Expt-2:

Run a basic Word Count Map Reduce program to understand Map Reduce Paradigm.

AIM:

To run a basic Word Count MapReduce program using Hadoop.

PROCEDURE:

1. **Create Data File**:

   nano word_count_data.txt

   Example content for word_count_data.txt:

   Hadoop is a framework that allows for distributed processing of large data sets.

2. **Mapper Program (mapper.py)**:

```
import sys
for line in sys.stdin:
    line = line.strip()
    words = line.split()
    for word in words:
        print(f'{word}\t1')
```

3. **Reducer Program (reducer.py)**:

```
import sys
current_word = None
current_count = 0
word = None

for line in sys.stdin:
    line = line.strip()
    word, count = line.split('\t', 1)

    try:
        count = int(count)
    except ValueError:
        continue
```

```
        if current_word == word:
            current_count += count
        else:
            if current_word:
                print(f'{current_word}\t{current_count}')
            current_count = count
            current_word = word

    if current_word == word:
        print(f'{current_word}\t{current_count}')
```

4. **Set Hadoop Environment**:

```
hdfs dfs -mkdir /word_count_input
hdfs dfs -copyFromLocal word_count_data.txt /word_count_input
```

5. **Run Word Count Program**:

```
hadoop jar $HADOOP_HOME/share/hadoop/tools/lib/hadoop-streaming-*.jar \
-input /word_count_input/word_count_data.txt \
-output /word_count_output \
-mapper mapper.py \
-reducer reducer.py
```

6. **Check Output**:

```
hdfs dfs -cat /word_count_output/part-00000
```

OUTPUT:

```
suriya@Ubuntu:~$ hdfs dfs -cat /WordCount/Output/part-r-00000
2024-09-28 23:06:32,621 WARN util.NativeCodeLoader: Unable to load native-hadoop
 library for your platform... using builtin-java classes where applicable
Hi      1
am      1
are     2
fine    2
hi      1
how     1
i       1
you     1
suriya@Ubuntu:~$
```

RESULT:

Thus, the program for basic Word Count Map Reduce has been executed successfully.