

Contenedores Docker. Imágenes multistage

1. Se pide modificar la práctica anterior con **React** de tal modo que el fichero **Dockerfile** permita construir la app React desde la imagen de node pero luego pasar esa construcción a un servidor nginx en esa misma Dockerfile.

Se muestra captura de pantalla del Dockerfile de ejemplo:

```
FROM node:18-alpine as react

WORKDIR /app

COPY package*.json .

RUN npm install

COPY . .

RUN npm run build

FROM nginx

COPY --from=react /app/dist /usr/share/nginx/html
```

Como en este caso, en el fichero **docker-compose.yml**, se debe exponer el puerto 80 del servicio nginx a la maquina Windows (por ejemplo, al puerto 8000), ya que la imagen de nginx usa este puerto por defecto

Se debe probar el acceso a la aplicación a través de la URL <http://localhost:8000>

1.- Dockerfile:

```
Dockerfile X docker-compose.yml
adivinar_numero > Dockerfile > ...
1  FROM node:18-alpine as react
2  WORKDIR /app
3  COPY package.json .
4  RUN npm install
5  COPY . .
6  RUN npm run build
7
8  FROM nginx
9
10 COPY --from=react /app/dist /usr/share/nginx/html
```

2.- docker-compose.yml:

```
adivinar_numero > docker-compose.yml
▷Run All Services
1 services:
  ▷ Run Service
2   app:
3     build: .
4     ports:
5       - 8000:80
```

3.- Resultado:

