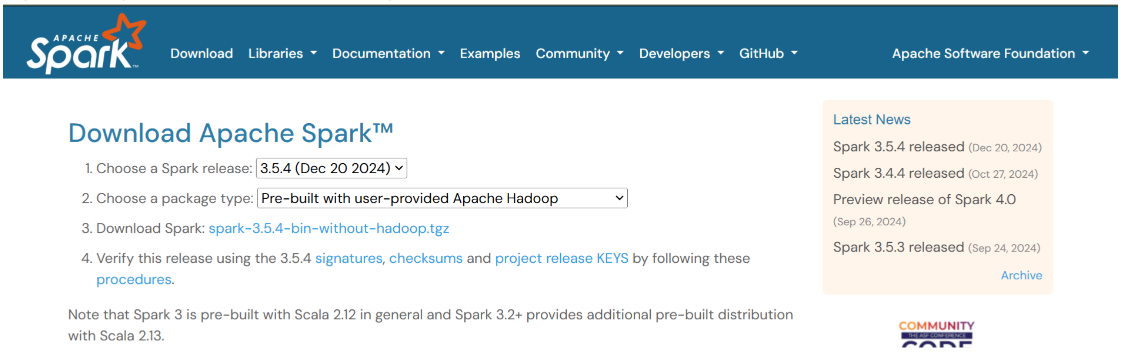
Big Data Aplicado

UD 6 - Apache Hadoop

Práctica 1 Spark

1. Configura un cluster como el explicado en clase. Tienes todas las instrucciones en la documentación del curso

   i. Primero debemos ir a la página oficial de Apache Spark y buscar la versión más acorde para nuestro cluster. Como ya tenemos Hadoop, usaremos la opción que no tiene hadoop.



Download Apache Spark™

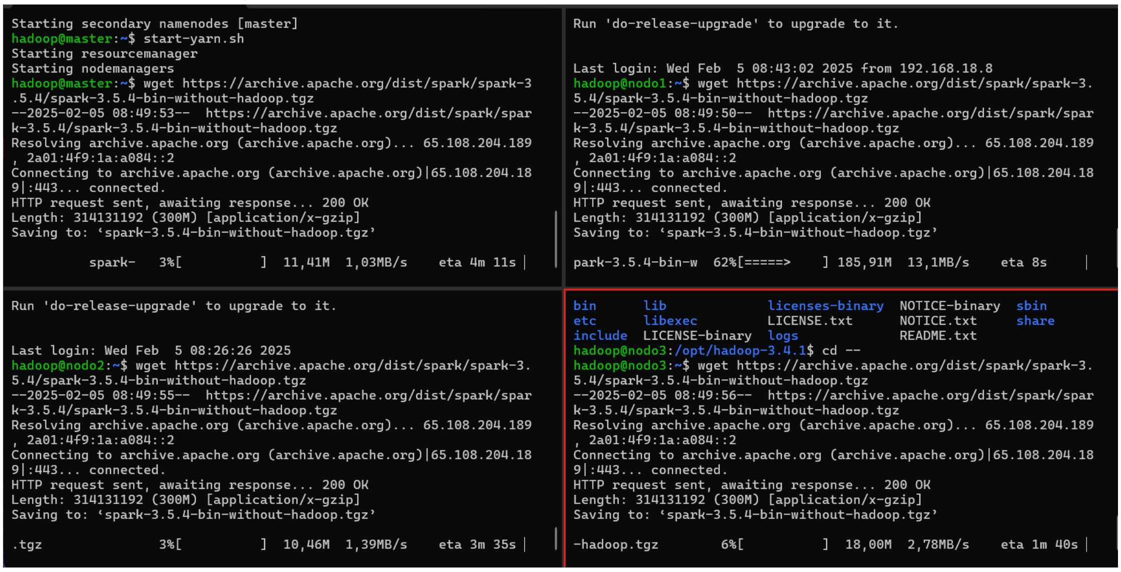
1. Choose a Spark release: 3.5.4 (Dec 20 2024)
2. Choose a package type: Pre-built with user-provided Apache Hadoop
3. Download Spark: spark-3.5.4-bin-without-hadoop.tgz
4. Verify this release using the 3.5.4 signatures, checksums and project release KEYS by following these procedures.

Note that Spark 3 is pre-built with Scala 2.12 in general and Spark 3.2+ provides additional pre-built distribution with Scala 2.13.

Latest News
Spark 3.5.4 released (Dec 20, 2024)
Spark 3.4.4 released (Oct 27, 2024)
Preview release of Spark 4.0 (Sep 26, 2024)
Spark 3.5.3 released (Sep 24, 2024)
Archive

```
wget https://archive.apache.org/dist/spark/spark-3.5.4/spark-3.5.4-bin-without-hadoop.tgz
```

Debemos descargar spark en los 4 nodos del cluster.



```
Starting secondary namenodes [master]
hadoop@master:~$ start-yarn.sh
Starting resourcemanager
Starting nodemanagers
hadoop@master:~$ wget https://archive.apache.org/dist/spark/spark-3
.5.4/spark-3.5.4-bin-without-hadoop.tgz
--2025-02-05 08:49:53--  https://archive.apache.org/dist/spark/spar
k-3.5.4/spark-3.5.4-bin-without-hadoop.tgz
Resolving archive.apache.org (archive.apache.org)... 65.108.204.189
, 2a01:4f9:1a:a084::2
Connecting to archive.apache.org (archive.apache.org)|65.108.204.18
9|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 314131192 (300M) [application/x-gzip]
Saving to: ‘spark-3.5.4-bin-without-hadoop.tgz’

          spark-   3%[          ]  11,41M  1,03MB/s    eta 4m 11s
```

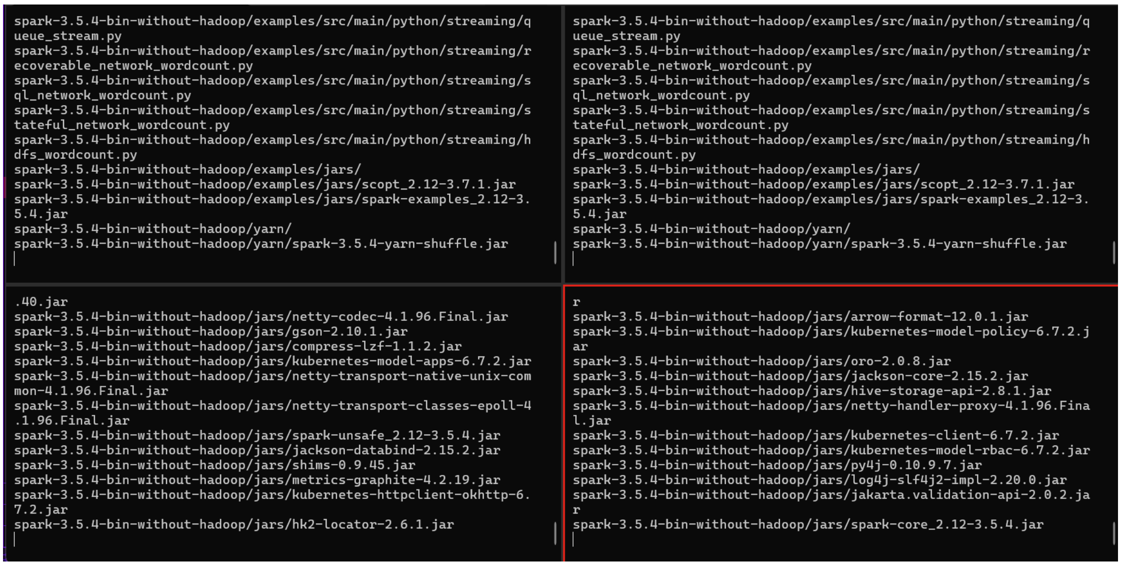
```
Run 'do-release-upgrade' to upgrade to it.


Last login: Wed Feb  5 08:43:02 2025 from 192.168.18.8
hadoop@nodo1:~$ wget https://archive.apache.org/dist/spark/spark-3.
5.4/spark-3.5.4-bin-without-hadoop.tgz
--2025-02-05 08:49:50--  https://archive.apache.org/dist/spark/spar
k-3.5.4/spark-3.5.4-bin-without-hadoop.tgz
Resolving archive.apache.org (archive.apache.org)... 65.108.204.189
, 2a01:4f9:1a:a084::2
Connecting to archive.apache.org (archive.apache.org)|65.108.204.18
9|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 314131192 (300M) [application/x-gzip]
Saving to: ‘spark-3.5.4-bin-without-hadoop.tgz’

park-3.5.4-bin-w  62%[=====>      ] 185,91M  13,1MB/s    eta 8s
```

```
Run 'do-release-upgrade' to upgrade to it.


Last login: Wed Feb  5 08:26:26 2025
hadoop@nodo2:~$ wget https://archive.apache.org/dist/spark/spark-3.
5.4/spark-3.5.4-bin-without-hadoop.tgz
--2025-02-05 08:49:55--  https://archive.apache.org/dist/spark/spar
k-3.5.4/spark-3.5.4-bin-without-hadoop.tgz
Resolving archive.apache.org (archive.apache.org)... 65.108.204.189
, 2a01:4f9:1a:a084::2
Connecting to archive.apache.org (archive.apache.org)|65.108.204.18
9|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 314131192 (300M) [application/x-gzip]
Saving to: ‘spark-3.5.4-bin-without-hadoop.tgz’

.tgz               3%[          ]  10,46M  1,39MB/s    eta 3m 35s
```

```
bin      lib              licenses-binary  NOTICE-binary  sbin
etc      libexec          LICENSE.txt      NOTICE.txt     share
include  LICENSE-binary   logs             README.txt
hadoop@nodo3:/opt/hadoop-3.4.1$ cd --
hadoop@nodo3:~$ wget https://archive.apache.org/dist/spark/spark-3.
5.4/spark-3.5.4-bin-without-hadoop.tgz
--2025-02-05 08:49:56--  https://archive.apache.org/dist/spark/spar
k-3.5.4/spark-3.5.4-bin-without-hadoop.tgz
Resolving archive.apache.org (archive.apache.org)... 65.108.204.189
, 2a01:4f9:1a:a084::2
Connecting to archive.apache.org (archive.apache.org)|65.108.204.18
9|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 314131192 (300M) [application/x-gzip]
Saving to: ‘spark-3.5.4-bin-without-hadoop.tgz’

-hadoop.tgz        6%[          ]  18,00M  2,78MB/s    eta 1m 40s
```

