

Servicios Amazon Web Services

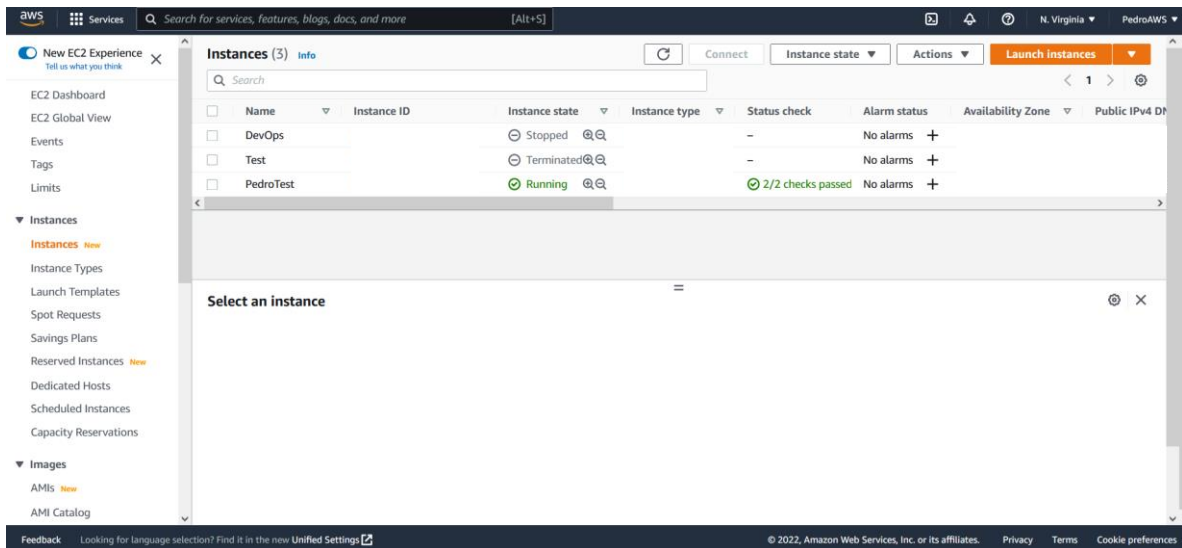


# Introducción

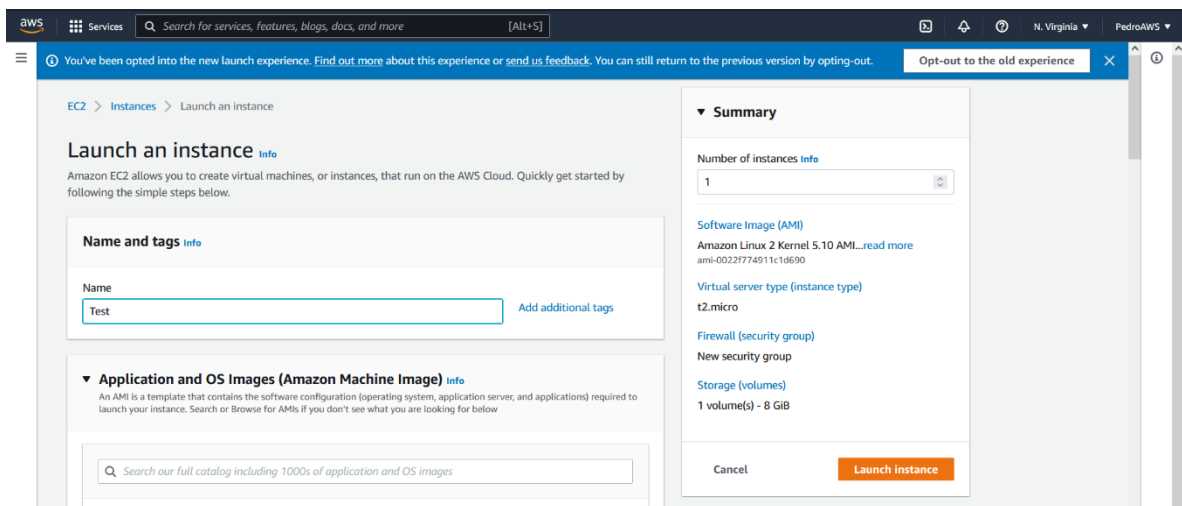
En este documento se explicará paso a paso que se realizó para poder crear una instancia y conectarnos a una base de datos MySQL que esté estructurada con la herramienta ‘Docker’, a su vez de como conectarnos a la base de datos.

