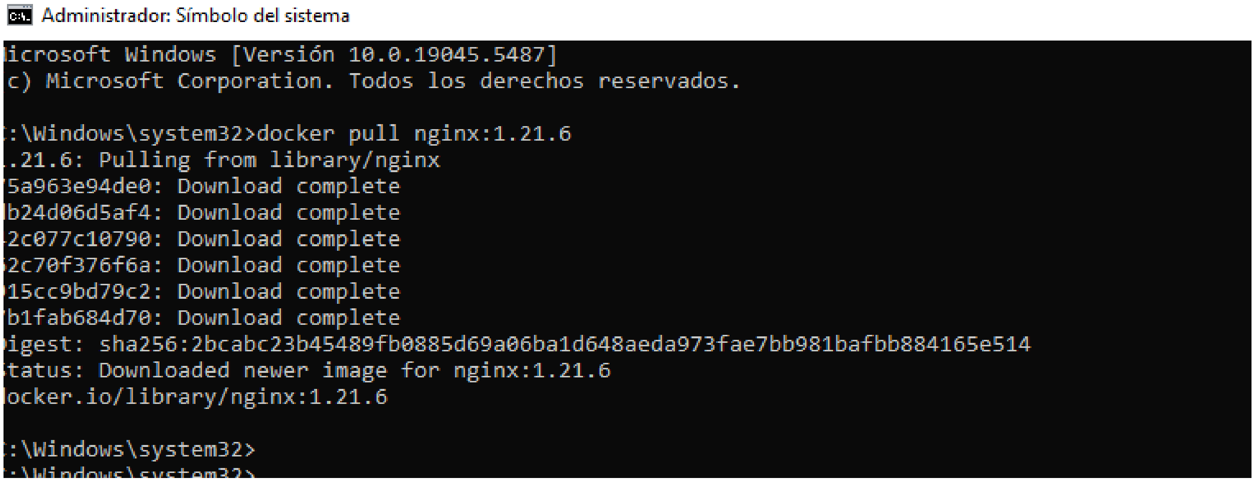
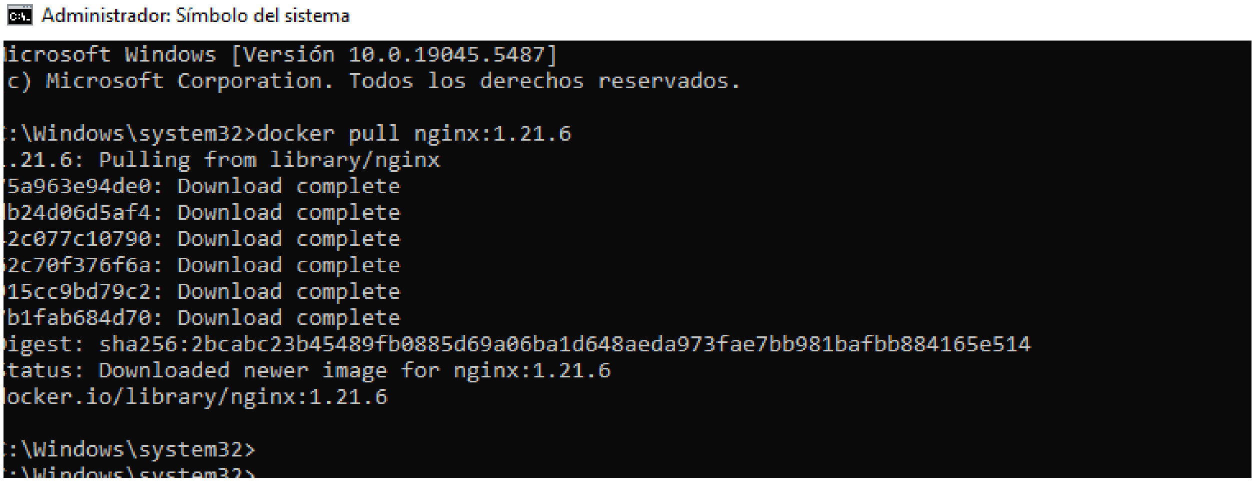
1. **Descarga de una imagen:**
   Utiliza comando docker pull para descargar imagen oficial de Nginx desde Docker Hub.



