Assisted Practice 18: Apache Spark Streaming

Problem Scenario: Create a real-time streaming application with the data provided to see the streaming output at different timestamps

Objective: In this demonstration, you will learn how to create a real-time Spark streaming application.

Tasks to create spark-streaming-example.py file:

1. The first step is to create a Streaming context
2. Then, create a text socket stream. The method socketTextStream returns a DStream object which represents a deserialized stream
3. Now, you will use the flatMap() method on DStream to break the line into words
4. Then, convert each word of **"spark-streaming-example.py"** into a tuple (containing word,1)
5. Finally, create a word count program using reduceByKey

Note: This file is already available in Course Resource Section.

Tasks to upload a Spark Streaming application step by step:

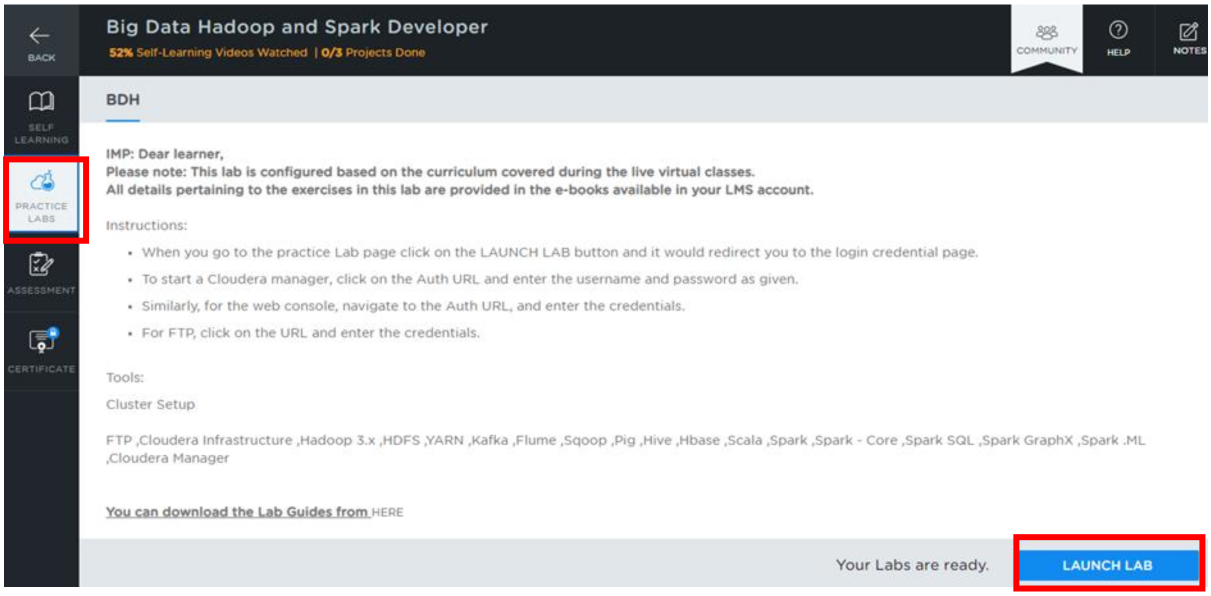
1. Upload the **"spark-streaming-example.py"** into the FTP
2. Open **"Webconsole"** and check the uploaded file
3. Run spark-submit command to execute the spark-streaming-example.py file

Steps to Perform:

Step 1: Log in to your LMS account

Step 2: Open the course **"Big Data Hadoop and Spark developer"**

Step 3: On the left side, click on the **"PRACTICE LABS"** tab and then click on the **"LAUNCH LAB"** button



Big Data Hadoop and Spark Developer

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

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:

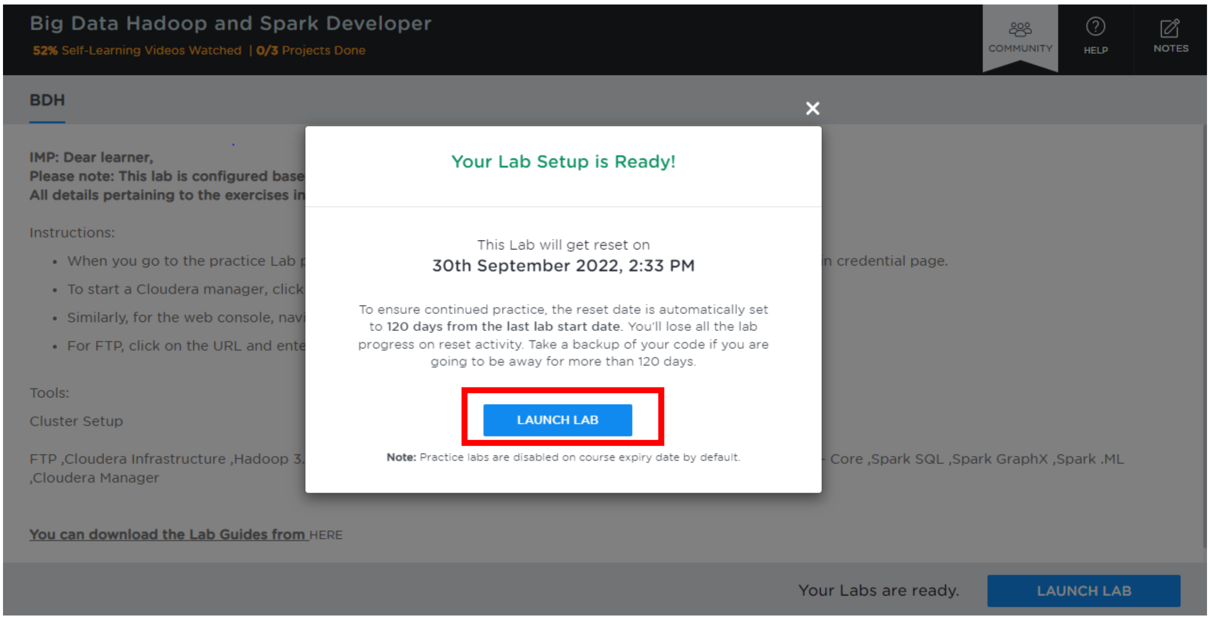
Cluster Setup

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.

Step 4: Again, click on the **"LAUNCH LAB"** button



Big Data Hadoop and Spark Developer

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

BDH

Your Lab Setup is Ready!

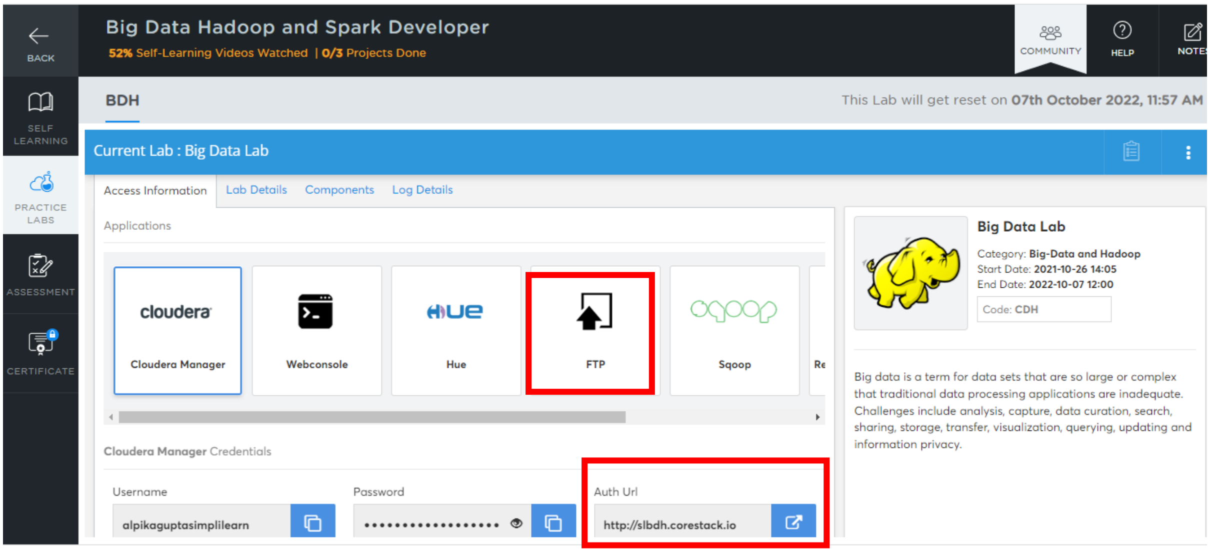
This Lab will get reset on
30th September 2022, 2:33 PM

