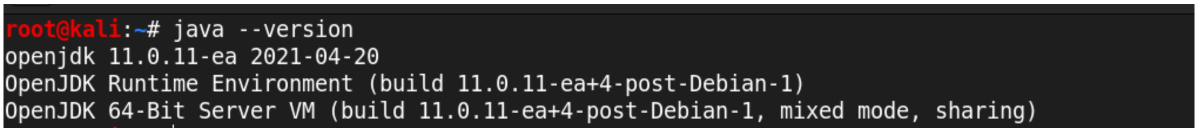
SPARK – INSTALLATION

The following steps show how to install Apache Spark.

Step 1: Verifying Java Installation

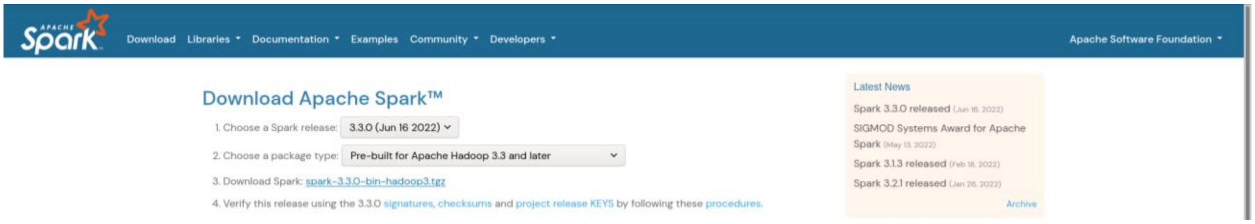
If Java is already installed on your system, you get to see the following response or some other versions.



```
root@kali:~# java --version
openjdk 11.0.11-ea 2021-04-20
OpenJDK Runtime Environment (build 11.0.11-ea+4-post-Debian-1)
OpenJDK 64-Bit Server VM (build 11.0.11-ea+4-post-Debian-1, mixed mode, sharing)
```

Step 2: Downloading Apache Spark

Download the latest version of Spark by visiting the following link Download Spark https://spark.apache.org/downloads.html . Select the latest version in Spark release and select pre-built for Apache Hadoop 3.3 and later . Click on the Download Spark link . It will navigate to one more page , and use HTTP to download the file . After downloading it, you will find the Spark tar file in the download folder.



Download Apache Spark™

1. Choose a Spark release: 3.3.0 (Jun 16 2022)
2. Choose a package type: Pre-built for Apache Hadoop 3.3 and later
3. Download Spark: spark-3.3.0-bin-hadoop3.tgz
4. Verify this release using the 3.3.0 signatures, checksums and project release KEYS by following these procedures.

Latest News

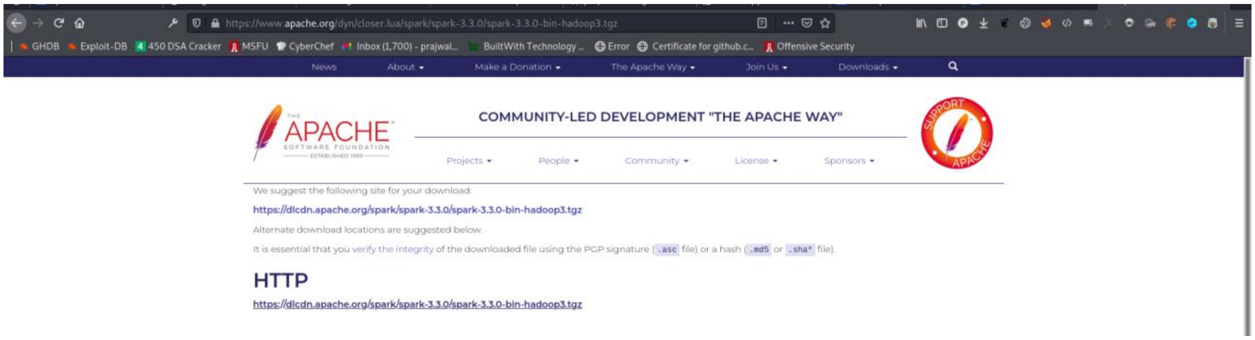
Spark 3.3.0 released (Jun 16, 2022)

SIGMOD Systems Award for Apache Spark (May 13, 2022)

Spark 3.1.3 released (Feb 18, 2022)

Spark 3.2.1 released (Jan 26, 2022)

Archive


https://www.apache.org/dyn/closer.lua/spark/spark-3.3.0/spark-3.3.0-bin-hadoop3.tgz
GHDB
Exploit-DB
450 DSA Cracker
MSFU
CyberChef
Inbox (1,700) - prajwal...
BuiltWith Technology ...
Error
Certificate for github.c...
Offensive Security
News
About
Make a Donation
The Apache Way
Join Us
Downloads
THE APACHE SOFTWARE FOUNDATION
COMMUNITY-LED DEVELOPMENT "THE APACHE WAY"
Projects
People
Community
License
Sponsors
SUPPORT APACHE

We suggest the following site for your download:

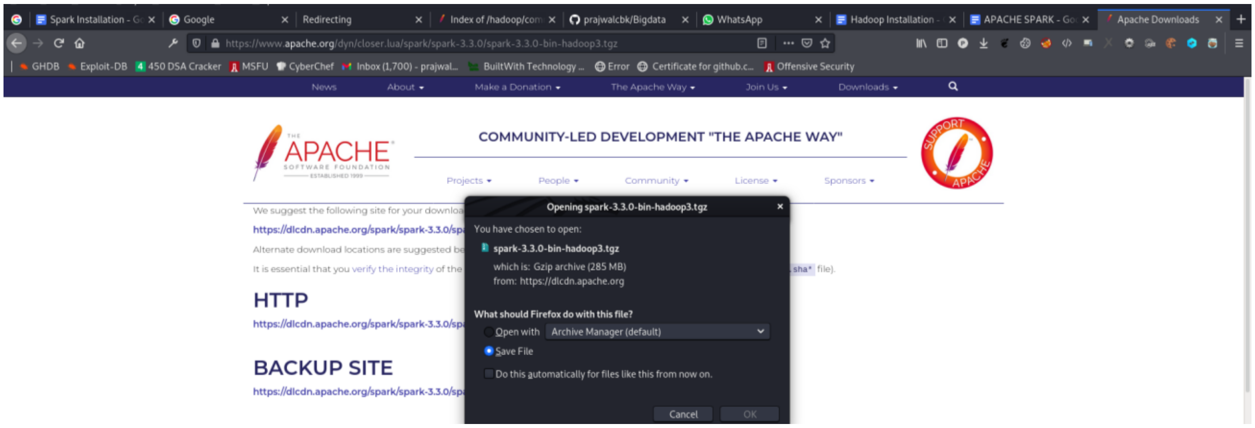
https://dlcdn.apache.org/spark/spark-3.3.0/spark-3.3.0-bin-hadoop3.tgz

Alternate download locations are suggested below.

It is essential that you verify the integrity of the downloaded file using the PGP signature (.asc file) or a hash (.md5 or .sha* file).

HTTP

https://dlcdn.apache.org/spark/spark-3.3.0/spark-3.3.0-bin-hadoop3.tgz


https://www.apache.org/dyn/closer.lua/spark/spark-3.3.0/spark-3.3.0-bin-hadoop3.tgz
COMMUNITY-LED DEVELOPMENT "THE APACHE WAY"
HTTP
BACKUP SITE
Cancel
OK

Opening spark-3.3.0-bin-hadoop3.tgz

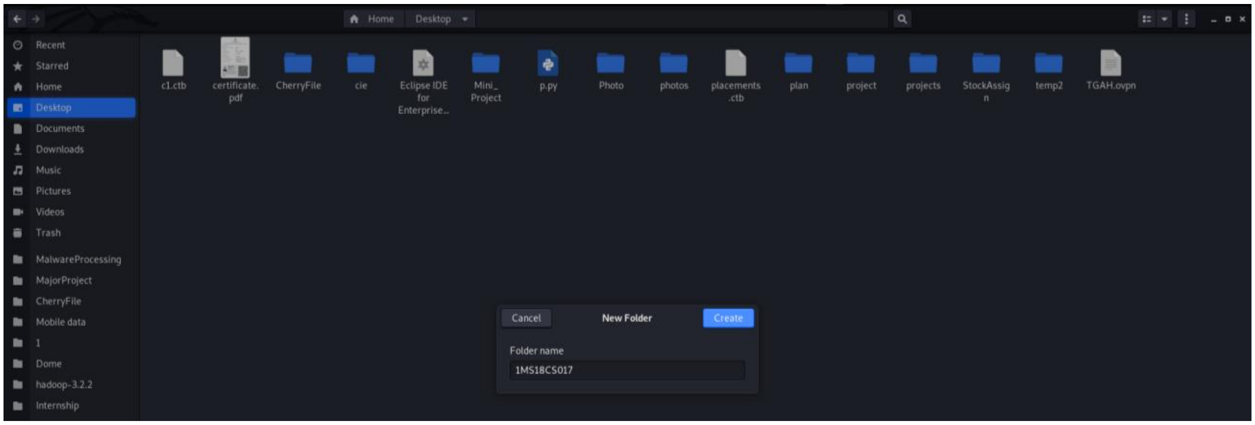
You have chosen to open:

spark-3.3.0-bin-hadoop3.tgz
which is: Gzip archive (285 MB)
from: https://dlcdn.apache.org

What should Firefox do with this file?

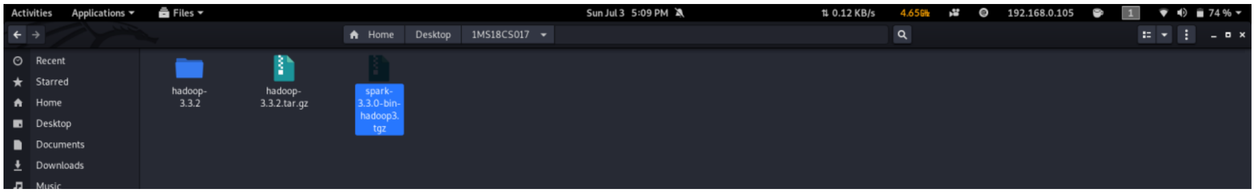
○ Open with Archive Manager (default)
● Save File
□ Do this automatically for files like this from now on.