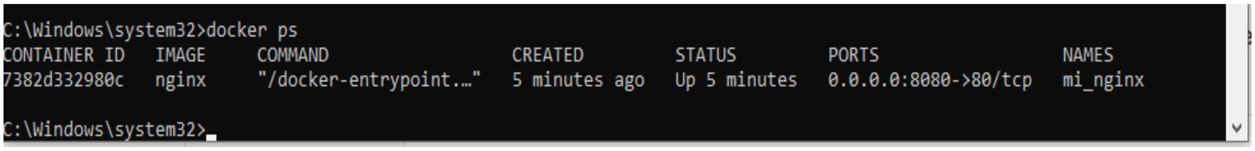
```
icrosoft Windows [Versión 10.0.19045.5487]
c) Microsoft Corporation. Todos los derechos reservados.

:\Windows\system32>docker pull nginx:1.21.6
.21.6: Pulling from library/nginx
5a963e94de0: Download complete
b24d06d5af4: Download complete
2c077c10790: Download complete
2c70f376f6a: Download complete
15cc9bd79c2: Download complete
b1fab684d70: Download complete
igest: sha256:2bcabc23b45489fb0885d69a06ba1d648aeda973fae7bb981bafbb884165e514
tatus: Downloaded newer image for nginx:1.21.6
ocker.io/library/nginx:1.21.6

:\Windows\system32>
:\Windows\system32>
```

2.**Ejecución de un contenedor**
Utiliza el comando docker run para crear y ejecutar un contenedor basado en la imagen de Nginx.
Mapea el puerto 80 del contenedor al puerto 8080 de tu maquina local.

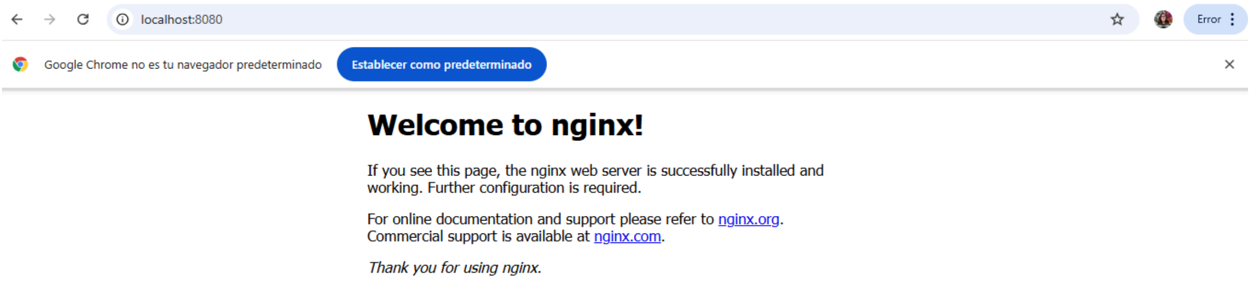


```
icrosoft Windows [Versión 10.0.19045.5487]
c) Microsoft Corporation. Todos los derechos reservados.

:\Windows\system32>docker pull nginx:1.21.6
.21.6: Pulling from library/nginx
5a963e94de0: Download complete
b24d06d5af4: Download complete
2c077c10790: Download complete
2c70f376f6a: Download complete
15cc9bd79c2: Download complete
b1fab684d70: Download complete
igest: sha256:2bcabc23b45489fb0885d69a06ba1d648aeda973fae7bb981bafbb884165e514
tatus: Downloaded newer image for nginx:1.21.6
ocker.io/library/nginx:1.21.6

:\Windows\system32>
:\Windows\system32>
```

3.**Verificación del contenedor:**
Utiliza comando docker ps para verificar que el contenedor se esta ejecutando.
Abre tu navegador y accede a http://localhost: 8080 para comprobar que Nginx esta sirviendo la pagina por defecto.



```
C:\Windows\system32>docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED         STATUS         PORTS                  NAMES
7382d332980c   nginx     "/docker-entrypoint.…"   5 minutes ago   Up 5 minutes   0.0.0.0:8080->80/tcp   mi_nginx

C:\Windows\system32>_
```


localhost:8080
Error
Google Chrome no es tu navegador predeterminado
Establecer como predeterminado

Welcome to nginx!