## Paso 1:

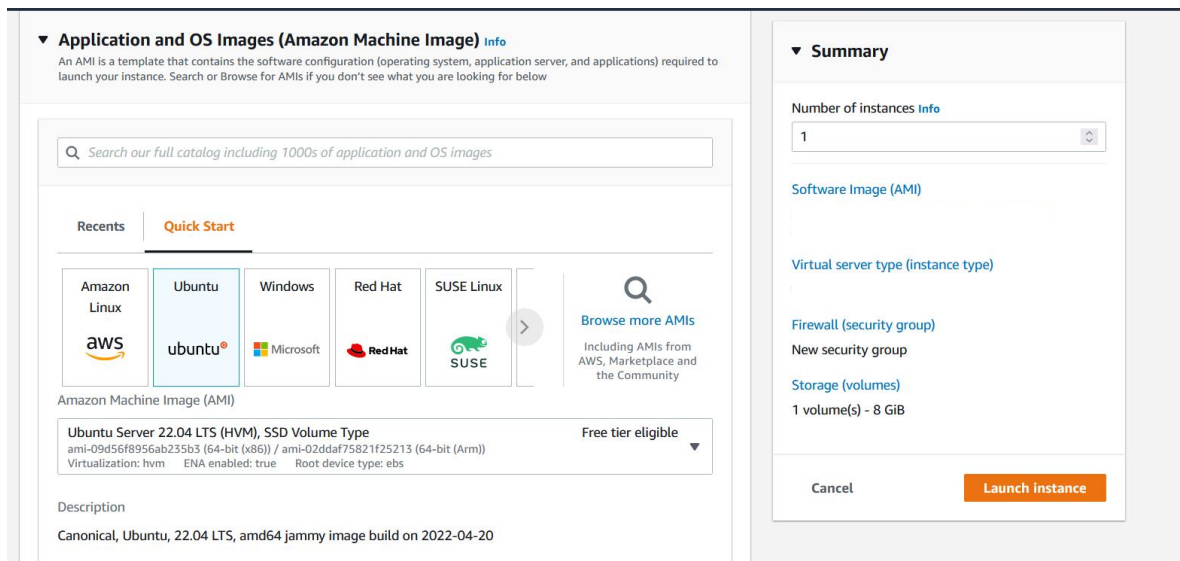
Tenemos que tener una cuenta de ‘Amazon Web Services’, tendríamos que ir al buscador de AWS y buscar el servicio ‘EC2’



Tendremos esta pantalla, y nos dirigiremos a ‘instances’ para crear nuestra instancia, daremos click en el boton naranja en la parte superior derecha de nuestra pantalla



Rellenaremos los campos necesarios como nombre, seleccionaremos un sistema operativo, que en esta ocasión será ‘Ubuntu 22.04 LTS’



▼ **Application and OS Images (Amazon Machine Image)** [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Search our full catalog including 1000s of application and OS images

Recents Quick Start

Amazon Linux Ubuntu Windows Red Hat SUSE Linux

Amazon Machine Image (AMI)

Ubuntu Server 22.04 LTS (HVM), SSD Volume Type  
ami-09d56f8956ab235b3 (64-bit (x86)) / ami-02ddaf75821f25213 (64-bit (Arm))  
Virtualization: hvm ENA enabled: true Root device type: ebs

Free tier eligible

Description

Canonical, Ubuntu, 22.04 LTS, amd64 jammy image build on 2022-04-20

▼ **Summary**

Number of instances [Info](#)

1

Software Image (AMI)

Virtual server type (instance type)

Firewall (security group)