Luego descomprimimos y movemos a la carpeta `spark-3.5.4`.

```
tar -zxvf spark-3.5.4-bin-without-hadoop.tgz
mv spark-3.5.4-bin-without-hadoop /opt/hadoop-3.4.1/spark-3.5.4
```

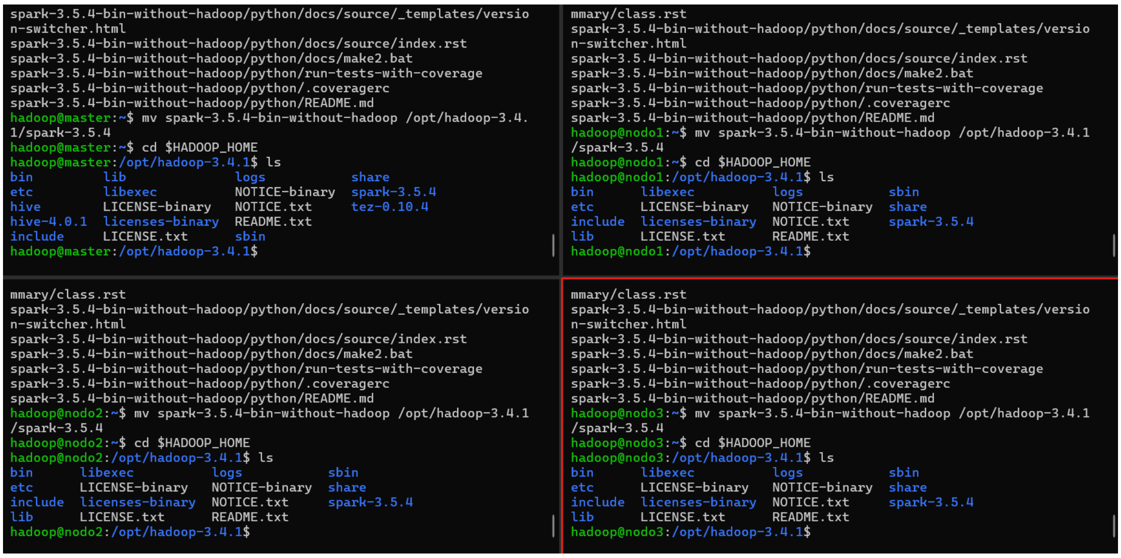


```
spark-3.5.4-bin-without-hadoop/examples/src/main/python/streaming/q
ueue_stream.py
spark-3.5.4-bin-without-hadoop/examples/src/main/python/streaming/r
ecoverable_network_wordcount.py
spark-3.5.4-bin-without-hadoop/examples/src/main/python/streaming/s
ql_network_wordcount.py
spark-3.5.4-bin-without-hadoop/examples/src/main/python/streaming/s
tateful_network_wordcount.py
spark-3.5.4-bin-without-hadoop/examples/src/main/python/streaming/h
dfs_wordcount.py
spark-3.5.4-bin-without-hadoop/examples/jars/
spark-3.5.4-bin-without-hadoop/examples/jars/scopt_2.12-3.7.1.jar
spark-3.5.4-bin-without-hadoop/examples/jars/spark-examples_2.12-3.
5.4.jar
spark-3.5.4-bin-without-hadoop/yarn/
spark-3.5.4-bin-without-hadoop/yarn/spark-3.5.4-yarn-shuffle.jar
```

```
spark-3.5.4-bin-without-hadoop/examples/src/main/python/streaming/q
ueue_stream.py
spark-3.5.4-bin-without-hadoop/examples/src/main/python/streaming/r
ecoverable_network_wordcount.py
spark-3.5.4-bin-without-hadoop/examples/src/main/python/streaming/s
ql_network_wordcount.py
spark-3.5.4-bin-without-hadoop/examples/src/main/python/streaming/s
tateful_network_wordcount.py
spark-3.5.4-bin-without-hadoop/examples/src/main/python/streaming/h
dfs_wordcount.py
spark-3.5.4-bin-without-hadoop/examples/jars/
spark-3.5.4-bin-without-hadoop/examples/jars/scopt_2.12-3.7.1.jar
spark-3.5.4-bin-without-hadoop/examples/jars/spark-examples_2.12-3.
5.4.jar
spark-3.5.4-bin-without-hadoop/yarn/
spark-3.5.4-bin-without-hadoop/yarn/spark-3.5.4-yarn-shuffle.jar
```

```
.40.jar
spark-3.5.4-bin-without-hadoop/jars/netty-codec-4.1.96.Final.jar
spark-3.5.4-bin-without-hadoop/jars/gson-2.10.1.jar
spark-3.5.4-bin-without-hadoop/jars/compress-lzf-1.1.2.jar
spark-3.5.4-bin-without-hadoop/jars/kubernetes-model-apps-6.7.2.jar
spark-3.5.4-bin-without-hadoop/jars/netty-transport-native-unix-com
mon-4.1.96.Final.jar
spark-3.5.4-bin-without-hadoop/jars/netty-transport-classes-epoll-4
.1.96.Final.jar
spark-3.5.4-bin-without-hadoop/jars/spark-unsafe_2.12-3.5.4.jar
spark-3.5.4-bin-without-hadoop/jars/jackson-databind-2.15.2.jar
spark-3.5.4-bin-without-hadoop/jars/shims-0.9.45.jar
spark-3.5.4-bin-without-hadoop/jars/metrics-graphite-4.2.19.jar
spark-3.5.4-bin-without-hadoop/jars/kubernetes-httpclient-okhttp-6.
7.2.jar
spark-3.5.4-bin-without-hadoop/jars/hk2-locator-2.6.1.jar
```

```
r
spark-3.5.4-bin-without-hadoop/jars/arrow-format-12.0.1.jar
spark-3.5.4-bin-without-hadoop/jars/kubernetes-model-policy-6.7.2.j
ar
spark-3.5.4-bin-without-hadoop/jars/oro-2.0.8.jar
spark-3.5.4-bin-without-hadoop/jars/jackson-core-2.15.2.jar
spark-3.5.4-bin-without-hadoop/jars/hive-storage-api-2.8.1.jar
spark-3.5.4-bin-without-hadoop/jars/netty-handler-proxy-4.1.96.Fina
l.jar
spark-3.5.4-bin-without-hadoop/jars/kubernetes-client-6.7.2.jar
spark-3.5.4-bin-without-hadoop/jars/kubernetes-model-rbac-6.7.2.jar
spark-3.5.4-bin-without-hadoop/jars/py4j-0.10.9.7.jar
spark-3.5.4-bin-without-hadoop/jars/log4j-slf4j2-impl-2.20.0.jar
spark-3.5.4-bin-without-hadoop/jars/jakarta.validation-api-2.0.2.ja
r
spark-3.5.4-bin-without-hadoop/jars/spark-core_2.12-3.5.4.jar
```



```
spark-3.5.4-bin-without-hadoop/python/docs/source/_templates/versio
n-switcher.html
spark-3.5.4-bin-without-hadoop/python/docs/source/index.rst
spark-3.5.4-bin-without-hadoop/python/docs/make2.bat
spark-3.5.4-bin-without-hadoop/python/run-tests-with-coverage
spark-3.5.4-bin-without-hadoop/python/.coveragerc
spark-3.5.4-bin-without-hadoop/python/README.md
hadoop@master:~$ mv spark-3.5.4-bin-without-hadoop /opt/hadoop-3.4.
1/spark-3.5.4
hadoop@master:~$ cd $HADOOP_HOME
hadoop@master:/opt/hadoop-3.4.1$ ls
bin         lib              logs           share
etc         libexec          NOTICE-binary  spark-3.5.4
hive        LICENSE-binary   NOTICE.txt     tez-0.10.4
hive-4.0.1  licenses-binary  README.txt
include     LICENSE.txt      sbin
hadoop@master:/opt/hadoop-3.4.1$
```