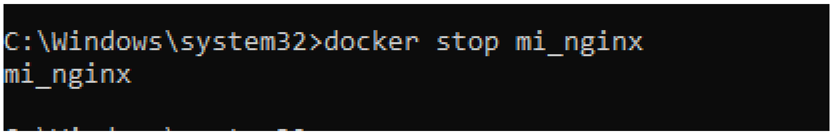
If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

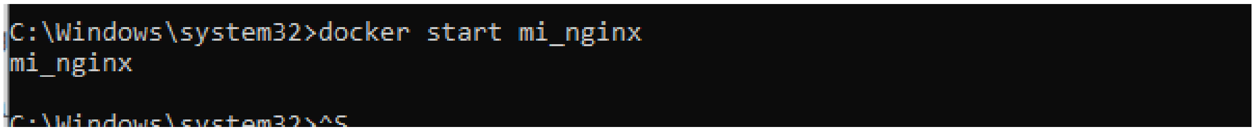
Thank you for using nginx.

4 .Detención y Reinicio del Contenedor:

Utiliza el comando docker stop para detener el contenedor.
Utiliza el comando Docker start para volver a iniciar el contenedor.



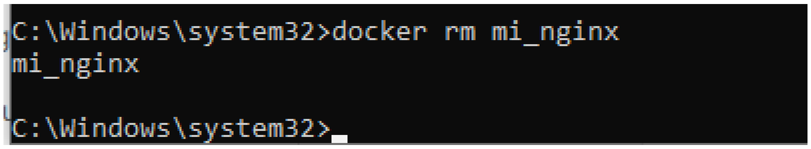
```
C:\Windows\system32>docker stop mi_nginx
mi_nginx
```



```
C:\Windows\system32>docker start mi_nginx
mi_nginx

C:\Windows\system32>^S
```

5 .Eliminación del Contenedor:

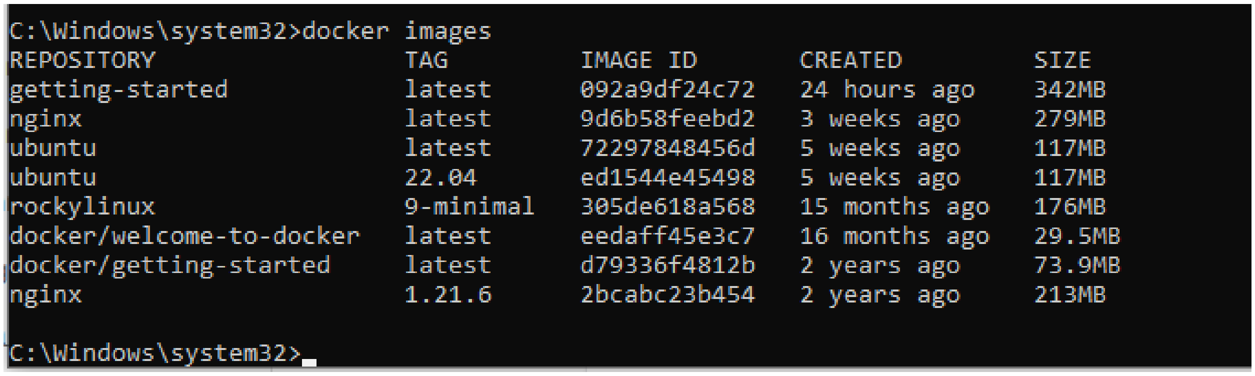
Utiliza el comando docker rm para eliminar el contenedor.



```
C:\Windows\system32>docker rm mi_nginx
mi_nginx

C:\Windows\system32>_
```

6.Listado de imagenes:

