
Big Data Hadoop and Spark Developer

Lesson-End Project Solution



Get Certified. Get Ahead.

Count the Number of Words Using MapReduce

Steps to Perform:

Step 1: Create a sample file with the name mapreducedemo.java

Step 2: Create two functions Mapper and Reduce with the main method

Mapper implementation:

```
public void map(Object key, Text value, Context context) throws IOException,
InterruptedException {
```

```
    StringTokenizer itr = new StringTokenizer(value.toString());
    while (itr.hasMoreTokens()) {
        word.set(itr.nextToken());
        context.write(word, one);
    }
}
```

Reduce implementation:

```
public void reduce(Text key, Iterable<IntWritable> values, Context
context)
```

```
    throws IOException, InterruptedException {
    int sum = 0;
    for (IntWritable val : values) {
        sum += val.get();
    }
    result.set(sum);
    context.write(key, result);
}
```

Main method:

```
package org.simplilearn.demo.mapreduce.wordcount;
import java.io.IOException;
import java.util.StringTokenizer;

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.Mapper;
import org.apache.hadoop.mapreduce.Reducer;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;

public class WordCount {

    public static class TokenizerMapper extends Mapper<Object, Text, Text,
IntWritable>{

        private final static IntWritable one = new IntWritable(1);
        private Text word = new Text();

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

            StringTokenizer itr = new StringTokenizer(value.toString());
            while (itr.hasMoreTokens()) {
                word.set(itr.nextToken());
                context.write(word, one);
            }
        }
    }

    public static class IntSumReducer extends Reducer<Text, IntWritable, Text,
IntWritable> {
        private IntWritable result = new IntWritable();

        public void reduce(Text key, Iterable<IntWritable> values, Context context)
            throws IOException, InterruptedException {
            int sum = 0;
            for (IntWritable val : values) {
                sum += val.get();
            }
        }
    }
}

```

```

        result.set(sum);
        context.write(key, result);
    }
}

public static void main(String[] args) throws Exception {
    Configuration conf = new Configuration();
    Job job = Job.getInstance(conf, "wordcount");
    job.setJarByClass(WordCount.class);
    job.setMapperClass(TokenizerMapper.class);
    //job.setCombinerClass(IntSumReducer.class);
    job.setReducerClass(IntSumReducer.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);
}
}

```

Step 3: Convert the JAVA files into JAR files and give it the name **“hadoop-mapreduce-example.jar”**

Note: “hadoop-mapreduce-example.jar” and “wordcount.txt” files are available in the Course Resources Section

Step 4: Log in to your LMS account

Step 5: Open the course **“Big Data Hadoop and Spark Developer”**

Step 6: Click on the **“PRACTICE LABS”** tab on the left side and select **“LAUNCH LAB”**

Big Data Hadoop and Spark Developer

52% Self-Learning Videos Watched | 0/3 Projects Done

COMMUNITY HELP NOTES

BDH

IMP: Dear learner,
Please note: This lab is configured based on the curriculum covered during the live virtual classes.
All details pertaining to the exercises in this lab are provided in the e-books available in your LMS account.

Instructions:

- When you go to the practice Lab page click on the LAUNCH LAB button and it would redirect you to the login credential page.
- To start a Cloudera manager, click on the Auth URL and enter the username and password as given.
- Similarly, for the web console, navigate to the Auth URL, and enter the credentials.
- For FTP, click on the URL and enter the credentials.

Tools:

Cluster Setup

FTP, Cloudera Infrastructure, Hadoop 3.x, HDFS, YARN, Kafka, Flume, Sqoop, Pig, Hive, Hbase, Scala, Spark, Spark - Core, Spark SQL, Spark GraphX, Spark .ML, Cloudera Manager

You can download the Lab Guides from [HERE](#)

Your Labs are ready. **LAUNCH LAB**

Step 7: Click on the “LAUNCH LAB” button

Big Data Hadoop and Spark Developer

52% Self-Learning Videos Watched | 0/3 Projects Done

COMMUNITY HELP NOTES

BDH

IMP: Dear learner,
Please note: This lab is configured based on the curriculum covered during the live virtual classes.
All details pertaining to the exercises in this lab are provided in the e-books available in your LMS account.

