

# GESTIÓN DE IMÁGENES Y PERSISTENCIA

## Parte 1 - Nextcloud

Nextcloud es una aplicación web que permite crear una nube privada en la cual ofrecer algunos servicios como almacenamiento de archivos, calendario, etc.

1. Busca y descarga, sin crear un contenedor, la distribución oficial de Docker de la imagen Nextcloud, concretamente la versión 30.0.8 que utiliza apache. Muestra cómo lo has hecho.

1. Para descargar la imagen Nextcloud versión 30.0.8 con Apache, se utiliza el siguiente comando en la terminal: **docker pull nextcloud:30.0.8-apache**

```
C:\Users\solia> docker pull nextcloud:30.0.8-apache
30.0.8-apache: Pulling from library/nextcloud
6e909acdb790: Already exists
5db2c4b6137b: Pull complete
dacb60b59038: Pull complete
64450047668b: Pull complete
4d6386e035f7: Pull complete
e4ca7ebe0914: Pull complete
3dc5d9089396: Pull complete
1778f52baa09: Pull complete
a5d9cb1c80ec: Pull complete
b116c8459f57: Pull complete
8f8dda3587a: Pull complete
f432ea27fc70: Pull complete
6a5a37a900f3: Pull complete
4f4fb700ef54: Pull complete
02c1344e00ef: Pull complete
fdbff194bf0c: Pull complete
e5fdceebbfb5: Pull complete
7639f2a1ec44: Pull complete
5f8171920ece: Pull complete
db90c97c0ba6: Pull complete
e0c0861b3193: Pull complete
2719dd8b3581: Pull complete
Digest: sha256:c3d8804b06988205ccb4e102b254cf9f987712d96a23c320003e56bbca90663
Status: Downloaded newer image for nextcloud:30.0.8-apache
docker.io/library/nextcloud:30.0.8-apache
```

2. Para ver si la imagen se haya descargado correctamente: **docker images**

```
C:\Users\solia> docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
nextcloud            30.0.8-apache      e76c48e387cc       13 days ago        1.28GB
httpd                latest             83d938198316       2 months ago       148MB
hello-world          latest             74cc54e27dc4       2 months ago       10.1kB
fauria/vsftpd        latest             9bfb39139661       2 years ago        394MB
```

2. Crea un contenedor con las siguientes características:

- La imagen de nextcloud descargada
- Contenedor de tipo demonio
- Vincula el puerto 80 del host con el puerto 80 del contenedor
- Crea un volumen Docker vinculado a la carpeta que recomienda la documentación de nextcloud con nombre “**tunombre-nextvolume**”

### e. Ponle de nombre tunombre-nextcloud

```
C:\Users\solia>docker run -d --name sofia-nextcloud -p 80:80 -v sofia-nextvolume:/var/www/html nextcloud:30.0.8-apache 4bfbb75fec73ce2bd3e5fcfd3f2eae97c4f248bd3c614368c7a3e0fe5d7567e
```

sofia-nextcloud 4bfbb75fec73 nextcloud:30.0.8-apache 80:80 0.01% 5 minutes ago

Comando: **docker run -d --name sofia-nextcloud -p 80:80 -v sofia-nextvolume:/var/www/html nextcloud:30.0.8-apache**

Explica los parámetros del comando utilizado:

- **docker run**: Ejecuta un contenedor.
- **-d**: Ejecuta el contenedor en segundo plano (modo demonio).
- **--name sofia-nextcloud**: Asigna el nombre sofia-nextcloud al contenedor.
- **-p 80:80**: Mapea el puerto 80 del host al puerto 80 del contenedor.
- **-v sofia-nextvolume:/var/www/html**: Crea un volumen llamado sofia-nextvolume vinculado a la carpeta /var/www/html dentro del contenedor.
- **nextcloud:30.0.8-apache**: Usa la imagen descargada de Nextcloud.

2. Para verificar que el contenedor está corriendo: **docker ps**

```
C:\Users\solia>docker ps
CONTAINER ID   IMAGE                  COMMAND                  CREATED        STATUS        PORTS                    NA
4bfbb75fec73   nextcloud:30.0.8-apache "/entrypoint.sh apac...  5 minutes ago Up 5 minutes   0.0.0.0:80->80/tcp      so
fia-nextcloud
```

3. Entra en la aplicación desde el navegador del host e instala la VPC (Virtual Private Cloud) utilizando SQLite, tu nombre y la contraseña '1234'.