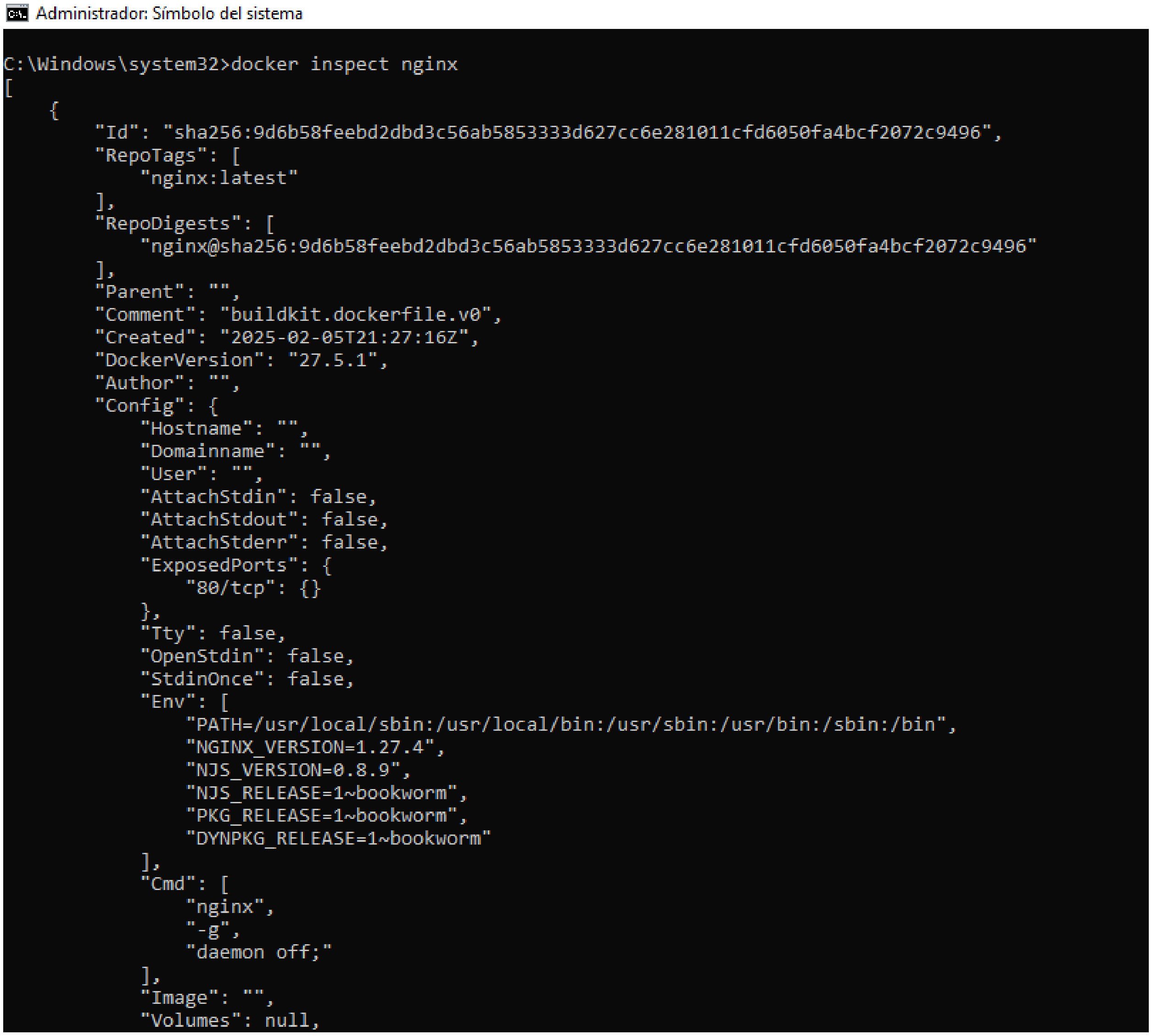
Utiliza el comando docker images para listar las imagenes Docker disponibles en tu maquina local.