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.

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

Your Labs are ready.

Step 5: Click on "**FTP**" and on the **"Auth Url"**



Big Data Hadoop and Spark Developer

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

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

Current Lab : Big Data Lab

Big Data Lab

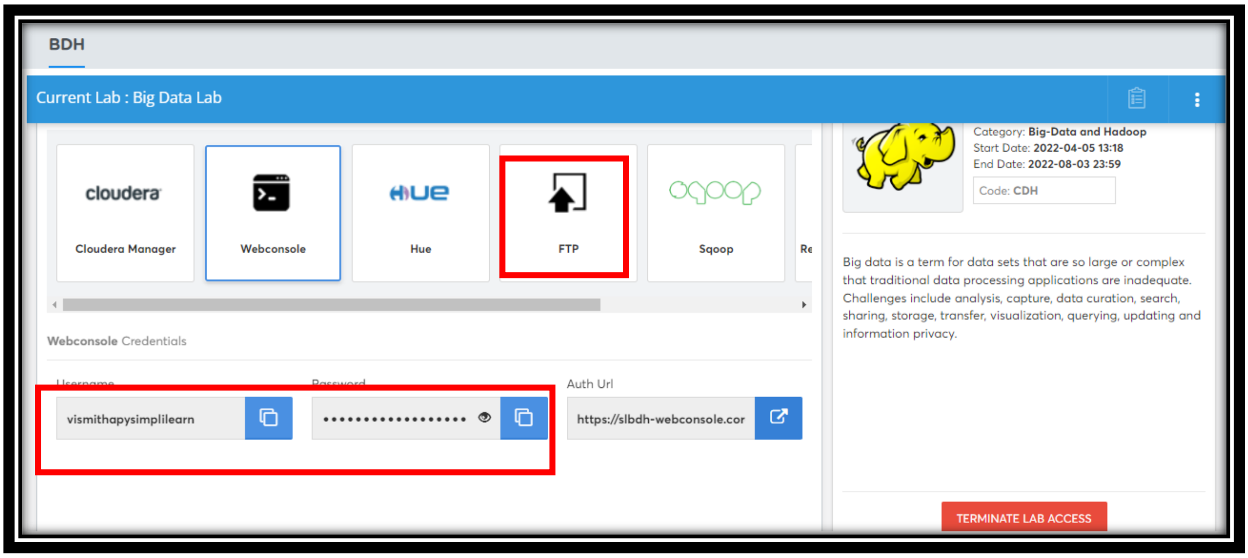
Category: Big-Data and Hadoop
Start Date: 2021-10-26 14:05
End Date: 2022-10-07 12:00
Code: CDH

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.