Buscando "http://localhost" en el buscador del navegador, debe aparecer esto:

Crear una **cuenta de administrador**

Nuevo nombre para la cuenta de administrador

Nueva contraseña de administrador

Almacenamiento y base de datos ▾

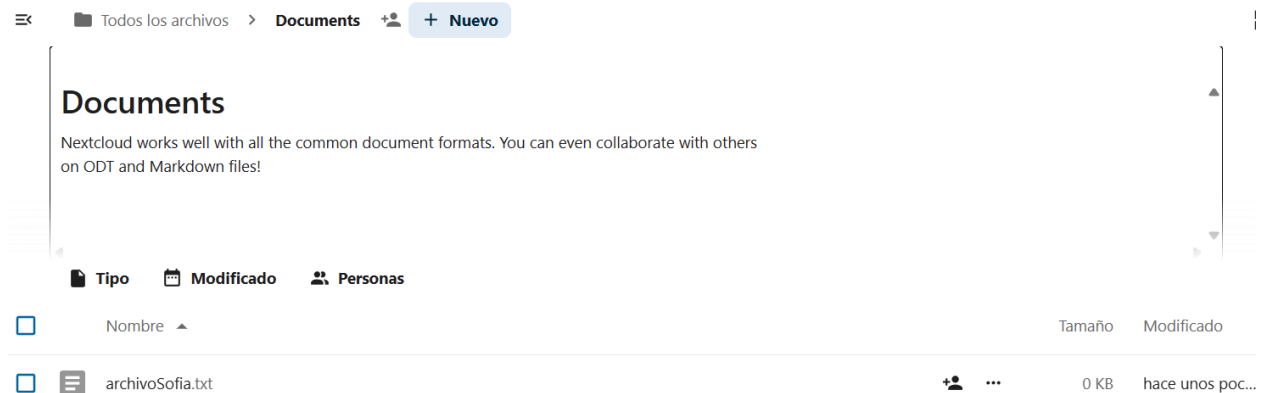
Carpeta de datos

Configurar la base de datos

SQLite **MySQL/MariaDB** PostgreSQL

4. Ve a la sección de documentos y sube un archivo. Después, desde el terminal, elimina el contenedor, vuelve a crearlo con el mismo volumen (puedes utilizar el mismo comando que antes) y comprueba si el archivo sigue ahí. Muestra los comandos utilizados.

1. Desde la interfaz web de Nextcloud, en la sección Archivos, subir cualquier archivo:



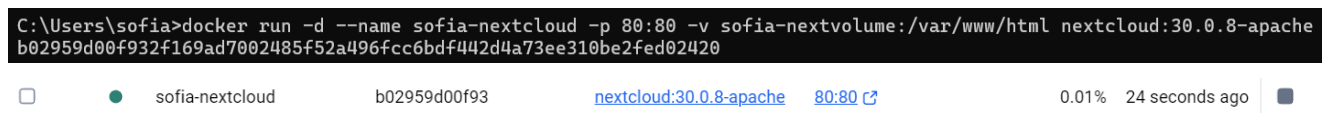
2. Desde la terminal, eliminar el contenedor: **docker rm -f sofia-nextcloud**

```
C:\Users\sofia>docker rm -f sofia-nextcloud
sofia-nextcloud

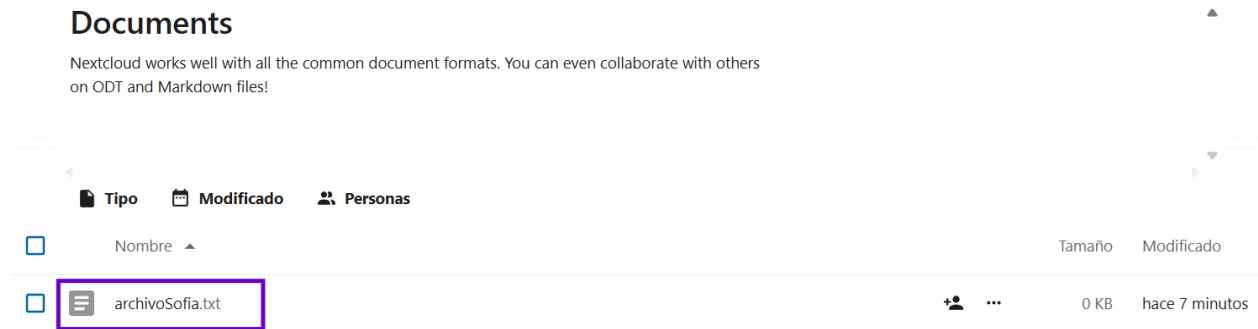
C:\Users\sofia>
```

3. Después, volver a crear el contenedor con el mismo comando que antes:

**docker run -d --name sofia-nextcloud -p 80:80 -v sofia-nextvolume:/var/www/html nextcloud:30.0.8-apache**




4. Volver a buscar “<http://localhost>” para ver si el archivo subido sigue ahí:



## Parte 2 - Entorno de desarrollo y uso de bibliotecas Python

En esta parte vamos a desplegar un entorno de desarrollo de Python capaz de ejecutar código que utiliza bibliotecas específicas.

