Exp No: 11 **Date:**

HADOOP
IMPLEMENT THE MAX TEMPERATURE MAPREDUCE PROGRAM TO IDENTIFY THE YEAR WISE MAXIMUM TEMPERATURE FROM SENSOR DATA

AIM

To implement the Max temperature MapReduce program to identify the year-wise maximum temperature from the sensor data.

Description

Sensors senses weather data in big text format containing station ID, year, date, time, temperature, quality etc. from each sensor and store it in a single line. Suppose thousands of data sensors are there, then we have thousands of records with no particular order. We require only a year and maximum temperature of particular quality in that year.

For example:

Input string from sensor:
002902907099999*1902*010720004+64333+023450
FM-12+
000599999V0202501N027819999999N0000001N9-*00331*+
99999098351ADDGF102991999999999999999999

Here: 1902 is year

0033 is temperature

1 is measurement quality (Range between 0 or 1 or 4 or 5 or 9)

Here each mapper takes the input key as "byte offset of line" and value as "one weather sensor read i.e one line". and parse each line and produce an intermediate key "year" and intermediate value as "temperature of certain measurement qualities" for that year.

The combiner will form set values of temperature. Year and set of values of temperatures is given as input <key, value> to reducer and Reducer will produce year and maximum temperature for that year from the set of temperature values.

PROGRAM

*/

```
import  org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
import java.io.IOException;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
importorg.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;
importorg.apache.hadoop.mapreduce.Reducer;

//Mapper class


class MaxTemperatureMapper
extends Mapper<LongWritable, Text, Text, IntWritable> { private static final int MISSING

= 9999;

@Override
public void map(LongWritable key, Text value, Context context) throws IOException,
InterruptedException {

String line = value.toString(); String year = line.substring(15, 19); int airTemperature;
if (line.charAt(87) == '+') { // parseInt doesn't like leading plus signs airTemperature =
Integer.parseInt(line.substring(88, 92));
} else {
airTemperature = Integer.parseInt(line.substring(87, 92));
}
String quality = line.substring(92, 93);
if (airTemperature != MISSING && quality.matches("[01459]")) { context.write(new
Text(year), new IntWritable(airTemperature));
}
}
}

//Reducer class
class MaxTemperatureReducer
extends Reducer<Text, IntWritable, Text, IntWritable> {

@Override
public void reduce(Text key, Iterable<IntWritable> values, Context context)
throws IOException, InterruptedException {
```

```
int maxValue = Integer.MIN_VALUE; for (IntWritable value : values) {
maxValue = Math.max(maxValue, value.get());
}
context.write(key, new IntWritable(maxValue));
}
}
//Driver Class

public class MaxTemperature {

public static void main(String[] args) throws Exception { if (args.length != 2) {
System.err.println("Usage: MaxTemperature <input path=""> <output path>"); System.exit(-
1);
}

Job job = Job.getInstance(new Configuration()); job.setJarByClass(MaxTemperature.class);
job.setJobName("Max temperature");

FileInputFormat.addInputPath(job, new Path(args[0])); FileOutputFormat.setOutputPath(job,
new Path(args[1]));


job.setMapperClass(MaxTemperatureMapper.class);
job.setReducerClass(MaxTemperatureReducer.class);

job.setOutputKeyClass(Text.class); job.setOutputValueClass(IntWritable.class);

job.submit();
}
}
```

OUTPUT:

Input for String :