Step 3: Create a new Folder inside Desktop , name the Folder as your USN <1ms18cs017>.

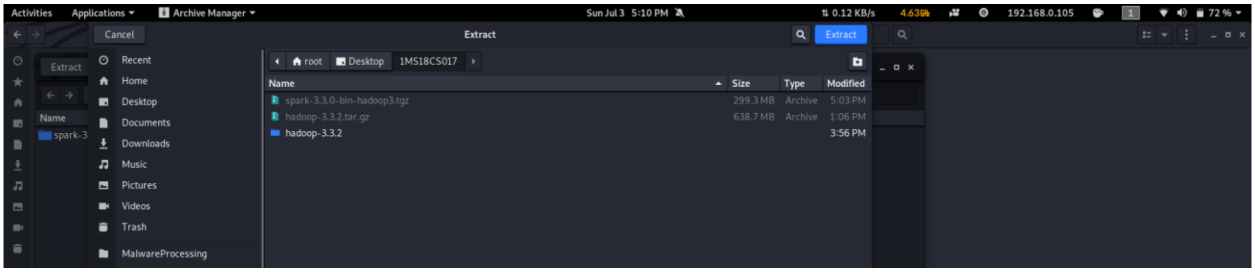


Folder name
1MS18CS017

Step 4 . Move the Downloaded Spark File to USN <1ms18cs017> Folder.



Step 5. Right Click on that File and Extract inside the USN <1ms18cs017> Folder.

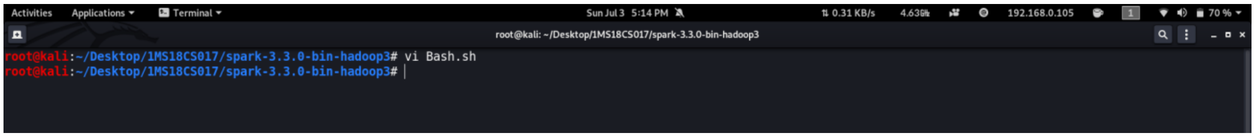


Name	Size	Type	Modified
spark-3.3.0-bin-hadoop3.tgz	299.3 MB	Archive	5:03 PM
hadoop-3.3.2.tar.gz	638.7 MB	Archive	1:06 PM
hadoop-3.3.2			3:56 PM

Step 6:. Open Terminal

Navigate to Extracted Hadoop Folder cd ~/Desktop/<1ms18cs017>/spark-3.3.0-bin-hadoop3

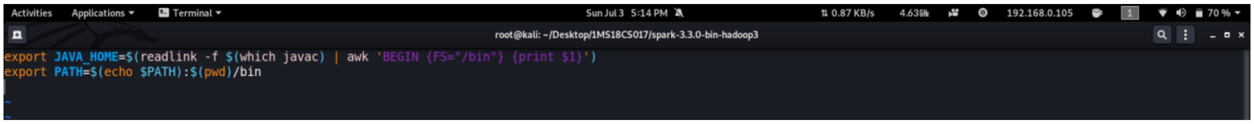
7. Create a New File named Bash.sh



```
root@kali:~/Desktop/1MS18CS017/spark-3.3.0-bin-hadoop3# vi Bash.sh
root@kali:~/Desktop/1MS18CS017/spark-3.3.0-bin-hadoop3# |
```

8. Copy the Below code and Paste inside Bash.sh and save that File.

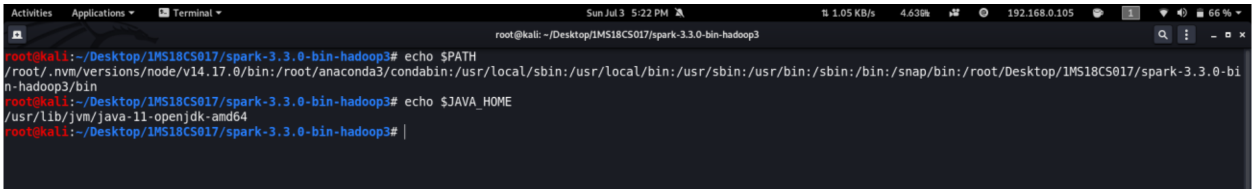
```
export JAVA_HOME=$(readlink -f $(which javac) | awk 'BEGIN {FS="/bin"} {print $1}')
if ! command -v spark-shell --version &> /dev/null
then
    export PATH=$(echo $PATH):$(pwd)/bin
fi
```



```
export JAVA_HOME=$(readlink -f $(which javac) | awk 'BEGIN {FS="/bin"} {print $1}')
export PATH=$(echo $PATH):$(pwd)/bin
```

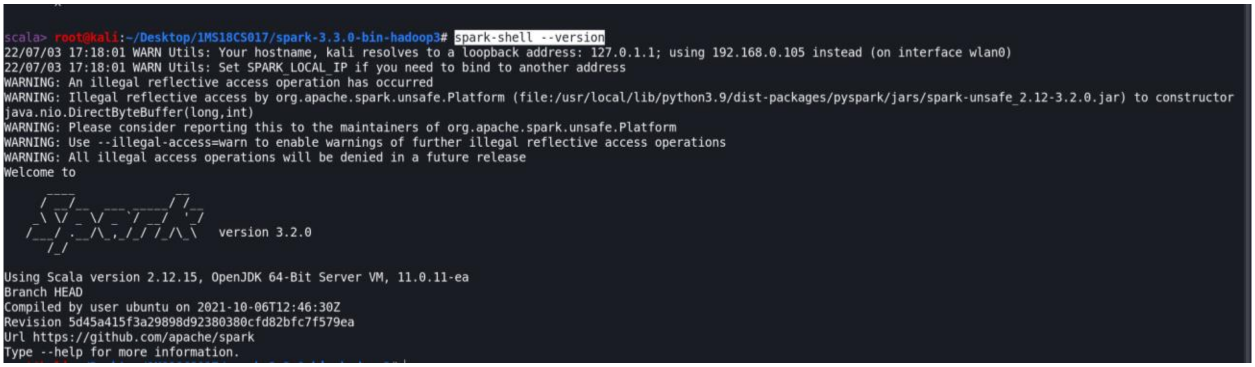
9. Execute the bash.sh File using following command source Bash.sh.
NOTE: Make source before compiling or running spark compile this file.

10. Verify JAVA_HOME variable to be set to Java Path and PATH variable has your USN Spark Folder.If any previous PATH set to Spark Folder remove that inside .bashrc file.



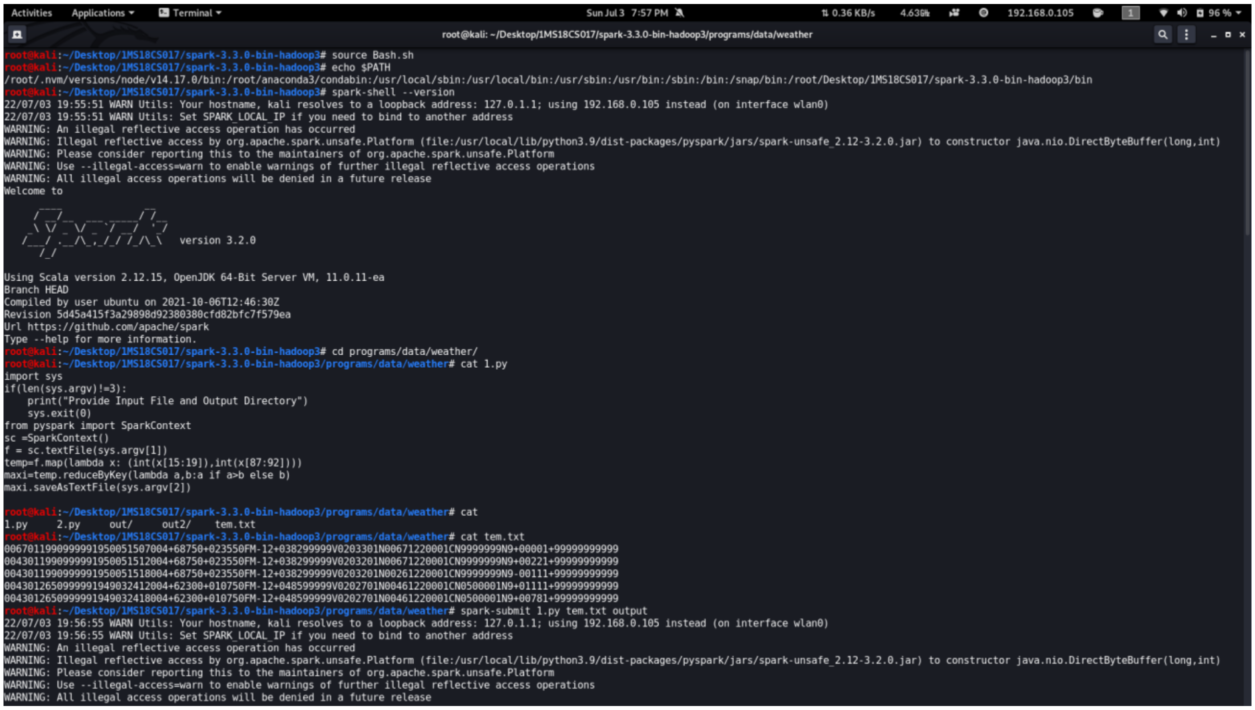
```
root@kali:~/Desktop/1MS18CS017/spark-3.3.0-bin-hadoop3# echo $PATH
/root/.nvm/versions/node/v14.17.0/bin:/root/anaconda3/condabin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin:/root/Desktop/1MS18CS017/spark-3.3.0-bi
n-hadoop3/bin
root@kali:~/Desktop/1MS18CS017/spark-3.3.0-bin-hadoop3# echo $JAVA_HOME
/usr/lib/jvm/java-11-openjdk-amd64
root@kali:~/Desktop/1MS18CS017/spark-3.3.0-bin-hadoop3# |
```