Cloudera Manager Credentials

Username: alpikaguptasimplilearn
Password: ••••••••••••••••••
Auth Url: http://slbdh.corestack.io

Step 6: Copy the **"Username"** and the **"Password"** provided to log in to the **"FTP"**



Current Lab : Big Data Lab

Category: Big-Data and Hadoop
Start Date: 2022-04-05 13:18
End Date: 2022-08-03 23:59
Code: CDH

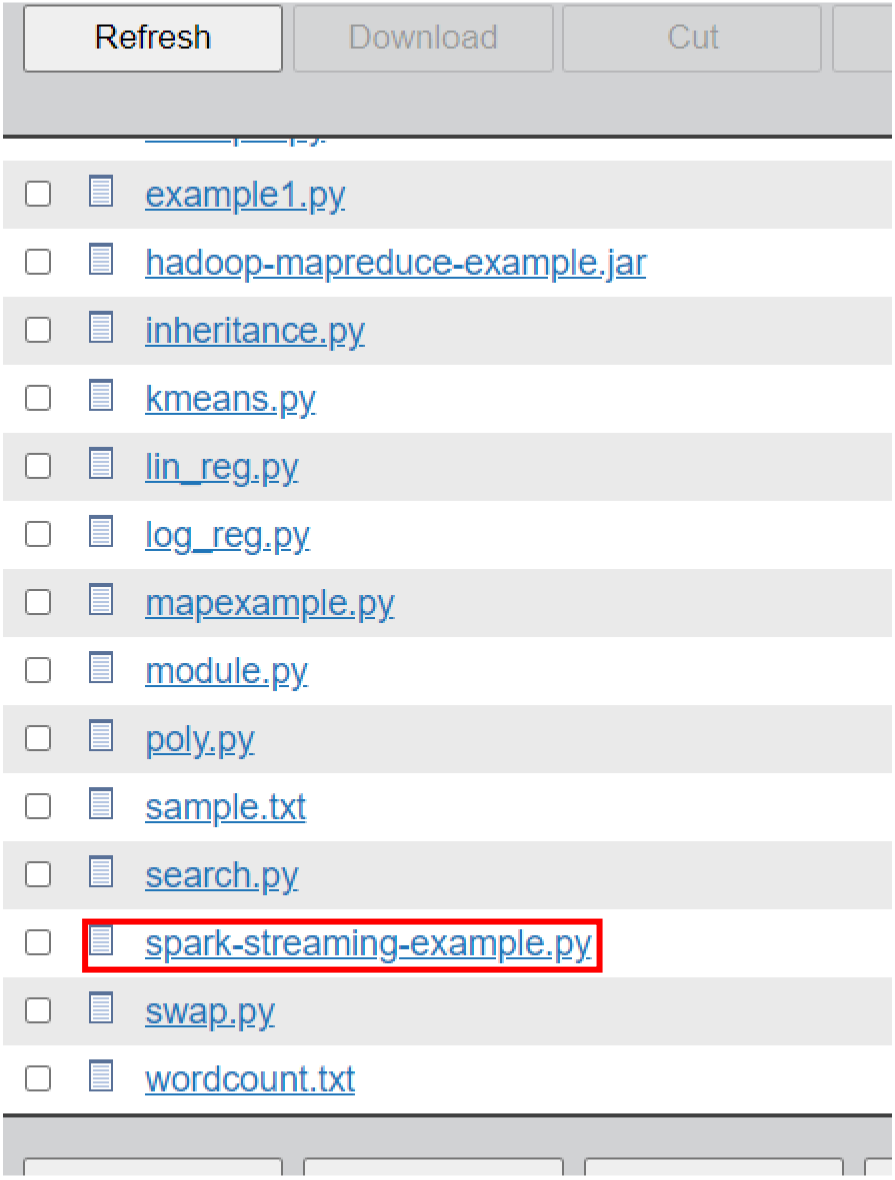
Big data is a term for data sets that are so large or complex that traditional data processing applications are inadequate. Challenges include analysis, capture, data curation, search, sharing, storage, transfer, visualization, querying, updating and information privacy.

