Exp 11 **IMPLEMENTATION OF APACHE SPARK**
30/09/24

Aim:

To install the spark in Linux and execute the wordcount program in the spark shell.

Procedure:

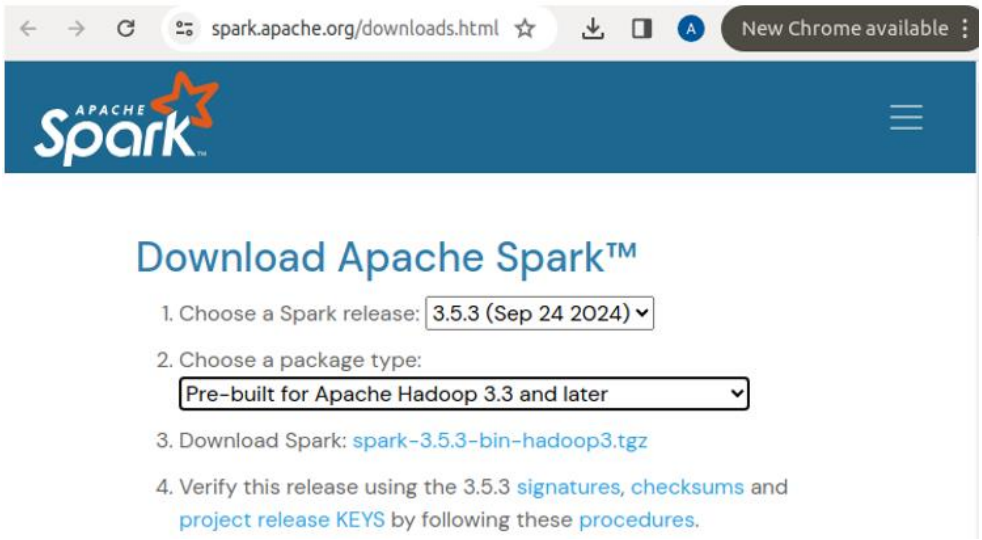
Step 1: Verify if Java is installed

```
tce@tce-VirtualBox:~$ java -version
openjdk version "11.0.19" 2023-04-18
OpenJDK Runtime Environment (build 11.0.19+7-post-Ubuntu-0ubuntu118.04.1)
OpenJDK 64-Bit Server VM (build 11.0.19+7-post-Ubuntu-0ubuntu118.04.1, mixed mo
de, sharing)
```

Step 2 : Verify if Spark is installed

```
tce@tce-VirtualBox:~$ sudo apt-get install scala
[sudo] password for tce:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  gir1.2-goa-1.0 gir1.2-snapd-1
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  libhawtjni-runtime-java libjansi-java libjansi-native-java libjline2-java
  scala-library scala-parser-combinators scala-xml
Suggested packages:
  scala-doc
The following NEW packages will be installed:
```

Step 3: Download and Install Apache Spark : https://spark.apache.org/downloads.html



Step 4: Check spark file using *ls* in Downloads and extract the spark file.

```
tce@tce-VirtualBox:~$ cd Downloads
tce@tce-VirtualBox:~/Downloads$ ls
'a (1).php'         'JPS Virus Maker1-20240405T180900Z-001.zip'
 a.php              'JPS Virus Maker.tar.gz'
 hadoop              spark-3.5.3-bin-hadoop3.tgz
 Hadoop.docx         Spark_Manual.docx
'JPS Virus Maker'
tce@tce-VirtualBox:~/Downloads$ tar xvf spark-3.5.3-bin-hadoop3.tgz
spark-3.5.3-bin-hadoop3/
spark-3.5.3-bin-hadoop3/data/
spark-3.5.3-bin-hadoop3/data/graphx/
spark-3.5.3-bin-hadoop3/data/graphx/users.txt
spark-3.5.3-bin-hadoop3/data/graphx/followers.txt
spark-3.5.3-bin-hadoop3/data/mllib/
spark-3.5.3-bin-hadoop3/data/mllib/sample_linear_regression_data.txt
spark-3.5.3-bin-hadoop3/data/mllib/sample_fpgrowth.txt
spark-3.5.3-bin-hadoop3/data/mllib/sample_libsvm_data.txt
```

```
tce@tce-VirtualBox:~/Downloads$ ls
'a (1).php'         'JPS Virus Maker1-20240405T180900Z-001.zip'
 a.php              'JPS Virus Maker.tar.gz'
 hadoop              spark-3.5.3-bin-hadoop3
 Hadoop.docx         spark-3.5.3-bin-hadoop3.tgz
'JPS Virus Maker'    Spark_Manual.docx
tce@tce-VirtualBox:~/Downloads$
```