11. Verify Hadoop is Installed or not by executing spark-shell --version command.if command gives Information about Hadoop command then Hadoop is Successfully Installed.



```
scala> root@kali:~/Desktop/1MS18CS017/spark-3.3.0-bin-hadoop3# spark-shell --version
22/07/03 17:18:01 WARN Utils: Your hostname, kali resolves to a loopback address: 127.0.1.1; using 192.168.0.105 instead (on interface wlan0)
22/07/03 17:18:01 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:/usr/local/lib/python3.9/dist-packages/pyspark/jars/spark-unsafe_2.12-3.2.0.jar) to constructor
java.nio.DirectByteBuffer(long,int)
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
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /___/ .__/\_,_/_/ /_/\_\   version 3.2.0
      /_/

Using Scala version 2.12.15, OpenJDK 64-Bit Server VM, 11.0.11-ea
Branch HEAD
Compiled by user ubuntu on 2021-10-06T12:46:30Z
Revision 5d45a415f3a29898d92380380cfd82bfc7f579ea
Url https://github.com/apache/spark
Type --help for more information.
```

Execute all spark python files with spark-submit<python_filename>.py <inputFile> <outputfolder>



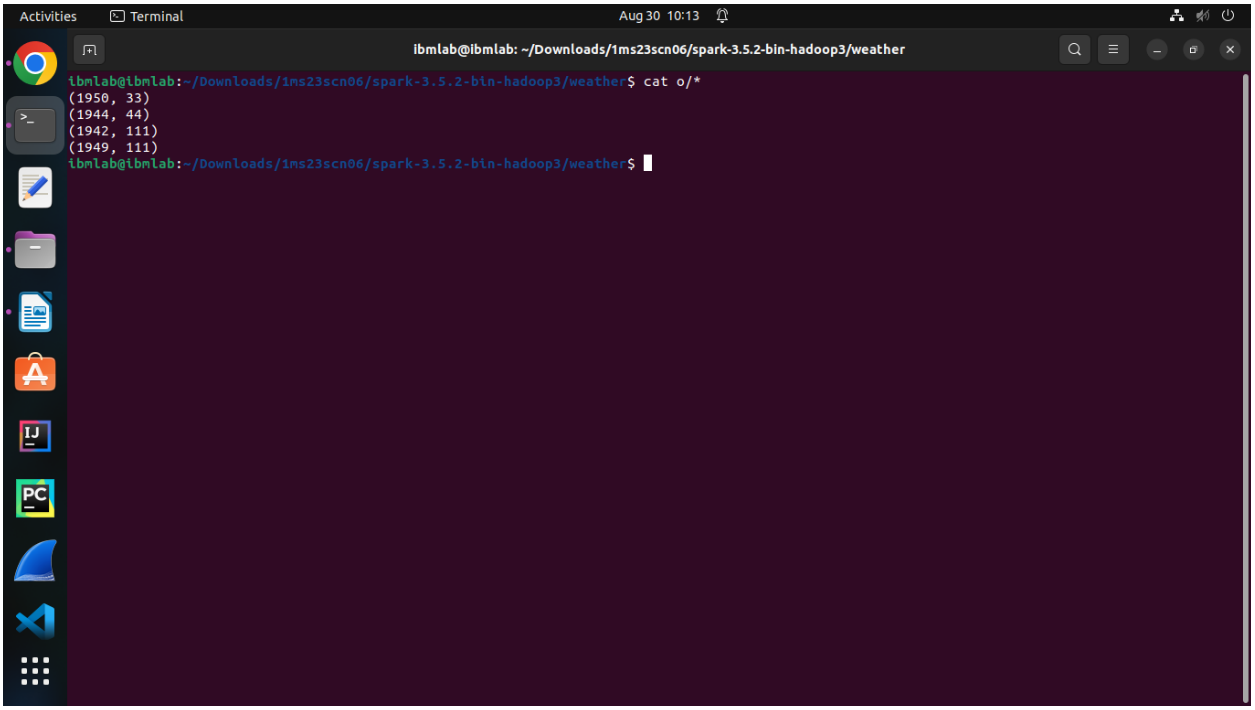
```
root@kali:~/Desktop/1MS18CS017/spark-3.3.0-bin-hadoop3# source Bash.sh
root@kali:~/Desktop/1MS18CS017/spark-3.3.0-bin-hadoop3# echo $PATH
/root/.nvm/versions/node/v14.17.0/bin:/root/anaconda3/condabin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin:/root/Desktop/1MS18CS017/spark-3.3.0-bin-hadoop3/bin
root@kali:~/Desktop/1MS18CS017/spark-3.3.0-bin-hadoop3# spark-shell --version
22/07/03 19:55:51 WARN Utils: Your hostname, kali resolves to a loopback address: 127.0.1.1; using 192.168.0.105 instead (on interface wlan0)
22/07/03 19:55:51 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:/usr/local/lib/python3.9/dist-packages/pyspark/jars/spark-unsafe_2.12-3.2.0.jar) to constructor java.nio.DirectByteBuffer(long,int)
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
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /___/ .__/\_,_/_/ /_/\_\   version 3.2.0
      /_/

Using Scala version 2.12.15, OpenJDK 64-Bit Server VM, 11.0.11-ea
Branch HEAD
Compiled by user ubuntu on 2021-10-06T12:46:30Z
Revision 5d45a415f3a29898d92380380cfd82bfc7f579ea
Url https://github.com/apache/spark
Type --help for more information.
root@kali:~/Desktop/1MS18CS017/spark-3.3.0-bin-hadoop3# cd programs/data/weather/
root@kali:~/Desktop/1MS18CS017/spark-3.3.0-bin-hadoop3/programs/data/weather# cat 1.py
import sys
if(len(sys.argv)!=3):
    print("Provide Input File and Output Directory")
    sys.exit(0)
from pyspark import SparkContext
sc =SparkContext()
f = sc.textFile(sys.argv[1])
temp=f.map(lambda x: (int(x[15:19]),int(x[87:92])))
maxi=temp.reduceByKey(lambda a,b:a if a>b else b)
maxi.saveAsTextFile(sys.argv[2])

root@kali:~/Desktop/1MS18CS017/spark-3.3.0-bin-hadoop3/programs/data/weather# cat
1.py     2.py     out/     out2/    tem.txt
root@kali:~/Desktop/1MS18CS017/spark-3.3.0-bin-hadoop3/programs/data/weather# cat tem.txt
0067011990999991950051507004+68750+023550FM-12+038299999V0203301N00671220001CN9999999N9+00001+99999999999
0043011990999991950051512004+68750+023550FM-12+038299999V0203201N00671220001CN9999999N9+00221+99999999999
0043011990999991950051518004+68750+023550FM-12+038299999V0203201N00261220001CN9999999N9-00111+99999999999
0043012650999991949032412004+62300+010750FM-12+048599999V0202701N00461220001CN0500001N9+01111+99999999999
0043012650999991949032418004+62300+010750FM-12+048599999V0202701N00461220001CN0500001N9+00781+99999999999
root@kali:~/Desktop/1MS18CS017/spark-3.3.0-bin-hadoop3/programs/data/weather# spark-submit 1.py tem.txt output
22/07/03 19:56:55 WARN Utils: Your hostname, kali resolves to a loopback address: 127.0.1.1; using 192.168.0.105 instead (on interface wlan0)
22/07/03 19:56:55 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:/usr/local/lib/python3.9/dist-packages/pyspark/jars/spark-unsafe_2.12-3.2.0.jar) to constructor java.nio.DirectByteBuffer(long,int)
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
```

SPARK Programs

-Jeevan Raj H (1MS23SCN06)

- **Write a spark to analyze the given weather report data and to generate a report with cities having maximum temperature for a particular year.**

```
import sys
if(len(sys.argv)!=3):
    print("Provide Input File and Output Directory")
    sys.exit(0)
from pyspark import SparkContext
sc =SparkContext()
f = sc.textFile(sys.argv[1])
temp=f.map(lambda x: (int(x[15:19]),int(x[87:92])))
maxi=temp.reduceByKey(lambda a,b:a if a>b else b)
maxi.saveAsTextFile(sys.argv[2])
```

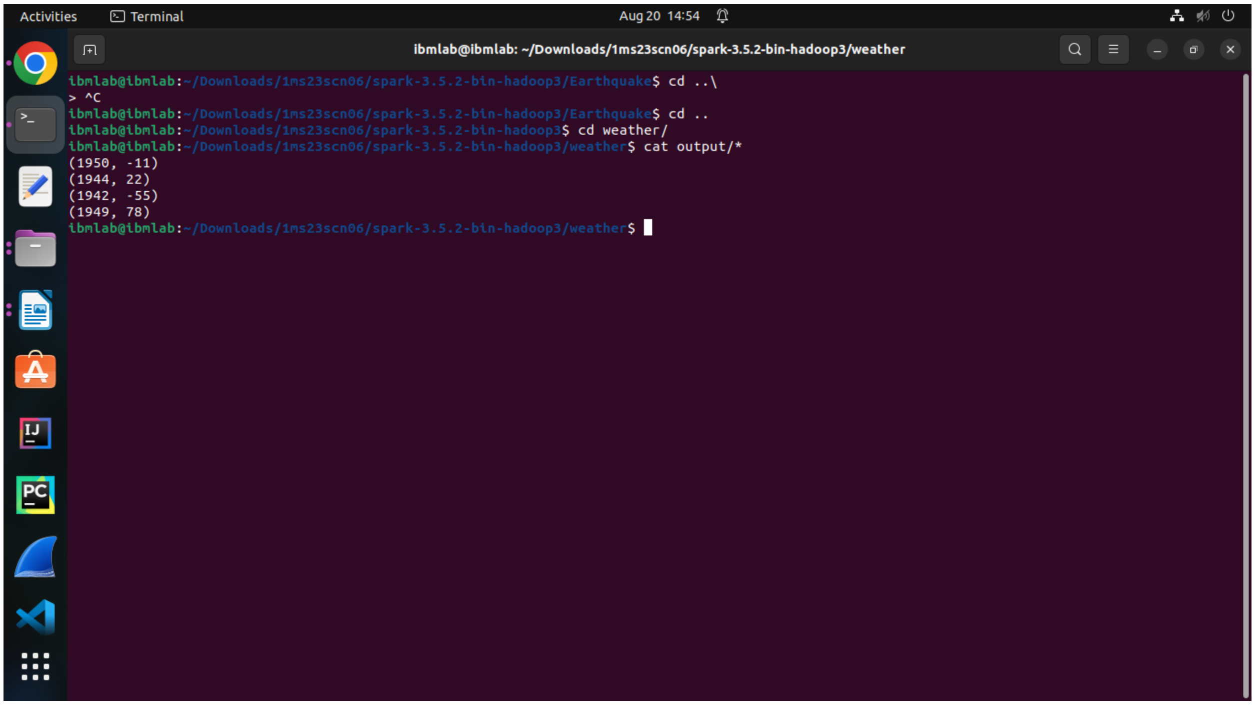

Activities
Terminal
Aug 30 10:13
ibmlab@ibmlab: ~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/weather

```
ibmlab@ibmlab:~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/weather$ cat o/*
(1950, 33)
(1944, 44)
(1942, 111)
(1949, 111)
ibmlab@ibmlab:~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/weather$ 
```

