**LAB-8**

**VIRTUAL LAB**

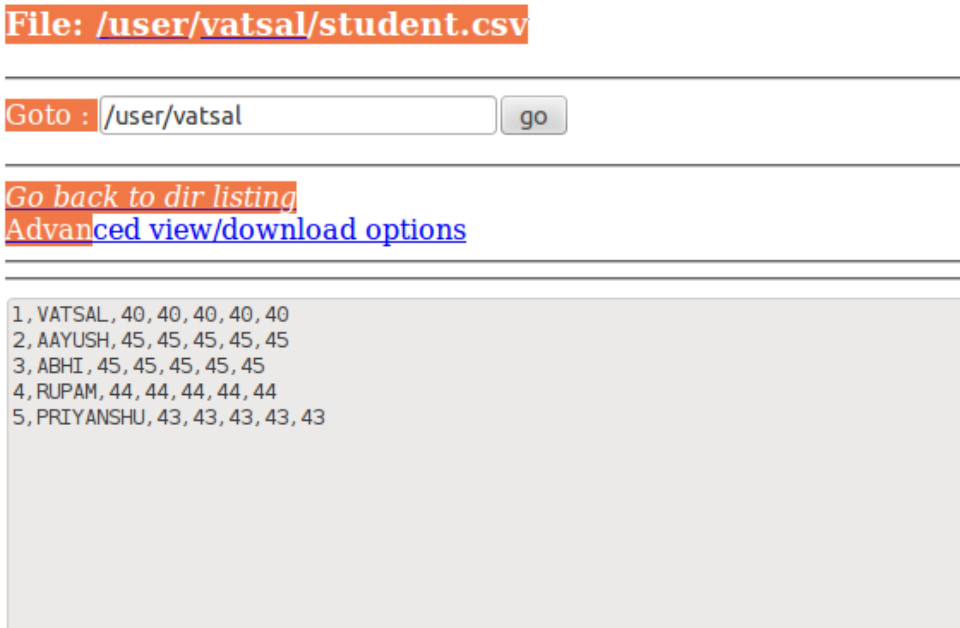
**PRIYANSHU SHARMA**

**15BCE1282**

**CODE**

1. SIDE DATA CONFIG

INPUT:



**CODE:**

**import** java.io.\*;

**import** java.util.\*;

**import** org.apache.hadoop.fs.Path;

**import** org.apache.hadoop.io.\*;

**import** org.apache.hadoop.conf.\*;

**import** org.apache.hadoop.mapreduce.\*;

**import** org.apache.hadoop.mapreduce.Mapper.Context;

**import** org.apache.hadoop.mapreduce.lib.input.\*;

**import** org.apache.hadoop.mapreduce.lib.output.\*;

**public** **class** virtualOne {

**public** **static** **class** Map **extends** Mapper<LongWritable, Text, Text, Text>

{

String name1;

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

name1 = context.getConfiguration().get("name");

String[] line = value.toString().split(",");

String name = line[1];

**if**(name.equalsIgnoreCase(name1)){

context.write(**new** Text(name), **new** Text (line[0] + "," + line[2] + "," + line[3] + "," + line[4] + "," + line[5] + "," + line[6]));

}

}

}

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

{

Configuration conf = **new** Configuration();

conf.set("name", args[2]);

Job job = **new** Job(conf, "virtualOne");

//job.setPartitionerClass(dpart.class);

//job.setNumReduceTasks(3);

job.setJarByClass(virtualOne.**class**);

job.setOutputKeyClass(Text.**class**);

job.setOutputValueClass(Text.**class**);

job.setMapperClass(Map.**class**);

//job.setReducerClass(Reduce.class);

job.setInputFormatClass(TextInputFormat.**class**);

job.setOutputFormatClass(TextOutputFormat.**class**);

