

Q. Find average temperature for each year from NDBC data set.

```
package temp;
```

```
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;
```

```
@
```

```
public class AverageDriver {
```

```
    public static void main (String[] args) throws Exception {
```

```
        if (args.length != 2) {
```

```
            System.out.println ("please enter input & output  
parameters");
```

```
            System.exit (-1);
```

```
        }
```

```
        Job job = new Job();
```

```
        job.setJarByClass (AverageDriver.class);
```

```
        job.setJobName ("max temperature");
```

```
        FileInputFormat.addInputPath (job, new  
            Path (args[0]));
```

```
        FileOutputFormat.setOutputPath (job,  
            new Path (args[1]));
```

```
        job.setMapperClass (AverageMapper.class);
```

```
        job.setReducerClass (AverageMap
```

```
}
```

SPR
29/4/24

```
}
```

Reducer Code :-

/* import all the required library */

```
public class WCReducer extends MapReducer.  
Base implements Reducer<Text, implements  
Text, IntWritable> {
```

```
public void reduce (Text key, Iterator  
<IntWritable> value, OutputCollector<  
Text, IntWritable> output,  
Reporter rep) throws IOException
```

```
{
```

```
    int count = 0;
```

```
    while (value.hasNext()) {
```

```
        IntWritable i = value.next();
```

```
        count += i.get();
```

```
    }
```

```
    output.collect (key, new  
        IntWritable (count));
```

```
}
```

```
}
```

S.P. J.

Word count program map reduce:

Mappercode . java .

Public class Mappercode mapper.
& long variable

WC Driver code

import java.io.IOException;

public class WCMapper extends MapReduce.
Base implements Mapper<LongWritable, Text
Text, IntWritable> {

public void map(LongWritable key, Text
value, OutputCollector<Text> Int
Writable> output, Reporter rep)
throws IOException {

String line = value.toString();

for (String word : line.split(" "))

{

if (word.length() > 0) {

output.collect(new Text(word),
new IntWritable(1));

}

}

}

}

}