












# Instalacion de Apache Spark en Ubuntu

Autor: Anderson Tenorio Acha

Paso1: verificamos la versión de jdk y hadoop que tenemos instaladas en nuestra distro, para eso usamos el comando “**java -version**” y “**hadoop version**”

```
tenorio@nodo1:~$ java -version
openjdk version "1.8.0_452"
OpenJDK Runtime Environment (build 1.8.0_452-8u452-ga-us1-0ubuntu1~24.04-b09)
OpenJDK 64-Bit Server VM (build 25.452-b09, mixed mode)
tenorio@nodo1:~$ hadoop version
Hadoop 2.10.2
Subversion Unknown -r 965fd380006fa78b2315668fbc7eb432e1d8200f
Compiled by ubuntu on 2022-05-24T22:35Z
Compiled with protoc 2.5.0
From source with checksum d3ab737f7788f05d467784f0a86573fe
This command was run using /opt/hadoop/share/hadoop/common/hadoop-common-2.10.2.jar
tenorio@nodo1:~$
```

Paso 2: una vez verificado entramos al siguiente enlace <https://archive.apache.org/dist/spark/> y descargamos la versión que es mas compatible para nuestra versión de hadoop, en mi caso usare la version 3.3.2

	<a href="#">spark-3.2.2/</a>	2022-07-15 14:43	-
	<a href="#">spark-3.2.3/</a>	2022-11-28 18:04	-
	<a href="#">spark-3.2.4/</a>	2023-04-13 13:46	-
	<a href="#">spark-3.3.0/</a>	2022-06-17 11:11	-
	<a href="#">spark-3.3.1/</a>	2022-10-25 07:22	-
	<a href="#">spark-3.3.2/</a>	2023-02-15 21:15	-
	<a href="#">spark-3.3.3/</a>	2023-08-21 08:23	-
	<a href="#">spark-3.3.4/</a>	2023-12-16 00:26	-
	<a href="#">spark-3.4.0/</a>	2023-04-13 17:54	-
	<a href="#">spark-3.4.1/</a>	2023-06-23 08:23	-
	<a href="#">...</a>	----	----

Name	Last modified	Size	Description
Parent Directory	-	-	-
SparkR_3.3.2.tar.gz	2023-02-10 21:28	344K	
SparkR_3.3.2.tar.gz.asc	2023-02-10 21:28	687	
SparkR_3.3.2.tar.gz.sha512	2023-02-10 21:28	150	
pyspark-3.3.2.tar.gz	2023-02-10 21:28	268M	
pyspark-3.3.2.tar.gz.asc	2023-02-10 21:28	687	
<b>spark-3.3.2-bin-hadoop2.tgz</b>	2023-02-10 21:28	261M	
spark-3.3.2-bin-hadoop2.tgz.sha512	2023-02-10 21:28	158	
spark-3.3.2-bin-hadoop3-scale2.13.tgz	2023-02-10 21:28	292M	
spark-3.3.2-bin-hadoop3-scale2.13.tgz.asc	2023-02-10 21:28	687	
spark-3.3.2-bin-hadoop3-scale2.13.tgz.sha512	2023-02-10 21:28	168	
spark-3.3.2-bin-hadoop3.tgz	2023-02-10 21:28	285M	
spark-3.3.2-bin-hadoop3.tgz.asc	2023-02-10 21:28	687	
spark-3.3.2-bin-hadoop3.tgz.sha512	2023-02-10 21:28	158	
spark-3.3.2-bin-without-hadoop.tgz	2023-02-10 21:28	201M	
spark-3.3.2-bin-without-hadoop.tgz.asc	2023-02-10 21:28	687	
spark-3.3.2-bin-without-hadoop.tgz.sha512	2023-02-10 21:28	165	
spark-3.3.2.tgz	2023-02-10 21:28	28M	
spark-3.3.2.tgz.asc	2023-02-10 21:28	687	
spark-3.3.2.tgz.sha512	2023-02-10 21:28	146	

Paso 3: una vez descargado el spark, entramos a la carpeta Descargas desde nuestra terminal usando el comando “cd ~/Descargas” y verificamos si está ahí con el comando “ls”

```
tenorio@nodo1:~$ cd ~/Descargas
tenorio@nodo1:~/Descargas$ ls
hadoop-2.10.2.tar.gz  prueba_final.txt  spark-3.3.2-bin-hadoop2.tgz
tenorio@nodo1:~/Descargas$
```

