

Steps:

1.- Crear contenedor de Node-RED: `docker run -it -p 1880:1880 -v node-red_data:/data --name [containerName] nodered/node-red`

Hablar sobre los puertos, volumen y nombre del contenedor

2.- Levantar contenedor haciendo uso de Docker-compose (Docker-compose.yml)

Hablar sobre puertos, volumen, uso de redes y explicar property restart.

3.- Uso de dockerfile o como dockerizar una aplicación o script

En la carpeta Uso de dockerfile hay un ejemplo de api en Python desarrollada con Flask. El comando para levantarla es: `docker-compose up --build -d api`

Importante el build vamos a generar de esta forma la imagen al vuelo. El -d es para que el proceso no este atacheado. Es decir, si cierro la consola pueda seguir y no se detenga.

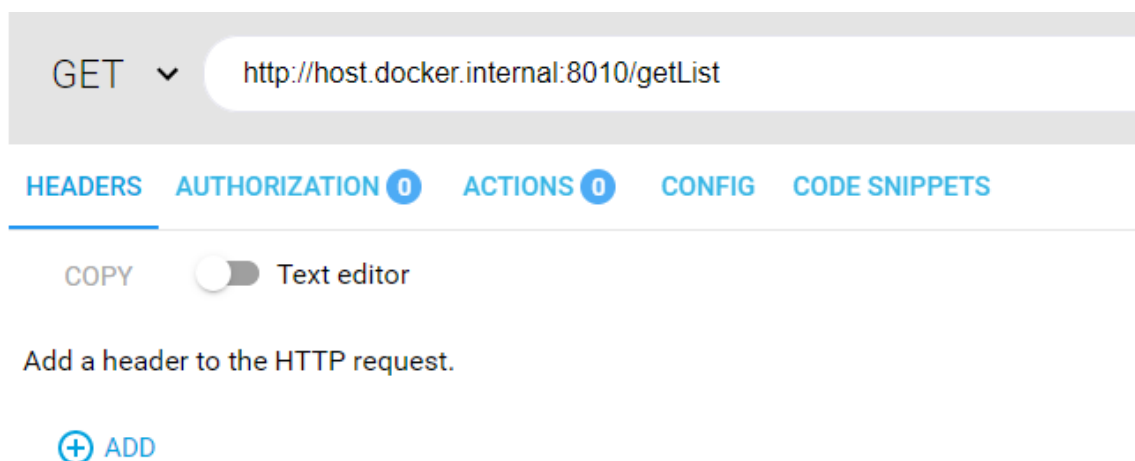
Explicar los pasos del dockerfile y limpieza del código instalando librerías en fichero requirements.txt

Una vez ejecutado. Haciendo: `docker logs -f api` vemos el log del contenedor de la API.

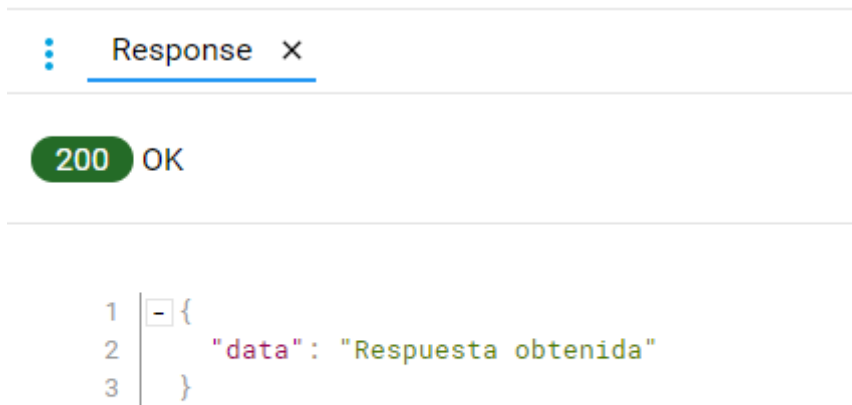
```
PS C:\Users\Thor\Desktop\Tavernes\Sesiones\5.- Docker\Uso de dockerfile\folder> docker logs -f api
* Serving Flask app 'script'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:8010
* Running on http://172.19.0.2:8010
Press CTRL+C to quit
```

Nuestra API tienes dos metodos. Podemos probarlo desde postman, arc o cualquier cliente de API REST. Se ejecuta en el puerto 8010

Vamos a probar el GET.



