

Instalación de Docker. Generar un docker-spark-cluster



Néstor Batista Díaz

INSTALAR DOCKER

Install Docker Desktop on Windows

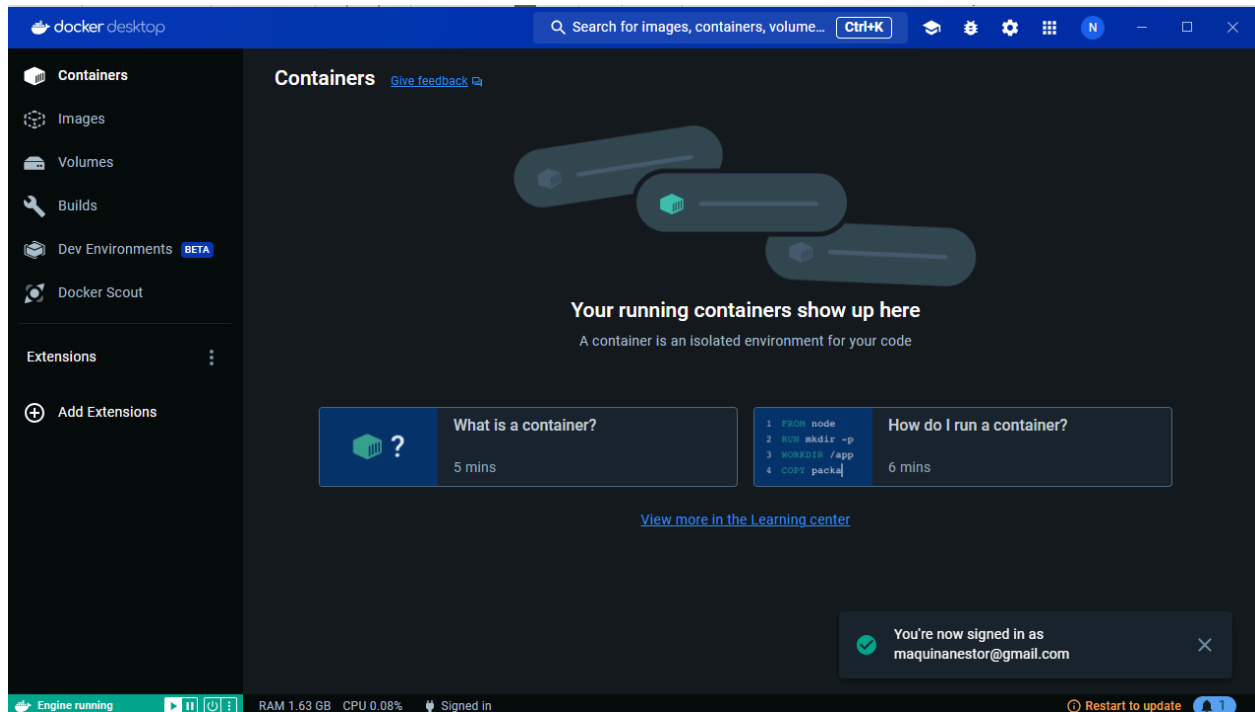
i Docker Desktop terms

Commercial use of Docker Desktop in larger enterprises (more than 250 employees OR more than \$10 million USD in annual revenue) requires a [paid subscription](#).

This page contains the download URL, information about system requirements, and instructions on how to install Docker Desktop for Windows.