Paso 4: ahí mismo usamos el comando “tar -xvzf spark-3.3.2-bin-hadoop2.tgz” para extraer los archivos dentro de la carpeta, recordemos que el “x” representación la extracción, el “v” muestra en pantalla los archivos que se extraen, el “z” indica que el archivo esta comprimido en gzip y por ultimo el “f” le indica a tar que se usara un archivo.

```
24 de jun 16:36
tenorio@nodo1: ~/Descargas
spark-3.3.2-bin-hadoop2/python/test_support/userlibrary.py
spark-3.3.2-bin-hadoop2/sbin/
spark-3.3.2-bin-hadoop2/sbin/decommission-slave.sh
spark-3.3.2-bin-hadoop2/sbin/decommission-worker.sh
spark-3.3.2-bin-hadoop2/sbin/slaves.sh
spark-3.3.2-bin-hadoop2/sbin/spark-config.sh
spark-3.3.2-bin-hadoop2/sbin/spark-daemon.sh
spark-3.3.2-bin-hadoop2/sbin/spark-daemons.sh
spark-3.3.2-bin-hadoop2/sbin/start-all.sh
spark-3.3.2-bin-hadoop2/sbin/start-history-server.sh
spark-3.3.2-bin-hadoop2/sbin/start-master.sh
spark-3.3.2-bin-hadoop2/sbin/start-mesos-dispatcher.sh
spark-3.3.2-bin-hadoop2/sbin/start-mesos-shuffle-service.sh
spark-3.3.2-bin-hadoop2/sbin/start-slave.sh
spark-3.3.2-bin-hadoop2/sbin/start-slaves.sh
spark-3.3.2-bin-hadoop2/sbin/start-thriftserver.sh
spark-3.3.2-bin-hadoop2/sbin/start-worker.sh
spark-3.3.2-bin-hadoop2/sbin/start-workers.sh
spark-3.3.2-bin-hadoop2/sbin/stop-all.sh
spark-3.3.2-bin-hadoop2/sbin/stop-history-server.sh
spark-3.3.2-bin-hadoop2/sbin/stop-master.sh
spark-3.3.2-bin-hadoop2/sbin/stop-mesos-dispatcher.sh
spark-3.3.2-bin-hadoop2/sbin/stop-mesos-shuffle-service.sh
spark-3.3.2-bin-hadoop2/sbin/stop-slave.sh
spark-3.3.2-bin-hadoop2/sbin/stop-slaves.sh
spark-3.3.2-bin-hadoop2/sbin/stop-thriftserver.sh
spark-3.3.2-bin-hadoop2/sbin/stop-worker.sh
spark-3.3.2-bin-hadoop2/sbin/stop-workers.sh
spark-3.3.2-bin-hadoop2/sbin/workers.sh
spark-3.3.2-bin-hadoop2/yarn/
spark-3.3.2-bin-hadoop2/yarn/spark-3.3.2-yarn-shuffle.jar
tenorio@nodo1: ~/Descargas$
```

Paso 5: ahora movemos el archivo descomprimido con el comando “mv mv spark-3.3.2-bin-hadoop2 ~/spark” esto cambiara el nombre de la carpeta a “spark” y lo alojara en “home(el inicio de todo el terminal”

```
24 de jun 16:45
tenorio@nodo1: ~
tenorio@nodo1:~$ ls
Descargas  Escritorio  Música     Público   spark
Documentos Imágenes   Plantillas snap      videos
tenorio@nodo1:~$
```

Paso 6: editamos el archivo bashrc con el comando “nano ~/.bashrc” y le añadimos lo siguiente en la parte final los siguiente y lo guardamos con el “ctrl + O”, una vez guardado usamos el comando “source ~/.bashrc” esto permitirá usar comandos de “spark” desde cualquier parte.

```
24 de jun 10:52
tenorio@nodo1: ~
GNU nano 7.2 /home/tenorio/.bashrc *
if [ -f /usr/share/bash-completion/bash_completion ]; then
    . /usr/share/bash-completion/bash_completion
elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
fi

export HADOOP_PREFEX=/opt/hadoop
export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64/jre
export HADOOP_CONF_DIR=/opt/hadoop/etc/hadoop
export PATH=$HADOOP_PREFEX/sbin:$PATH:$HADOOP_PREFEX/bin
export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
export PATH=$PATH:$JAVA_HOME/bin
# Apache Spark
export SPARK_HOME=~/.spark
export PATH=$PATH:$SPARK_HOME/bin:$SPARK_HOME/sbin
export HADOOP_CONF_DIR=$HADOOP_HOME/etc/hadoop

^G Ayuda      ^O Guardar    ^W Buscar     ^K Cortar     ^T Ejecutar   ^C Ubicación
^X Salir      ^R Leer fich. ^\ Reemplazar  ^U Pegar      ^J Justificar ^/ Ir a línea
```

```
tenorio@nodo1: ~
tenorio@nodo1:~$ source ~/.bashrc
tenorio@nodo1:~$
```

Paso 7: probamos si esta todo correcto usando el comando “spark-shell” y el comando para Python con el comando “pyspark”

```
tenorio@nodo1:~$ source ~/.bashrc
tenorio@nodo1:~$ spark-shell
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
25/06/24 17:01:14 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Spark context Web UI available at http://nodo1:4040
Spark context available as 'sc' (master = local[*], app id = local-1750802475694).
Spark session available as 'spark'.
Welcome to

  ____      _
 / ___|    / \
| |  | |  / _ \
| |  | | / ___ \
| |  | || |___| \
 \___|_||_|___|_|
                    version 3.3.2

Using Scala version 2.12.15 (OpenJDK 64-Bit Server VM, Java 1.8.0_452)
Type in expressions to have them evaluated.
Type :help for more information.

scala>
```

```
24 de jun 17:10
tenorio@nodo1: ~/spark
tenorio@nodo1:~/spark$ pyspark
Python 3.12.3 (main, Feb 4 2025, 14:48:35) [GCC 13.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
25/06/24 17:08:51 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Welcome to

  ____      _
 / ___|    / \
| |  | |  / _ \
| |  | | / ___ \
| |  | || |___| \
 \___|_||_|___|_|
                    version 3.3.2

Using Python version 3.12.3 (main, Feb 4 2025 14:48:35)
Spark context Web UI available at http://nodo1:4040
Spark context available as 'sc' (master = local[*], app id = local-1750802932372).
SparkSession available as 'spark'.
>>>
```