- **Write a spark to analyze the given weather report data and to generate a report with cities having minimum temperature for a particular year.**

```
import sys
if(len(sys.argv)!=3):
    print("Provide Input File and Output Directory")
    sys.exit(0)
```

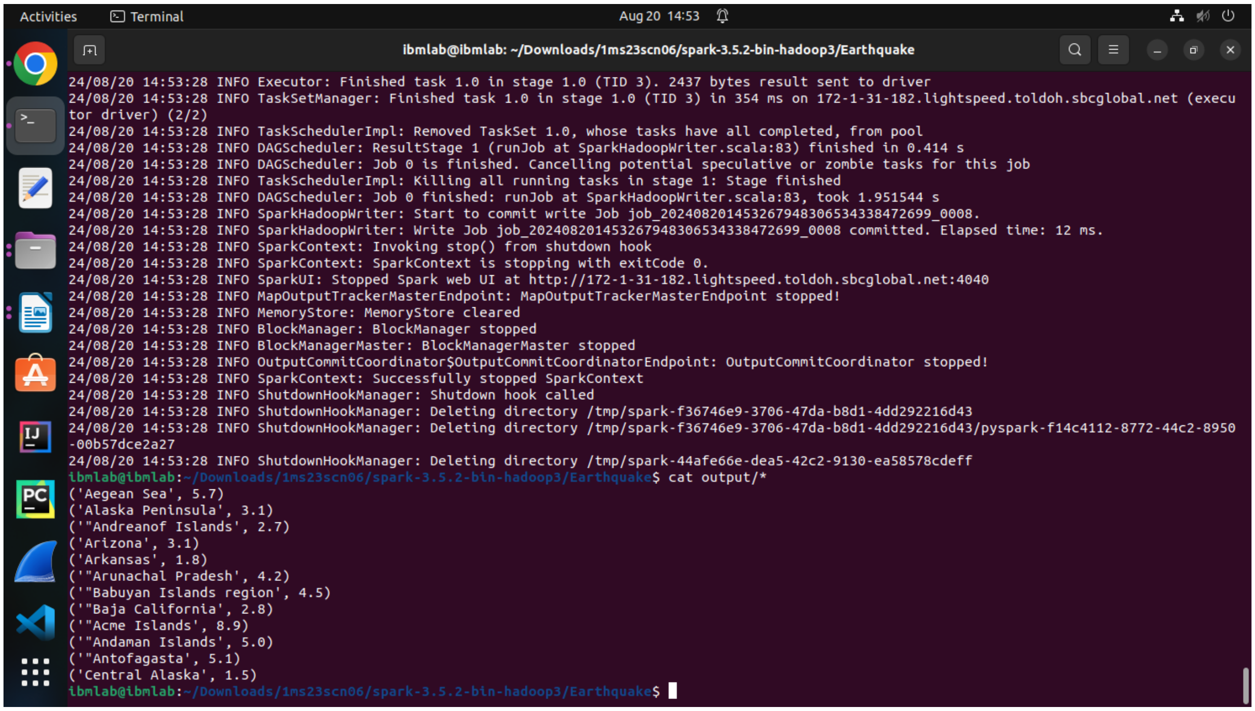
```
from pyspark import SparkContext
sc =SparkContext()
f = sc.textFile(sys.argv[1])
temp=f.map(lambda x: (int(x[15:19]),int(x[87:92])))
mini=temp.reduceByKey(lambda a,b:a if a<b else b)
mini.saveAsTextFile(sys.argv[2])
```


Activities
Terminal
Aug 20 14:54
ibmlab@ibmlab: ~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/weather

```
ibmlab@ibmlab:~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/Earthquake$ cd ..\
> ^C
ibmlab@ibmlab:~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/Earthquake$ cd ..
ibmlab@ibmlab:~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3$ cd weather/
ibmlab@ibmlab:~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/weather$ cat output/*
(1950, -11)
(1944, 22)
(1942, -55)
(1949, 78)
ibmlab@ibmlab:~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/weather$ 
```

- **Write a spark program to analyze the given Earthquake data and generate statistics with region and magnitud**e.

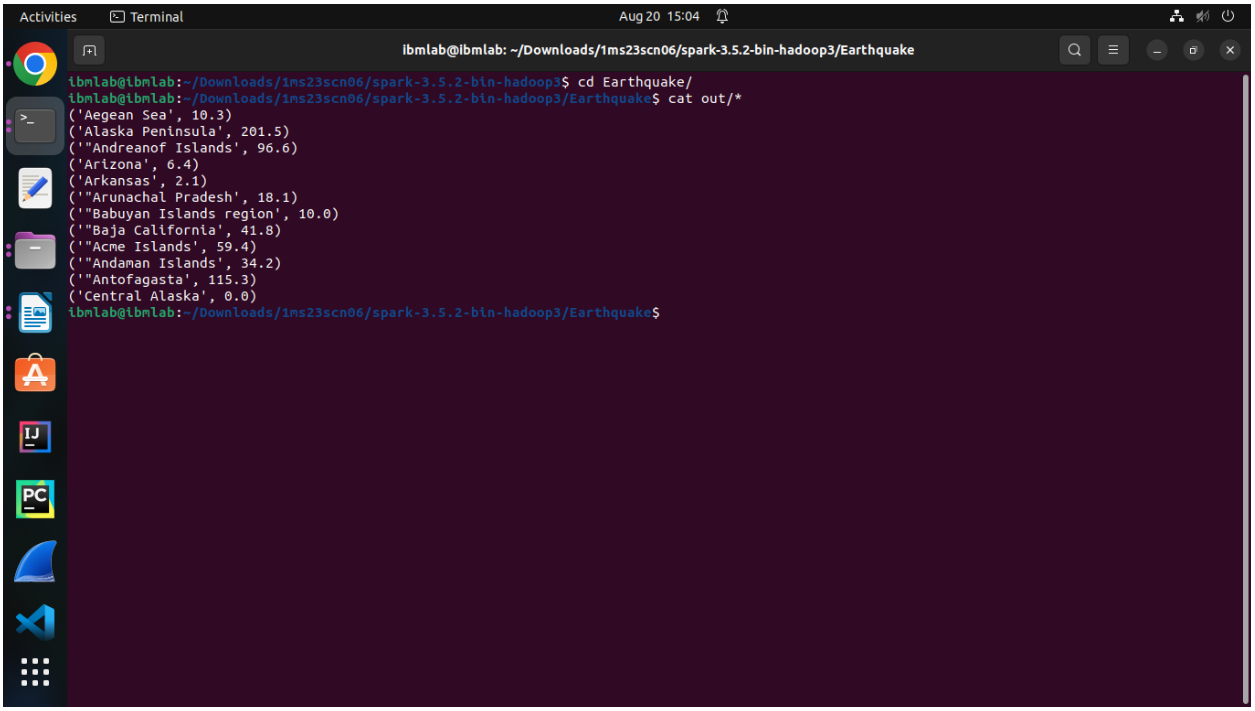
```
import sys
if(len(sys.argv)!=3):
    print("Provide Input File and Output Directory")
    sys.exit(0)
from pyspark import SparkContext
sc =SparkContext()
f = sc.textFile(sys.argv[1])
temp=f.map(lambda x: (x.split(',')[11],float(x.split(',')[8])))
maxi=temp.reduceByKey(lambda a,b:a if a>b else b)
maxi.saveAsTextFile(sys.argv[2])\
```


Activities
Terminal
Aug 20 14:53
ibmlab@ibmlab: ~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/Earthquake

```
24/08/20 14:53:28 INFO Executor: Finished task 1.0 in stage 1.0 (TID 3). 2437 bytes result sent to driver
24/08/20 14:53:28 INFO TaskSetManager: Finished task 1.0 in stage 1.0 (TID 3) in 354 ms on 172-1-31-182.lightspeed.toldoh.sbcglobal.net (execu
tor driver) (2/2)
24/08/20 14:53:28 INFO TaskSchedulerImpl: Removed TaskSet 1.0, whose tasks have all completed, from pool
24/08/20 14:53:28 INFO DAGScheduler: ResultStage 1 (runJob at SparkHadoopWriter.scala:83) finished in 0.414 s
24/08/20 14:53:28 INFO DAGScheduler: Job 0 is finished. Cancelling potential speculative or zombie tasks for this job
24/08/20 14:53:28 INFO TaskSchedulerImpl: Killing all running tasks in stage 1: Stage finished
24/08/20 14:53:28 INFO DAGScheduler: Job 0 finished: runJob at SparkHadoopWriter.scala:83, took 1.951544 s
24/08/20 14:53:28 INFO SparkHadoopWriter: Start to commit write Job job_202408201453267948306534338472699_0008.
24/08/20 14:53:28 INFO SparkHadoopWriter: Write Job job_202408201453267948306534338472699_0008 committed. Elapsed time: 12 ms.
24/08/20 14:53:28 INFO SparkContext: Invoking stop() from shutdown hook
24/08/20 14:53:28 INFO SparkContext: SparkContext is stopping with exitCode 0.
24/08/20 14:53:28 INFO SparkUI: Stopped Spark web UI at http://172-1-31-182.lightspeed.toldoh.sbcglobal.net:4040
24/08/20 14:53:28 INFO MapOutputTrackerMasterEndpoint: MapOutputTrackerMasterEndpoint stopped!
24/08/20 14:53:28 INFO MemoryStore: MemoryStore cleared
24/08/20 14:53:28 INFO BlockManager: BlockManager stopped
24/08/20 14:53:28 INFO BlockManagerMaster: BlockManagerMaster stopped
24/08/20 14:53:28 INFO OutputCommitCoordinator$OutputCommitCoordinatorEndpoint: OutputCommitCoordinator stopped!
24/08/20 14:53:28 INFO SparkContext: Successfully stopped SparkContext
24/08/20 14:53:28 INFO ShutdownHookManager: Shutdown hook called
24/08/20 14:53:28 INFO ShutdownHookManager: Deleting directory /tmp/spark-f36746e9-3706-47da-b8d1-4dd292216d43
24/08/20 14:53:28 INFO ShutdownHookManager: Deleting directory /tmp/spark-f36746e9-3706-47da-b8d1-4dd292216d43/pyspark-f14c4112-8772-44c2-8950
-00b57dce2a27
24/08/20 14:53:28 INFO ShutdownHookManager: Deleting directory /tmp/spark-44afe66e-dea5-42c2-9130-ea58578cdeff
ibmlab@ibmlab:~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/Earthquake$ cat output/*
('Aegean Sea', 5.7)
('Alaska Peninsula', 3.1)
('"Andreanof Islands', 2.7)
('Arizona', 3.1)
('Arkansas', 1.8)
('"Arunachal Pradesh', 4.2)
('"Babuyan Islands region', 4.5)
('"Baja California', 2.8)
('"Acme Islands', 8.9)
('"Andaman Islands', 5.0)
('"Antofagasta', 5.1)
('Central Alaska', 1.5)
ibmlab@ibmlab:~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/Earthquake$ 
```