Webconsole Credentials

Username: vismithapysimplilearn
Password: ••••••••••••••••••
Auth Url: https://slbdh-webconsole.cor

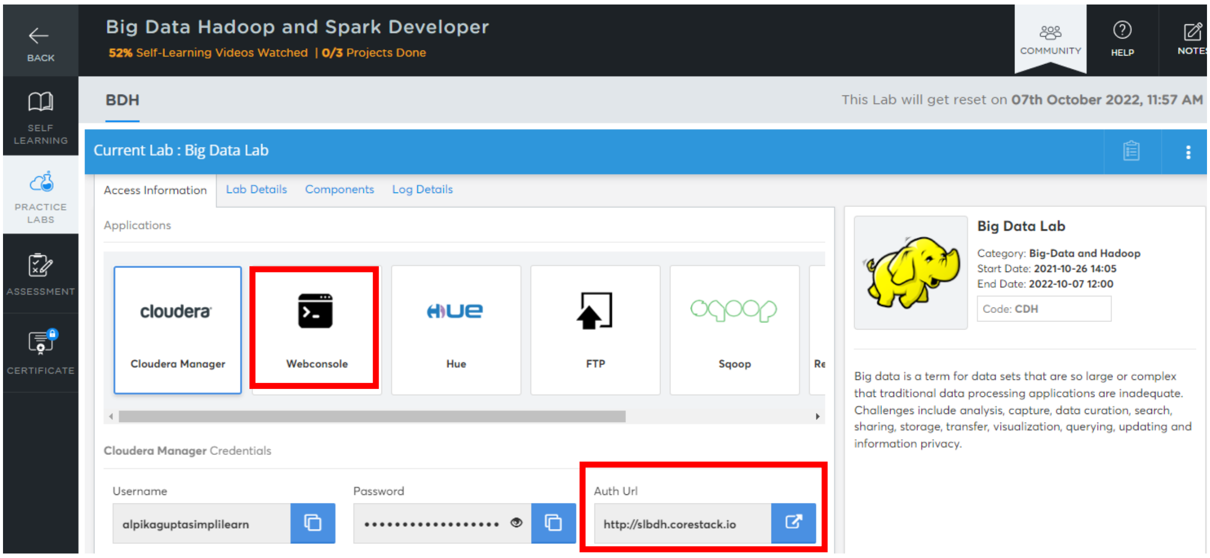
Step 7: Paste the "**Username**" and the "**Password**" and click log in