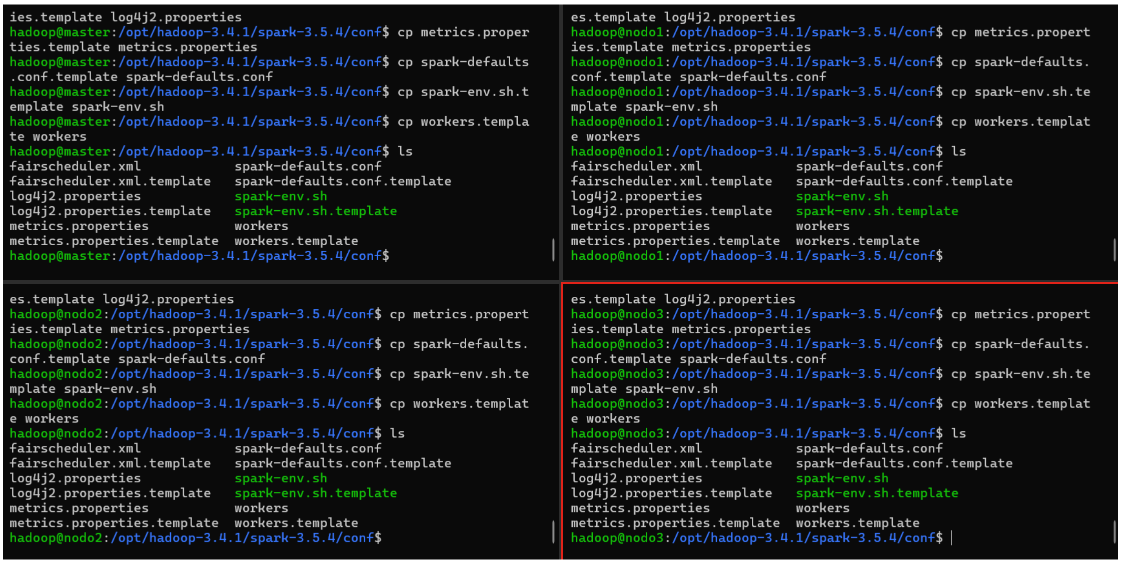
```
mmary/class.rst
spark-3.5.4-bin-without-hadoop/python/docs/source/_templates/versio
n-switcher.html
spark-3.5.4-bin-without-hadoop/python/docs/source/index.rst
spark-3.5.4-bin-without-hadoop/python/docs/make2.bat
spark-3.5.4-bin-without-hadoop/python/run-tests-with-coverage
spark-3.5.4-bin-without-hadoop/python/.coveragerc
spark-3.5.4-bin-without-hadoop/python/README.md
hadoop@nodo1:~$ mv spark-3.5.4-bin-without-hadoop /opt/hadoop-3.4.1
/spark-3.5.4
hadoop@nodo1:~$ cd $HADOOP_HOME
hadoop@nodo1:/opt/hadoop-3.4.1$ ls
bin      libexec          logs           sbin
etc      LICENSE-binary   NOTICE-binary  share
include  licenses-binary  NOTICE.txt     spark-3.5.4
lib      LICENSE.txt      README.txt
hadoop@nodo1:/opt/hadoop-3.4.1$
```

```
mmary/class.rst
spark-3.5.4-bin-without-hadoop/python/docs/source/_templates/versio
n-switcher.html
spark-3.5.4-bin-without-hadoop/python/docs/source/index.rst
spark-3.5.4-bin-without-hadoop/python/docs/make2.bat
spark-3.5.4-bin-without-hadoop/python/run-tests-with-coverage
spark-3.5.4-bin-without-hadoop/python/.coveragerc
spark-3.5.4-bin-without-hadoop/python/README.md
hadoop@nodo2:~$ mv spark-3.5.4-bin-without-hadoop /opt/hadoop-3.4.1
/spark-3.5.4
hadoop@nodo2:~$ cd $HADOOP_HOME
hadoop@nodo2:/opt/hadoop-3.4.1$ ls
bin      libexec          logs           sbin
etc      LICENSE-binary   NOTICE-binary  share
include  licenses-binary  NOTICE.txt     spark-3.5.4
lib      LICENSE.txt      README.txt
hadoop@nodo2:/opt/hadoop-3.4.1$
```

```
mmary/class.rst
spark-3.5.4-bin-without-hadoop/python/docs/source/_templates/versio
n-switcher.html
spark-3.5.4-bin-without-hadoop/python/docs/source/index.rst
spark-3.5.4-bin-without-hadoop/python/docs/make2.bat
spark-3.5.4-bin-without-hadoop/python/run-tests-with-coverage
spark-3.5.4-bin-without-hadoop/python/.coveragerc
spark-3.5.4-bin-without-hadoop/python/README.md
hadoop@nodo3:~$ mv spark-3.5.4-bin-without-hadoop /opt/hadoop-3.4.1
/spark-3.5.4
hadoop@nodo3:~$ cd $HADOOP_HOME
hadoop@nodo3:/opt/hadoop-3.4.1$ ls
bin      libexec          logs           sbin
etc      LICENSE-binary   NOTICE-binary  share
include  licenses-binary  NOTICE.txt     spark-3.5.4
lib      LICENSE.txt      README.txt
hadoop@nodo3:/opt/hadoop-3.4.1$
```

ii. Hacemos las templates dentro de la carpeta `conf` de Spark.

```
cp fairscheduler.xml.template fairscheduler.xml
cp log4j2.properties.template log4j2.properties
cp metrics.properties.template metrics.properties
cp spark-defaults.conf.template spark-defaults.conf
cp spark-env.sh.template spark-env.sh
cp workers.template workers
```



```
ies.template log4j2.properties
hadoop@master:/opt/hadoop-3.4.1/spark-3.5.4/conf$ cp metrics.proper
ties.template metrics.properties
hadoop@master:/opt/hadoop-3.4.1/spark-3.5.4/conf$ cp spark-defaults
.conf.template spark-defaults.conf
hadoop@master:/opt/hadoop-3.4.1/spark-3.5.4/conf$ cp spark-env.sh.t
emplate spark-env.sh
hadoop@master:/opt/hadoop-3.4.1/spark-3.5.4/conf$ cp workers.templa
te workers
hadoop@master:/opt/hadoop-3.4.1/spark-3.5.4/conf$ ls
fairscheduler.xml             spark-defaults.conf
fairscheduler.xml.template    spark-defaults.conf.template
log4j2.properties             spark-env.sh
log4j2.properties.template    spark-env.sh.template
metrics.properties            workers
metrics.properties.template   workers.template
hadoop@master:/opt/hadoop-3.4.1/spark-3.5.4/conf$
```

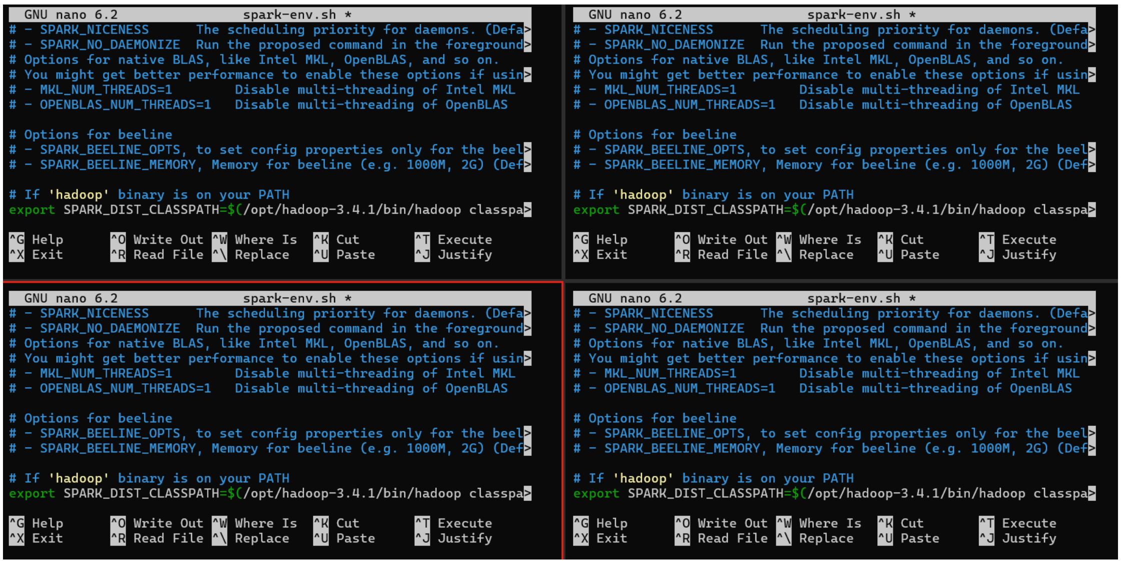
```
es.template log4j2.properties
hadoop@nodo1:/opt/hadoop-3.4.1/spark-3.5.4/conf$ cp metrics.propert
ies.template metrics.properties
hadoop@nodo1:/opt/hadoop-3.4.1/spark-3.5.4/conf$ cp spark-defaults.
conf.template spark-defaults.conf
hadoop@nodo1:/opt/hadoop-3.4.1/spark-3.5.4/conf$ cp spark-env.sh.te
mplate spark-env.sh
hadoop@nodo1:/opt/hadoop-3.4.1/spark-3.5.4/conf$ cp workers.templat
e workers
hadoop@nodo1:/opt/hadoop-3.4.1/spark-3.5.4/conf$ ls
fairscheduler.xml             spark-defaults.conf
fairscheduler.xml.template    spark-defaults.conf.template
log4j2.properties             spark-env.sh
log4j2.properties.template    spark-env.sh.template
metrics.properties            workers
metrics.properties.template   workers.template
hadoop@nodo1:/opt/hadoop-3.4.1/spark-3.5.4/conf$
```

```
es.template log4j2.properties
hadoop@nodo2:/opt/hadoop-3.4.1/spark-3.5.4/conf$ cp metrics.propert
ies.template metrics.properties
hadoop@nodo2:/opt/hadoop-3.4.1/spark-3.5.4/conf$ cp spark-defaults.
conf.template spark-defaults.conf
hadoop@nodo2:/opt/hadoop-3.4.1/spark-3.5.4/conf$ cp spark-env.sh.te
mplate spark-env.sh
hadoop@nodo2:/opt/hadoop-3.4.1/spark-3.5.4/conf$ cp workers.templat
e workers
hadoop@nodo2:/opt/hadoop-3.4.1/spark-3.5.4/conf$ ls
fairscheduler.xml             spark-defaults.conf
fairscheduler.xml.template    spark-defaults.conf.template
log4j2.properties             spark-env.sh
log4j2.properties.template    spark-env.sh.template
metrics.properties            workers
metrics.properties.template   workers.template
hadoop@nodo2:/opt/hadoop-3.4.1/spark-3.5.4/conf$
```

```
es.template log4j2.properties
hadoop@nodo3:/opt/hadoop-3.4.1/spark-3.5.4/conf$ cp metrics.propert
ies.template metrics.properties
hadoop@nodo3:/opt/hadoop-3.4.1/spark-3.5.4/conf$ cp spark-defaults.
conf.template spark-defaults.conf
hadoop@nodo3:/opt/hadoop-3.4.1/spark-3.5.4/conf$ cp spark-env.sh.te
mplate spark-env.sh
hadoop@nodo3:/opt/hadoop-3.4.1/spark-3.5.4/conf$ cp workers.templat
e workers
hadoop@nodo3:/opt/hadoop-3.4.1/spark-3.5.4/conf$ ls
fairscheduler.xml             spark-defaults.conf
fairscheduler.xml.template    spark-defaults.conf.template
log4j2.properties             spark-env.sh
log4j2.properties.template    spark-env.sh.template
metrics.properties            workers
metrics.properties.template   workers.template
hadoop@nodo3:/opt/hadoop-3.4.1/spark-3.5.4/conf$
```

