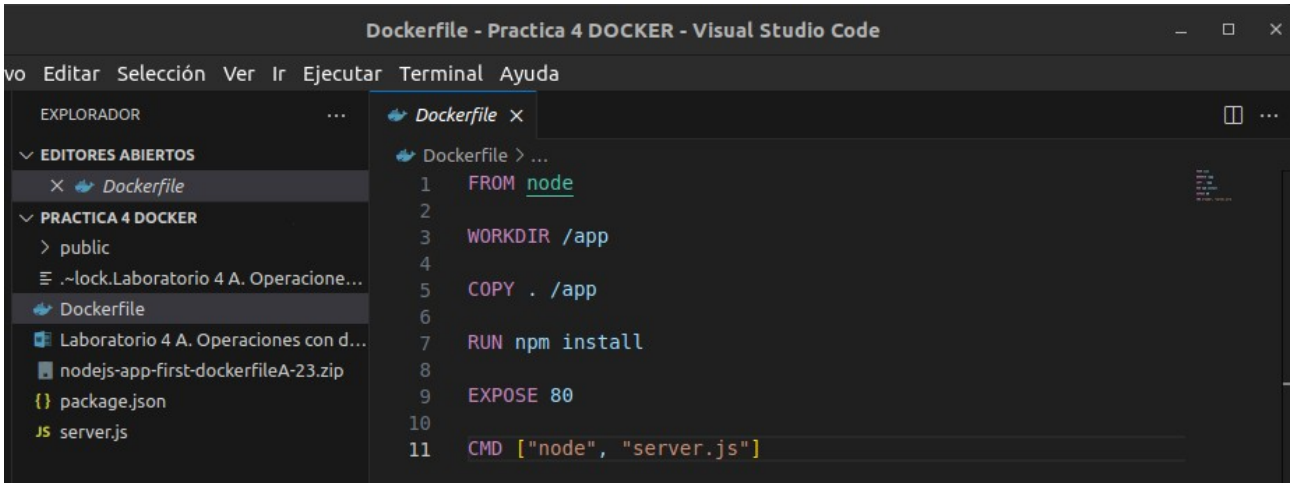
Laboratorio 4A.
Crear un archivo Dockerfile. Operaciones sobre contenedores.

Paso 1. Descargamos la carpeta comprimida con los archivos.
Creamos el **Dockerfile** sin la extensión.

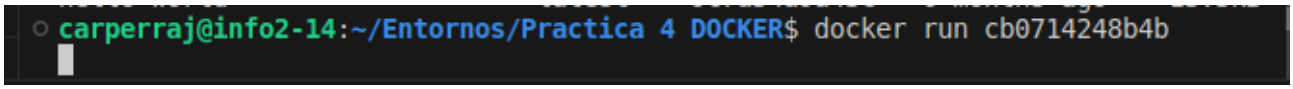


Paso 2. Creamos la imagen sobre el Dockerfile:
Utilizamos el comando ya conocido:

docker build . (dejamos un espacio y colocamos un punto)

```
carperraj@info2-14:~/Entornos/Practica 4 DOCKER$ docker build .
[+] Building 51.1s (10/10) FINISHED
 => [internal] load build definition from Dockerfile                    0.1s
 => => transferring dockerfile: 128B                                    0.0s
 => [internal] load .dockerignore                                       0.1s
 => => transferring context: 2B                                         0.0s
 => [internal] load metadata for docker.io/library/node:latest          2.1s
 => [auth] library/node:pull token for registry-1.docker.io             0.0s
 => [1/4] FROM docker.io/library/node@sha256:84bb4077fd52933a935e7057b  44.0s
 => => resolve docker.io/library/node@sha256:84bb4077fd52933a935e7057ba  0.0s
 => => sha256:07308918cc5b9f838e11922e6626826b380730bc8 7.53kB / 7.53kB  0.0s
 => => sha256:27e1a8ca91d35598fbae8dee7f1c211f0f93cec 24.05MB / 24.05MB  4.9s
 => => sha256:d3a767d1d12e57724b9f254794e359f3b04d4d 64.13MB / 64.13MB  21.5s
 => => sha256:84bb4077fd52933a935e7057ba9991e7cb18487b0 1.21kB / 1.21kB  0.0s
 => => sha256:06facfb3aace8ca5f47e2895901e4f1bbea25f9b1 2.00kB / 2.00kB  0.0s
 => => sha256:90e5e7d8b87a34877f61c2b86d053db1c4f440 49.58MB / 49.58MB  15.0s
 => => sha256:711be5dc50448ab08ccab0b44d65962f3657 211.07MB / 211.07MB  37.5s
 => => sha256:22956530cc64ef2361591684e23e3b8e5bb5910d 3.37kB / 3.37kB  15.2s
 => => extracting sha256:90e5e7d8b87a34877f61c2b86d053db1c4f440b9054cf4  1.0s
 => => sha256:f72e9eb5a1b39e6b6969a2b02e57c8be809fac 49.33MB / 49.33MB  24.9s
 => => extracting sha256:27e1a8ca91d35598fbae8dee7f1c211f0f93cec529f680  0.3s
 => => extracting sha256:d3a767d1d12e57724b9f254794e359f3b04d4d5ad96600  1.4s
 => => sha256:96d22460c6db052a986434847a4a217967597989 2.23MB / 2.23MB  22.4s
 => => sha256:4e772c30a8d40e1deba91b6961381e3b931fa4839cc4 451B / 451B  22.6s
 => => extracting sha256:711be5dc50448ab08ccab0b44d65962f36574d341749ab  3.5s
 => => extracting sha256:22956530cc64ef2361591684e23e3b8e5bb5910da23197  0.0s
 => => extracting sha256:f72e9eb5a1b39e6b6969a2b02e57c8be809fac89205a98  1.2s
 => => extracting sha256:96d22460c6db052a986434847a4a217967597989687e0b  0.0s
 => => extracting sha256:4e772c30a8d40e1deba91b6961381e3b931fa4839cc4ca  0.0s
 => [internal] load build context                                       0.1s
 => => transferring context: 310.29kB                                   0.0s
 => [2/4] WORKDIR /app                                                  0.0s
 => [3/4] COPY . /app                                                   0.0s
 => [4/4] RUN npm install                                               3.9s
 => exporting to image                                                  1.1s
 => => exporting layers                                                 1.1s
 => => writing image sha256:cb0714248b4b3d263574e848cb1ae3facd8908393ab  0.0s
carperraj@info2-14:~/Entornos/Practica 4 DOCKER$
```

Podemos observar los pasos que hemos especificado en el Dockerfile, se han ejecutado sin errores.
Paso 3. Buscamos el nombre de la imagen y creamos una instancia un proceso, es decir un contendor. Por lo tanto, copiamos el ID de la imagen y en la terminal de visual estudio code escribimos:

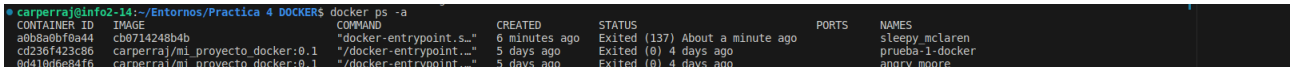


El contenedor no se para. El motivo es que le hemos indicado esta instrucción de start un nuevo servidor:

Paso 5. Para ver todos los procesos activos, en la terminal nueva escribimos el comando docker ps -a
Ahora el siguiente paso es parar nuestro contenedor. Lo haremos con el comando

docker stop nombre del contenedor o ID

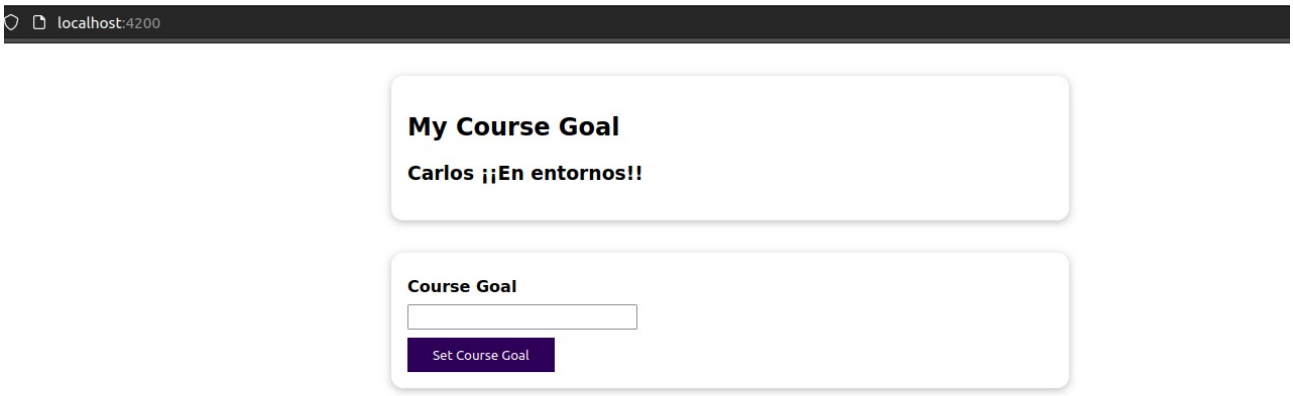
Y después veremos el comando ***exited*** que significa el proceso finalizado.