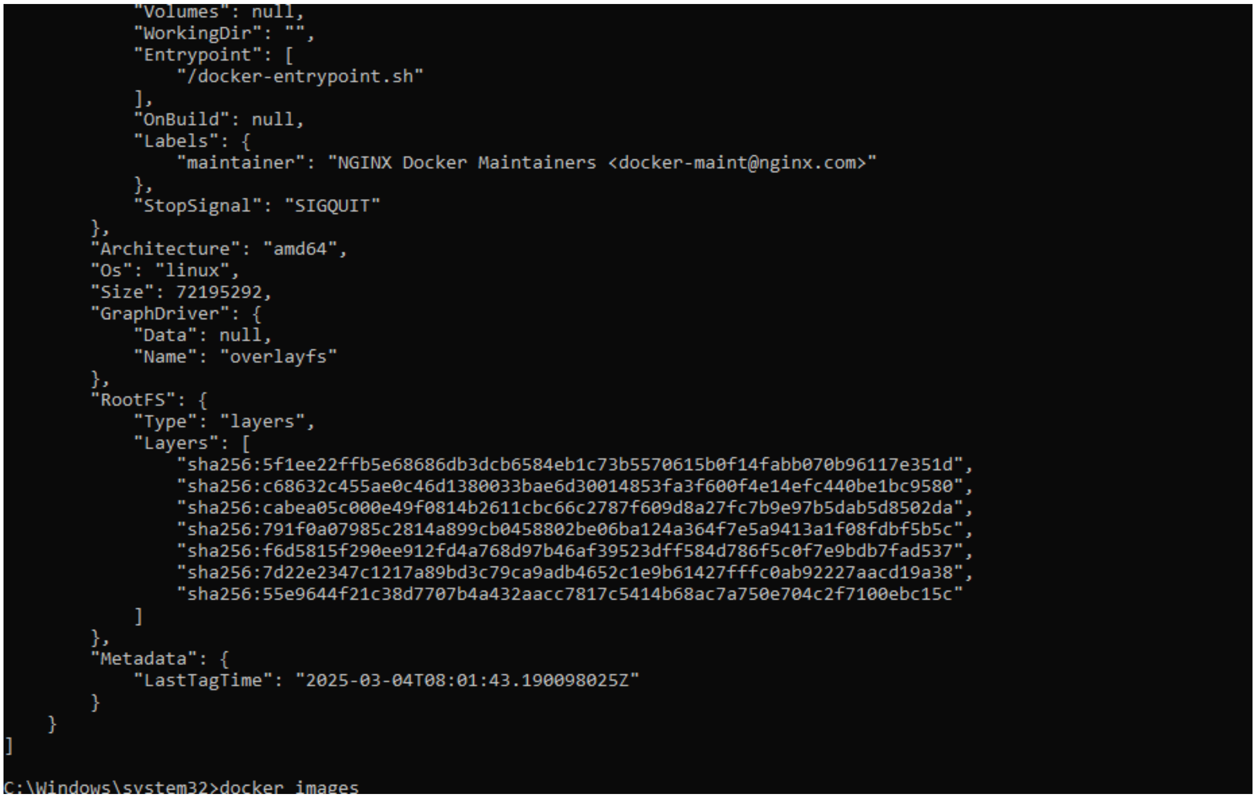
```
C:\Windows\system32>docker images
REPOSITORY                TAG         IMAGE ID       CREATED         SIZE
getting-started           latest      092a9df24c72   24 hours ago    342MB
nginx                     latest      9d6b58feebd2   3 weeks ago     279MB
ubuntu                    latest      72297848456d   5 weeks ago     117MB
ubuntu                    22.04       ed1544e45498   5 weeks ago     117MB
rockylinux                9-minimal   305de618a568   15 months ago   176MB
docker/welcome-to-docker  latest      eedaff45e3c7   16 months ago   29.5MB
docker/getting-started    latest      d79336f4812b   2 years ago     73.9MB
nginx                     1.21.6      2bcabc23b454   2 years ago     213MB

C:\Windows\system32>_
```

7.Información sobre una imagen:

Utiliza el comando docker inspect para obtener informacion detallada sobre la imagen de Nginx (puedes utilizar el id o el nombre de la imagen).

