SL-V BE IT
EXP 5 Part A [According to new revised assignments]

Aim: Design a distributed application using MapReduce under Hadoop for:
a) Character counting in a given text file.

Steps:
First install hadoop (if not installed yet) by,
https://sl6it.wordpress.com/2015/12/04/1-study-and-configure-hadoop-for-big-data/

Download **sample.txt** file (attached with this post)
Paste sample.txt in your **home** folder
Open terminal
whoami
It will display your user name, we will use it later.

Open eclipse->new java project->project name **exp5a**->new class-> **CharMap**
Add following code in that class

```
package exp5a;
import java.io.IOException;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;

public class CharMap extends Mapper<LongWritable, Text, Text, IntWritable> {
    public void map(LongWritable key, Text value, Context context)
            throws IOException, InterruptedException {

        String line = value.toString();
        char[] carr = line.toCharArray();
        for (char c : carr) {
            System.out.println(c);
            context.write(new Text(String.valueOf(c)), new IntWritable(1));
        }

    }

}
```

Save the file

It will display some errors, so we are going to import three jar files in our project.

Copy hadoop-mapreduce-client-core-2.7.1.jar from ~/hadoop/share/hadoop/mapreduce directory
In eclipse-> right click on exp5a project- >paste
Right click on pasted hadoop-mapreduce-client-core-2.7.1.jar-> Buid path-> add to buid path

#Copy hadoop-common-2.7.1.jar from ~/hadoop/share/hadoop/common directory
In eclipse-> right click on exp5a project- >paste
Right click on pasted hadoop-common-2.7.1.jar-> Buid path-> add to buid path

#Copy commons-cli-1.2.jar from ~/hadoop/share/hadoop/common/lib directory
In eclipse-> right click on exp5a project- >paste
Right click on pasted commons-cli-1.2.jar-> Buid path-> add to buid path

In eclipse->right click on project **exp5a**->new class-> **CharReduce**
Add following code in that class

```
package exp5a;
import java.io.IOException;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Reducer;

public class CharReduce extends Reducer<Text, IntWritable, Text, IntWritable> {
    public void reduce(Text key,Iterable<IntWritable> values,Context
context)throws IOException,InterruptedException{
        int count = 0;
        IntWritable result = new IntWritable();
        for (IntWritable val : values) {
            count +=val.get();
            result.set(count);
        }
        context.write(key, result);
     }
}
```

Save the file

In eclipse->right click on project **exp5a**->new class-> **CharCount**
Add following code in that class

```
package exp5a;
import org.apache.hadoop.conf.Configuration;
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.input.TextInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
import org.apache.hadoop.mapreduce.lib.output.TextOutputFormat;

public class CharCount {
    public static void main(String[] args) throws Exception {
        // TODO Auto-generated method stub
        Configuration conf = new Configuration();
        @SuppressWarnings("deprecation")
            Job job = new Job(conf, "Charcount");
        job.setJarByClass(CharCount.class);
        job.setMapperClass(CharMap.class);
        job.setReducerClass(CharReduce.class);
        job.setInputFormatClass(TextInputFormat.class);
        job.setOutputFormatClass(TextOutputFormat.class);
        job.setMapOutputKeyClass(Text.class);
        job.setMapOutputValueClass(IntWritable.class);
        job.setOutputKeyClass(Text.class);
        job.setOutputValueClass(IntWritable.class);
        FileInputFormat.addInputPath(job, new Path(args[0]));
        FileOutputFormat.setOutputPath(job, new Path(args[1]));
        System.exit(job.waitForCompletion(true) ? 0 : 1);
    }
}
```

Save the file

In eclipse->Right click on project exp5a-> export->java->jar file->next-> select the export destination -> /home/**your_user_name**/exp5a.jar -> next -> next -> select main class ->browse -> **`CharCount`** -> finish

exp5a.jar file will be created in your home folder

Open terminal

Now Start NameNode daemon and DataNode daemon:

```
~/hadoop/sbin/start-dfs.sh
```

Make the HDFS directories required to execute MapReduce jobs

```
~/hadoop/bin/hdfs dfs -mkdir /user
```

```
~/hadoop/bin/hdfs dfs -mkdir /user/your_user_name
```

Put sample.txt file in hdfs

```
~/hadoop/bin/hdfs dfs -put ~/sample.txt input_data
```

Perform MapReduce job

```
~/hadoop/bin/hadoop jar ~/exp5a.jar input_data output_data
```

Output

```
~/hadoop/bin/hdfs dfs -cat output_data/*
```

Our task is done, so delete the distributed files (input_data & output_data)

```
~/hadoop/bin/hdfs dfs -rm -r input_data output_data
```

Stop haddop

```
~/hadoop/sbin/stop-dfs.sh
```

```
jps
```

Reference : Hadoop the definitive guide, O'Reilly Publications, by Tom White

---

Prof. S. T. Kolhe
(Department of I.T – S.R.E.S C.O.E Kopargaon)