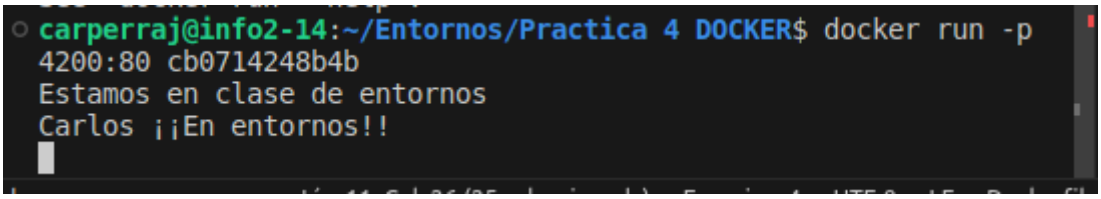


Paso 6. Si buscamos ver la respuesta en nuestro localhost 80, el resultado nos drá un error. Necesitamos realizar unos pasos opcionales. Volver a ejecutar el contenedor, pero necesitamos agregar una bandera adicional. Es decir:

Añadimos la *bandera -p delante del nombre contenedor.*

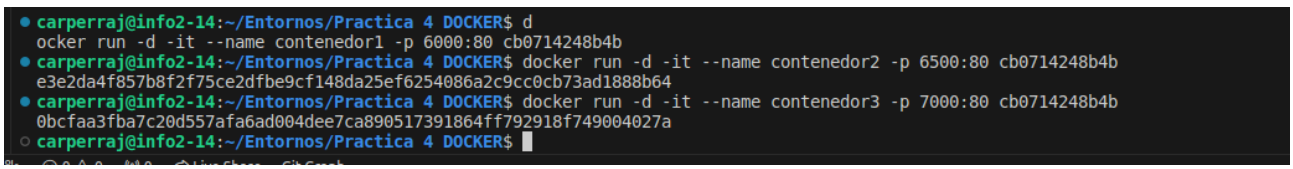
Paso 7. Cada alumno debe colocar un mensaje personalizado en la categoría **"My Course Goal".** En otras palabras, una frase, y también realizar una captura de pantalla demostrando que el código y la configuración están funcionando correctamente.






```
docker run -d -it --name contenedor1 -p 6000:puerto_interno nombre_de_tu_imagen
docker run -d -it --name contenedor2 -p 6500:puerto_interno nombre_de_tu_imagen
docker run -d -it --name contenedor3 -p 7000:puerto_interno nombre_de_tu_imagen
```

Asegúrate de reemplazar "puerto_interno" ***busca en dockerhub*** ***https://hub.docker.com/layers/library/node/lts-slim/images/sha256-18aacc7993a16f1d766c21e3bff922e830bcdc7b549bbb789ceb7374a6138480?context=explore*** ***, o en*** ***https://es.stackoverflow.com/search?q=pue rto+expose+node*** ***el puerto de la imagen! con el número o de puerto que tu aplicación dentro del contenedor está utilizando.***



```
carperraj@info2-14:~/Entornos/Practica 4 DOCKER$ docker ps -a
CONTAINER ID   IMAGE                  COMMAND                  CREATED          STATUS                      PORTS                                       NAMES
0bcfaa3fba7c   cb0714248b4b           "docker-entrypoint.s…"   45 seconds ago   Up 44 seconds               0.0.0.0:7000->80/tcp, :::7000->80/tcp       contenedor3
e3e2da4f857b   cb0714248b4b           "docker-entrypoint.s…"   58 seconds ago   Up 57 seconds               0.0.0.0:6500->80/tcp, :::6500->80/tcp       contenedor2
caaffb09f5b8   cb0714248b4b           "docker-entrypoint.s…"   2 minutes ago    Up 2 minutes                0.0.0.0:6000->80/tcp, :::6000->80/tcp       contenedor1
```