Ahora en `spark-env.sh` debemos incluir los paquetes `jar` de Hadoop para añadir Spark a Hadoop. Lo hacemos con esta línea.

```
# If 'hadoop' binary is on your PATH
export SPARK_DIST_CLASSPATH=$(/opt/hadoop-3.4.1/bin/hadoop classpat
```



```
  GNU nano 6.2                    spark-env.sh *
# - SPARK_NICENESS      The scheduling priority for daemons. (Defa>
# - SPARK_NO_DAEMONIZE  Run the proposed command in the foreground>
# Options for native BLAS, like Intel MKL, OpenBLAS, and so on.
# You might get better performance to enable these options if usin>
# - MKL_NUM_THREADS=1        Disable multi-threading of Intel MKL
# - OPENBLAS_NUM_THREADS=1   Disable multi-threading of OpenBLAS

# Options for beeline
# - SPARK_BEELINE_OPTS, to set config properties only for the beel>
# - SPARK_BEELINE_MEMORY, Memory for beeline (e.g. 1000M, 2G) (Def>

# If 'hadoop' binary is on your PATH
export SPARK_DIST_CLASSPATH=$(/opt/hadoop-3.4.1/bin/hadoop classpa>

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify
```

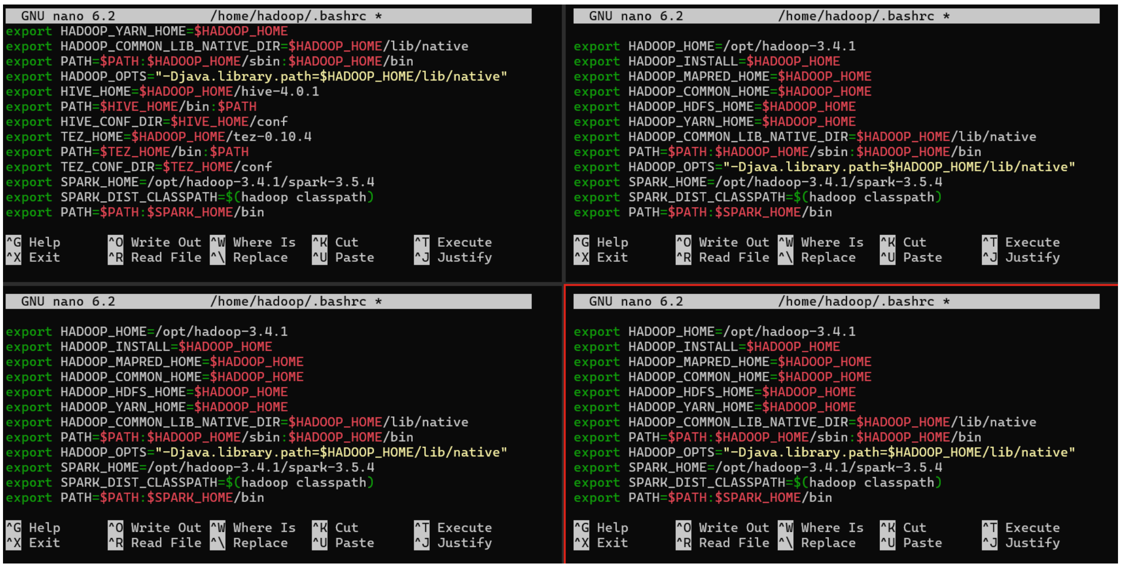
```
  GNU nano 6.2                    spark-env.sh *
# - SPARK_NICENESS      The scheduling priority for daemons. (Defa>
# - SPARK_NO_DAEMONIZE  Run the proposed command in the foreground>
# Options for native BLAS, like Intel MKL, OpenBLAS, and so on.
# You might get better performance to enable these options if usin>
# - MKL_NUM_THREADS=1        Disable multi-threading of Intel MKL
# - OPENBLAS_NUM_THREADS=1   Disable multi-threading of OpenBLAS

# Options for beeline
# - SPARK_BEELINE_OPTS, to set config properties only for the beel>
# - SPARK_BEELINE_MEMORY, Memory for beeline (e.g. 1000M, 2G) (Def>

# If 'hadoop' binary is on your PATH
export SPARK_DIST_CLASSPATH=$(/opt/hadoop-3.4.1/bin/hadoop classpa>

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify
```

```
  GNU nano 6.2                    spark-env.sh *
# - SPARK_NICENESS      The scheduling priority for daemons. (Defa>
# - SPARK_NO_DAEMONIZE  Run the proposed command in the foreground>
# Options for native BLAS, like Intel MKL, OpenBLAS, and so on.
# You might get better performance to enable these options if usin>
# - MKL_NUM_THREADS=1        Disable multi-threading of Intel MKL
# - OPENBLAS_NUM_THREADS=1   Disable multi-threading of OpenBLAS

# Options for beeline
# - SPARK_BEELINE_OPTS, to set config properties only for the beel>
# - SPARK_BEELINE_MEMORY, Memory for beeline (e.g. 1000M, 2G) (Def>

# If 'hadoop' binary is on your PATH
export SPARK_DIST_CLASSPATH=$(/opt/hadoop-3.4.1/bin/hadoop classpa>

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify
```

```
  GNU nano 6.2                    spark-env.sh *
# - SPARK_NICENESS      The scheduling priority for daemons. (Defa>
# - SPARK_NO_DAEMONIZE  Run the proposed command in the foreground>
# Options for native BLAS, like Intel MKL, OpenBLAS, and so on.
# You might get better performance to enable these options if usin>
# - MKL_NUM_THREADS=1        Disable multi-threading of Intel MKL
# - OPENBLAS_NUM_THREADS=1   Disable multi-threading of OpenBLAS

# Options for beeline
# - SPARK_BEELINE_OPTS, to set config properties only for the beel>
# - SPARK_BEELINE_MEMORY, Memory for beeline (e.g. 1000M, 2G) (Def>

# If 'hadoop' binary is on your PATH
export SPARK_DIST_CLASSPATH=$(/opt/hadoop-3.4.1/bin/hadoop classpa>

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify
```

El último paso en común en todos los nodos es en el archivo `.bashrc` incluir Spark. Debemos incluir el directorio `bin` porque `sbin` tiene comandos similares a los de Hadoop y podría haber conflicto.

```
export SPARK_HOME=/opt/hadoop-3.4.1/spark-3.5.4
export SPARK_DIST_CLASSPATH=$(hadoop classpath)
export PATH=$PATH:$SPARK_HOME/bin
```



```
  GNU nano 6.2                  /home/hadoop/.bashrc *
export HADOOP_YARN_HOME=$HADOOP_HOME
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
export PATH=$PATH:$HADOOP_HOME/sbin:$HADOOP_HOME/bin
export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib/native"
export HIVE_HOME=$HADOOP_HOME/hive-4.0.1
export PATH=$HIVE_HOME/bin:$PATH
export HIVE_CONF_DIR=$HIVE_HOME/conf
export TEZ_HOME=$HADOOP_HOME/tez-0.10.4
export PATH=$TEZ_HOME/bin:$PATH
export TEZ_CONF_DIR=$TEZ_HOME/conf
export SPARK_HOME=/opt/hadoop-3.4.1/spark-3.5.4
export SPARK_DIST_CLASSPATH=$(hadoop classpath)
export PATH=$PATH:$SPARK_HOME/bin

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify
```

```
  GNU nano 6.2                  /home/hadoop/.bashrc *

export HADOOP_HOME=/opt/hadoop-3.4.1
export HADOOP_INSTALL=$HADOOP_HOME
export HADOOP_MAPRED_HOME=$HADOOP_HOME
export HADOOP_COMMON_HOME=$HADOOP_HOME
export HADOOP_HDFS_HOME=$HADOOP_HOME
export HADOOP_YARN_HOME=$HADOOP_HOME
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
export PATH=$PATH:$HADOOP_HOME/sbin:$HADOOP_HOME/bin
export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib/native"
export SPARK_HOME=/opt/hadoop-3.4.1/spark-3.5.4
export SPARK_DIST_CLASSPATH=$(hadoop classpath)
export PATH=$PATH:$SPARK_HOME/bin

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify
```

```
  GNU nano 6.2                  /home/hadoop/.bashrc *

export HADOOP_HOME=/opt/hadoop-3.4.1
export HADOOP_INSTALL=$HADOOP_HOME
export HADOOP_MAPRED_HOME=$HADOOP_HOME
export HADOOP_COMMON_HOME=$HADOOP_HOME
export HADOOP_HDFS_HOME=$HADOOP_HOME
export HADOOP_YARN_HOME=$HADOOP_HOME
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
export PATH=$PATH:$HADOOP_HOME/sbin:$HADOOP_HOME/bin
export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib/native"
export SPARK_HOME=/opt/hadoop-3.4.1/spark-3.5.4
export SPARK_DIST_CLASSPATH=$(hadoop classpath)
export PATH=$PATH:$SPARK_HOME/bin

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify
```