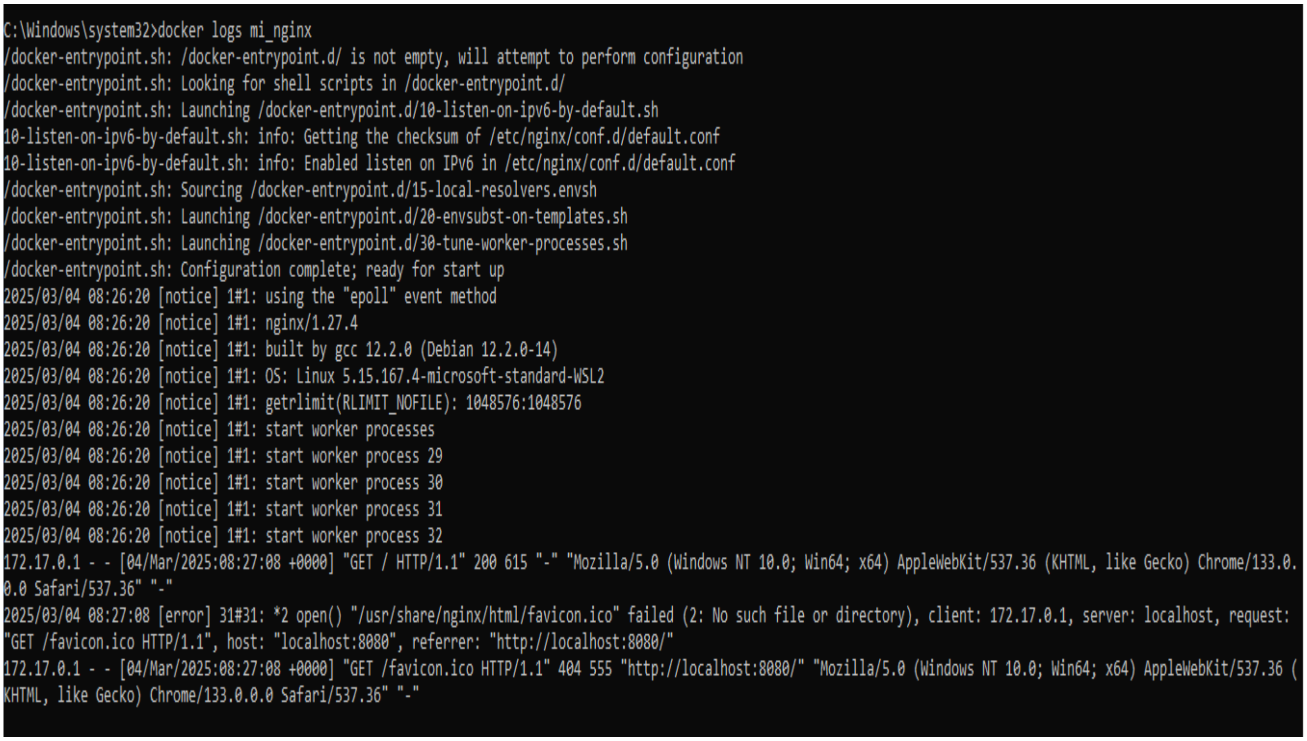


```
C:\Windows\system32>docker inspect nginx
[
    {
        "Id": "sha256:9d6b58feebd2dbd3c56ab5853333d627cc6e281011cfd6050fa4bcf2072c9496",
        "RepoTags": [
            "nginx:latest"
        ],
        "RepoDigests": [
            "nginx@sha256:9d6b58feebd2dbd3c56ab5853333d627cc6e281011cfd6050fa4bcf2072c9496"
        ],
        "Parent": "",
        "Comment": "buildkit.dockerfile.v0",
        "Created": "2025-02-05T21:27:16Z",
        "DockerVersion": "27.5.1",
        "Author": "",
        "Config": {
            "Hostname": "",
            "Domainname": "",
            "User": "",
            "AttachStdin": false,
            "AttachStdout": false,
            "AttachStderr": false,
            "ExposedPorts": {
                "80/tcp": {}
            },
            "Tty": false,
            "OpenStdin": false,
            "StdinOnce": false,
            "Env": [
                "PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin",
                "NGINX_VERSION=1.27.4",
                "NJS_VERSION=0.8.9",
                "NJS_RELEASE=1~bookworm",
                "PKG_RELEASE=1~bookworm",
                "DYNPKG_RELEASE=1~bookworm"
            ],
            "Cmd": [
                "nginx",
                "-g",
                "daemon off;"
            ],
            "Image": "",
            "Volumes": null,
```



```
            "Volumes": null,
            "WorkingDir": "",
            "Entrypoint": [
                "/docker-entrypoint.sh"
            ],
            "OnBuild": null,
            "Labels": {
                "maintainer": "NGINX Docker Maintainers <docker-maint@nginx.com>"
            },
            "StopSignal": "SIGQUIT"
        },
        "Architecture": "amd64",
        "Os": "linux",
        "Size": 72195292,
        "GraphDriver": {
            "Data": null,
            "Name": "overlayfs"
        },
        "RootFS": {
            "Type": "layers",
            "Layers": [
                "sha256:5f1ee22ffb5e68686db3dcb6584eb1c73b5570615b0f14fabb070b96117e351d",
                "sha256:c68632c455ae0c46d1380033bae6d30014853fa3f600f4e14efc440be1bc9580",
                "sha256:cabea05c000e49f0814b2611cbc66c2787f609d8a27fc7b9e97b5dab5d8502da",
                "sha256:791f0a07985c2814a899cb0458802be06ba124a364f7e5a9413a1f08fdbf5b5c",
                "sha256:f6d5815f290ee912fd4a768d97b46af39523dff584d786f5c0f7e9bdb7fad537",
                "sha256:7d22e2347c1217a89bd3c79ca9adb4652c1e9b61427fffc0ab92227aacd19a38",
                "sha256:55e9644f21c38d7707b4a432aacc7817c5414b68ac7a750e704c2f7100ebc15c"
            ]
        },
        "Metadata": {
            "LastTagTime": "2025-03-04T08:01:43.190098025Z"
        }
    }
]

C:\Windows\system32>docker images
```