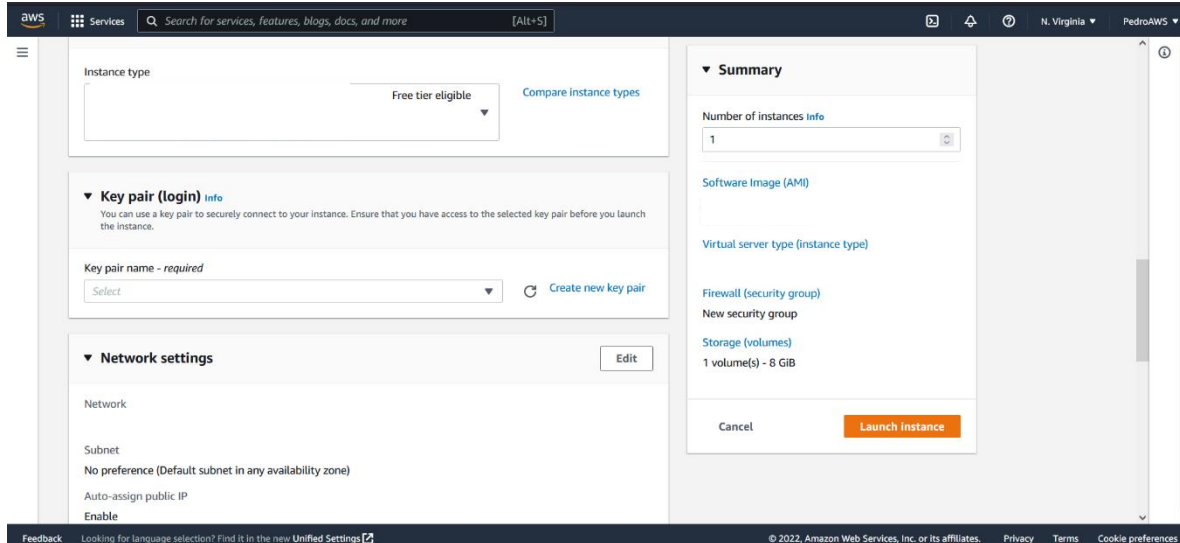
New security group

Storage (volumes)

1 volume(s) - 8 GiB

Cancel Launch instance

Crearemos nuestra llave y la guardaremos en una carpeta donde nadie pueda obtenerla



aws Services Search for services, features, blogs, docs, and more [Alt+S] N. Virginia PedroAWS

Instance type

Free tier eligible Compare instance types

▼ **Key pair (login)** [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required

Select Create new key pair

▼ **Network settings** [Edit](#)

Network

Subnet

No preference (Default subnet in any availability zone)

Auto-assign public IP

Enable

▼ **Summary**

Number of instances [Info](#)

1

Software Image (AMI)

Virtual server type (instance type)

Firewall (security group)

New security group

Storage (volumes)