- **Write a spark program to analyze the given Earthquake data and generate statistics with region and depth.**

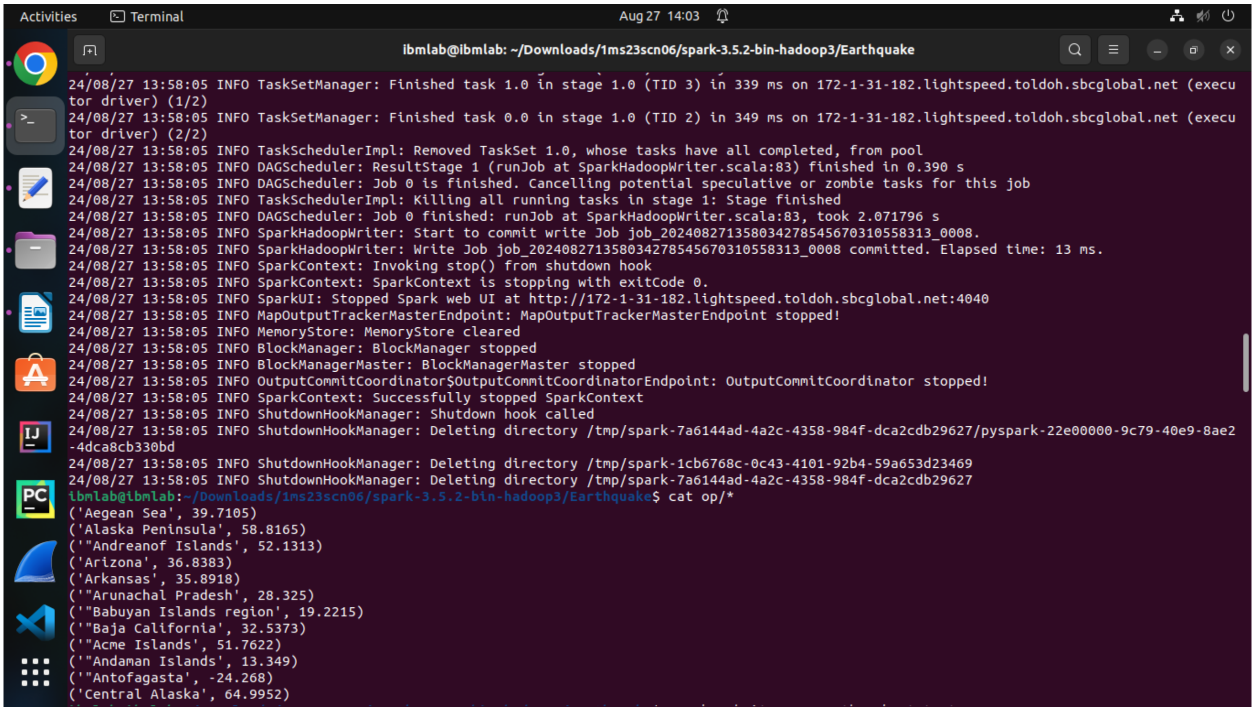
```
import sys
if(len(sys.argv)!=3):
      print("Provide Input File and Output Directory")
      sys.exit(0)
from pyspark import SparkContext
sc =SparkContext()
f = sc.textFile(sys.argv[1])
temp=f.map(lambda x: (x.split(',')[11],float(x.split(',')[9])))
maxi=temp.reduceByKey(lambda a,b:a if a>b else b
maxi.saveAsTextFile(sys.argv[2])
```


Activities
Terminal
Aug 20 15:04
ibmlab@ibmlab: ~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/Earthquake

```
ibmlab@ibmlab:~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3$ cd Earthquake/
ibmlab@ibmlab:~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/Earthquake$ cat out/*
('Aegean Sea', 10.3)
('Alaska Peninsula', 201.5)
('"Andreanof Islands', 96.6)
('Arizona', 6.4)
('Arkansas', 2.1)
('"Arunachal Pradesh', 18.1)
('"Babuyan Islands region', 10.0)
('"Baja California', 41.8)
('"Acme Islands', 59.4)
('"Andaman Islands', 34.2)
('"Antofagasta', 115.3)
('Central Alaska', 0.0)
ibmlab@ibmlab:~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/Earthquake$
```

- **Write a spark program to analyze the given Earthquake data and generate statistics with region and latitude.**

```
import sys
if(len(sys.argv)!=3):
    print("Provide Input File and Output Directory")
    sys.exit(0)
from pyspark import SparkContext
sc =SparkContext()
f = sc.textFile(sys.argv[1])
temp=f.map(lambda x: (x.split(',')[11],float(x.split(',')[6])))
maxi=temp.reduceByKey(lambda a,b:a if a>b else b)
maxi.saveAsTextFile(sys.argv[2])
```