Paso B2. Cambiar Nombre del Contenedor:

``` **terminal**
docker rename contenedor1 nuevo_nombre_contenedor1
```

```
caaffb09f5b8   cb0714248b4b   "docker-entrypoint.s…"   3 minutes ago   Up 3 minutes   0.0.0.0:6000->80/tcp, :::6000->80/tcp   contenedor_carlos
```

Paso B3. **Adjuntar un Archivo al Contenedor en Ejecución:**

``` **terminal**
docker cp archivo_adjunto.txt contenedor2:/ruta/destino
```

```
Successfully copied 2.56kB to contenedor2:/Entornos/Practica_4_DOCKER
carperraj@info2-14:~/Entornos/Practica 4 DOCKER$ le /Entornos in container contenedor2
carperraj@info2-14:~/Entornos/Practica 4 DOCKER$
```

Paso B4. **Detener uno de los Contenedores:**

``` **terminal**
docker stop contenedor3
```

```
carperraj@info2-14:~/Entornos/Practica 4 DOCKER$ docker stop contenedor3
contenedor3
carperraj@info2-14:~/Entornos/Practica 4 DOCKER$ docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED          STATUS                      PORTS     NAMES
0bcfaa3fba7c   cb0714248b4b   "docker-entrypoint.s…"   8 minutes ago    Exited (137) 10 seconds ago           contenedor3
```

Paso B5. **Abrir Terminal Interactiva (bash) en uno de los Contenedores:**

``` **terminal**
docker exec -it contenedor1 /bin/bash
```

Esto abrirá una terminal interactiva en el contenedor1.

```
carperraj@info2-14:~/Entornos/Practica 4 DOCKER$ docker exec -it contenedor_carlos /bin/bash
root@caaffb09f5b8:/app#
```

Paso B6. Sugerencia Adicional: Verificar Estadísticas del Contenedor:

``` **terminal**
docker stats contenedor2
```

Este comando muestra estadísticas en tiempo real del uso de recursos del contenedor2.

```
CONTAINER ID   NAME          CPU %     MEM USAGE / LIMIT     MEM %     NET I/O       BLOCK I/O   PIDS
e3e2da4f857b   contenedor2   0.00%     11.33MiB / 7.534GiB   0.15%     1.01kB / 0B   0B / 0B     10
```

Paso B7. Ver Historial de la Imagen:

``` **terminal**
docker history nombre_de_tu_imagen
```

Este comando mostrará el historial de capas de la imagen Docker.

```
carperraj@info2-14:~/Entornos/Practica 4 DOCKER$ docker history cb0714248b4b
IMAGE          CREATED          CREATED BY
 SIZE      COMMENT
cb0714248b4b   48 minutes ago   CMD ["node" "server.js"]
 0B        buildkit.dockerfile.v0
<missing>      48 minutes ago   EXPOSE map[80/tcp:{}]
 0B        buildkit.dockerfile.v0
<missing>      48 minutes ago   RUN /bin/sh -c npm install # buildkit
 7.07MB    buildkit.dockerfile.v0
<missing>      48 minutes ago   COPY . /app # buildkit
 310kB     buildkit.dockerfile.v0
<missing>      48 minutes ago   WORKDIR /app
 0B        buildkit.dockerfile.v0
<missing>      7 days ago       /bin/sh -c #(nop)  CMD ["node"]
 0B
<missing>      7 days ago       /bin/sh -c #(nop)  ENTRYPOINT ["docker-entry…
 0B
<missing>      7 days ago       /bin/sh -c #(nop) COPY file:4d192565a7220e13…
 388B
<missing>      7 days ago       /bin/sh -c set -ex   && export GNUPGHOME="$(…
 7.58MB
<missing>      7 days ago       /bin/sh -c #(nop)  ENV YARN_VERSION=1.22.19
 0B
<missing>      7 days ago       /bin/sh -c ARCH= && dpkgArch="$(dpkg --print…
 166MB
<missing>      7 days ago       /bin/sh -c #(nop)  ENV NODE_VERSION=21.2.0
 0B
<missing>      7 days ago       /bin/sh -c groupadd --gid 1000 node   && use…
 8.94kB
<missing>      8 days ago       /bin/sh -c set -ex;  apt-get update;  apt-ge…
 587MB
<missing>      8 days ago       /bin/sh -c apt-get update && apt-get install…
 177MB
<missing>      8 days ago       /bin/sh -c set -eux;  apt-get update;  apt-g…
 48.4MB
<missing>      8 days ago       /bin/sh -c #(nop)  CMD ["bash"]
 0B
<missing>      8 days ago       /bin/sh -c #(nop) ADD file:39d17d28c5de0bd62…
 117MB
carperraj@info2-14:~/Entornos/Practica 4 DOCKER$
```