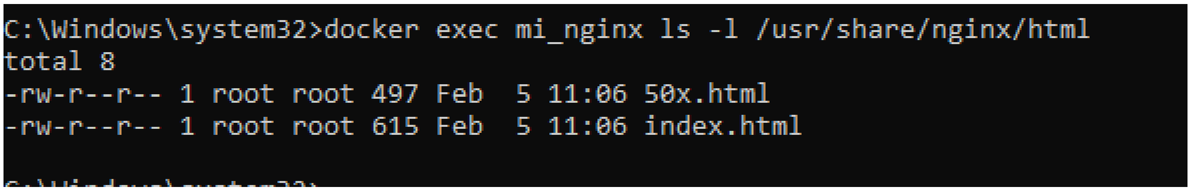
9.**Visualización de logs:**
Utiliza el comando docker logs para ver los logs del contenedor Nginx.
Comando utilizado: **docker logs mi_nginx**



```
C:\Windows\system32>docker logs mi_nginx
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
/docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
/docker-entrypoint.sh: Configuration complete; ready for start up
2025/03/04 08:26:20 [notice] 1#1: using the "epoll" event method
2025/03/04 08:26:20 [notice] 1#1: nginx/1.27.4
2025/03/04 08:26:20 [notice] 1#1: built by gcc 12.2.0 (Debian 12.2.0-14)
2025/03/04 08:26:20 [notice] 1#1: OS: Linux 5.15.167.4-microsoft-standard-WSL2
2025/03/04 08:26:20 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2025/03/04 08:26:20 [notice] 1#1: start worker processes
2025/03/04 08:26:20 [notice] 1#1: start worker process 29
2025/03/04 08:26:20 [notice] 1#1: start worker process 30
2025/03/04 08:26:20 [notice] 1#1: start worker process 31
2025/03/04 08:26:20 [notice] 1#1: start worker process 32
172.17.0.1 - - [04/Mar/2025:08:27:08 +0000] "GET / HTTP/1.1" 200 615 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/133.0.
0.0 Safari/537.36" "-"
2025/03/04 08:27:08 [error] 31#31: *2 open() "/usr/share/nginx/html/favicon.ico" failed (2: No such file or directory), client: 172.17.0.1, server: localhost, request:
"GET /favicon.ico HTTP/1.1", host: "localhost:8080", referrer: "http://localhost:8080/"
172.17.0.1 - - [04/Mar/2025:08:27:08 +0000] "GET /favicon.ico HTTP/1.1" 404 555 "http://localhost:8080/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (
KHTML, like Gecko) Chrome/133.0.0.0 Safari/537.36" "-"
```