Step 8: Upload the **"spark-streaming-example.py"** file into the FTP

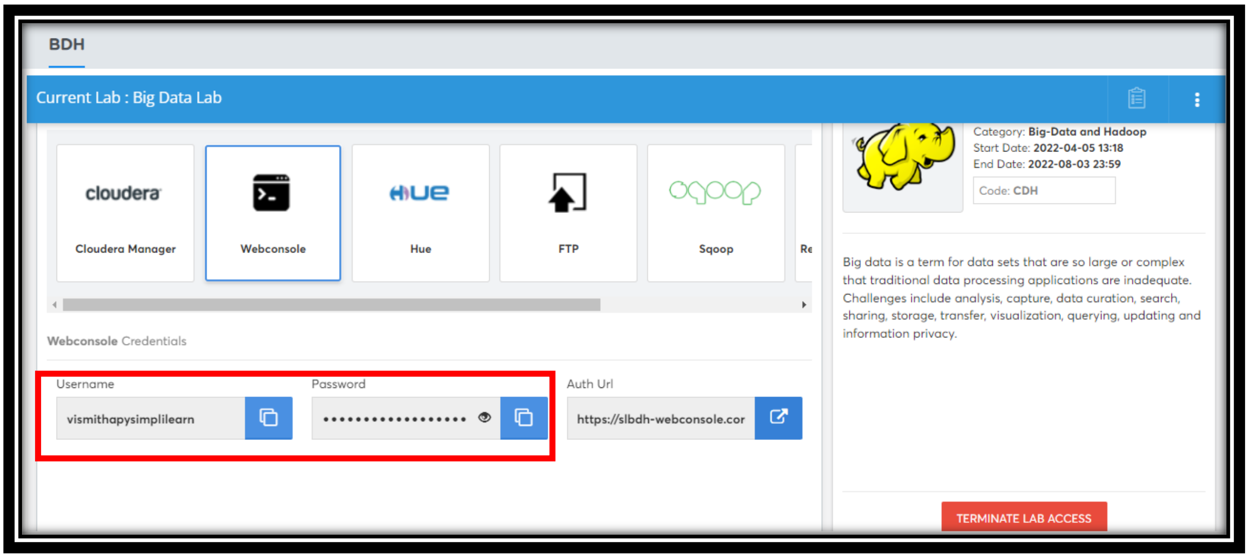


□ example1.py
□ hadoop-mapreduce-example.jar
□ inheritance.py
□ kmeans.py
□ lin_reg.py
□ log_reg.py
□ mapexample.py
□ module.py
□ poly.py
□ sample.txt
□ search.py
□ spark-streaming-example.py
□ swap.py
□ wordcount.txt

Step 9: Click on the "**Webconsole**" and on the **"Auth Url"**


BACK
SELF LEARNING
PRACTICE LABS
ASSESSMENT
CERTIFICATE
Big Data Hadoop and Spark Developer
52% Self-Learning Videos Watched | 0/3 Projects Done
COMMUNITY
HELP
NOTES
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 Manager
Webconsole
Hue
FTP
Sqoop
Cloudera Manager Credentials
Username
alpikaguptasimplilearn
Password
Auth Url
http://slbdh.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 10: Copy the **"Username"** and the **"Password"** provided to log in to the **"Webconsole"**


BDH
Current Lab : Big Data Lab
Cloudera Manager
Webconsole
Hue
FTP
Sqoop
Webconsole Credentials
Username
vismithapysimplilearn
Password
Auth Url
https://slbdh-webconsole.cor
Category: Big-Data and Hadoop
Start Date: 2022-04-05 13:18
End Date: 2022-08-03 23:59
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.
TERMINATE LAB ACCESS

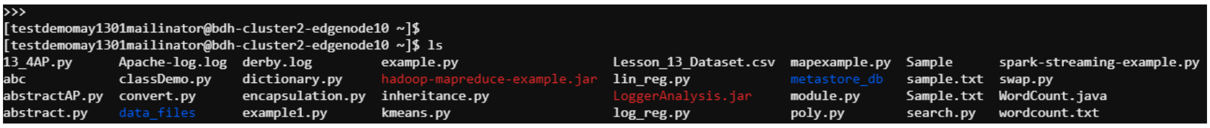
Step 11: Paste the "**Username**" and the "**Password**" on the console and click on Enter

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

Step 12: Now, check the uploaded file using the below command:

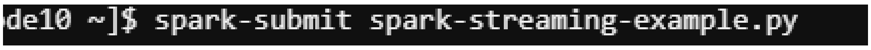
Command:

```
ls
```



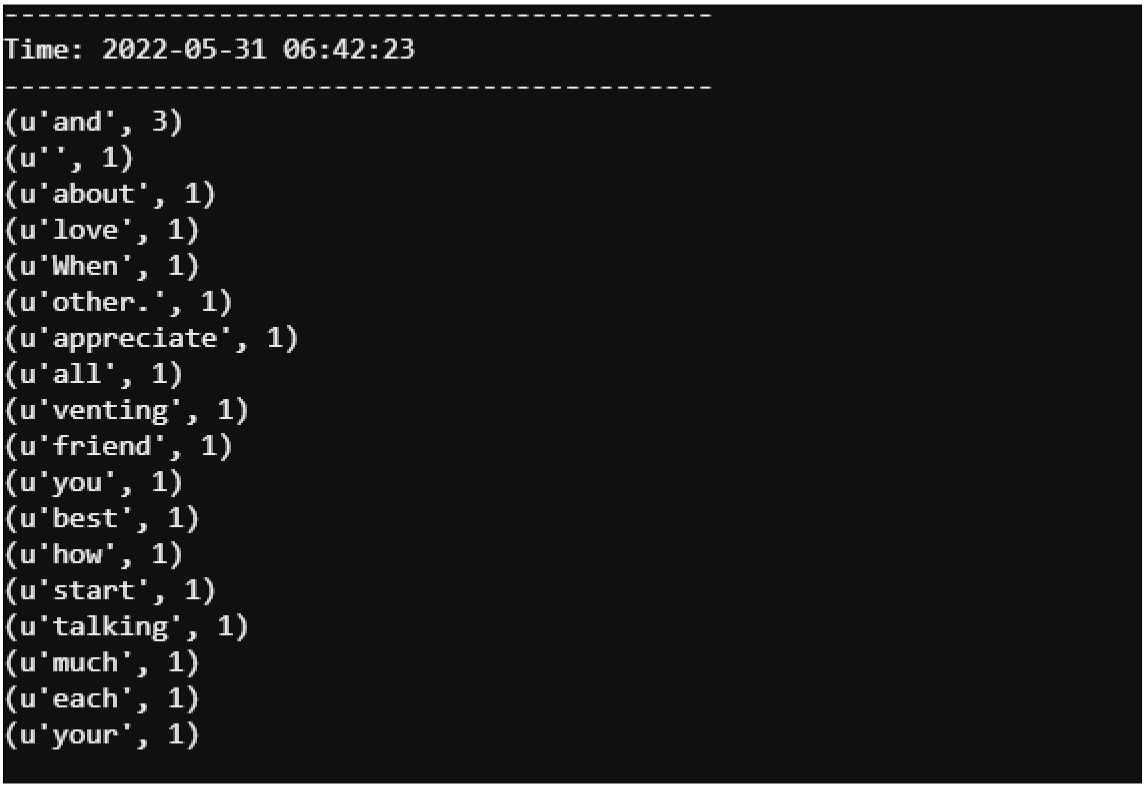
```
>>>
[testdemomay1301mailinator@bdh-cluster2-edgenode10 ~]$
[testdemomay1301mailinator@bdh-cluster2-edgenode10 ~]$ ls
13_4AP.py      Apache-log.log  derby.log          example.py                  Lesson_13_Dataset.csv  mapexample.py  Sample      spark-streaming-example.py
abc            classDemo.py    dictionary.py      hadoop-mapreduce-example.jar  lin_reg.py             metastore_db   sample.txt  swap.py
abstractAP.py  convert.py      encapsulation.py   inheritance.py              LoggerAnalysis.jar     module.py      Sample.txt  WordCount.java
abstract.py    data_files      example1.py        kmeans.py                   log_reg.py             poly.py        search.py   wordcount.txt
```

Step 13: Run the spark-submit command to execute the **“spark-streaming-example.py”** file