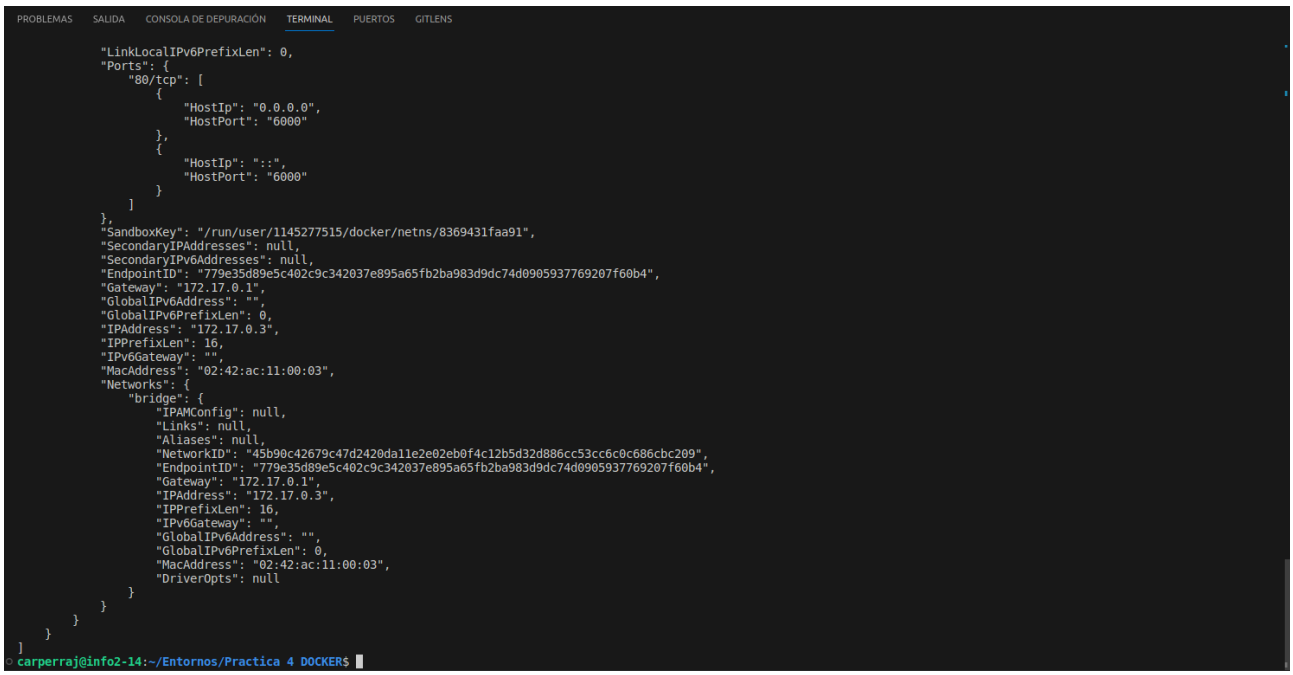
Paso B8. Obtener Detalles del Contenedor mediante `docker inspect`:

``` **terminal**
docker inspect contenedor1
```

```
carperraj@info2-14:~/Entornos/Practica 4 DOCKER$ docker inspect contenedor_carlos
[
    {
        "Id": "caaffb09f5b8fd2ac3e0de1a3c55d631b20f805f0249e6507091893802f6ed41",
        "Created": "2023-11-29T11:05:10.569014044Z",
        "Path": "docker-entrypoint.sh",
        "Args": [
            "node",
            "server.js"
        ],
        "State": {
            "Status": "running",
            "Running": true,
            "Paused": false,
            "Restarting": false,
            "OOMKilled": false,
            "Dead": false,
            "Pid": 14899,
            "ExitCode": 0,
            "Error": "",
            "StartedAt": "2023-11-29T11:05:10.95083454Z",
            "FinishedAt": "0001-01-01T00:00:00Z"
        },
        "Image": "sha256:cb0714248b4b3d263574e848cb1ae3facd8908393ab24f6d6c4031d8736e1667",
        "ResolvConfPath": "/home/carperraj/.local/share/docker/containers/caaffb09f5b8fd2ac3e0de1a3c55d631b20f805f0249e6507091893802f6ed41/resolv.conf",
        "HostnamePath": "/home/carperraj/.local/share/docker/containers/caaffb09f5b8fd2ac3e0de1a3c55d631b20f805f0249e6507091893802f6ed41/hostname",
        "HostsPath": "/home/carperraj/.local/share/docker/containers/caaffb09f5b8fd2ac3e0de1a3c55d631b20f805f0249e6507091893802f6ed41/hosts",
        "LogPath": "/home/carperraj/.local/share/docker/containers/caaffb09f5b8fd2ac3e0de1a3c55d631b20f805f0249e6507091893802f6ed41/caaffb09f5b8fd2ac3e0de1a3c55d631b20f805f0249e65070
91893802f6ed41-json.log",
        "Name": "/contenedor_carlos",
        "RestartCount": 0,
        "Driver": "overlay2",
        "Platform": "linux",
        "MountLabel": "",
        "ProcessLabel": "",
        "AppArmorProfile": "",
        "ExecIDs": [
            "4e8bd801e8af0a3a3740d00a9cab6e00110f053479c6d6cee19101eab3f46263"
        ],
```

```
PROBLEMAS   SALIDA   CONSOLA DE DEPURACIÓN   TERMINAL   PUERTOS   GITLENS

        "Driver": "overlay2",
        "Platform": "linux",
        "MountLabel": "",
        "ProcessLabel": "",
        "AppArmorProfile": "",
        "ExecIDs": [
            "4e8bd801e8af0a3a3740d00a9cab6e00110f053479c6d6cee19101eab3f46263"
        ],
        "HostConfig": {
            "Binds": null,
            "ContainerIDFile": "",
            "LogConfig": {
                "Type": "json-file",
                "Config": {}
            },
            "NetworkMode": "default",
            "PortBindings": {
                "80/tcp": [
                    {
                        "HostIp": "",
                        "HostPort": "6000"
                    }
                ]
            },
            "RestartPolicy": {
                "Name": "no",
                "MaximumRetryCount": 0
            },
            "AutoRemove": false,
            "VolumeDriver": "",
            "VolumesFrom": null,
            "ConsoleSize": [
                33,
                49
            ],
            "CapAdd": null,
            "CapDrop": null,
            "CgroupnsMode": "private",
            "Dns": [],
            "DnsOptions": [],
            "DnsSearch": [],
            "ExtraHosts": null,
            "GroupAdd": null,
            "IpcMode": "private",
            "Cgroup": "",
```