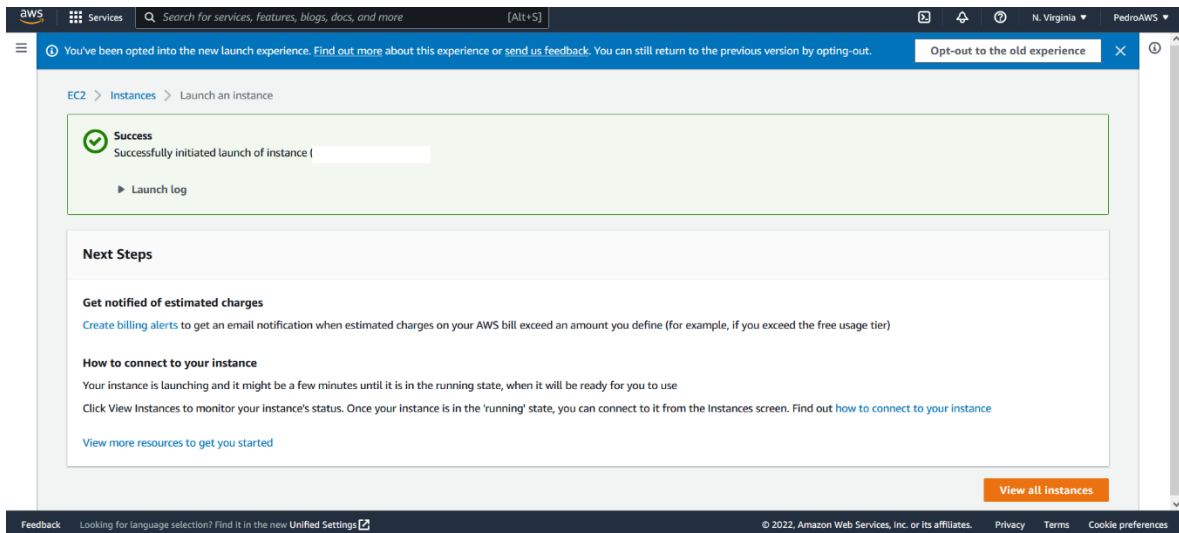
1 volume(s) - 8 GiB

Cancel Launch instance

Feedback Looking for language selection? Find it in the new Unified Settings

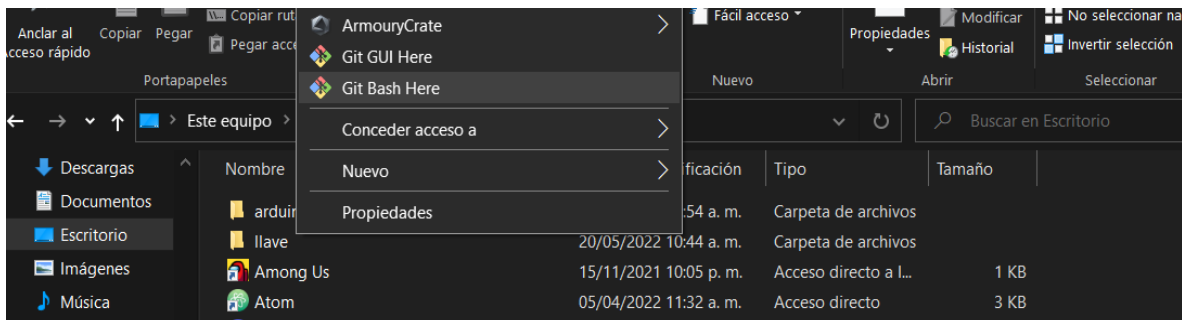
© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Le daremos click en ‘Launch Instance’ y estaremos listos para conectarnos a nuestro servicio para que levantemos mediante Docker una base de datos



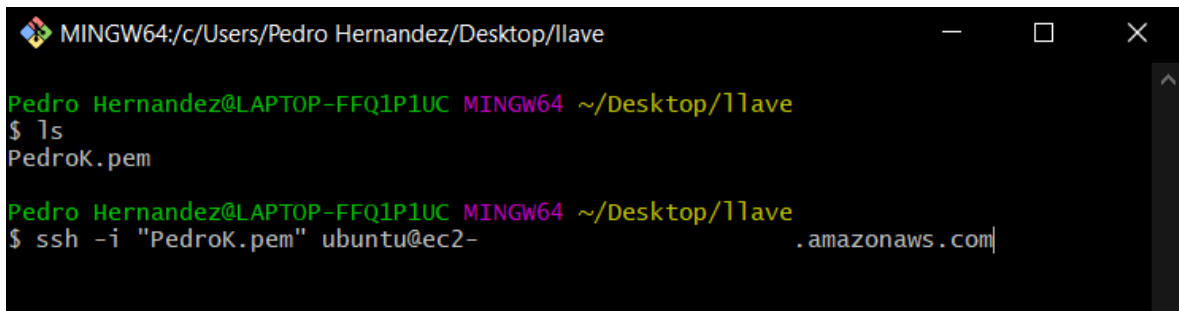
## Paso 2

Tendremos que tener igual Git instalado de preferencia para simular un “SO Linux”



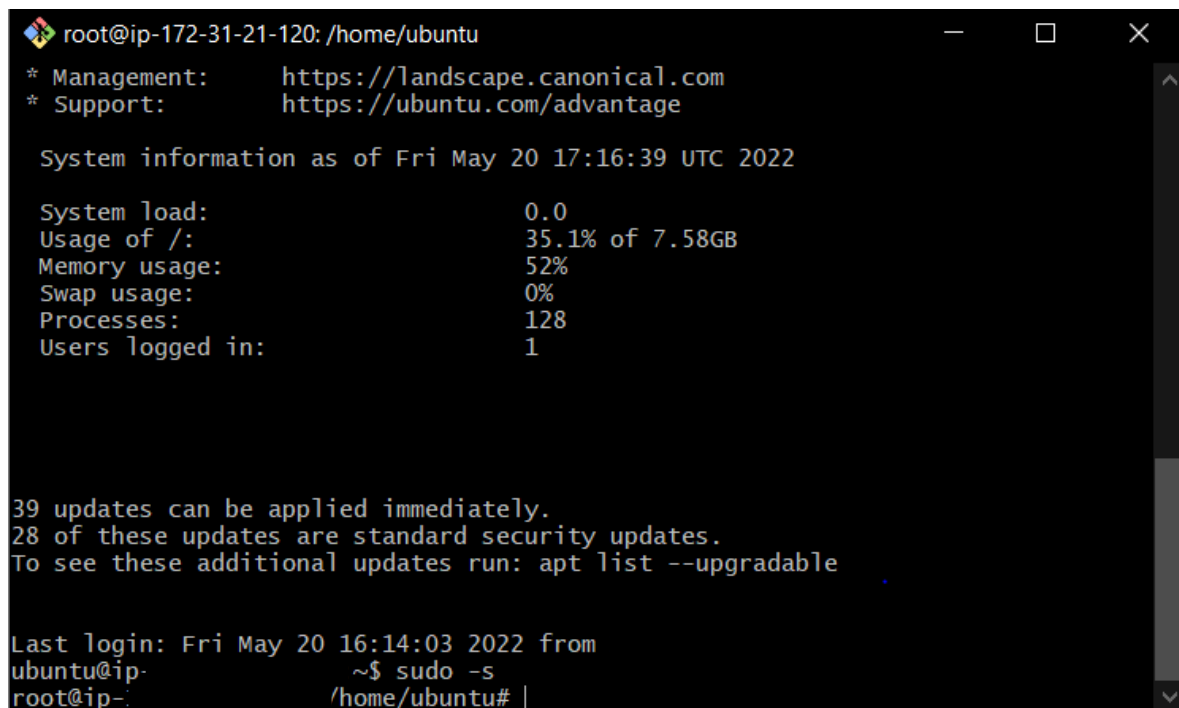
Le daremos click a ‘Git Bash Here’ para acceder a nuestra carpeta donde esté nuestra llave de conexión

