## **Práctica en Clase**

1. Realizar la instalación de VirtualBox: <https://www.virtualbox.org/wiki/Downloads>
2. Realizar la instalación de Putty: <https://www.putty.org/> - <https://www.compuhoy.com/como-descargo-putty-en-linux/>
3. Realizar la instalación de WinSCP: <https://winscp.net/eng/download.php> (FileZilla es una alternativa si no usas sistema operativo Windows)
4. En el archivo "Servidor\_Ubuntu.zip" hay disponible un archivo VDI necesario para crear una máquina virtual Linux en VirtualBox. Esta máquina virtual es un servidor Ubuntu:  
   usuario: ubuntu  
   contraseña: ubuntu  
   Ya contiene una instalación de Docker. Por lo que a y b son opcionales:  
   a) Instalación de Docker en tu sistema operativo: <https://hub.docker.com/>b) Si el sistema operativo usado es Linux: <https://docs.docker.com/engine/install/>sudo apt install -y docker-compose
5. Comenzar a familiarizarse con los comandos de Linux:  
   [Tutorial] (<https://www.tutorialspoint.com/unix_commands/index.htm>)  
   [Interactivo] (<https://cli-boot.camp/?id=1dbj970vv4n>)

#### **Ejecutar en la consola el contenedor “hello-world” del Docker-Hub y luego verificar si está ejecutando:**

1. $ docker run hello-world (corro el contenedor hello-world)
2. $ docker ps (muestra los contenedores activos)

#### **Ejecutar una inspección de un contenedor específico**

1. $ docker ps -a (muestra todos los contenedores)
2. $ docker inspect <container ID> (muestra el detalle completo de un contenedor)
3. $ docker inspect <name> (igual que el anterior pero invocado con el nombre)

#### **Ejecutar la eliminación de un contenedor (usar rm y prune)**

1. $ docker rm <ID o nombre> (borro un contenedor)
2. $ docker container prune (borro todos lo contenedores que esten parados)
3. Explorar Docker Hub y probar ejecutar alguna de las imagenes. <https://hub.docker.com/>

### **11) Ejecutar la imagen “ubuntu”:**

$ docker run -it ubuntu (lo corre y entro al shell de ubuntu)  
-i: interactivo  
-t: abre la consola

corre el siguiente comando en la consola de linux $ cat /etc/lsb-release (veo la versión)

### **Ejecutar la imagen “nginx” y probar los comandos “stop” y “rm”**

1. $ docker run -d --name proxy nginx (corro un nginx)
2. $ docker stop proxy (apaga el contenedor)
3. $ docker rm proxy (borro el contenedor)
4. $ docker rm -f <contenedor> (lo para y lo borra)

### **12) Ejecutar nginx exponiendo el puerto 8080 de mi máquina**

1. Exponer contenedores:

$ docker run -d --name proxy -p 8081:80 nginx (corro un nginx y expongo el puerto 80 del contenedor en el puerto 8080 de mi máquina)

1. localhost:8081 (desde mi navegador compruebo que funcione)

### **13) Ejecutar el comando logs para ver los logs del contenedor de nginx:**

1. $ docker logs proxy (veo los logs)
2. $ docker logs -f proxy (hago un follow del log)

### **14) Ejecutar comando “logs –tail” para ver las últimas N entradas de log**

1. $ docker logs --tail 10 -f proxy (veo y sigo solo las 10 últimas entradas del log)

### **14) Elimina todo en Docker**

Sudo Docker rmi -f $(sudo Docker images -aq)