PROBLEMAS
SALIDA
CONSOLA DE DEPURACIÓN
TERMINAL
PUERTOS
GITLENS
"LinkLocalIPv6PrefixLen": 0,
"Ports": {
"80/tcp": [
{
"HostIp": "0.0.0.0",
"HostPort": "6000"
},
{
"HostIp": "::",
"HostPort": "6000"
}
]
},
"SandboxKey": "/run/user/1145277515/docker/netns/8369431faa91",
"SecondaryIPAddresses": null,
"SecondaryIPv6Addresses": null,
"EndpointID": "779e35d89e5c402c9c342037e895a65fb2ba983d9dc74d0905937769207f60b4",
"Gateway": "172.17.0.1",
"GlobalIPv6Address": "",
"GlobalIPv6PrefixLen": 0,
"IPAddress": "172.17.0.3",
"IPPrefixLen": 16,
"IPv6Gateway": "",
"MacAddress": "02:42:ac:11:00:03",
"Networks": {
"bridge": {
"IPAMConfig": null,
"Links": null,
"Aliases": null,
"NetworkID": "45b90c42679c47d2420da11e2e02eb0f4c12b5d32d886cc53cc6c0c686cbc209",
"EndpointID": "779e35d89e5c402c9c342037e895a65fb2ba983d9dc74d0905937769207f60b4",
"Gateway": "172.17.0.1",
"IPAddress": "172.17.0.3",
"IPPrefixLen": 16,
"IPv6Gateway": "",
"GlobalIPv6Address": "",
"GlobalIPv6PrefixLen": 0,
"MacAddress": "02:42:ac:11:00:03",
"DriverOpts": null
}
}
}
}
]
carperraj@info2-14:~/Entornos/Practica 4 DOCKER$