002902907099999*1902*010720004+64333+023450FM-12+
000599999V0202501N027819999999N0000001N9-*00331*+
99999098351ADDGF10299199999999999999’

```
hayagreevan@fedora:~/cc$ start-all.sh
WARNING: Attempting to start all Apache Hadoop daemons as hayagreevan in 10 seco
nds.
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
hayagreevan@fedora:~/cc$
```

```
hayagreevan@fedora:~/cc$ jps
4736 SecondaryNameNode
4515 DataNode
5636 Jps
4358 NameNode
5067 ResourceManager
5212 NodeManager
hayagreevan@fedora:~/cc$
```

```
hayagreevan@fedora:~/cc$ hdfs dfs -ls /exp3
Found 2 items
-rw-r--r--   1 hayagreevan supergroup      79568 2024-08-28 12:27 /exp3/dataset.
txt
drwxr-xr-x   - hayagreevan supergroup          0 2024-08-28 12:29 /exp3/output
hayagreevan@fedora:~/cc$
```

```
hayagreevan@fedora:~/da_lab/exp3$ hadoop jar ~/da_lab/exp3/hadoop-streaming-3.3.6.jar -input /exp3/dataset.txt -output /exp3/new_output -mapper ~/da_lab/exp3/mapper.py -reducer ~/da_lab/exp3/reducer.py

packageJobJar: [/tmp/hadoop-unjar8521919454352244797/] [] /tmp/streamjob5327111103252164825.jar tmpDir=null
2024-11-17 00:28:26,737 INFO client.DefaultNoHARMFailoverProxyProvider: Connecting to ResourceManager at /0.0.0.0:8032
2024-11-17 00:28:28,608 INFO client.DefaultNoHARMFailoverProxyProvider: Connecting to ResourceManager at /0.0.0.0:8032
2024-11-17 00:28:30,688 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/hayagreevan/.staging/job_1731783187439_0001
2024-11-17 00:28:34,577 INFO mapred.FileInputFormat: Total input files to process : 1
2024-11-17 00:28:35,836 INFO mapreduce.JobSubmitter: number of splits:2
2024-11-17 00:28:39,081 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1731783187439_0001
2024-11-17 00:28:39,082 INFO mapreduce.JobSubmitter: Executing with tokens: []
2024-11-17 00:28:41,235 INFO conf.Configuration: resource-types.xml not found
2024-11-17 00:28:41,237 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2024-11-17 00:28:45,315 INFO impl.YarnClientImpl: Submitted application application_1731783187439_0001
2024-11-17 00:28:45,662 INFO mapreduce.Job: The url to track the job: http://fedora:8088/proxy/application_1731783187439_0001/
2024-11-17 00:28:45,669 INFO mapreduce.Job: Running job: job_1731783187439_0001
2024-11-17 00:29:22,616 INFO mapreduce.Job: Job job_1731783187439_0001 running in uber mode : false
2024-11-17 00:29:22,623 INFO mapreduce.Job:  map 0% reduce 0%
2024-11-17 00:29:53,005 INFO mapreduce.Job:  map 100% reduce 0%
2024-11-17 00:30:17,134 INFO mapreduce.Job:  map 100% reduce 100%
2024-11-17 00:30:20,264 INFO mapreduce.Job: Job job_1731783187439_0001 completed successfully
2024-11-17 00:30:21,871 INFO mapreduce.Job: Counters: 54
        File System Counters
                FILE: Number of bytes read=102094
                FILE: Number of bytes written=1041712
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=83844
                HDFS: Number of bytes written=96
                HDFS: Number of read operations=11
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
                HDFS: Number of bytes read erasure-coded=0
        Job Counters
                Launched map tasks=2
                Launched reduce tasks=1
                Data-local map tasks=2
                Total time spent by all maps in occupied slots (ms)=51317
                Total time spent by all reduces in occupied slots (ms)=21181
                Total time spent by all map tasks (ms)=51317
                Total time spent by all reduce tasks (ms)=21181
                Total vcore-milliseconds taken by all map tasks=51317
                Total vcore-milliseconds taken by all reduce tasks=21181
                Total megabyte-milliseconds taken by all map tasks=52548608
                Total megabyte-milliseconds taken by all reduce tasks=21689344
        Map-Reduce Framework
```

```
        Map-Reduce Framework
                Map input records=365
                Map output records=10220
                Map output bytes=81648
                Map output materialized bytes=102100
                Input split bytes=180
                Combine input records=0
                Combine output records=0
                Reduce input groups=12
                Reduce shuffle bytes=102100
                Reduce input records=10220
                Reduce output records=12
                Spilled Records=20440
                Shuffled Maps =2
                Failed Shuffles=0
                Merged Map outputs=2
                GC time elapsed (ms)=1132
                CPU time spent (ms)=17300
                Physical memory (bytes) snapshot=877088768
                Virtual memory (bytes) snapshot=7757271040
                Total committed heap usage (bytes)=695205888
                Peak Map Physical memory (bytes)=323620864
                Peak Map Virtual memory (bytes)=2586337280
                Peak Reduce Physical memory (bytes)=230580224
                Peak Reduce Virtual memory (bytes)=2589831168
        Shuffle Errors
                BAD_ID=0
                CONNECTION=0
                IO_ERROR=0
                WRONG_LENGTH=0
                WRONG_MAP=0
                WRONG_REDUCE=0
        File Input Format Counters
                Bytes Read=83664
        File Output Format Counters
                Bytes Written=96
2024-11-17 00:30:21,880 INFO streaming.StreamJob: Output directory: /exp3/new_output
hayagreevan@fedora:~/da_lab/exp3$
```

```
hayagreevan@fedora:~/da_lab/exp3$ hdfs dfs -cat /exp3/new_output/*
01      26.5
02      26.6
03      29.1
04      30.8
05      31.1
06      33.6
07      38.5
08      40.2
09      36.5
10      36.9
11      27.6
12      25.9
```

RESULT

Thus a java program has been implemented to identify the year-wise maximum temperature from the sensor data.