

# Práctica

# DOCKER

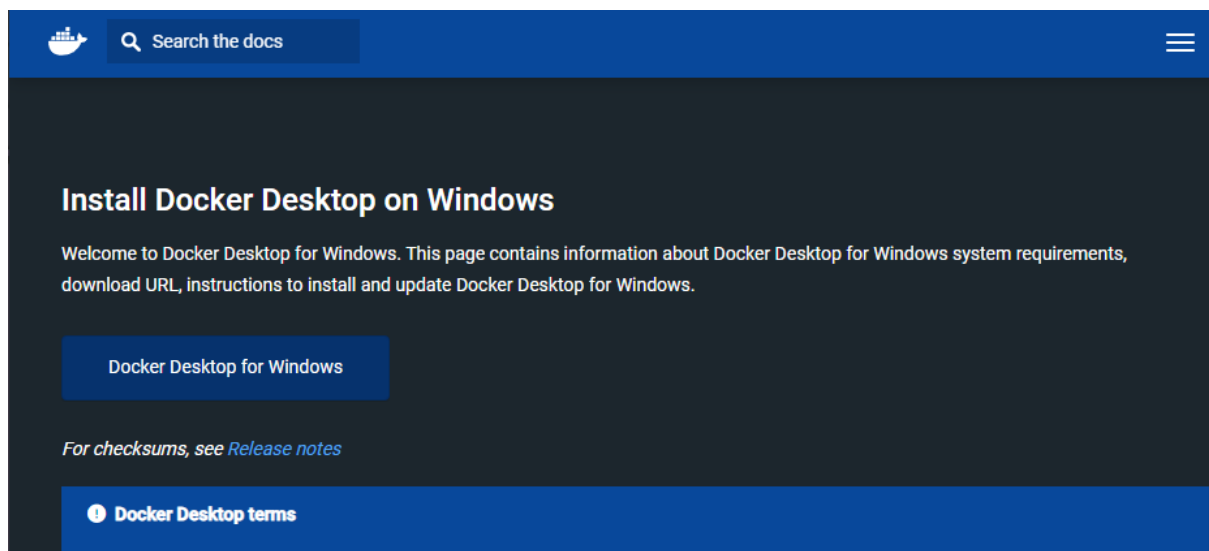
---

## Instalación de servidor Docker:

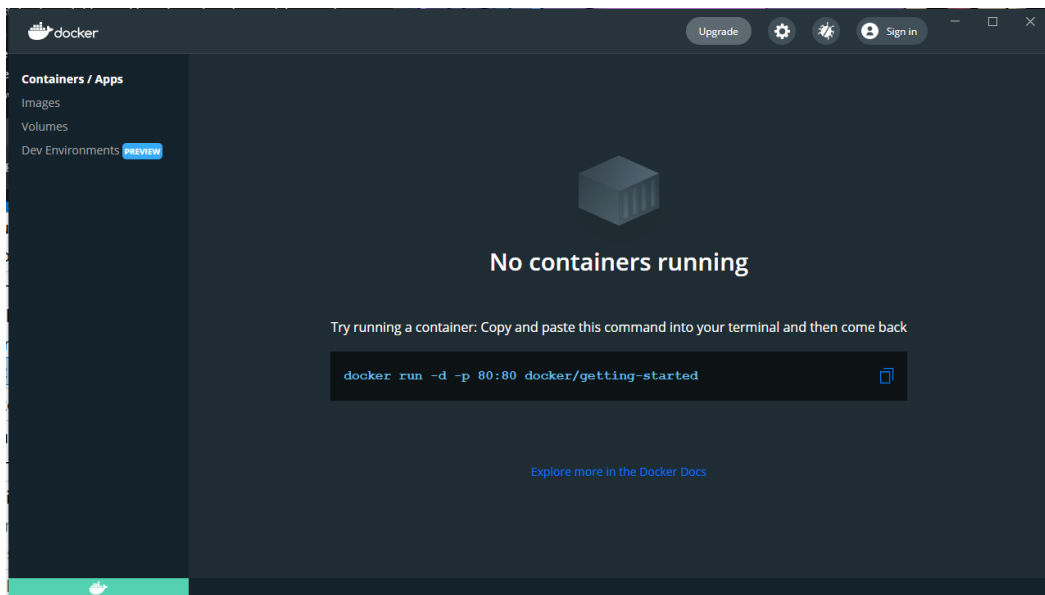
Antes de comenzar con la instalación de docker en el caso de windows deberías descargar primero [Windows Server](#), y añadido a esto una maquina virtual con azure para poder guardar de manera más sencilla los contenedores.

La ventaja de usar Image Builder es que la configuración se realiza en tiempo de compilación y no tendría ningún efecto en la carga de trabajo en tiempo de ejecución; cuando el conjunto de escalado de máquinas virtuales crea una instancia de una nueva máquina virtual a partir de la imagen personalizada, la imagen ya se habrá preparado, por lo que no se debe emplear tiempo aquí y estará lista inmediatamente para ejecutar contenedores.

Comenzando con la instalación de docker, descargaremos Docker Desktop,



Una vez se instale se verá así y lanzaremos el comando siguiente para establecer un contenedor predeterminado.



```
C:\Users\Pablo>& $Env:ProgramFiles\Docker\Docker\DockerCli.exe -SwitchDaemon
```

## Creación de contenedor con imagen Docker:

Abre una ventana de símbolo del sistema y, a continuación, ejecuta el siguiente comando para descargar e instalar la imagen base:

```
C:\Users\Pablo>docker pull amd64/ubuntu
```

Ahora confirmaremos que se descargó correctamente cuando el comando **docker images**.

```
C:\Users\Pablo>docker images
REPOSITORY    TAG       IMAGE ID      CREATED        SIZE
amd64/ubuntu  latest   1f6ddc1b2547  3 weeks ago   77.8MB
```

## Creación de contenedor Docker-Compose:

Crea un archivo llamado `docker-compose.yml` y ábrelo con un editor de texto. Agrega el siguiente contenido al archivo `docker-compose.yml`:

```
C: > Users > Pablo > Desktop > docker > 📄 docker-compose.yml
1  version: '3'
2  services:
3    dhcp-server:
4      image: networkboot/dhcpd
5      restart: always
6      volumes:
7        - ./dhcpd.conf:/etc/dhcp/dhcpd.conf
8        - ./dhcpd.leases:/var/lib/dhcp/dhcpd.leases
9      ports:
10       - "67:67/udp"
11     networks:
12       - dhcp-network
13
14    dns-server:
15      image: bind
16      restart: always
17      volumes:
18        - ./named.conf:/etc/bind/named.conf
19        - ./zones:/etc/bind/zones
20      ports:
21       - "53:53/udp"
22     networks:
23       - dhcp-network
24
25  networks:
26    dhcp-network:
27      driver: bridge
28
```

Crea una carpeta llamada dhcp y dentro de ella crea un archivo llamado dhcpd.conf. Este archivo contendrá la configuración del servidor DHCP. Puedes personalizarlo según tus necesidades. Aquí tienes un ejemplo básico:

```
C: > Users > Pablo > Desktop > docker > dhcp > ⚙️ dhcpd.conf
1  default-lease-time 600;
2  max-lease-time 7200;
3  subnet 192.168.1.0 netmask 255.255.255.0 {
4      range 192.168.1.100 192.168.1.200;
5      option routers 192.168.1.1;
6      option domain-name-servers 192.168.1.10;
7  }
8
```

Crea una carpeta llamada dns y dentro de ella crea un archivo llamado named.conf. Este archivo contendrá la configuración del servidor DNS. Aquí tienes un ejemplo básico:

```
C: > Users > Pablo > Desktop > docker > dns > ⚙️ ConfiguracionDNS.conf
1  options {
2      directory "/etc/bind";
3      forwarders {
4          8.8.8.8;
5          8.8.4.4;
6      };
7  };
8
9  zone "DockerPractica.com" {
10     type master;
11     file "/etc/bind/zones/DockerPractica.com.zone";
12 };
13
14 zone "1.168.192.in-addr.arpa" {
15     type master;
16     file "/etc/bind/zones/1.168.192.zone";
17 };
18
```

Una vez completado todo esto crearemos 2 archivos ejemplo con las direcciones pertinentes y simplemente tendremos que hacer un **docker-compose up -d**

```
C:\Users\Pablo\Desktop\docker>docker-compose up -d
Creating network "docker_dhcp-network" with driver "bridge"
Pulling dhcp-server (networkboot/dhcpd)...
latest: Pulling from networkboot/dhcpd
d19f32bd9e41: Pull complete
1f23f0afd8a5: Pull complete
4928050fa1d6: Pull complete
7de3f1d09837: Pull complete
Digest: sha256:e99bbfbd6fb2f7e19fa4ce18aeebea02ab6377d6e909adc2b303a694edfb7219
Status: Downloaded newer image for networkboot/dhcpd:latest
Pulling dns-server (bind:)...

```

## Usos de Docker:

Una vez completado todo esto en mi opinión docker es muy útil ya que al usar solo una imagen y no el servicio entero, o evitarte la creación de una máquina virtual, ahorras mucho espacio de almacenamiento, además de poder tener varios servicios levantados al mismo tiempo.