Copiaremos el comando para entrar en SSH

A terminal window titled 'MINGW64:/c/Users/Pedro Hernandez/Desktop/llave'. The prompt is 'Pedro Hernandez@LAPTOP-FFQ1P1UC MINGW64 ~/Desktop/llave'. The user enters '\$ ls' and the output is 'PedroK.pem'. Then the user enters '\$ ssh -i "PedroK.pem" ubuntu@ec2-'. The cursor is at the end of the command, and '.amazonaws.com' is visible in the background of the terminal.

```
MINGW64:/c/Users/Pedro Hernandez/Desktop/llave
Pedro Hernandez@LAPTOP-FFQ1P1UC MINGW64 ~/Desktop/llave
$ ls
PedroK.pem
Pedro Hernandez@LAPTOP-FFQ1P1UC MINGW64 ~/Desktop/llave
$ ssh -i "PedroK.pem" ubuntu@ec2- .amazonaws.com|
```

Lo pegaremos para poder entrar a nuestra instancia y damos click en la tecla ‘Enter‘

A terminal window titled 'root@ip-172-31-21-120: /home/ubuntu'. It shows management and support links, system information as of Fri May 20 17:16:39 UTC 2022, and a list of updates that can be applied immediately. The prompt is 'root@ip-: /home/ubuntu#'.

```
root@ip-172-31-21-120: /home/ubuntu
* Management:      https://landscape.canonical.com
* Support:         https://ubuntu.com/advantage

System information as of Fri May 20 17:16:39 UTC 2022

System load:                0.0
Usage of /:                  35.1% of 7.58GB
Memory usage:               52%
Swap usage:                 0%
Processes:                  128
Users logged in:            1

39 updates can be applied immediately.
28 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Last login: Fri May 20 16:14:03 2022 from
ubuntu@ip-: ~$ sudo -s
root@ip-: /home/ubuntu# |
```

Una vez dentro de la instancia tendremos que poner el siguiente comando ‘sudo -s’ para que podamos ejecutar comando que necesiten privilegios de administrador

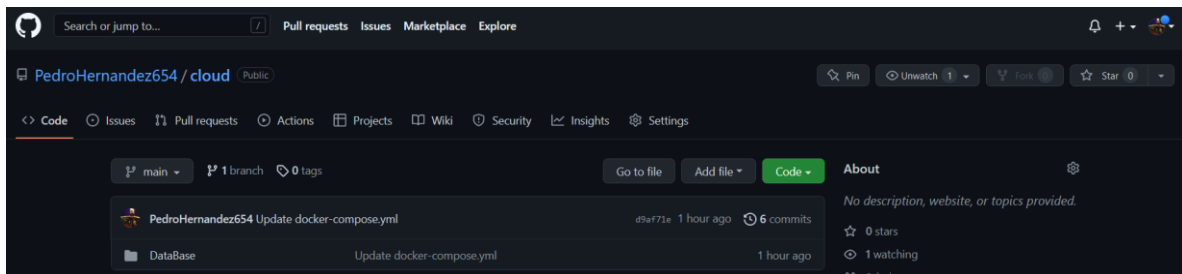
## Paso 3

Con nuestro editor de de código que más nos agrade, tendremos que crear un archivo ‘docker-compose.yml’ para que podamos introducir los comandos necesarios para nuestra herramienta

