Ex No 4

Create UDF (User Defined Functions) in Apache Pig and execute it in MapReduce / HDFS mode

AIM:

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

PROCEDURE:

1. Install Apache Pig

Download Pig:

1. Download Pig from the Apache Pig download page:

Link: Apache Pig 0.17.0 Download

Extract the downloaded file (assuming you downloaded `pig-0.17.0.tar.gz`):`tar -`

`xzf pig-0.17.0.tar.gz`

Move the extracted folder to a directory, such as `/usr/local/`:`sudo`

`mv pig-0.17.0 /usr/local/pig`

2. Set Up Environment Variables for Pig

Edit your `~/._profile` or `~/.zshrc` to include Pig in the PATH.

`nano ~/.zshrc`

Add the following lines:

`export PIG_HOME=/usr/local/pig`
`export PATH=$PIG_HOME/bin:$PATH`

Apply the changes:

`source ~/.zshrc`

3. Verify Pig Installation

Run the following command to check if Pig is installed correctly:

```
pig -x local
```

You should see the Pig Grunt shell prompt:

```
grunt>
```

Type `quit` to exit the shell.

4. Start Hadoop Services

Make sure your Hadoop is up and running. Start the required services:

```
cd /usr/local/hadoop/sbin
./start-dfs.sh
./start-yarn.sh
```

5. Prepare Input Data (`ex4.txt`)

Create a sample text file for testing the UDF, named `ex4.txt`:nano

```
ex4.txt
```

Example content:

```
1,John
2,Soniya
3,Vijay
4,Sonu
```

Upload the file to HDFS:

```
hdfs dfs -mkdir /UDF
hdfs dfs -put ex4.txt /UDF/
```

6. Create UDF in Python

Now, you need to write your Python UDF.

Create a Python file `uppercase_udf.py`:

```
nano uppercase_udf.py
```

Add the following code to `uppercase_udf.py`:

```python
#!/usr/bin/python3
def uppercase(text):
    return text.upper()


if___name___== "___main___":
    import sys
    for line in sys.stdin:
        line = line.strip()
        result = uppercase(line)
        print(result)
```

Upload the Python UDF to HDFS:

```
hdfs dfs -mkdir /UDF/udfs
hdfs dfs -put uppercase_udf.py /UDF/udfs/
```

Make sure the file is in the correct HDFS directory by running:

```
hdfs dfs -ls /UDF/udfs
```

7. Write Pig Script (`UDF.pig`)

Create a Pig script to apply your UDF.Create

`UDF.pig`:

```
nano UDF.pig
```

Add the following Pig script to `UDF.pig`:

```
-- Register the UDF
REGISTER hdfs:///UDF/udfs/uppercase_udf.py USING jython AS myudfs;

-- Load the ex4.txt file from HDFS
data = LOAD 'hdfs:///UDF/ex4.txt' USING PigStorage(',') AS
(id:int,name:chararray);

-- Apply the UDF to each line
uppercase_data = FOREACH data GENERATE myudfs.uppercase(name) AS
upper_line;
```

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

Save the file and exit.

8. Run the Pig Script in MapReduce Mode

Now that everything is set up, execute the Pig script in MapReduce mode:

hdfs dfs -chmod 755 /UDF/udfs/uppercase_udf.pyhdfs dfs
-chmod 755 /UDF
hdfs dfs -chmod 755 /UDF/ex4.txt

```
pig -x mapreduce UDF.pig
```

9. Check the Output

After the job finishes, you can view the output in HDFS.List the

output directory:

```
hdfs dfs -ls /UDF/output
```

You should see something like:

```
Found 1 items
-rw-r--r--   3 user group      123 2024-09-11 12:00
/UDF/output/part-m-00000
```

View the output file:

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

You should see the content in uppercase

OUTPUT:

```
Last login: Tue Sep 10 20:08:42 on ttys002
nativewit@Nativewits-MacBook-Air ~ %

cd /usr/local/Cellar/hadoop/3.4.0/libexec/sbin

nativewit@Nativewits-MacBook-Air sbin % ./start-dfs.sh

Starting namenodes on [localhost]
Starting datanodes
Starting secondary namenodes [Nativewits-MacBook-Air.local]
2024-09-10 20:35:20,894 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
nativewit@Nativewits-MacBook-Air sbin % ./start-yarn.sh

Starting resourcemanager
Starting nodemanagers
nativewit@Nativewits-MacBook-Air sbin % nano ex4.txt

nativewit@Nativewits-MacBook-Air sbin % hdfs dfs -mkdir /UDF

2024-09-10 20:36:25,288 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
nativewit@Nativewits-MacBook-Air sbin % hdfs dfs -put ex4.txt /UDF/

2024-09-10 20:36:31,300 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
nativewit@Nativewits-MacBook-Air sbin % nano uppercase_udf.py

nativewit@Nativewits-MacBook-Air sbin % hdfs dfs -mkdir /UDF/udfs

2024-09-10 20:37:00,810 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
nativewit@Nativewits-MacBook-Air sbin % hdfs dfs -put uppercase_udf.py /UDF/udfs/

2024-09-10 20:37:06,897 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
nativewit@Nativewits-MacBook-Air sbin % hdfs dfs -ls /UDF/udfs

2024-09-10 20:37:12,402 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 1 items
-rw-r--r--   1 nativewit supergroup        219 2024-09-10 20:37 /UDF/udfs/uppercase_udf.py
nativewit@Nativewits-MacBook-Air sbin % nano UDF.pig

nativewit@Nativewits-MacBook-Air sbin % hdfs dfs -chmod 755 /UDF/udfs/uppercase_udf.py

2024-09-10 20:38:16,221 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
nativewit@Nativewits-MacBook-Air sbin % hdfs dfs -chmod 755 /UDF

2024-09-10 20:38:20,614 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
nativewit@Nativewits-MacBook-Air sbin % hdfs dfs -chmod 755 /UDF/ex4.txt

2024-09-10 20:38:25,425 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
nativewit@Nativewits-MacBook-Air sbin % pig -x mapreduce UDF.pig

2024-09-10 20:38:36,793 INFO pig.ExecTypeProvider: Trying ExecType : LOCAL
2024-09-10 20:38:36,795 INFO pig.ExecTypeProvider: Trying ExecType : MAPREDUCE
2024-09-10 20:38:36,795 INFO pig.ExecTypeProvider: Picked MAPREDUCE as the ExecType
```

RESULT:

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