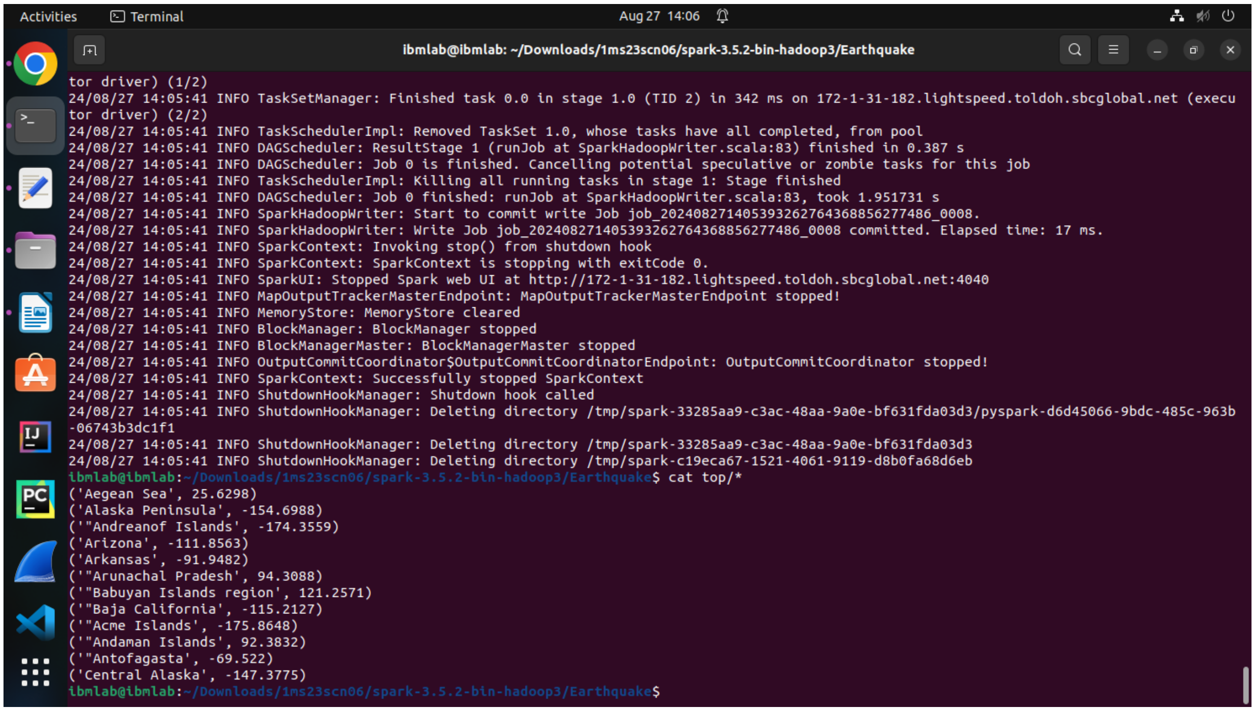


Activities
Terminal
Aug 27 14:03
ibmlab@ibmlab: ~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/Earthquake

```
24/08/27 13:58:05 INFO TaskSetManager: Finished task 1.0 in stage 1.0 (TID 3) in 339 ms on 172-1-31-182.lightspeed.toldoh.sbcglobal.net (execu
tor driver) (1/2)
24/08/27 13:58:05 INFO TaskSetManager: Finished task 0.0 in stage 1.0 (TID 2) in 349 ms on 172-1-31-182.lightspeed.toldoh.sbcglobal.net (execu
tor driver) (2/2)
24/08/27 13:58:05 INFO TaskSchedulerImpl: Removed TaskSet 1.0, whose tasks have all completed, from pool
24/08/27 13:58:05 INFO DAGScheduler: ResultStage 1 (runJob at SparkHadoopWriter.scala:83) finished in 0.390 s
24/08/27 13:58:05 INFO DAGScheduler: Job 0 is finished. Cancelling potential speculative or zombie tasks for this job
24/08/27 13:58:05 INFO TaskSchedulerImpl: Killing all running tasks in stage 1: Stage finished
24/08/27 13:58:05 INFO DAGScheduler: Job 0 finished: runJob at SparkHadoopWriter.scala:83, took 2.071796 s
24/08/27 13:58:05 INFO SparkHadoopWriter: Start to commit write Job job_202408271358034278545670310558313_0008.
24/08/27 13:58:05 INFO SparkHadoopWriter: Write Job job_202408271358034278545670310558313_0008 committed. Elapsed time: 13 ms.
24/08/27 13:58:05 INFO SparkContext: Invoking stop() from shutdown hook
24/08/27 13:58:05 INFO SparkContext: SparkContext is stopping with exitCode 0.
24/08/27 13:58:05 INFO SparkUI: Stopped Spark web UI at http://172-1-31-182.lightspeed.toldoh.sbcglobal.net:4040
24/08/27 13:58:05 INFO MapOutputTrackerMasterEndpoint: MapOutputTrackerMasterEndpoint stopped!
24/08/27 13:58:05 INFO MemoryStore: MemoryStore cleared
24/08/27 13:58:05 INFO BlockManager: BlockManager stopped
24/08/27 13:58:05 INFO BlockManagerMaster: BlockManagerMaster stopped
24/08/27 13:58:05 INFO OutputCommitCoordinator$OutputCommitCoordinatorEndpoint: OutputCommitCoordinator stopped!
24/08/27 13:58:05 INFO SparkContext: Successfully stopped SparkContext
24/08/27 13:58:05 INFO ShutdownHookManager: Shutdown hook called
24/08/27 13:58:05 INFO ShutdownHookManager: Deleting directory /tmp/spark-7a6144ad-4a2c-4358-984f-dca2cdb29627/pyspark-22e00000-9c79-40e9-8ae2
-4dca8cb330bd
24/08/27 13:58:05 INFO ShutdownHookManager: Deleting directory /tmp/spark-1cb6768c-0c43-4101-92b4-59a653d23469
24/08/27 13:58:05 INFO ShutdownHookManager: Deleting directory /tmp/spark-7a6144ad-4a2c-4358-984f-dca2cdb29627
ibmlab@ibmlab:~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/Earthquake$ cat op/*
('Aegean Sea', 39.7105)
('Alaska Peninsula', 58.8165)
('"Andreanof Islands', 52.1313)
('Arizona', 36.8383)
('Arkansas', 35.8918)
('"Arunachal Pradesh', 28.325)
('"Babuyan Islands region', 19.2215)
('"Baja California', 32.5373)
('"Acme Islands', 51.7622)
('"Andaman Islands', 13.349)
('"Antofagasta', -24.268)
('Central Alaska', 64.9952)
```

- **Write a spark program to analyze the given Earthquake data and generate statistics with region and longitude.**

```
import sys
if(len(sys.argv)!=3):
    print("Provide Input File and Output Directory")
    sys.exit(0)
from pyspark import SparkContext
sc =SparkContext()
f = sc.textFile(sys.argv[1])
temp=f.map(lambda x: (x.split(',')[11],float(x.split(',')[7])))
maxi=temp.reduceByKey(lambda a,b:a if a>b else b)
maxi.saveAsTextFile(sys.argv[2])
```

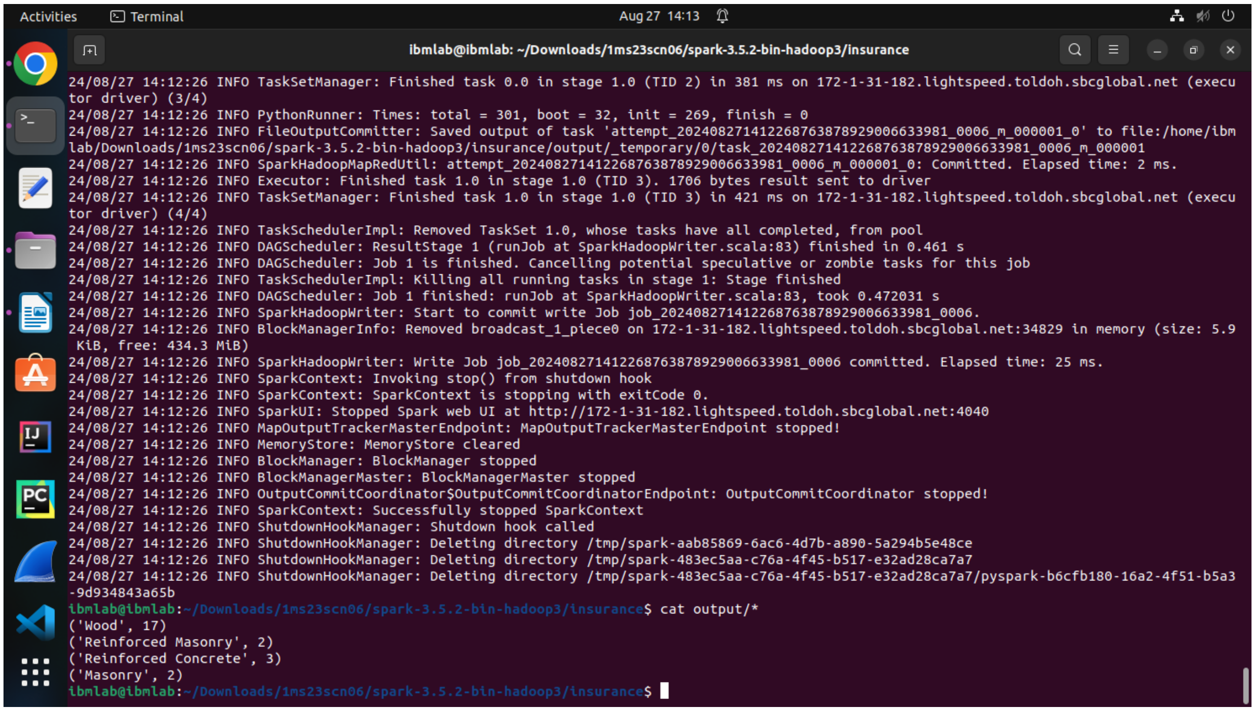


Activities
Terminal
Aug 27 14:06
ibmlab@ibmlab: ~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/Earthquake

```
tor driver) (1/2)
24/08/27 14:05:41 INFO TaskSetManager: Finished task 0.0 in stage 1.0 (TID 2) in 342 ms on 172-1-31-182.lightspeed.toldoh.sbcglobal.net (execu
tor driver) (2/2)
24/08/27 14:05:41 INFO TaskSchedulerImpl: Removed TaskSet 1.0, whose tasks have all completed, from pool
24/08/27 14:05:41 INFO DAGScheduler: ResultStage 1 (runJob at SparkHadoopWriter.scala:83) finished in 0.387 s
24/08/27 14:05:41 INFO DAGScheduler: Job 0 is finished. Cancelling potential speculative or zombie tasks for this job
24/08/27 14:05:41 INFO TaskSchedulerImpl: Killing all running tasks in stage 1: Stage finished
24/08/27 14:05:41 INFO DAGScheduler: Job 0 finished: runJob at SparkHadoopWriter.scala:83, took 1.951731 s
24/08/27 14:05:41 INFO SparkHadoopWriter: Start to commit write Job job_202408271405393262764368856277486_0008.
24/08/27 14:05:41 INFO SparkHadoopWriter: Write Job job_202408271405393262764368856277486_0008 committed. Elapsed time: 17 ms.
24/08/27 14:05:41 INFO SparkContext: Invoking stop() from shutdown hook
24/08/27 14:05:41 INFO SparkContext: SparkContext is stopping with exitCode 0.
24/08/27 14:05:41 INFO SparkUI: Stopped Spark web UI at http://172-1-31-182.lightspeed.toldoh.sbcglobal.net:4040
24/08/27 14:05:41 INFO MapOutputTrackerMasterEndpoint: MapOutputTrackerMasterEndpoint stopped!
24/08/27 14:05:41 INFO MemoryStore: MemoryStore cleared
24/08/27 14:05:41 INFO BlockManager: BlockManager stopped
24/08/27 14:05:41 INFO BlockManagerMaster: BlockManagerMaster stopped
24/08/27 14:05:41 INFO OutputCommitCoordinator$OutputCommitCoordinatorEndpoint: OutputCommitCoordinator stopped!
24/08/27 14:05:41 INFO SparkContext: Successfully stopped SparkContext
24/08/27 14:05:41 INFO ShutdownHookManager: Shutdown hook called
24/08/27 14:05:41 INFO ShutdownHookManager: Deleting directory /tmp/spark-33285aa9-c3ac-48aa-9a0e-bf631fda03d3/pyspark-d6d45066-9bdc-485c-963b
-06743b3dc1f1
24/08/27 14:05:41 INFO ShutdownHookManager: Deleting directory /tmp/spark-33285aa9-c3ac-48aa-9a0e-bf631fda03d3
24/08/27 14:05:41 INFO ShutdownHookManager: Deleting directory /tmp/spark-c19eca67-1521-4061-9119-d8b0fa68d6eb
ibmlab@ibmlab:~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/Earthquake$ cat top/*
('Aegean Sea', 25.6298)
('Alaska Peninsula', -154.6988)
('"Andreanof Islands', -174.3559)
('Arizona', -111.8563)
('Arkansas', -91.9482)
('"Arunachal Pradesh', 94.3088)
('"Babuyan Islands region', 121.2571)
('"Baja California', -115.2127)
('"Acme Islands', -175.8648)
('"Andaman Islands', 92.3832)
('"Antofagasta', -69.522)
('Central Alaska', -147.3775)
ibmlab@ibmlab:~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/Earthquake$
```

- **Write a spark program to analyze the given Insurance data and generate a statistics report with the construction building name and the count of building.**