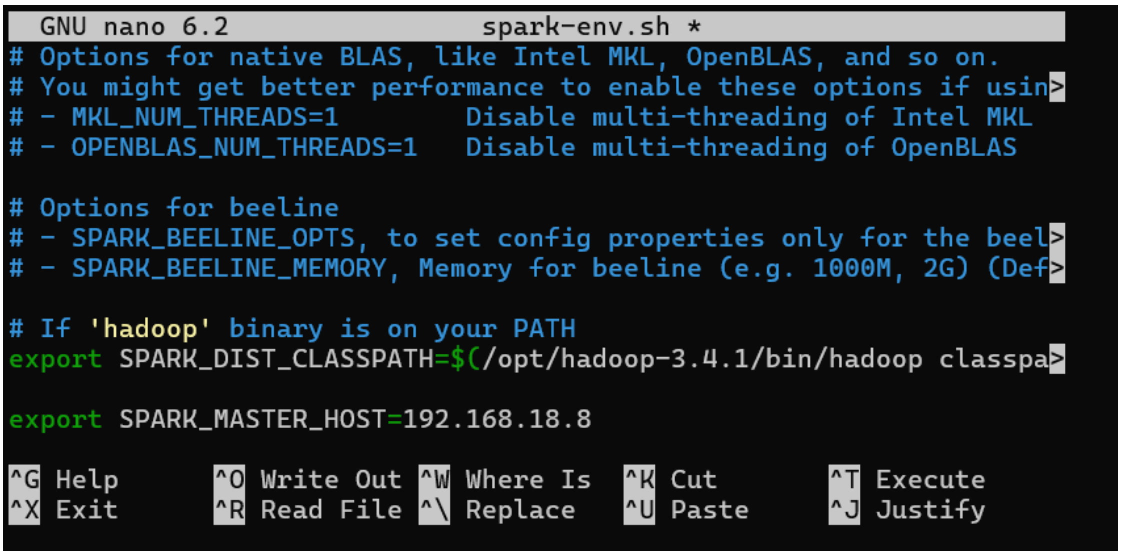
```
  GNU nano 6.2                  /home/hadoop/.bashrc *

export HADOOP_HOME=/opt/hadoop-3.4.1
export HADOOP_INSTALL=$HADOOP_HOME
export HADOOP_MAPRED_HOME=$HADOOP_HOME
export HADOOP_COMMON_HOME=$HADOOP_HOME
export HADOOP_HDFS_HOME=$HADOOP_HOME
export HADOOP_YARN_HOME=$HADOOP_HOME
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
export PATH=$PATH:$HADOOP_HOME/sbin:$HADOOP_HOME/bin
export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib/native"
export SPARK_HOME=/opt/hadoop-3.4.1/spark-3.5.4
export SPARK_DIST_CLASSPATH=$(hadoop classpath)
export PATH=$PATH:$SPARK_HOME/bin

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify
```

iii. Ahora Spark se podría iniciar, pero para que funcione como clúster debemos indicar que nodo es el master y cuáles son workers. Para ello indicaremos en el nodo donde se lance Spark master la IP de nuestro master.

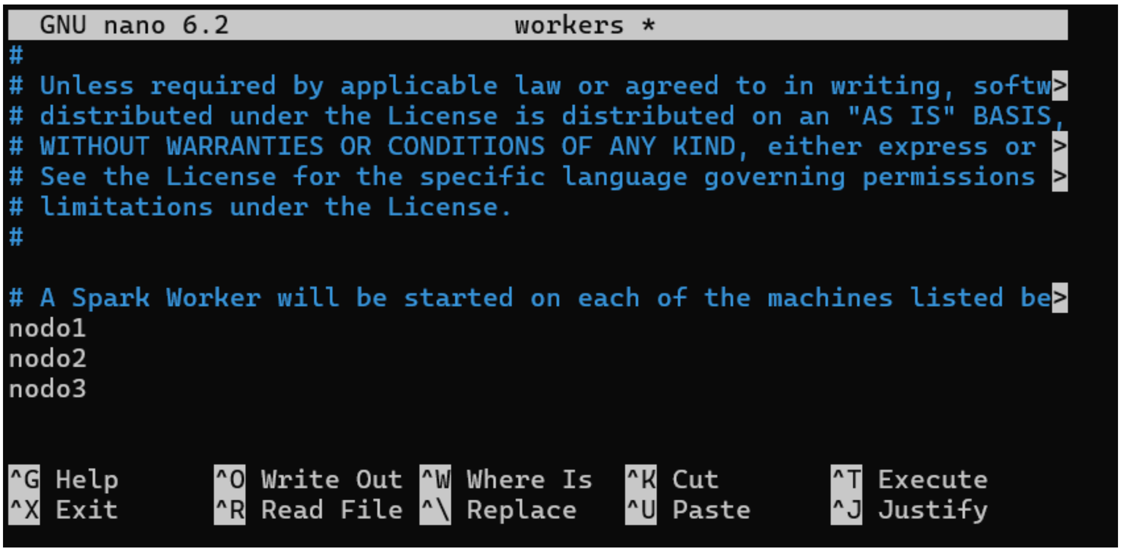
```
export SPARK_MASTER_HOST=192.168.18.8
```

Y en `/conf/workers` indicaremos los nodos workers. Debemos eliminar el `localhost`.

```
nodo1
nodo2
nodo3
```



```
  GNU nano 6.2                    spark-env.sh *
# Options for native BLAS, like Intel MKL, OpenBLAS, and so on.
# You might get better performance to enable these options if usin>
# - MKL_NUM_THREADS=1        Disable multi-threading of Intel MKL
# - OPENBLAS_NUM_THREADS=1   Disable multi-threading of OpenBLAS

# Options for beeline
# - SPARK_BEELINE_OPTS, to set config properties only for the beel>
# - SPARK_BEELINE_MEMORY, Memory for beeline (e.g. 1000M, 2G) (Def>

# If 'hadoop' binary is on your PATH
export SPARK_DIST_CLASSPATH=$(/opt/hadoop-3.4.1/bin/hadoop classpa>

export SPARK_MASTER_HOST=192.168.18.8

^G Help       ^O Write Out ^W Where Is  ^K Cut       ^T Execute
^X Exit       ^R Read File ^\ Replace   ^U Paste     ^J Justify
```



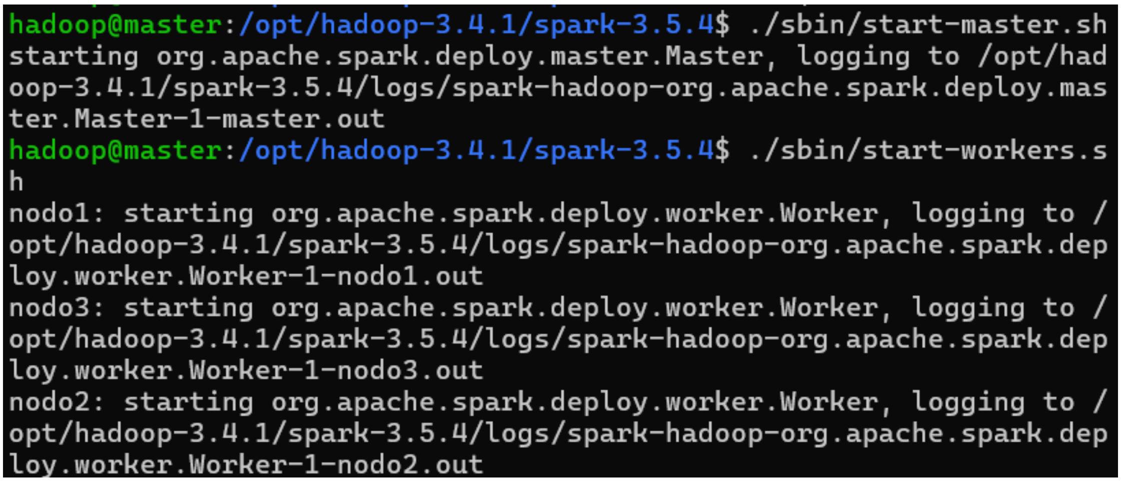
```
  GNU nano 6.2                      workers *
#
# Unless required by applicable law or agreed to in writing, softw>
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or >
# See the License for the specific language governing permissions >
# limitations under the License.
#

# A Spark Worker will be started on each of the machines listed be>
nodo1
nodo2
nodo3


^G Help       ^O Write Out ^W Where Is  ^K Cut       ^T Execute
^X Exit       ^R Read File ^\ Replace   ^U Paste     ^J Justify
```

Ahora iniciamos el master con este comando:

```
./sbin/start-master.sh
```

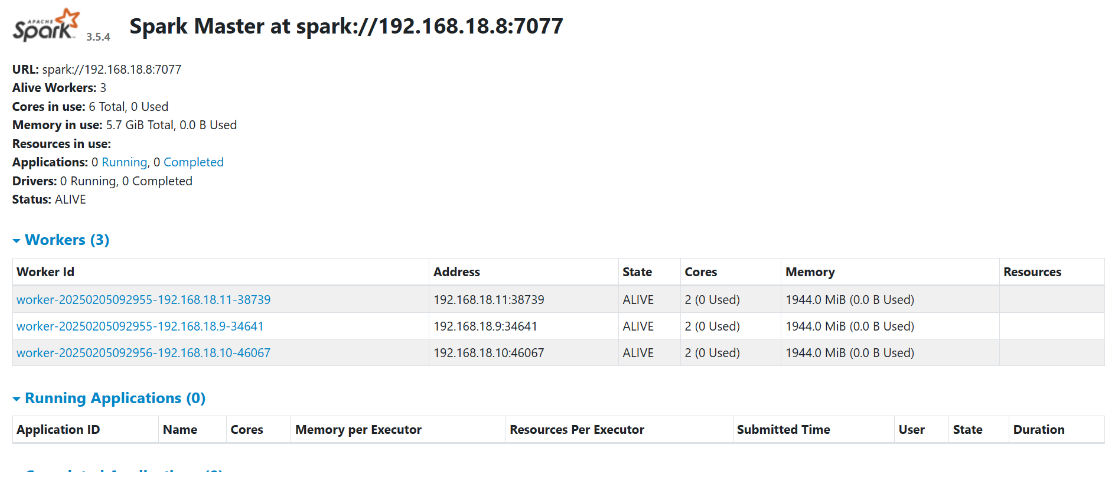
Y los workers con este otro:

```
./sbin/start-workers.sh
```



```
hadoop@master:/opt/hadoop-3.4.1/spark-3.5.4$ ./sbin/start-master.sh
starting org.apache.spark.deploy.master.Master, logging to /opt/had
oop-3.4.1/spark-3.5.4/logs/spark-hadoop-org.apache.spark.deploy.mas
ter.Master-1-master.out
hadoop@master:/opt/hadoop-3.4.1/spark-3.5.4$ ./sbin/start-workers.s
h
nodo1: starting org.apache.spark.deploy.worker.Worker, logging to /
opt/hadoop-3.4.1/spark-3.5.4/logs/spark-hadoop-org.apache.spark.dep
loy.worker.Worker-1-nodo1.out
nodo3: starting org.apache.spark.deploy.worker.Worker, logging to /
opt/hadoop-3.4.1/spark-3.5.4/logs/spark-hadoop-org.apache.spark.dep
loy.worker.Worker-1-nodo3.out
nodo2: starting org.apache.spark.deploy.worker.Worker, logging to /
opt/hadoop-3.4.1/spark-3.5.4/logs/spark-hadoop-org.apache.spark.dep
loy.worker.Worker-1-nodo2.out
```

Si todo ha ido bien, en la url `192.168.165.8:8080` deberíamos tener la UI de Spark.