Step 5: Move the spark-3.5.3-bin-hadoop3 to spark directory.

```
tce@tce-VirtualBox:~/Downloads$ cd
tce@tce-VirtualBox:~$ sudo su
[sudo] password for tce:
root@tce-VirtualBox:/home/tce# cd /home/tce/Downloads/
root@tce-VirtualBox:/home/tce/Downloads# mv spark-3.5.3-bin-hadoop3 /usr/local/
spark
root@tce-VirtualBox:/home/tce/Downloads# ls
'a (1).php'         'JPS Virus Maker1-20240405T180900Z-001.zip'
 a.php              'JPS Virus Maker.tar.gz'
 hadoop              spark-3.5.3-bin-hadoop3.tgz
 Hadoop.docx         Spark_Manual.docx
'JPS Virus Maker'
root@tce-VirtualBox:/home/tce/Downloads# cd /usr/local
root@tce-VirtualBox:/usr/local# ls
bin  etc  games  include  lib  man  sbin  share  spark  src
```

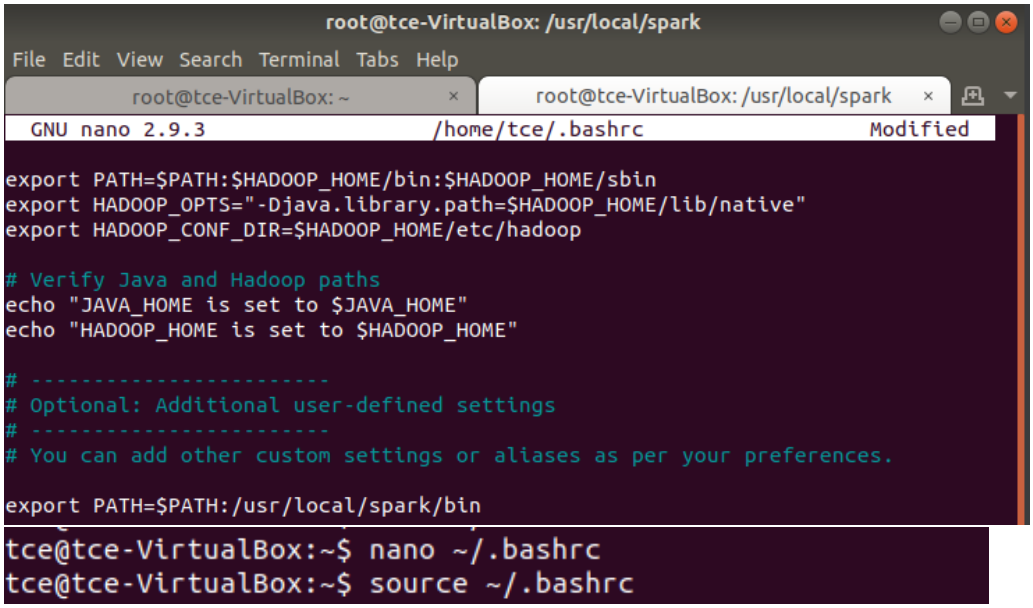
```
root@tce-VirtualBox:~# cd /usr/local/spark
root@tce-VirtualBox:/usr/local/spark# ls
bin    data      jars         LICENSE   NOTICE  R           RELEASE  yarn
conf   examples  kubernetes   licenses  python  README.md   sbin
root@tce-VirtualBox:/usr/local/spark#
```

STEP 6: Open the bashrc

```
tce@tce-VirtualBox:~$ nano ~/.bashrc
tce@tce-VirtualBox:~$
```