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

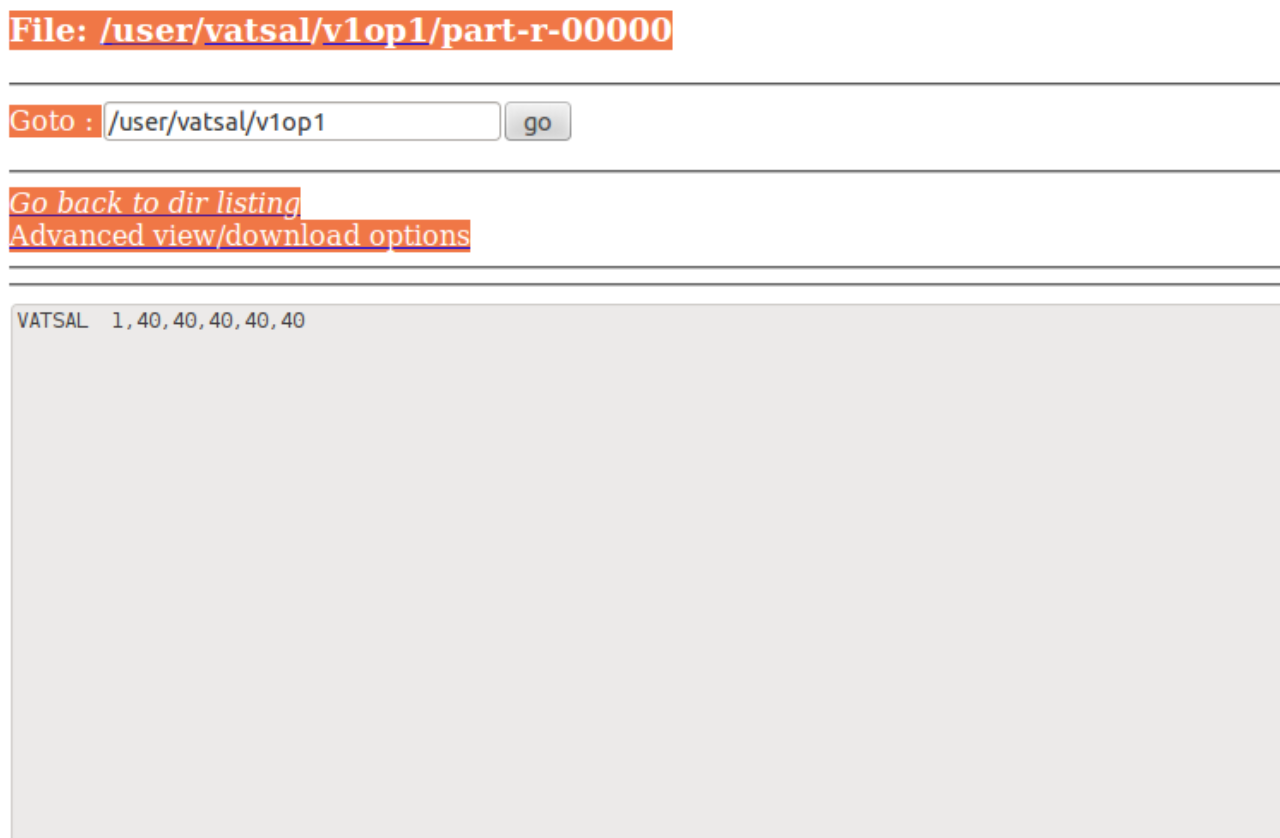
FileOutputFormat.*setOutputPath*(job,**new** Path(args[1]));

job.waitForCompletion(**true**);