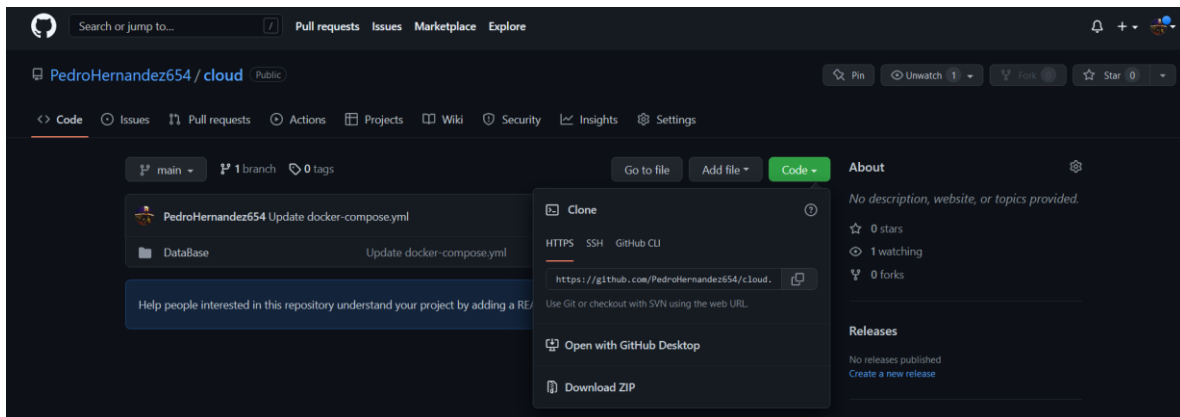
```
version: "3"

services:
  mysql:
    container_name: mysql
    image: mysql:5.7
    command: --default-authentication-plugin=mysql_native_password
    volumes:
      - ./mysql-data:/var/lib/mysql
    ports:
      - 3336:3306
    environment:
      MYSQL_ROOT_PASSWORD: 
      MYSQL_DATABASE: "db_test"
```

Este será el código que meteremos a un repositorio de GitHub para poder clonarlo dentro de nuestra instancia de ‘EC2’



Crearemos un nuevo repositorio y meteremos la carpeta donde está nuestro archivo ‘docker-compose.yml’



Daremos click en el botón verde para obtener el link de nuestro repositorio para poder clonarlo en la instancia, junto con código ‘git clone’

```
root@ip-172-31-21-120: /home/ubuntu
* Support:      https://ubuntu.com/advantage

System information as of Fri May 20 17:22:27 UTC 2022

System load:                0.0
Usage of /:                  35.1% of 7.58GB
Memory usage:               53%
Swap usage:                 0%
Processes:                  133
Users logged in:            1
...

39 updates can be applied immediately.
28 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Last login: Fri May 20 17:22:28 2022 from 189.220.43.42
ubuntu@ip-172-31-21-120:~$ sudo -s
root@ip-172-31-21-120:/home/ubuntu# git clone https://github.com/PedroHernandez654/cloud.git
```

Daremos al botón ‘Enter para clonarlo’, nos meteremos a la carpeta como la hayamos llamado con comandos ‘cd + \*Nombre de la carpeta\*’

## Paso 4 Docker

Antes de poder utilizar nuestra instancia debemos instalar Docker y Docker Compose dentro de la instancia para que sea más fácil la creación de la base de datos.

Utilizaremos estos comandos para instalar primero Docker

```
sudo apt update
```

```
sudo apt install apt-transport-https ca-certificates curl software-properties-common
```

```
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
```

```
sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu  
focal stable"
```

```
sudo apt update
```

```
apt-cache policy docker-ce
```

```
sudo apt install docker-ce
```

```
sudo systemctl status Docker
```