Spark 3.5.4 **Spark Master at spark://192.168.18.8:7077**

URL: spark://192.168.18.8:7077
Alive Workers: 3
Cores in use: 6 Total, 0 Used
Memory in use: 5.7 GiB Total, 0.0 B Used
Resources in use:
Applications: 0 Running, 0 Completed
Drivers: 0 Running, 0 Completed
Status: ALIVE

Workers (3)

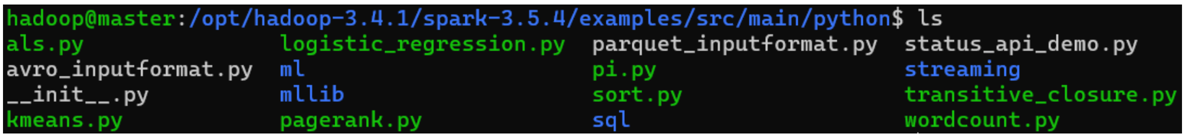
Worker Id	Address	State	Cores	Memory	Resources
worker-20250205092955-192.168.18.11-38739	192.168.18.11:38739	ALIVE	2 (0 Used)	1944.0 MiB (0.0 B Used)	
worker-20250205092955-192.168.18.9-34641	192.168.18.9:34641	ALIVE	2 (0 Used)	1944.0 MiB (0.0 B Used)	
worker-20250205092956-192.168.18.10-46067	192.168.18.10:46067	ALIVE	2 (0 Used)	1944.0 MiB (0.0 B Used)	

Running Applications (0)

Application ID	Name	Cores	Memory per Executor	Resources Per Executor	Submitted Time	User	State	Duration

Podemos ver que en la página se indican todos los workers y su estado actual. También hay apartados para las aplicaciones que se están ejecutando y las que se han completado.

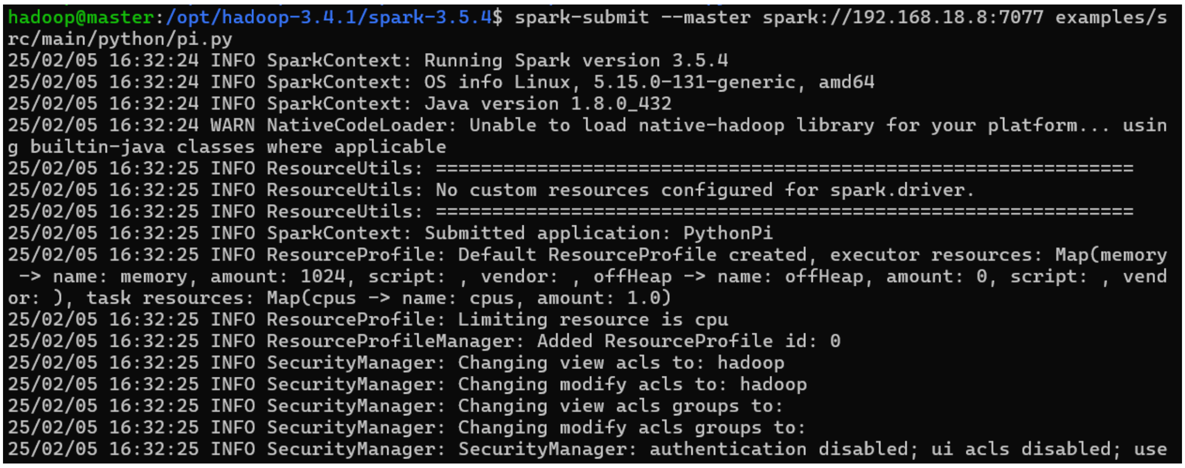
2. Observa el directorio de ejemplos de aplicaciones que ya tenemos al instalar Spark. Se encuentra en el directorio `$SPARK_HOME/examples/src/main/python`



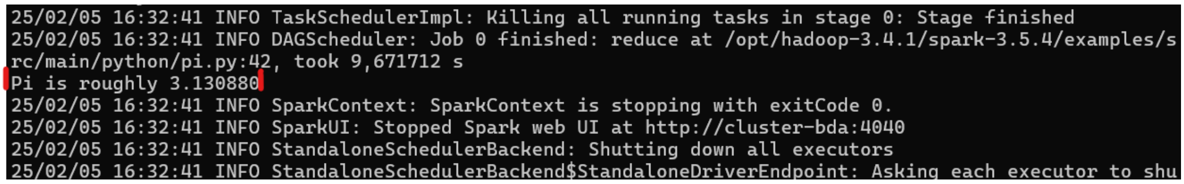
```
hadoop@master:/opt/hadoop-3.4.1/spark-3.5.4/examples/src/main/python$ ls
als.py             logistic_regression.py  parquet_inputformat.py  status_api_demo.py
avro_inputformat.py ml                     pi.py                   streaming
__init__.py        mllib                   sort.py                 transitive_closure.py
kmeans.py          pagerank.py             sql                     wordcount.py
```

3. Elige uno para ejecutarlo, que no sea **wordcount** De los muchos ejemplos que tenemos, podemos ejecutar `pi.py` que mostrará el número PI aproximadamente. para ello haremos el siguiente comando:

```
spark-submit --master spark://192.168.18.8:7077 examples/src/main/python/pi.py
```



```
hadoop@master:/opt/hadoop-3.4.1/spark-3.5.4$ spark-submit --master spark://192.168.18.8:7077 examples/s
rc/main/python/pi.py
25/02/05 16:32:24 INFO SparkContext: Running Spark version 3.5.4
25/02/05 16:32:24 INFO SparkContext: OS info Linux, 5.15.0-131-generic, amd64
25/02/05 16:32:24 INFO SparkContext: Java version 1.8.0_432
25/02/05 16:32:24 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... usin
g builtin-java classes where applicable
25/02/05 16:32:25 INFO ResourceUtils: ==============================================================
25/02/05 16:32:25 INFO ResourceUtils: No custom resources configured for spark.driver.
25/02/05 16:32:25 INFO ResourceUtils: ==============================================================
25/02/05 16:32:25 INFO SparkContext: Submitted application: PythonPi
25/02/05 16:32:25 INFO ResourceProfile: Default ResourceProfile created, executor resources: Map(memory
 -> name: memory, amount: 1024, script: , vendor: , offHeap -> name: offHeap, amount: 0, script: , vend
or: ), task resources: Map(cpus -> name: cpus, amount: 1.0)
25/02/05 16:32:25 INFO ResourceProfile: Limiting resource is cpu
25/02/05 16:32:25 INFO ResourceProfileManager: Added ResourceProfile id: 0
25/02/05 16:32:25 INFO SecurityManager: Changing view acls to: hadoop
25/02/05 16:32:25 INFO SecurityManager: Changing modify acls to: hadoop
25/02/05 16:32:25 INFO SecurityManager: Changing view acls groups to:
25/02/05 16:32:25 INFO SecurityManager: Changing modify acls groups to:
25/02/05 16:32:25 INFO SecurityManager: SecurityManager: authentication disabled; ui acls disabled; use
```

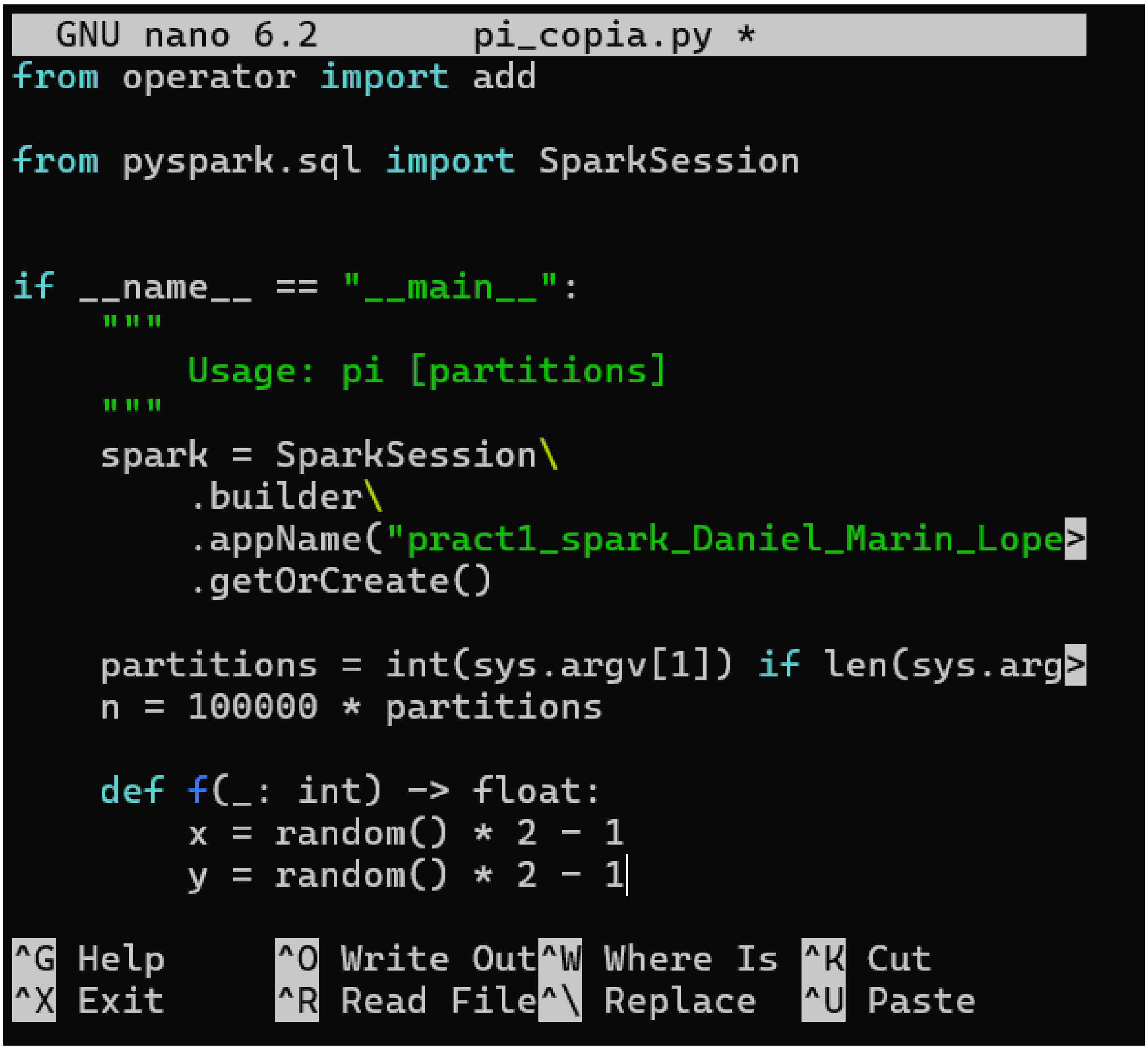


```
25/02/05 16:32:41 INFO TaskSchedulerImpl: Killing all running tasks in stage 0: Stage finished
25/02/05 16:32:41 INFO DAGScheduler: Job 0 finished: reduce at /opt/hadoop-3.4.1/spark-3.5.4/examples/s
rc/main/python/pi.py:42, took 9,671712 s
Pi is roughly 3.130880
25/02/05 16:32:41 INFO SparkContext: SparkContext is stopping with exitCode 0.
25/02/05 16:32:41 INFO SparkUI: Stopped Spark web UI at http://cluster-bda:4040
25/02/05 16:32:41 INFO StandaloneSchedulerBackend: Shutting down all executors
25/02/05 16:32:41 INFO StandaloneSchedulerBackend$StandaloneDriverEndpoint: Asking each executor to shu
```