STEP 7: Edit bashrc

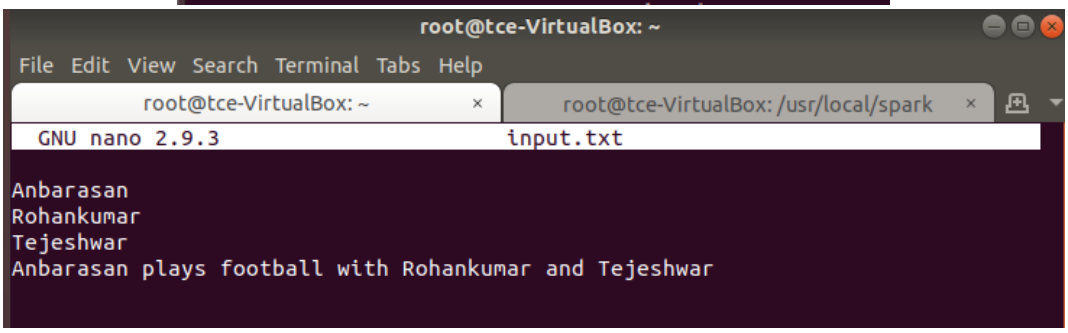
export PATH =$PATH /usr/local/spark/bin




```
tce@tce-VirtualBox:~$ nano ~/.bashrc
tce@tce-VirtualBox:~$ source ~/.bashrc
```

STEP 9: Save the input.txt file

```
tce@tce-VirtualBox:~$ nano input.txt
```



STEP 8: Open the spark-shell

```
tce@tce-VirtualBox:~$ spark-shell
24/10/05 15:59:10 WARN Utils: Your hostname, tce-VirtualBox resolves to a loopb
ack address: 127.0.1.1; using 10.0.2.15 instead (on interface enp0s3)
24/10/05 15:59:10 WARN Utils: Set SPARK_LOCAL_IP if you need to bind to another
 address
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLev
el(newLevel).
24/10/05 15:59:22 WARN NativeCodeLoader: Unable to load native-hadoop library f
or your platform... using builtin-java classes where applicable
Spark context Web UI available at http://10.0.2.15:4040
Spark context available as 'sc' (master = local[*], app id = local-172812416526
9).
Spark session available as 'spark'.
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /___/ .__/\_,_/_/ /_/\_\   version 3.5.3
      /_/

Using Scala version 2.12.18 (OpenJDK 64-Bit Server VM, Java 11.0.19)
Type in expressions to have them evaluated.
Type :help for more information.

scala>
```

STEP 9: Open the file map it and split and save to sparkoutput.txt

```
scala> val textFile = sc.textFile("file:///home/tce/input.txt")
textFile: org.apache.spark.rdd.RDD[String] = file:///home/tce/input.txt MapPart
itionsRDD[32] at textFile at <console>:23

scala> val words = textFile.flatMap(line => line.split(" "))
words: org.apache.spark.rdd.RDD[String] = MapPartitionsRDD[33] at flatMap at <c
onsole>:23

scala> val wordPairs = words.map(word => (word, 1))
wordPairs: org.apache.spark.rdd.RDD[(String, Int)] = MapPartitionsRDD[34] at ma
p at <console>:23

scala> val wordCounts = wordPairs.reduceByKey((a, b) => a + b)
wordCounts: org.apache.spark.rdd.RDD[(String, Int)] = ShuffledRDD[35] at reduce
ByKey at <console>:23

scala> wordCounts.saveAsTextFile("file:///home/tce/sparkoutput")

scala> val output = sc.textFile("file:///home/tce/sparkoutput")
output: org.apache.spark.rdd.RDD[String] = file:///home/tce/sparkoutput MapPart
itionsRDD[38] at textFile at <console>:23
```

STEP 10: Print the output in console by below command

```
scala> output.collect().foreach(println)
(football,1)
(plays,1)
(Rohankumar,2)
(with,1)
(Tejeshwar,2)
(Anbarasan,2)
(and,1)
```

Result:

Thus the installation of Spark in linux and execution of sample program has been executed successfully and output has been verified.