}

}

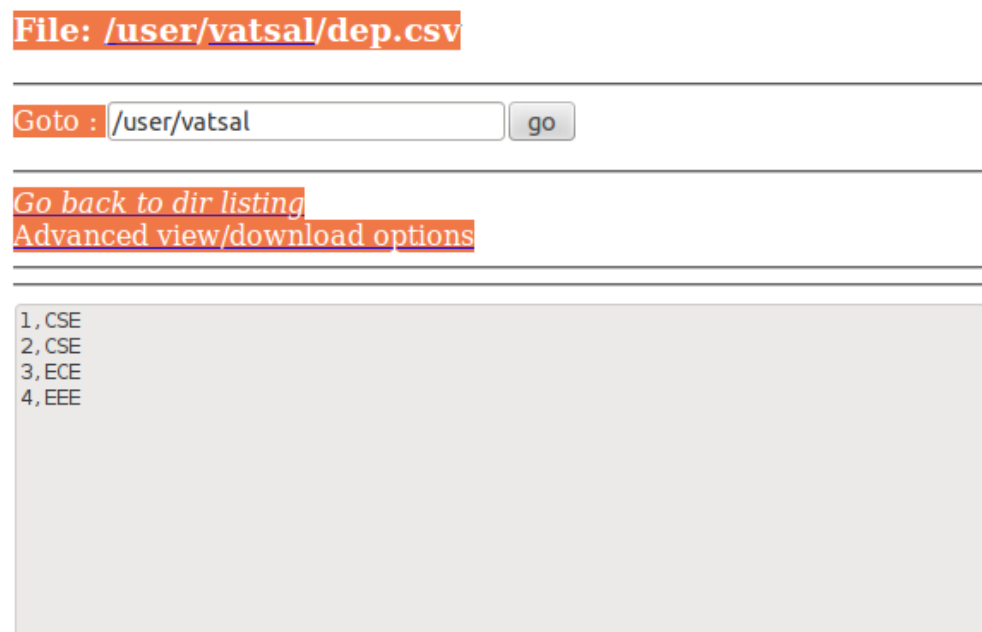
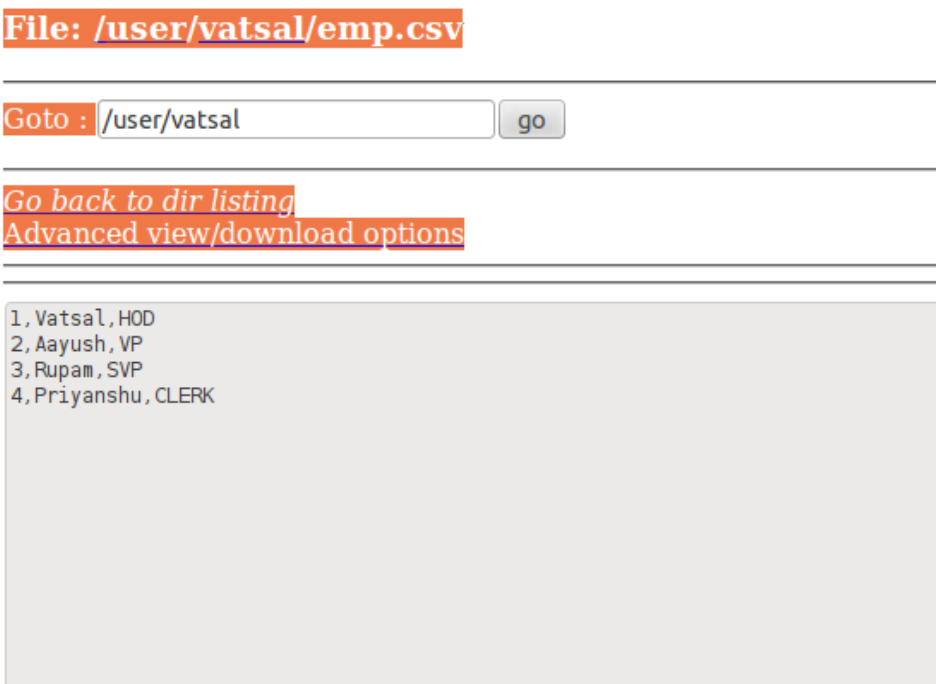
**OUTPUT:**



NAME = VATSAL

1. Joining two files.

**INPUT:**



**CODE:**

**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.Mapper.Context;

**import** org.apache.hadoop.mapreduce.lib.input.MultipleInputs;

**import** org.apache.hadoop.mapreduce.lib.input.TextInputFormat;

**import** org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;

**public** **class** virtualTwo {

**public** **static** **class** Map **extends** Mapper<LongWritable, Text, IntWritable, Text>

{

**public** **void** map(IntWritable key, Text value, Context context) **throws** IOException, InterruptedException

{

String[] line = value.toString().split(",");

**int** i = Integer.*parseInt*(line[0]);

context.write(**new** IntWritable(i), value);

}

}

**public** **static** **class** Map2 **extends** Mapper<LongWritable, Text, IntWritable, Text>

{

**public** **void** map(IntWritable key, Text value, Context context) **throws** IOException, InterruptedException

{

String[] line = value.toString().split(",");

**int** i = Integer.*parseInt*(line[0]);

context.write(**new** IntWritable(i), value);

}

}

**public** **static** **class** Reduce **extends** Reducer<LongWritable, Text, IntWritable, Text>

{

**public** **void** reduce(IntWritable key, Iterable<Text> values, Context context) **throws** IOException, InterruptedException

{

//String element = values.toString();

//String[] line = element.split(",");

String line1 = **null**;

**for**(Text val:values)

{

line1 = val.toString();

}

context.write(key, **new** Text(line1));

}

}

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

Configuration conf = **new** Configuration();

// conf.set("fs.default.name", "hdfs://localhost:8020");

Job job = **new** Job(conf, "virtualTwo");

job.setJarByClass(virtualTwo.**class**);

job.setJobName("Compare Two Files and Identify the Difference");

FileOutputFormat.*setOutputPath*(job, **new** Path(args[2]));

job.setReducerClass(Reduce.**class**);

job.setOutputKeyClass(LongWritable.**class**);

job.setOutputValueClass(Text.**class**);

MultipleInputs.*addInputPath*(job, **new** Path(args[0]),

TextInputFormat.**class**, Map.**class**);

MultipleInputs.*addInputPath*(job, **new** Path(args[1]),

TextInputFormat.**class**, Map2.**class**);

//FileOutputFormat.setOutputPath(job,new Path(args[3]));

job.waitForCompletion(**true**);

}

}

**OUTPUT:**