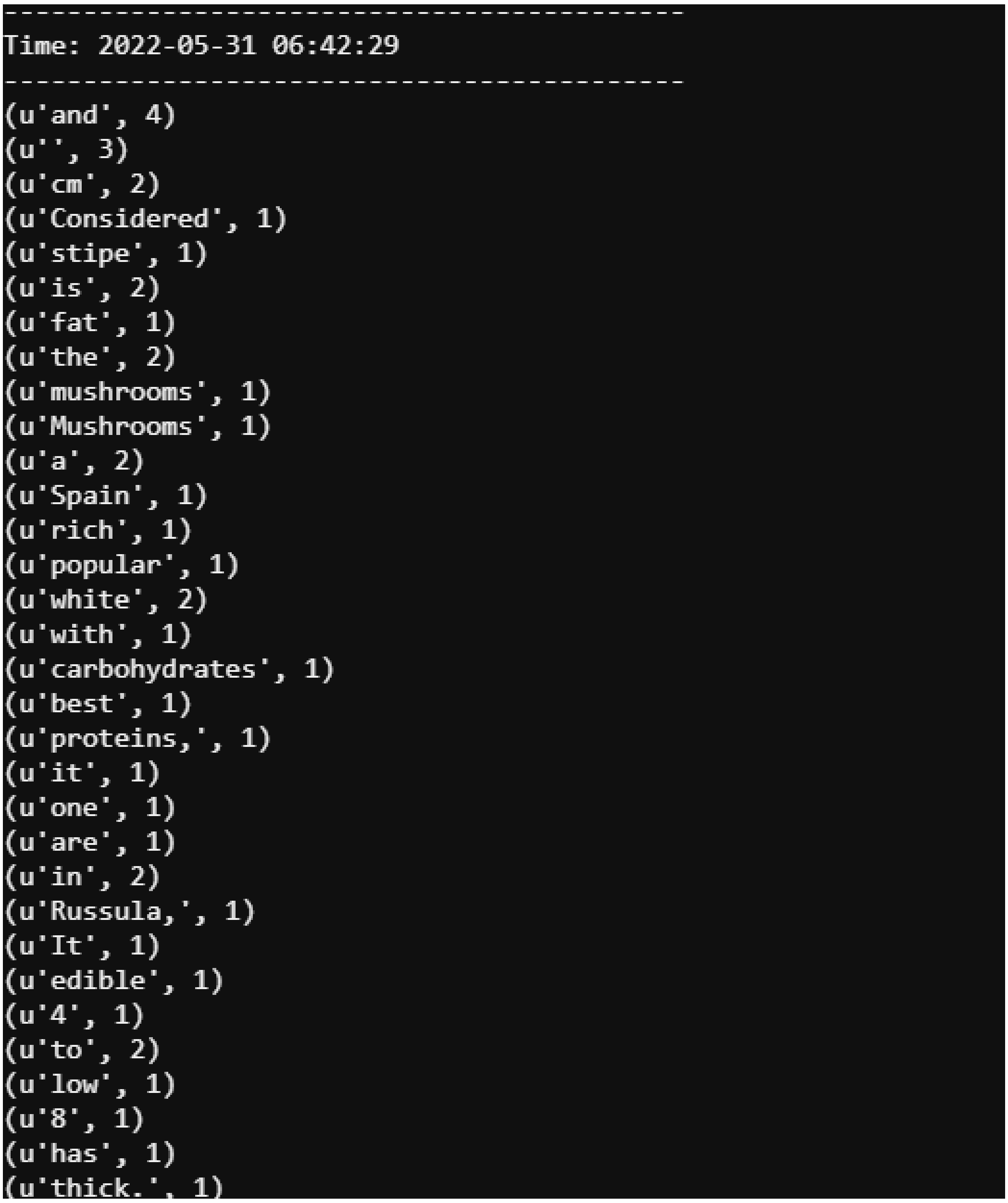


```
de10 ~]$ spark-submit spark-streaming-example.py
```

Step 14: You will be able to see the streaming output at different timestamps



```
-------------------------------------------
Time: 2022-05-31 06:42:23
-------------------------------------------
(u'and', 3)
(u'', 1)
(u'about', 1)
(u'love', 1)
(u'When', 1)
(u'other.', 1)
(u'appreciate', 1)
(u'all', 1)
(u'venting', 1)
(u'friend', 1)
(u'you', 1)
(u'best', 1)
(u'how', 1)
(u'start', 1)
(u'talking', 1)
(u'much', 1)
(u'each', 1)
(u'your', 1)
```



```
-------------------------------------------
Time: 2022-05-31 06:42:29
-------------------------------------------
(u'and', 4)
(u'', 3)
(u'cm', 2)
(u'Considered', 1)
(u'stipe', 1)
(u'is', 2)
(u'fat', 1)
(u'the', 2)
(u'mushrooms', 1)
(u'Mushrooms', 1)
(u'a', 2)
(u'Spain', 1)
(u'rich', 1)
(u'popular', 1)
(u'white', 2)
(u'with', 1)
(u'carbohydrates', 1)
(u'best', 1)
(u'proteins,', 1)
(u'it', 1)
(u'one', 1)
(u'are', 1)
(u'in', 2)
(u'Russula,', 1)
(u'It', 1)
(u'edible', 1)
(u'4', 1)
(u'to', 2)
(u'low', 1)
(u'8', 1)
(u'has', 1)
(u'thick.', 1)
```