



Formação Engenheiro de Dados

Aplicação Prática

Fluxo



**Na maquina virtual, baixar do ambiente do
curso arquivo hadoop.zip**

Este arquivo contém

- WordCount.java
- pesquisa.txt



**/home/Cloudera/Downloads
O arquivo deve ser descompactado**

pesquisa.txt

1222 linhas

5	BigWhell
6	Elite
7	Auge555
8	Auge555
9	Auge555
10	SL429F
11	Auge555
12	Auge555
13	Auge555
14	Elite
15	Elite
16	Elite
17	BigWhell
18	BigWhell
19	Riff50SR
20	Riff50SR
21	Elite
22	Elite
23	Riff50SR

```

import java.io.IOException;
import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.Mapper;
import org.apache.hadoop.mapreduce.Reducer;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
import org.apache.hadoop.util.GenericOptionsParser;

public class WordCount {

```

```

    public static class MapForWordCount extends Mapper<LongWritable, Text, Text, IntWritable>{
        public void map(LongWritable key, Text value, Context con) throws IOException, InterruptedException {
            String line = value.toString();
            String[] words=line.split(" ");
            for(String word: words) {
                Text outputKey = new Text(word.toUpperCase().trim());
                IntWritable outputValue = new IntWritable(1);
                con.write(outputKey, outputValue);
            }
        }
        public static class ReduceForWordCount extends Reducer<Text, IntWritable, Text, IntWritable>
        {
            public void reduce(Text word, Iterable<IntWritable> values, Context con) throws IOException, InterruptedException {
                int sum = 0;
                for(IntWritable value : values)
                {
                    sum += value.get();
                }
                con.write(word, new IntWritable(sum));
            }
        }
    }

```

WordCount.java

Seguindo o Fluxo

- ◆ Vamos criar uma pasta no hdfs

```
hdfs dfs -mkdir /contar/
```

- ◆ Vamos copiar o arquivo

```
hdfs dfs -put  
/home/cloudera/Downloads/pesquisa.txt  
/contar/pesquisa.txt
```

- ◆ Vamos criar um jar usando o Eclipse

- ◆ IDE Java já instalada na VM

- ◆ Vamos salvar o jar criado em

```
/home/cloudera
```

Contagem de Palavras

- ◈ Vamos rodar uma aplicação Map Reduce no Hadoop
- ◈ Local do jar /Package /Arquivo /Arquivo /Diretório saída

```
hadoop jar  
/home/cloudera/MRProgramDemo.jar  
PackageDemo.WordCount  
/contar/pesquisa.txt /contar2
```

```
hdfs dfs -ls /contar2
```

- ◈ Verificamos o diretório criado

```
hdfs dfs -ls /contar2
```

- ◈ Abrimos o arquivo de resultado

```
hdfs dfs -cat /contar2/part-r-00000
```