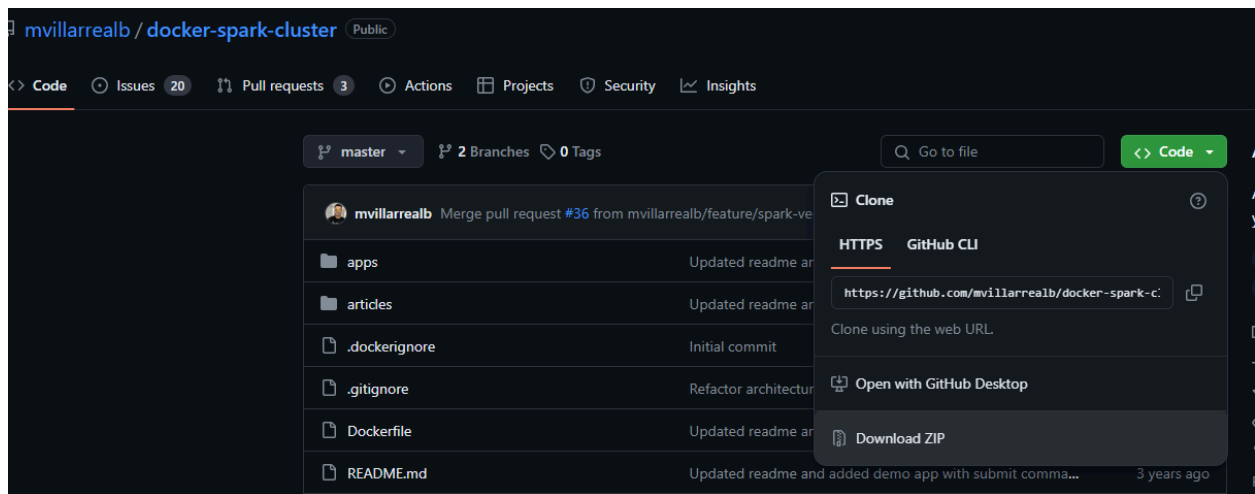
[Docker Desktop for Windows](#)

For checksums, see [Release notes](#)



INSTALAR EL CLUSTER DE SPARK

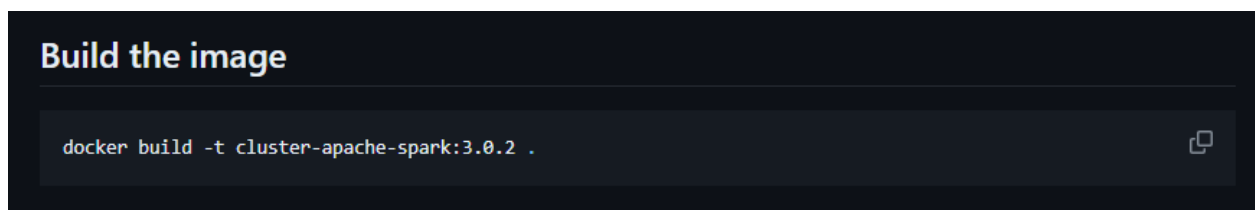
Descargamos el proyecto de github



Descomprimos

	Nombre	Fecha de modificación	Tipo	Tamaño
	apps	27/06/2021 2:41	Carpeta de archivos	
	articles	27/06/2021 2:41	Carpeta de archivos	
	.dockerignore	27/06/2021 2:41	Archivo DOCKERI...	0 KB
	.gitignore	27/06/2021 2:41	Documento de te...	1 KB
	docker-compose.yml	27/06/2021 2:41	Archivo de origen ...	2 KB
	Dockerfile	27/06/2021 2:41	Archivo	2 KB
	README.md	27/06/2021 2:41	Archivo de origen ...	6 KB
	start-spark.sh	27/06/2021 2:41	Shell Script	1 KB