Aquí vemos que el resultado arrojado es 3.130880.

4. Haz una copia y modifica el código fuente para que el nombre de la aplicación sea " `pract1_spark_Nombre_Apellido1_Apellido2` " (en mi caso, por ejemplo, debería añadir " `pract1_spark_Jaime_Rabasco_Ronda` ").

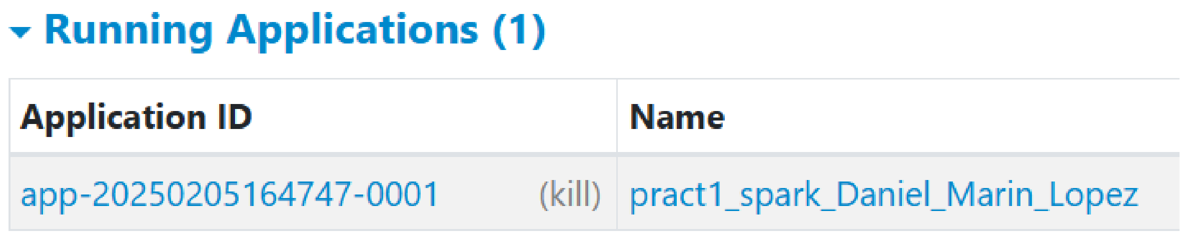
Modificamos el archivo y ponemos en appName nuestro nombre:



```
  GNU nano 6.2          pi_copia.py *
from operator import add

from pyspark.sql import SparkSession


if __name__ == "__main__":
    """
        Usage: pi [partitions]
    """
    spark = SparkSession\
        .builder\
        .appName("pract1_spark_Daniel_Marin_Lope>
        .getOrCreate()

    partitions = int(sys.argv[1]) if len(sys.arg>
    n = 100000 * partitions

    def f(_: int) -> float:
        x = random() * 2 - 1
        y = random() * 2 - 1

^G Help       ^O Write Out^W Where Is  ^K Cut
^X Exit       ^R Read File^\ Replace   ^U Paste
```

Y volvemos a ejecutar el comando con la copia:

```
spark-submit --master spark://192.168.18.8:7077
examples/src/main/python/pi_copia.py
```



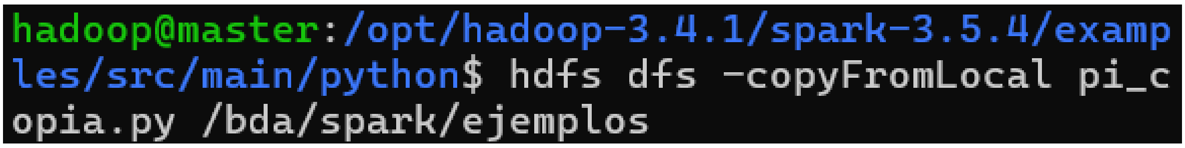
▾ Running Applications (1)

Application ID	Name
app-20250205164747-0001 (kill)	pract1_spark_Daniel_Marin_Lopez

Aquí en la UI se puede ver que en vez de tener su nombre aprace el que hemos puesto identificando la copia en cuestión.

5. Copia la aplicación elegida en hdfs Hacemos una copia del archivo con `copyFromLocal` .

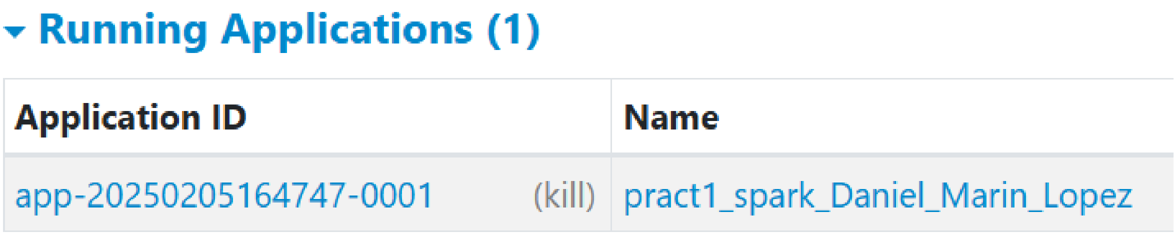
```
hdfs dfs -copyFromLocal pi_copia.py /bda/spark/ejemplos
```



```
hadoop@master:/opt/hadoop-3.4.1/spark-3.5.4/examp
les/src/main/python$ hdfs dfs -copyFromLocal pi_c
opia.py /bda/spark/ejemplos
```

6. Ejecuta el código. Recuerda añadir los parámetros que necesite, si los necesita (pueden estar en hdfs, local o internet)

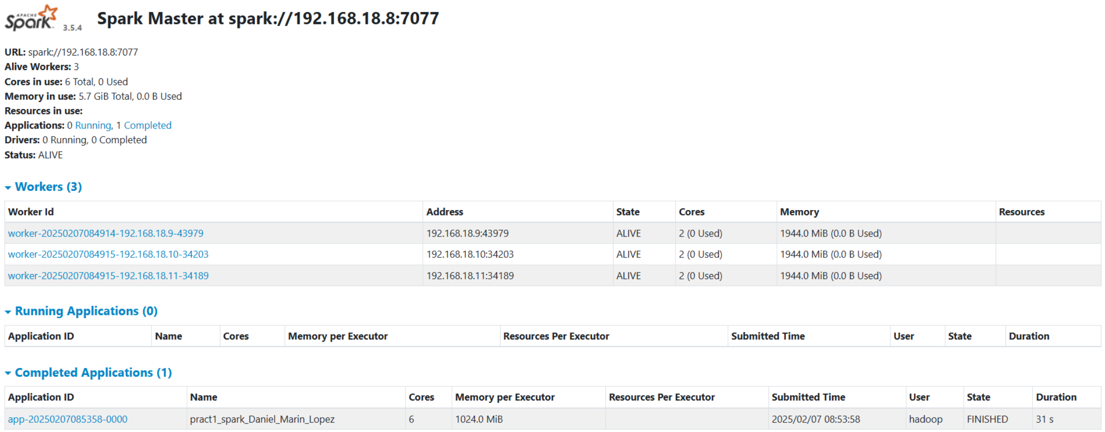
```
 spark-submit --master spark://192.168.18.8:7077
examples/src/main/python/pi_copia.py
```



Running Applications (1)

Application ID	Name
app-20250205164747-0001 (kill)	pract1_spark_Daniel_Marin_Lopez

7. Haz todas las capturas de SparkUI donde se vea claramente
   i. Master
   ii. Workers
   iii. Ejecución de la aplicación con tu nombre y apellidos La UI se ve de la siguiente forma:



Spark Master at spark://192.168.18.8:7077

URL: spark://192.168.18.8:7077
Alive Workers: 3
Cores in use: 6 Total, 0 Used
Memory in use: 5.7 GiB Total, 0.0 B Used
Resources in use:
Applications: 0 Running, 1 Completed
Drivers: 0 Running, 0 Completed
Status: ALIVE

Workers (3)

Worker Id	Address	State	Cores	Memory	Resources
worker-20250207084914-192.168.18.9-43979	192.168.18.9:43979	ALIVE	2 (0 Used)	1944.0 MiB (0.0 B Used)	
worker-20250207084915-192.168.18.10-34203	192.168.18.10:34203	ALIVE	2 (0 Used)	1944.0 MiB (0.0 B Used)	
worker-20250207084915-192.168.18.11-34189	192.168.18.11:34189	ALIVE	2 (0 Used)	1944.0 MiB (0.0 B Used)	

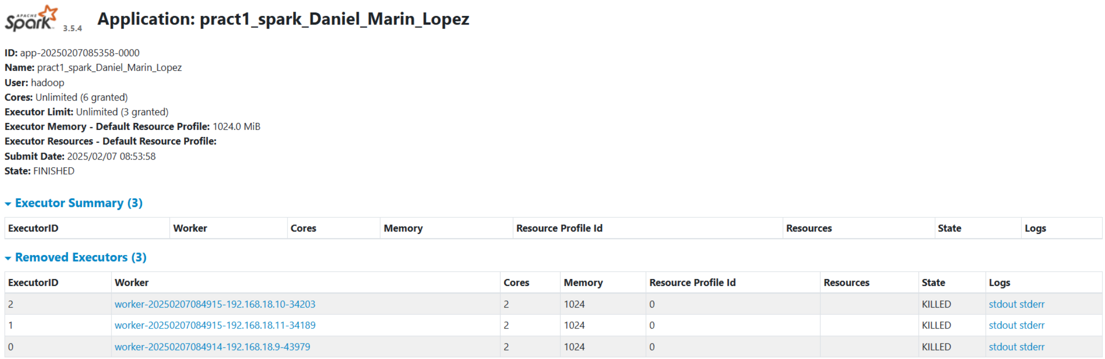
Running Applications (0)