1. Crea una carpeta en el escritorio con nombre “**tunombre-python**”:

 sofia-python

02/04/2025 16:36

2. Crea (sin ejecutar) un contenedor con las siguientes características:

- No es de tipo demonio
- Utiliza la imagen `python:3.11-slim`
- Interactuable desde el terminal
- Crea un bind mount que vincule la carpeta que creaste en el punto anterior con la carpeta del contenedor `/usr/src/app`
- Define `/usr/src/app` como directorio de trabajo del contenedor
- Ponle el nombre al contenedor ‘`tunombre-entornopython`’

Comando: **docker create -it --name sofia-entornopython**

**-v "C:\Users\sofia\Desktop\sofia-python:/usr/src/app" -w /usr/src/app python:3.11-slim**

- **docker create**: Crea un contenedor (sin ejecutarlo todavía).
- **-it**: Permite interactuar con el contenedor desde la terminal.
- **--name sofia-entornopython**: Le da el nombre `sofia-entornopython`.
- **-v "C:\Users\sofia\Desktop\sofia-python:/usr/src/app"**: Conecta la carpeta `sofia-python` con `/usr/src/app` dentro del contenedor.
- **-w /usr/src/app**: Establece `/usr/src/app` como directorio de trabajo dentro del contenedor.
- **python:3.11-slim**: Usa la imagen de Python 3.11 en su versión ligera.

```
PS C:\Users\sofia> docker create -it --name sofia-entornopython -v "C:\Users\sofia\Desktop\sofia-python:/usr/src/app" -w /usr/src/app python:3.11-slim
3a84b618f15f213a31a3110a4e97a570601a3e032861a226916b4ceb3051a3ac
PS C:\Users\sofia>
```

### 3. Arranca el contenedor:

Comando: **docker start sofia-entornopython**

```
PS C:\Users\sofia> docker start sofia-entornopython
sofia-entornopython
```

sofia-entornopython cf8f084765cc python:3.11-slim 0% 26 seconds ago

### 4. Conéctate al terminal del contenedor

Comando: **docker exec -it sofia-entornopython bash**

```
PS C:\Users\sofia> docker exec -it sofia-entornopython bash
root@69f35c22f45d:/usr/src/app#
```

### 5. Descarga la librería de Python ‘NumPy’

Dentro de la terminal del contenedor, utilizar el comando: **pip install numpy**

```
root@69f35c22f45d:/usr/src/app# pip install numpy
Collecting numpy
  Downloading numpy-2.2.4-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (62 kB)
     |-----| 62.0/62.0 kB 1.0 MB/s eta 0:00:00
  Downloading numpy-2.2.4-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (16.4 MB)
     |-----| 16.4/16.4 MB 7.4 MB/s eta 0:00:00
Installing collected packages: numpy
Successfully installed numpy-2.2.4
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is recommended to use a virtual environment instead: https://pip.pypa.io/warnings/venv

[notice] A new release of pip is available: 24.0 -> 25.0.1
[notice] To update, run: pip install --upgrade pip
root@69f35c22f45d:/usr/src/app#
```

### 6. Crea en la carpeta vinculada del host un archivo .py

Crear un archivo “script.py” dentro de la carpeta sofia-python con el siguiente código:

```

import numpy as np

# Crear un array de 10 números aleatorios entre 0 y 100
arr = np.random.randint(0, 100, size=10)
print("Array aleatorio:", arr)

# Obtener estadísticas básicas
print("Media:", np.mean(arr))
print("Mediana:", np.median(arr))
print("Desviación estándar:", np.std(arr))
print("Mínimo:", np.min(arr), "Máximo:", np.max(arr))

# Ordenar el array
arr_sorted = np.sort(arr)
print("Array ordenado:", arr_sorted)

# Crear una matriz 3x3 con valores aleatorios entre 0 y 10
matrix = np.random.randint(0, 10, size=(3, 3))
print("\nMatriz aleatoria 3x3:\n", matrix)

# Sumar y multiplicar filas y columnas
print("Suma por filas:", np.sum(matrix, axis=1))
print("Suma por columnas:", np.sum(matrix, axis=0))

# Producto punto de la matriz consigo misma
matrix_product = np.dot(matrix, matrix)
print("Producto punto:\n", matrix_product)

```

## 7. Ejecuta el archivo en el contenedor y muestra el resultado

Comando dentro de la terminal del contenedor: **python script.py**

```

root@3a84b618f15f:/usr/src/app# python script.py
Array aleatorio: [12 14 83 27 41 50 69 29 70 29]
Media: 42.4
Mediana: 35.0
Desviación estándar: 23.461457755220582
Mínimo: 12 Máximo: 83
Array ordenado: [12 14 27 29 29 41 50 69 70 83]

Matriz aleatoria 3x3:
[[0 9 6]
 [6 4 2]
 [9 9 6]]
Suma por filas: [15 12 24]
Suma por columnas: [15 22 14]
Producto punto:
[[108 90 54]
 [ 42 88 56]
 [108 171 108]]
root@3a84b618f15f:/usr/src/app#

```