Y su respuesta:



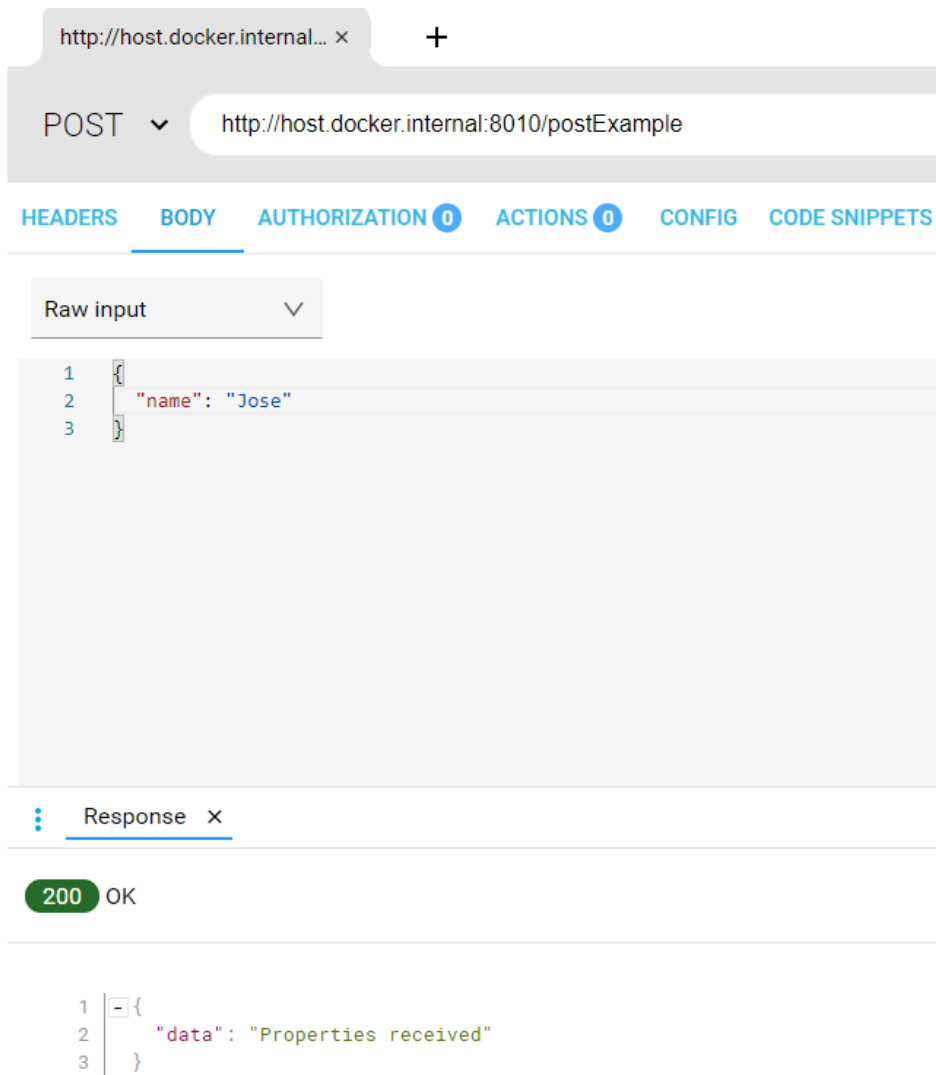
Explicar porque accedemos a la url con host.docker.internal (Windows, mi equipo) en Linux usaríamos 172.17.0.1 y a continuación el puerto.

Ejemplo POST y control de parámetros que recibe el body.

Hacer distintas pruebas invocando este endpoint sin el Content-Type adecuado, llamándolo como GET , sin pasar parámetros y pasando alguno que no sea el requerido.

Explicar el main del fichero Python y porque usamos 0.0.0.0.

Este pantallazo es de su ejecución correcta.



4.- Exponer API de Docker.

Intentar ejecutar este comando: <http://172.17.0.1:2375/images/json>

Este método nos lista las imágenes docker que tenemos en nuestro pc

Usar este link: <https://medium.com/@ssmak/how-to-enable-docker-remote-api-on-docker-host-7b73bd3278c6>

Probar a ejecutarlo ahora una vez habilitado su API.

Este link nos lleva al API de Docker para que vean los métodos y se puede hacer alguna prueba.