

## Lab 9 Scala and pySpark

1. Write a Scala program to print numbers from 1 to 100 using for loop.

```
scala> for(i <- 1 to 100){  
  | println(i)}  
1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18
```

2. Using RDD and FlatMap count how many times each word appears in a file and write out a list of words whose count is strictly greater than 4 using Spark.

```
GNU nano 6.2 wordCount.py *  
from pyspark import SparkContext  
  
sc = SparkContext("local", "SimpleWordCount")  
  
rdd = sc.textFile("text1.txt")  
  
counts = (rdd.flatMap(lambda line: line.split())  
          .map(lambda word: (word.lower(), 1))  
          .reduceByKey(lambda a, b: a + b)  
          .filter(lambda x: x[1] > 4))  
  
for word, count in counts.collect():  
    print(word, count)  
  
sc.stop()
```

## Spark Shell Execution Screenshots

```
bmscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/pyspark-wordcount$ sudo apt update
Hit:2 http://in.archive.ubuntu.com/ubuntu jammy InRelease
Get:3 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Get:4 http://in.archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]
Hit:5 https://repo.mongodb.org/apt/ubuntu jammy/mongodb-org/6.0 InRelease
Ign:1 https://downloads.apache.org/cassandra/debian 40x InRelease
Err:6 https://downloads.apache.org/cassandra/debian 40x Release
  404 Not Found [IP: 88.99.208.237 443]
Hit:7 http://in.archive.ubuntu.com/ubuntu jammy-backports InRelease
Reading package lists... Done
W: https://repo.mongodb.org/apt/ubuntu/dists/jammy/mongodb-org/6.0/InRelease: Key is stored in legacy trusted
E: The repository 'http://www.apache.org/dist/cassandra/debian 40x Release' does not have a Release file.
N: Updating from such a repository can't be done securely, and is therefore disabled by default.
N: See apt-secure(8) manpage for repository creation and user configuration details.
bmscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/pyspark-wordcount$ sudo apt install python3-pip -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
```

```
bmscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/pyspark-wordcount$ pip3 install pyspark
Defaulting to user installation because normal site-packages is not writeable
Collecting pyspark
  Downloading pyspark-3.5.5.tar.gz (317.2 MB)
    317.2/317.2 MB, 1.8 MB/s, eta 0:00:00
```

```
bmscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ mkdir ~/pyspark-wordcount
bmscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ cd ~/pyspark-wordcount
bmscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/pyspark-wordcount$ nano.txt
nano.txt: command not found
bmscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/pyspark-wordcount$ nano file.txt
bmscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/pyspark-wordcount$ nano wordcount.py
bmscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/pyspark-wordcount$ python3 wordcount.py
```

```
bmscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~/pyspark-wordcount$ python3 wordcount.py
25/05/20 11:41:52 WARN Utils: Your hostname, bmscecse-HP-Elite-Tower-600-G9-Desktop-PC resolves to a loopback
25/05/20 11:41:52 WARN Utils: Set SPARK_LOCAL_IP if you need to bind to another address
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.spark.unsafe.Platform (file:/opt/spark/jars/spark-unsafe_2
WARNING: Please consider reporting this to the maintainers of org.apache.spark.unsafe.Platform
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
25/05/20 11:41:52 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using bo
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
scala 4
```

3. Write a simple streaming program in Spark to receive text data streams on a particular port, perform basic text cleaning (like white space removal, stop words removal, lemmatization, etc.), and print the cleaned text on the screen.

```
GNU nano 6.2          streaming_cleaner.py *
from pyspark import SparkContext
from pyspark.streaming import StreamingContext
from nltk.corpus import stopwords
from nltk.stem import WordNetLemmatizer
import re

# Set up Spark context and streaming context
sc = SparkContext("local[2]", "TextCleanerStreaming")
sc.setLogLevel("ERROR")
ssc = StreamingContext(sc, 5) # 5-second batch interval

# Set of stop words and lemmatizer
stop_words = set(stopwords.words("english"))
lemmatizer = WordNetLemmatizer()

# Connect to TCP socket on localhost:9999
lines = ssc.socketTextStream("localhost", 9999)

def clean_text(line):
    # Lowercase and remove punctuation
    line = re.sub(r"^[a-zA-Z\s]", "", line.lower())
    words = line.split()
    # Remove stopwords and lemmatize
    cleaned = [lemmatizer.lemmatize(word) for word in words if word not in stop_words]
    return " ".join(cleaned)

# Clean each line and print
lines.map(clean_text).pprint()

# Start streaming
ssc.start()
ssc.awaitTermination()
```

```
bmscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC: ~
bmscecse@bmsce... x bmscecse@bmsce... x bmscecse@bmsce... x bmscecse@bmsce... x
scecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ pip3 install nltk
faulting to user installation because normal site-packages is not writeable
llecting nltk
Downloading nltk-3.9.1-py3-none-any.whl (1.5 MB)
1.5/1.5 MB 7.6 MB/s eta 0:00:00
```

Installation of Natural Language Toolkit (nltk)

```
04.5
bmscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ python3
Python 3.10.12 (main, Jun 11 2023, 05:26:28) [GCC 11.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import nltk
>>> nltk.download('stopwords')
[nltk_data] Downloading package stopwords to
[nltk_data] /home/bmscecse/nltk_data...
[nltk_data] Unzipping corpora/stopwords.zip.
True
>>> nltk.download('wordnet')
[nltk_data] Downloading package wordnet to /home/bmscecse/nltk_data...
True
>>> exit()
```

```
bmscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ nano streaming_cleaner.py
bmscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ python3 streaming_cleaner.py
25/05/20 12:05:10 WARN Utils: Your hostname, bmscecse-HP-Elite-Tower-600-G9-Desktop-PC resolves to a loopback address: 127.0.1.1; using 10.124.3.71 instead (on interface eno1)
25/05/20 12:05:10 WARN Utils: Set SPARK_LOCAL_IP if you need to bind to another address
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.spark.unsafe.Platform (file:/opt/spark/jars/spark-unsafe_2.12-3.0.3.jar) to constructor java.nio.DirectByteBuffer(long,int)
WARNING: Please consider reporting this to the maintainers of org.apache.spark.unsafe.Platform
```

Executing the streaming\_cleaner.py

```
bmscecse@bmsce... x bmscecse@bmsce... x bmscecse@bmsce... x bmscecse@bmsce... x
bmscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ nc -lk 9999
Spark is very powerful and fast for big data processing.
```

Starting a TCP server that listens for incoming connections on port 9999

```
-----
Time: 2025-05-20 12:05:55
-----
spark powerful fast big data processing
-----
Time: 2025-05-20 12:06:00
```

Output- cleaned data