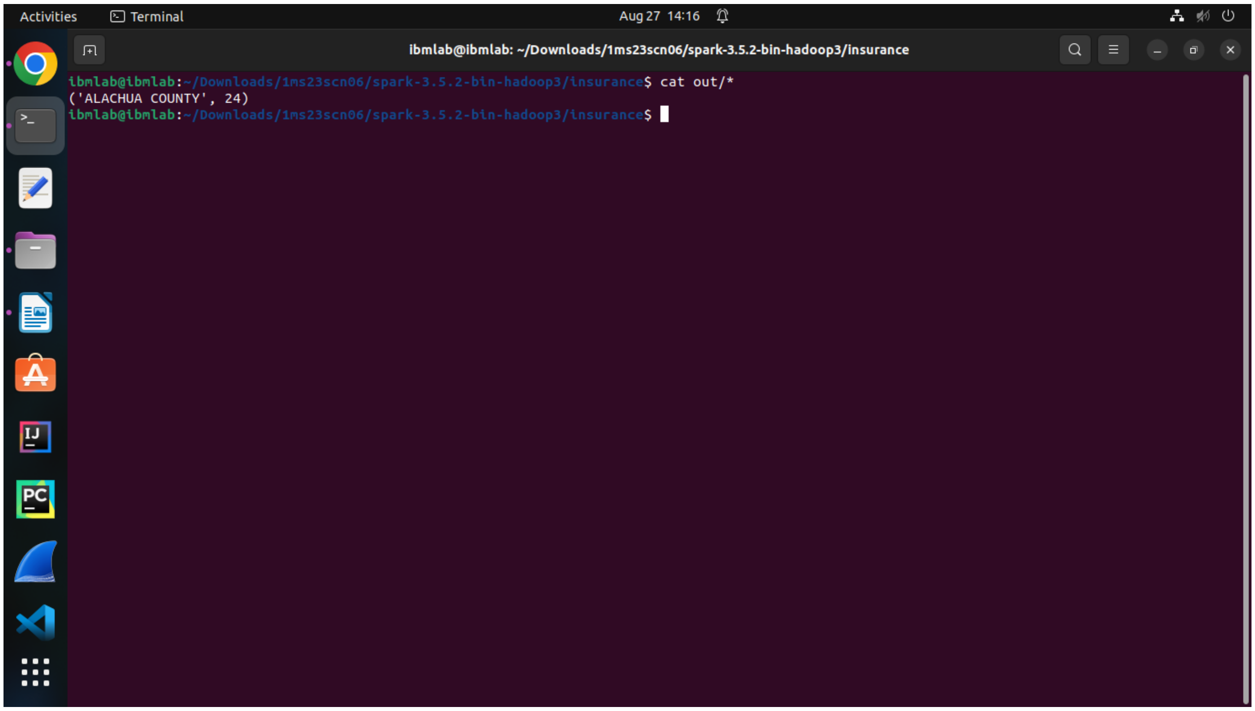
```
import sys
if(len(sys.argv)!=3):
    print("Provide Input File and Output Directory")
    sys.exit(0)
from pyspark import SparkContext
sc =SparkContext()
f = sc.textFile(sys.argv[1])
temp=f.map(lambda x: (x.split(',')[16],1))
data=temp.countByKey()
dd=sc.parallelize(data.items())
dd.saveAsTextFile(sys.argv[2])
```


Activities
Terminal
Aug 27 14:13
ibmlab@ibmlab: ~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/insurance

```
24/08/27 14:12:26 INFO TaskSetManager: Finished task 0.0 in stage 1.0 (TID 2) in 381 ms on 172-1-31-182.lightspeed.toldoh.sbcglobal.net (execu
tor driver) (3/4)
24/08/27 14:12:26 INFO PythonRunner: Times: total = 301, boot = 32, init = 269, finish = 0
24/08/27 14:12:26 INFO FileOutputCommitter: Saved output of task 'attempt_202408271412268763878929006633981_0006_m_000001_0' to file:/home/ibm
lab/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/insurance/output/_temporary/0/task_202408271412268763878929006633981_0006_m_000001
24/08/27 14:12:26 INFO SparkHadoopMapRedUtil: attempt_202408271412268763878929006633981_0006_m_000001_0: Committed. Elapsed time: 2 ms.
24/08/27 14:12:26 INFO Executor: Finished task 1.0 in stage 1.0 (TID 3). 1706 bytes result sent to driver
24/08/27 14:12:26 INFO TaskSetManager: Finished task 1.0 in stage 1.0 (TID 3) in 421 ms on 172-1-31-182.lightspeed.toldoh.sbcglobal.net (execu
tor driver) (4/4)
24/08/27 14:12:26 INFO TaskSchedulerImpl: Removed TaskSet 1.0, whose tasks have all completed, from pool
24/08/27 14:12:26 INFO DAGScheduler: ResultStage 1 (runJob at SparkHadoopWriter.scala:83) finished in 0.461 s
24/08/27 14:12:26 INFO DAGScheduler: Job 1 is finished. Cancelling potential speculative or zombie tasks for this job
24/08/27 14:12:26 INFO TaskSchedulerImpl: Killing all running tasks in stage 1: Stage finished
24/08/27 14:12:26 INFO DAGScheduler: Job 1 finished: runJob at SparkHadoopWriter.scala:83, took 0.472031 s
24/08/27 14:12:26 INFO SparkHadoopWriter: Start to commit write Job job_202408271412268763878929006633981_0006.
24/08/27 14:12:26 INFO BlockManagerInfo: Removed broadcast_1_piece0 on 172-1-31-182.lightspeed.toldoh.sbcglobal.net:34829 in memory (size: 5.9
 KiB, free: 434.3 MiB)
24/08/27 14:12:26 INFO SparkHadoopWriter: Write Job job_202408271412268763878929006633981_0006 committed. Elapsed time: 25 ms.
24/08/27 14:12:26 INFO SparkContext: Invoking stop() from shutdown hook
24/08/27 14:12:26 INFO SparkContext: SparkContext is stopping with exitCode 0.
24/08/27 14:12:26 INFO SparkUI: Stopped Spark web UI at http://172-1-31-182.lightspeed.toldoh.sbcglobal.net:4040
24/08/27 14:12:26 INFO MapOutputTrackerMasterEndpoint: MapOutputTrackerMasterEndpoint stopped!
24/08/27 14:12:26 INFO MemoryStore: MemoryStore cleared
24/08/27 14:12:26 INFO BlockManager: BlockManager stopped
24/08/27 14:12:26 INFO BlockManagerMaster: BlockManagerMaster stopped
24/08/27 14:12:26 INFO OutputCommitCoordinator$OutputCommitCoordinatorEndpoint: OutputCommitCoordinator stopped!
24/08/27 14:12:26 INFO SparkContext: Successfully stopped SparkContext
24/08/27 14:12:26 INFO ShutdownHookManager: Shutdown hook called
24/08/27 14:12:26 INFO ShutdownHookManager: Deleting directory /tmp/spark-aab85869-6ac6-4d7b-a890-5a294b5e48ce
24/08/27 14:12:26 INFO ShutdownHookManager: Deleting directory /tmp/spark-483ec5aa-c76a-4f45-b517-e32ad28ca7a7
24/08/27 14:12:26 INFO ShutdownHookManager: Deleting directory /tmp/spark-483ec5aa-c76a-4f45-b517-e32ad28ca7a7/pyspark-b6cfb180-16a2-4f51-b5a3
-9d934843a65b
ibmlab@ibmlab:~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/insurance$ cat output/*
('Wood', 17)
('Reinforced Masonry', 2)
('Reinforced Concrete', 3)
('Masonry', 2)
ibmlab@ibmlab:~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/insurance$
```

- **Write a spark program to analyze the given Insurance data and generate a statistics report with the county name and its frequency.**

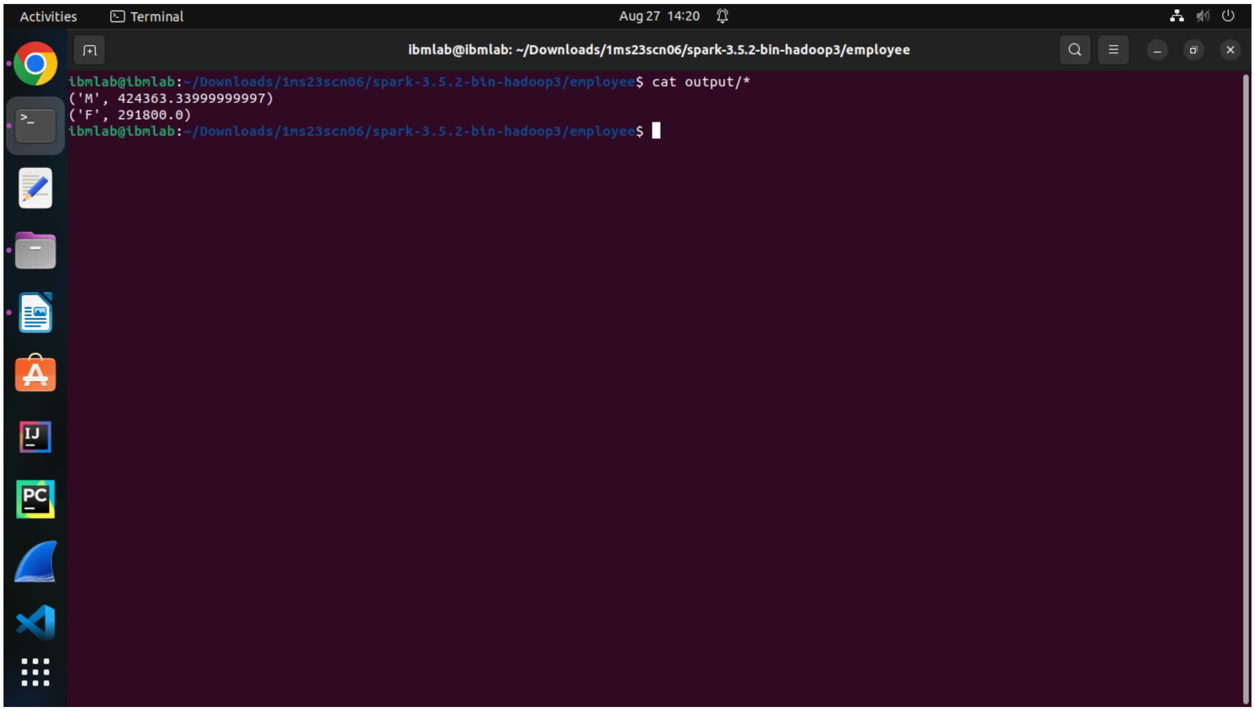
```
import sys
if(len(sys.argv)!=3):
    print("Provide Input File and Output Directory")
    sys.exit(0)
from pyspark import SparkContext
sc =SparkContext()
f = sc.textFile(sys.argv[1])
temp=f.map(lambda x: (x.split(',')[2],1))
data=temp.countByKey()
dd=sc.parallelize(data.items())
dd.saveAsTextFile(sys.argv[2])
```