Instructions:

- When you go to the practice Lab page click on the LAUNCH LAB button and it would redirect you to the login credential page.
- To start a Cloudera manager, click on the Auth URL and enter the username and password as given.
- Similarly, for the web console, navigate to the Auth URL, and enter the credentials.
- For FTP, click on the URL and enter the credentials.

Tools:

Cluster Setup

FTP, Cloudera Infrastructure, Hadoop 3.x, HDFS, YARN, Kafka, Flume, Sqoop, Pig, Hive, Hbase, Scala, Spark, Spark - Core, Spark SQL, Spark GraphX, Spark .ML, Cloudera Manager

You can download the Lab Guides from [HERE](#)

Your Labs are ready. **LAUNCH LAB**

Your Lab Setup is Ready!

This Lab will get reset on
30th September 2022, 2:33 PM

To ensure continued practice, the reset date is automatically set to 120 days from the last lab start date. You'll lose all the lab progress on reset activity. Take a backup of your code if you are going to be away for more than 120 days.

LAUNCH LAB

Note: Practice labs are disabled on course expiry date by default.

Step 8: Click on “HUE” to upload the datasets

Big Data Hadoop and Spark Developer

52% Self-Learning Videos Watched | 0/3 Projects Done

COMMUNITY HELP NOTE

BDH This Lab will get reset on 07th October 2022, 11:57 AM

Current Lab : Big Data Lab

Access Information Lab Details Components Log Details

Applications

cloudera Webconsole **Hue** FTP Sqoop

Cloudera Manager Credentials

Username Password Auth Url

alpikaguptasimplilearn http://sibdh.corestack.io

Big Data Lab

Category: Big-Data and Hadoop
Start Date: 2021-10-26 14:05
End Date: 2022-10-07 12:00
Code: CDH

Big data is a term for data sets that are so large or complex that traditional data processing applications are inadequate. Challenges include analysis, capture, data curation, search, sharing, storage, transfer, visualization, querying, updating and information privacy.

Step 9: Log in to **"HUE"** and click on create a directory named **"mapreducedemo"** and upload the **"wordcount.txt"** file into it

Home /user/testdemomay1301mailinator/mapreducedemo

	Name	Size	User	Group
	↑		testdemomay1301maili...	hadoop
	.		testdemomay1301maili...	hadoop
	wordcount.txt	790 bytes	testdemomay1301maili...	hadoop

Show 45 of 1 items Page 1

Step 10: Click on **"FTP"** and click on the **"Auth Url"** to upload the JAR file and copy the **"Username"** and **"Password"** provided to log in to **"FTP"**

Big Data Hadoop and Spark Developer
52% Self-Learning Videos Watched | 0/3 Projects Done

BDH
This Lab will get reset on 07th October 2022, 11:57 AM

Current Lab : Big Data Lab

Access Information Lab Details Components Log Details

Applications

cloudera Webconsole Hue FTP Sqoop

Cloudera Manager Credentials

Username Password Auth Url

alpikaguptasimplilearn http://sibdh.corestack.io

Big Data Lab

Category: Big-Data and Hadoop
Start Date: 2021-10-26 14:05
End Date: 2022-10-07 12:00
Code: CDH

Big data is a term for data sets that are so large or complex that traditional data processing applications are inadequate. Challenges include analysis, capture, data curation, search, sharing, storage, transfer, visualization, querying, updating and information privacy.