Utilizaremos estos comandos para instalar el Docker-Compose

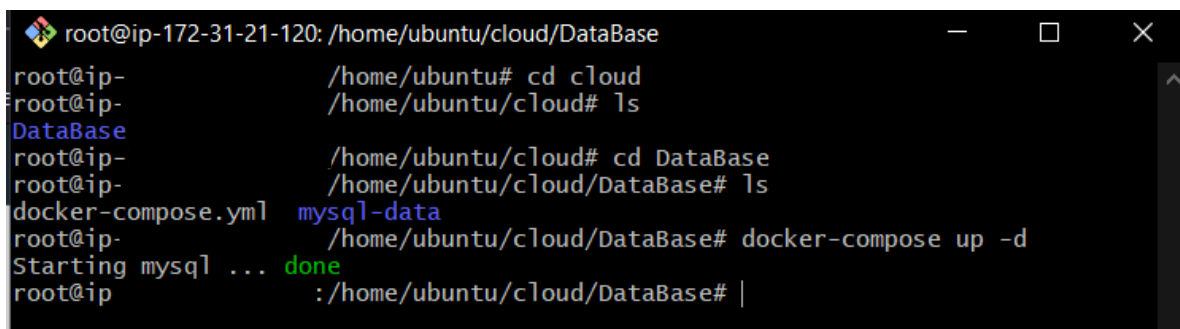
```
sudo curl -L "https://github.com/docker/compose/releases/download/1.26.0/docker-  
compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
```

```
sudo chmod +x /usr/local/bin/docker-compose
```

```
docker-compose --version
```

## Paso 5

Buscaremos nuestro documento ‘.yaml’ para poder darle el comando ‘docker-compose up -d’ para levantar la base de datos y poder acceder a ella



```
root@ip-172-31-21-120: /home/ubuntu/cloud/DataBase
root@ip- /home/ubuntu# cd cloud
root@ip- /home/ubuntu/cloud# ls
DataBase
root@ip- /home/ubuntu/cloud# cd DataBase
root@ip- /home/ubuntu/cloud/DataBase# ls
docker-compose.yml mysql-data
root@ip- /home/ubuntu/cloud/DataBase# docker-compose up -d
Starting mysql ... done
root@ip- :/home/ubuntu/cloud/DataBase# |
```

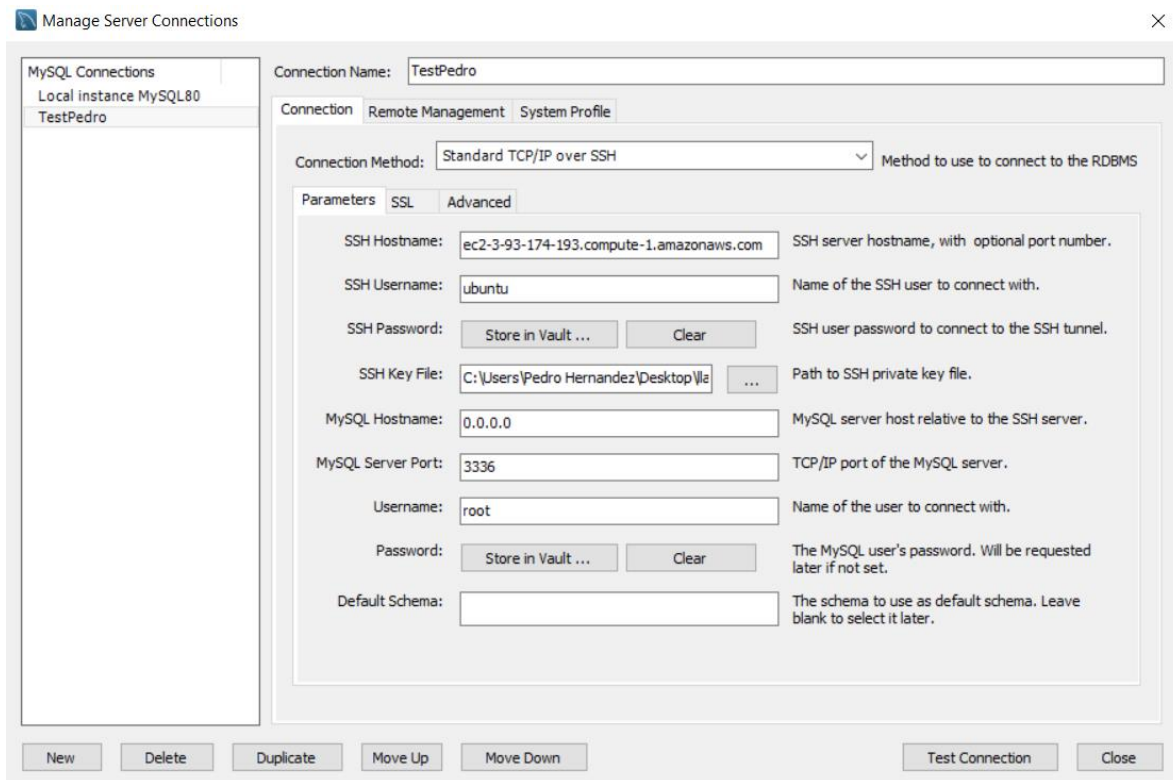
Una vez que levantamos la base de datos pasaremos al siguiente paso