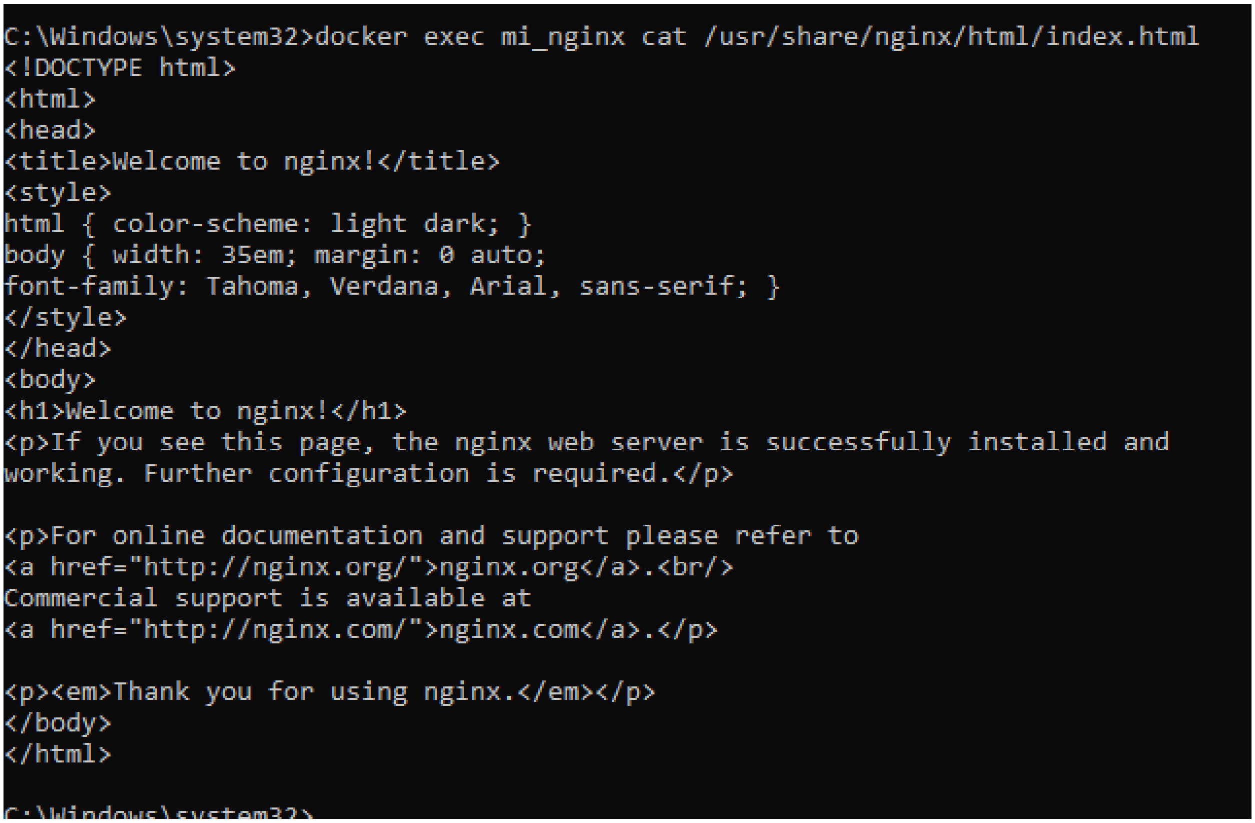
10. Ejecución de un comando dentro del contenedor

Utiliza el comando docker exec para ejecutar un comando simple dentro del contenedor Nginx (por ejemplo ls -l para listar los archivos en el directorio /usr/share/nginx/html).



```
C:\Windows\system32>docker exec mi_nginx ls -l /usr/share/nginx/html
total 8
-rw-r--r-- 1 root root 497 Feb  5 11:06 50x.html
-rw-r--r-- 1 root root 615 Feb  5 11:06 index.html

C:\Windows\system32>
```

Otro comando que podemos utilizar también para ver el archivo html es:
docker exec mi_nginx cat /usr/share/nginx/html/index.html



```
C:\Windows\system32>docker exec mi_nginx cat /usr/share/nginx/html/index.html
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif; }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>

C:\Windows\system32>
```