Abrimos CMD y ejecutamos el siguiente código en la carpeta del proyecto



```
C:\Windows\system32\cmd.exe
=> => sha256:567d410fadc4d7125fabe9cbac4ac3b566d498b3d8050c0405bdd78fdb4a384d 212B / 212B 0.6s
=> => sha256:24de726604f496a8d34cc960f39c3f3d825ebba522d8be7b256f8b289a448508 549B / 549B 0.0s
=> => sha256:891cdfaa81b9c4fd2d95f3539c4037c2ab928b2c0b90a00b2da92d26d0d4ea42 47.08MB / 47.08MB 7.2s
=> => extracting sha256:b4d181a07f8025e00e0cb28f1cc14613da2ce26450b80c54aea537fa93cf3bda 1.2s
=> => extracting sha256:3ee45ae9730633057cf9bd12924f9a1bf2b590631d3d085b14c96f4466557794 0.1s
=> => extracting sha256:567d410fadc4d7125fabe9cbac4ac3b566d498b3d8050c0405bdd78fdb4a384d 0.0s
=> => extracting sha256:891cdfaa81b9c4fd2d95f3539c4037c2ab928b2c0b90a00b2da92d26d0d4ea42 0.6s
=> [internal] load build context 0.0s
=> => transferring context: 716B 0.0s
=> [builder 2/4] RUN apt-get update && apt-get install -y curl vim wget software-properties-common ssh net-tool 81.6s
=> [builder 3/4] RUN update-alternatives --install "/usr/bin/python" "python" "$(which python3)" 1 0.3s
=> [builder 4/4] RUN wget --no-verbose -O apache-spark.tgz "https://archive.apache.org/dist/spark/spark-3.0.2/" 210.2s
=> [apache-spark 1/3] WORKDIR /opt/spark 0.0s
=> [apache-spark 2/3] RUN mkdir -p /opt/spark/logs && touch /opt/spark/logs/spark-master.out && touch /opt/spark 0.4s
=> [apache-spark 3/3] COPY start-spark.sh / 0.0s
=> exporting to image 2.2s
=> => exporting layers 2.2s
=> => writing image sha256:581dbef9776727bdb6c029ad2a37eec032f710a56c020724af718e7d9db02ec9 0.0s
=> => naming to docker.io/library/cluster-apache-spark:3.0.2 0.0s

What's Next?
View a summary of image vulnerabilities and recommendations → docker scout quickview
```

Ejecutamos Docker compose




























Run the docker-compose

The final step to create your test cluster will be to run the compose file:

```
docker-compose up -d
```

```
[+] Running 10/10er FP IA y BD\SPS\docker-spark\docker-spark-cluster-master>docker-compose up -d
demo-database 9 layers [00000000] 0B/0B Pulled 9.8s
  cdbbe7a5bc2a Pull complete 1.0s
  b52a8a2ca21a Pull complete 0.6s
  e36a19831e31 Pull complete 0.6s
  f1aa26821845 Pull complete 6.0s
  412d098142b4 Pull complete 1.2s
  75d5ef10726d Pull complete 1.5s
  ae3b5a8bbf62 Pull complete 2.0s
  e2f290791a5c Pull complete 2.2s
  187b81308ed8 Pull complete 2.6s
[+] Running 4/5
- Network docker-spark-cluster-master_default Create... 2.3s
  Container docker-spark-cluster-master-demo-database-1 Started 1.4s
  Container docker-spark-cluster-master-spark-master-1 Started 1.6s
  Container docker-spark-cluster-master-spark-worker-b-1 Started 2.0s
  Container docker-spark-cluster-master-spark-worker-a-1 Started 1.8s
```

Comprobamos que funciona

<input type="checkbox"/>		docker-spark-cluster-master		Running (4/4)	0.31%	6 minutes ago				
<input type="checkbox"/>		demo-database-1	postgres:11.7-alpine	Running	0%	5432-5432 	6 minutes ago			
<input type="checkbox"/>		spark-master-1	cluster-apache-spark:3.0.2	Running	0.09%	7077-7077  9090-8080  Show less	6 minutes ago			
<input type="checkbox"/>		spark-worker-a-1	cluster-apache-spark:3.0.2	Running	0.1%	7000-7000  9091-8080  Show less	6 minutes ago			
<input type="checkbox"/>		spark-worker-b-1	cluster-apache-spark:3.0.2	Running	0.12%	7001-7000  9092-8080  Show less	6 minutes ago			

Spark

3.0.2

Spark Master at spark://08ec0e71104d:7077

URL: spark://08ec0e71104d:7077

Alive Workers: 2

Cores in use: 2 Total, 0 Used

Memory in use: 2.0 GiB Total, 0.0 B Used

Resources in use:

Applications: 0 Running, 0 Completed

Drivers: 0 Running, 0 Completed

Status: ALIVE

Workers (2)

Worker Id	Address	State	Cores	Memory	Resources
worker-20240416172524-172.18.0.4-7000	172.18.0.4:7000	ALIVE	1 (0 Used)	1024.0 MiB (0.0 B Used)	
worker-20240416172524-172.18.0.5-7000	172.18.0.5:7000	ALIVE	1 (0 Used)	1024.0 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 (0)

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