## Paso 6

Tendremos que abrir ‘MySQL WorkBench’ para ingresar a la base de datos

Daremos click en ‘+’ para crear una nueva conexión





Rellenaremos los datos que se nos piden, en este caso es el nombre de la conexión, cambiaremos el método de conexión de Standart(TCP/IP) a Standart(TCP/IP) over SSH

El SHH Hostname será nuestra IPv4 DNS la cual podremos encontrar en nuestra instancia de EC2

SSH Username será ubuntu ya que es nuestro sistema operativo

SSH Key File aquí seleccionaremos en donde se ubica nuestra clave que creamos cuando creamos nuestra instancia

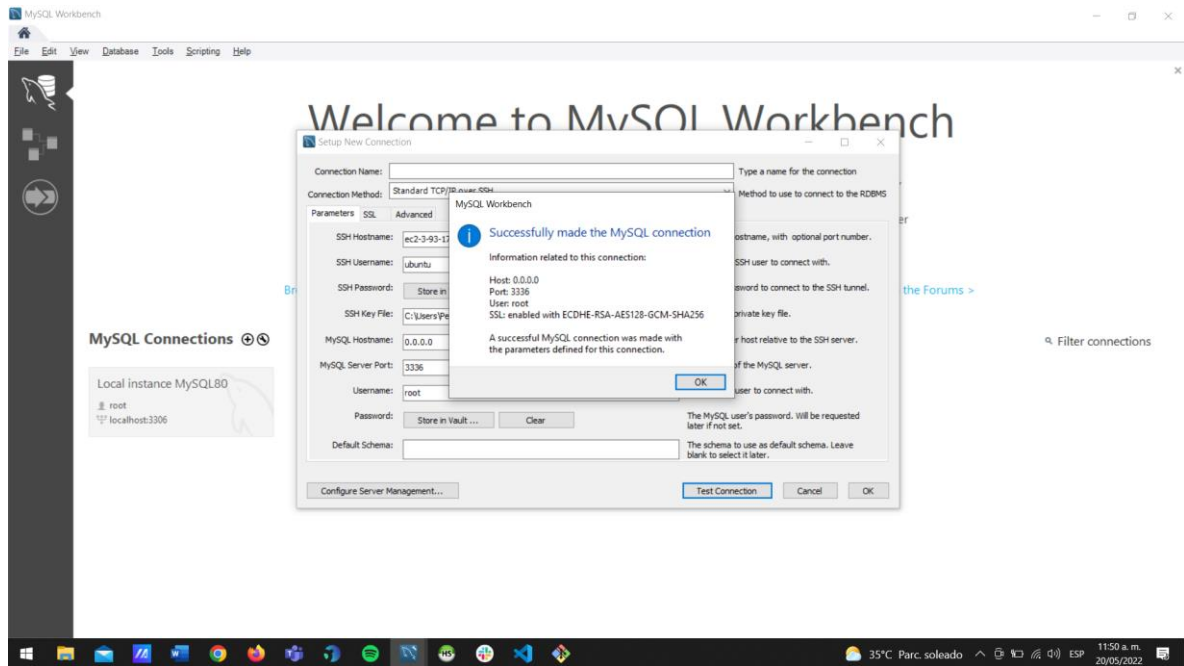
El Hostname tendrá 0.0.0.0 para que cualquier IP pueda entrar

Colocaremos el puerto que ubicamos en el anterior documento 'docker-compose.yml'

El usuario será root ya que se declaró en el anterior documento 'dockr-compose.yml'

SSH Pasword es la contraseña que colocamos en el anterior documento 'docker-compose.yml'

Probaremos la conexión primero y si todo resulta bien, aparecerá el siguiente mensaje



Y terminamos, Una vez que se termine la prueba podremos ingresar a nuestra base de datos en nuestra instancia de EC2