Activities
Terminal
Aug 27 14:16
ibmlab@ibmlab: ~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/insurance

```
ibmlab@ibmlab:~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/insurance$ cat out/*
('ALACHUA COUNTY', 24)
ibmlab@ibmlab:~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/insurance$ 
```

- **Write a map-reduce program to analyze the given employee record data and generate a statistics report with the total Sales for female and male employees.**

```
import sys
if(len(sys.argv)!=3):
    print("Provide Input File and Output Directory")
    sys.exit(0)
from pyspark import SparkContext
sc =SparkContext()
f = sc.textFile(sys.argv[1])
temp=f.map(lambda x: (x.split('\t')[3],float(x.split('\t')[8])))
total=temp.reduceByKey(lambda a,b : a+b)
total.saveAsTextFile(sys.argv[2])
```

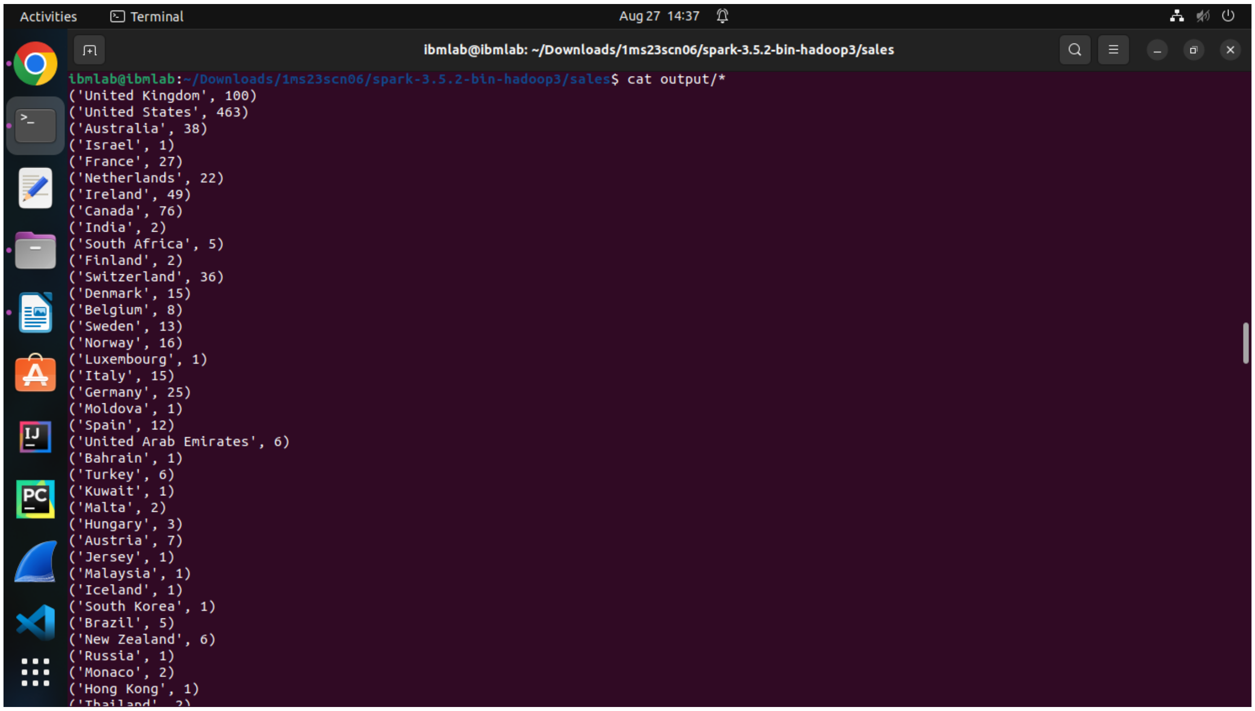


Activities
Terminal
Aug 27 14:20
ibmlab@ibmlab: ~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/employee

```
ibmlab@ibmlab:~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/employee$ cat output/*
('M', 424363.33999999997)
('F', 291800.0)
ibmlab@ibmlab:~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/employee$ 
```

- **Write a map-reduce program to analyze the given sales records over a period of time and generate data about the country’s total sales, and the total number of the products.**

```
import sys
if(len(sys.argv)!=3):
    print("Provide Input File and Output Directory")
    sys.exit(0)
from pyspark import SparkContext
sc =SparkContext()
f = sc.textFile(sys.argv[1])
temp=f.map(lambda x: (x.split(',')[7],1))
data=temp.countByKey()
dd=sc.parallelize(data.items())
dd.saveAsTextFile(sys.argv[2])
```

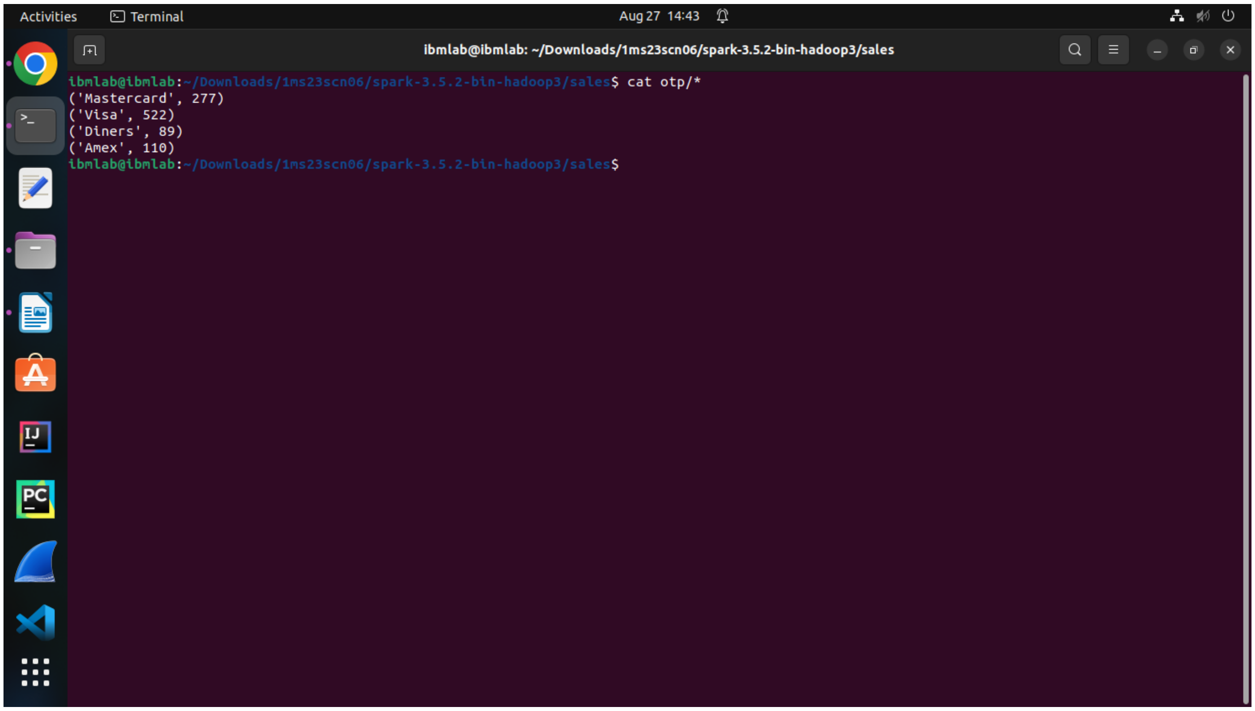


Activities
Terminal
Aug 27 14:37
ibmlab@ibmlab: ~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/sales

```
ibmlab@ibmlab:~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/sales$ cat output/*
('United Kingdom', 100)
('United States', 463)
('Australia', 38)
('Israel', 1)
('France', 27)
('Netherlands', 22)
('Ireland', 49)
('Canada', 76)
('India', 2)
('South Africa', 5)
('Finland', 2)
('Switzerland', 36)
('Denmark', 15)
('Belgium', 8)
('Sweden', 13)
('Norway', 16)
('Luxembourg', 1)
('Italy', 15)
('Germany', 25)
('Moldova', 1)
('Spain', 12)
('United Arab Emirates', 6)
('Bahrain', 1)
('Turkey', 6)
('Kuwait', 1)
('Malta', 2)
('Hungary', 3)
('Austria', 7)
('Jersey', 1)
('Malaysia', 1)
('Iceland', 1)
('South Korea', 1)
('Brazil', 5)
('New Zealand', 6)
('Russia', 1)
('Monaco', 2)
('Hong Kong', 1)
('Thailand', 2)
```

- **Write a map-reduce program to analyze the given sales records over a period of time and generate data about the country's total sales and the frequency of the payment mode.**

```
import sys
if(len(sys.argv)!=3):
    print("Provide Input File and Output Directory")
    sys.exit(0)
from pyspark import SparkContext
sc =SparkContext()
f = sc.textFile(sys.argv[1])
temp=f.map(lambda x: (x.split(',')[3],1))
data=temp.countByKey()
dd=sc.parallelize(data.items())
dd.saveAsTextFile(sys.argv[2])
```



Activities
Terminal
Aug 27 14:43
ibmlab@ibmlab: ~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/sales

```
ibmlab@ibmlab:~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/sales$ cat otp/*
('Mastercard', 277)
('Visa', 522)
('Diners', 89)
('Amex', 110)
ibmlab@ibmlab:~/Downloads/1ms23scn06/spark-3.5.2-bin-hadoop3/sales$
```