file "UDF.pig":

UDF - Notepad

File Edit Format View Help

```
-- udf_example.pig

-- Register the Python UDF script
REGISTER 'hdfs:///UDF/udfs/Uppercase_udf.py' USING jython AS udf;

-- Load some data
data = LOAD 'hdfs:///UDF/sample.txt' USING PigStorage(',') AS (id:int, name:chararray);

-- Use the Python UDF to convert names to uppercase
uppercased_data = FOREACH data GENERATE id, udf.uppercase(name) AS uppercase_name;

-- Store the result
STORE uppercased_data INTO 'hdfs:///UDF/output' USING PigStorage(',');
```

8. Execute Pig file

```
pig -x mapreduce C:/Users/user/Documents/Pig/UDF.pig
```

9. View the Output

hdfs dfs -ls /UDF/output

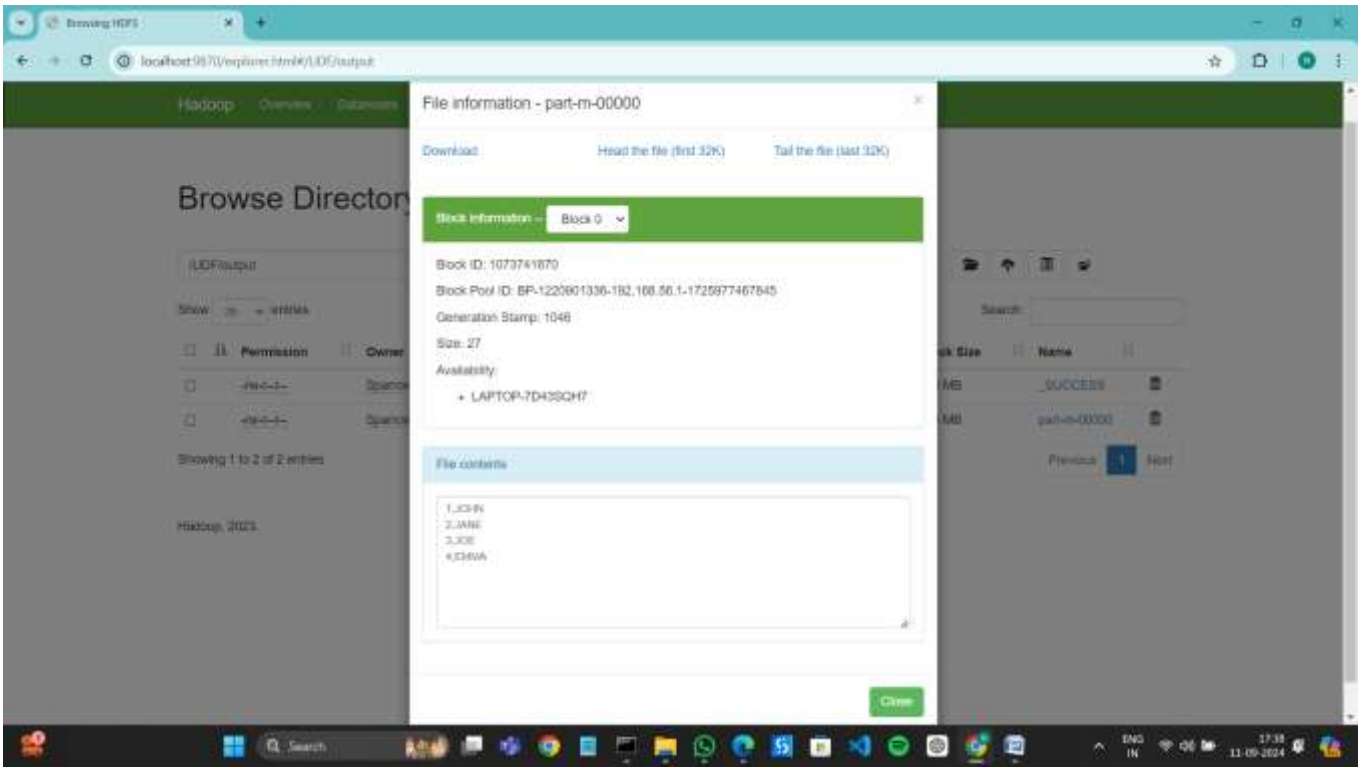
hdfs dfs -cat /UDF/output/part-m-00000

```
C:\hadoop\sbin>hdfs dfs -ls /UDF/output
Found 2 items
-rw-r--r--   1 user supergroup          0 2024-08-29 22:12 /UDF/output/_SUCCESS
-rw-r--r--   1 user supergroup         27 2024-08-29 22:12 /UDF/output/part-m-00000

C:\hadoop\sbin>hdfs dfs -cat /UDF/output/part-m-00000
1,JOHN
2,JANE
3,JOE
4,EMMA
```

10. Once the map reduce operations are performed successfully, the output will be present in the specified directory.

"/UDF/output/part-m-00000"



RESULT:

Thus, UDF in Apache Pig has been created and executed in MapReduce/HDFS mode successfully.