Step 11: Paste the **“Username”** and **“Password”** on the login window and click on **“Login”**

Cloud Lab FTP Server

Username:
testdemomay1301mailinator

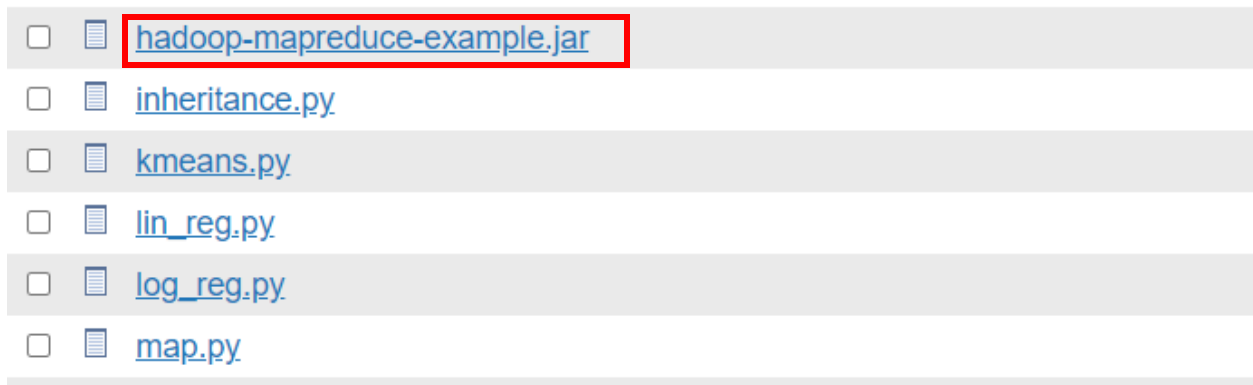
Password:
.....

Login

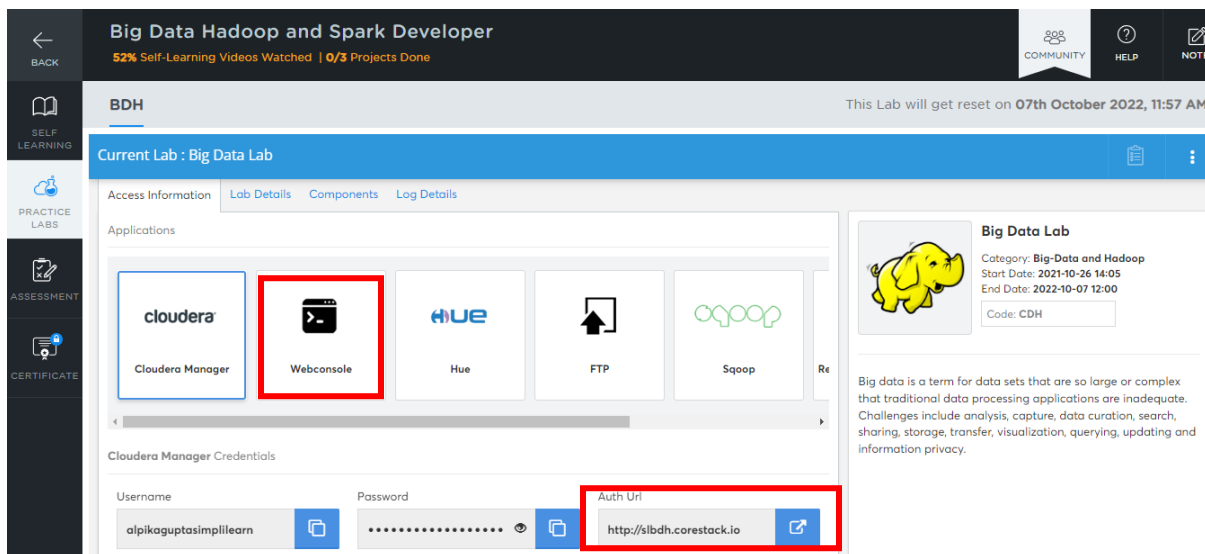
☐ Save login details

Language: English (US) ▼

Step 12: Click on the **“Upload Files”** icon and upload the **“hadoop-mapreduce-example.jar”** file into FTP