Application ID	Name	Cores	Memory per Executor	Resources Per Executor	Submitted Time	User	State	Duration

Completed Applications (1)

Application ID	Name	Cores	Memory per Executor	Resources Per Executor	Submitted Time	User	State	Duration
app-20250207085358-0000	pract1_spark_Daniel_Marin_Lopez	6	1024.0 MiB		2025/02/07 08:53:58	hadoop	FINISHED	31 s

Y si accedemos a la aplicación:



Application: pract1_spark_Daniel_Marin_Lopez

ID: app-20250207085358-0000
Name: pract1_spark_Daniel_Marin_Lopez
User: hadoop
Cores: Unlimited (6 granted)
Executor Limit: Unlimited (3 granted)
Executor Memory - Default Resource Profile: 1024.0 MiB
Executor Resources - Default Resource Profile:
Submit Date: 2025/02/07 08:53:58
State: FINISHED

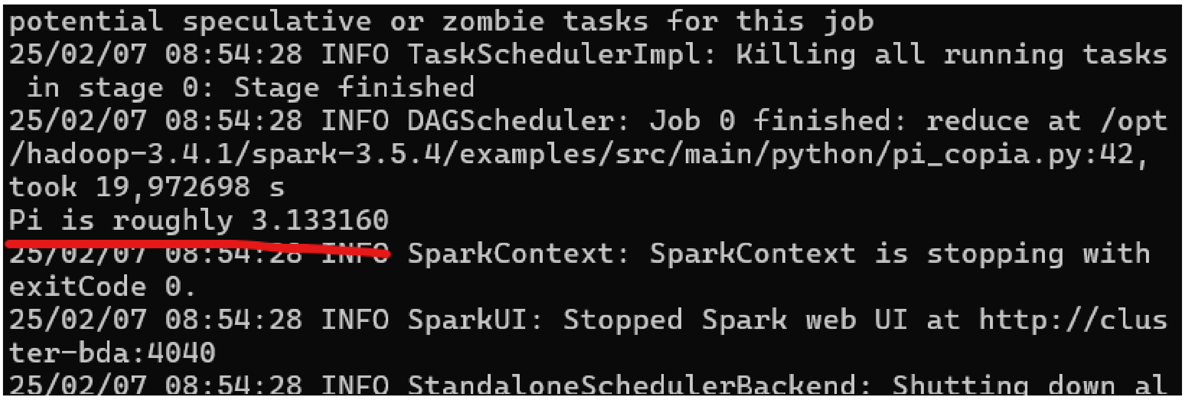
Executor Summary (3)

ExecutorID	Worker	Cores	Memory	Resource Profile Id	Resources	State	Logs

Removed Executors (3)

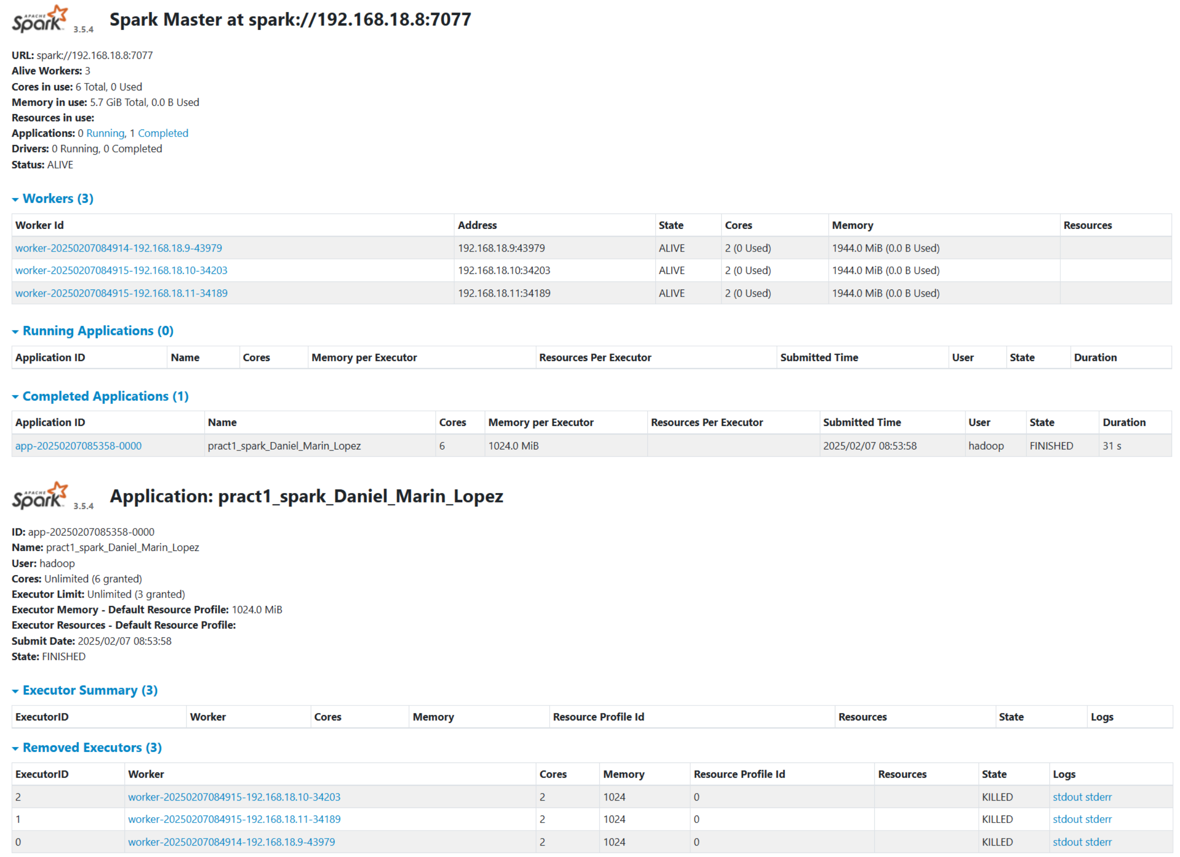
ExecutorID	Worker	Cores	Memory	Resource Profile Id	Resources	State	Logs
2	worker-20250207084915-192.168.18.10-34203	2	1024	0		KILLED	stdout stderr
1	worker-20250207084915-192.168.18.11-34189	2	1024	0		KILLED	stdout stderr
0	worker-20250207084914-192.168.18.9-43979	2	1024	0		KILLED	stdout stderr

8. Añade capturas también del resultado de la ejecución de la aplicación (puedes ser en Spark UI también o en terminal)



```
potential speculative or zombie tasks for this job
25/02/07 08:54:28 INFO TaskSchedulerImpl: Killing all running tasks
 in stage 0: Stage finished
25/02/07 08:54:28 INFO DAGScheduler: Job 0 finished: reduce at /opt
/hadoop-3.4.1/spark-3.5.4/examples/src/main/python/pi_copia.py:42,
took 19,972698 s
Pi is roughly 3.133160
25/02/07 08:54:28 INFO SparkContext: SparkContext is stopping with
exitCode 0.
25/02/07 08:54:28 INFO SparkUI: Stopped Spark web UI at http://clus
ter-bda:4040
25/02/07 08:54:28 INFO StandaloneSchedulerBackend: Shutting down al
```

9. Debe verse correctamente que tienes un cluster correctamente configurado y funcionando



Spark Master at spark://192.168.18.8:7077

URL: spark://192.168.18.8:7077
Alive Workers: 3
Cores in use: 6 Total, 0 Used
Memory in use: 5.7 GiB Total, 0.0 B Used
Resources in use:
Applications: 0 Running, 1 Completed
Drivers: 0 Running, 0 Completed
Status: ALIVE

Workers (3)

Worker Id	Address	State	Cores	Memory	Resources
worker-20250207084914-192.168.18.9-43979	192.168.18.9:43979	ALIVE	2 (0 Used)	1944.0 MiB (0.0 B Used)	
worker-20250207084915-192.168.18.10-34203	192.168.18.10:34203	ALIVE	2 (0 Used)	1944.0 MiB (0.0 B Used)	
worker-20250207084915-192.168.18.11-34189	192.168.18.11:34189	ALIVE	2 (0 Used)	1944.0 MiB (0.0 B Used)	

Running Applications (0)

Application ID	Name	Cores	Memory per Executor	Resources Per Executor	Submitted Time	User	State	Duration

Completed Applications (1)

Application ID	Name	Cores	Memory per Executor	Resources Per Executor	Submitted Time	User	State	Duration
app-20250207085358-0000	pract1_spark_Daniel_Marin_Lopez	6	1024.0 MiB		2025/02/07 08:53:58	hadoop	FINISHED	31 s

Application: pract1_spark_Daniel_Marin_Lopez

ID: app-20250207085358-0000
Name: pract1_spark_Daniel_Marin_Lopez
User: hadoop
Cores: Unlimited (6 granted)
Executor Limit: Unlimited (3 granted)
Executor Memory - Default Resource Profile: 1024.0 MiB
Executor Resources - Default Resource Profile:
Submit Date: 2025/02/07 08:53:58
State: FINISHED

Executor Summary (3)

ExecutorID	Worker	Cores	Memory	Resource Profile Id	Resources	State	Logs

Removed Executors (3)

ExecutorID	Worker	Cores	Memory	Resource Profile Id	Resources	State	Logs
2	worker-20250207084915-192.168.18.10-34203	2	1024	0		KILLED	stdout stderr
1	worker-20250207084915-192.168.18.11-34189	2	1024	0		KILLED	stdout stderr
0	worker-20250207084914-192.168.18.9-43979	2	1024	0		KILLED	stdout stderr