Step 13: Go back to the lab window and click on “Webconsole” and then on “Auth Url”



Step 14: Copy the “Username” and “Password” provided to log in to the “Webconsole”

Step 15: Paste the “Username” and “Password” on the console and click on enter

Note: The password will not be visible when pasted on the console.


```
bdh-cluster2-edgenode10 login: testdemomay1301mailinator
Password:
Last login: Tue Jun  7 10:44:11 on pts/0

do you want to continue?
=====
*                               :

Password for testdemomay1301mailinator@BDH-ENV.GNE4-RUTX.CLOUDERA.SITE:
[testdemomay1301mailinator@bdh-cluster2-edgenode10 ~]$
```

Step 16: Make sure you have the “**hadoop-mapreduce-example.jar**” file present in FTP using the ls command

```
[testdemomay1301mailinator@bdh-cluster2-edgenode10 ~]$ ls
13_4AP.py          convert.py          encapsulation.py
13_flights_graph_case_study  data.csv           example1.py
abc                data_files         example.py
abstractAP.py      data_files_CEP     flume.conf
abstract.py        demo11             hadoop-custom-demo.jar
Apache-log.log     derby.log          hadoop-mapreduce-example.jar
classDemo.py       dictionary.py      inheritance.py
[testdemomay1301mailinator@bdh-cluster2-edgenode10 ~]$
```

Step 17: Execute the below command and see if your job gets executed successfully

Note: Change the username of the Hadoop directory to “testdemomay1301mailinator” as assigned in your lab.





Command:

```
yarn jar hadoop-mapreduce-example.jar
/user/testdemomay1301mailinator/mapreducedemo/wordcount.txt
/user/testdemomay1301mailinator/mapreducedemo/Output
```





```
[testdemomay1301mailinator@bdh-cluster2-edgenode10 ~]$ yarn jar hadoop-mapreduce-example.jar /user/testdemomay1301mailinator/mapreducedemo/wordcount.txt /use
r/testdemomay1301mailinator/mapreducedemo/Output
```


Step 18: Log in to **HUE** and open the “**mapreducedemo**” directory. You will see one more folder named “**Output**”

[Home](#) /user/testdemomay1301mailinator/mapreducedemo

<input type="checkbox"/>	Name	Size	User	Group	P
<input type="checkbox"/>	 ↑		testdemomay1301maili...	hadoop	d
<input type="checkbox"/>	 .		testdemomay1301maili...	hadoop	d
<input type="checkbox"/>	 Output		testdemomay1301maili...	hadoop	d
<input type="checkbox"/>	 wordcount.txt	790 bytes	testdemomay1301maili...	hadoop	-r

Show of 2 items Page

<input type="checkbox"/>	Name	Size	User	Group	Permissions	Date
<input type="checkbox"/>	 ↑		testdemomay1301maili...	hadoop	drwxr-xr-x	June 12, 2022 0
<input type="checkbox"/>	 .		testdemomay1301maili...	hadoop	drwxr-xr-x	June 12, 2022 0
<input type="checkbox"/>	 _SUCCESS	0 bytes	testdemomay1301maili...	hadoop	-rw-r--r--	June 12, 2022 0
<input type="checkbox"/>	 part-r-00000	138 bytes	testdemomay1301maili...	hadoop	-rw-r--r--	June 12, 2022 0

Show of 2 items Page of 1 

Note: You will be able to see the part files with the count of each word.

API	2
Dataset	2
Foundation,	1
Originally	1
RDD	3
architectural	1
as	2
codebase	1
data	2
dataset	1
deprecated.	1
developed	1
fault-tolerant	1
foundation	1
framework.	1
in	1
interface	2
it	1
later	1
parallelism	1
programming	2