Actividades Docker. Contenedores en node

1. Se pide usar el contenedor de **`node:18-alpine`** y sobre él crear una imagen a partir de un Dockerfile que permita dockerizar la aplicación node que se entrega en el fichero app.js.

   a) Previamente se debe crear una carpeta para aplicación y copiar dentro de ella el fichero **`app.js`**
   b) Dentro de la carpeta anterior ejecutamos **`npm init -y`** para crear el fichero packages.json

   El fichero Dockerfile debe realizar las siguientes acciones:

   - Descargar la imagen de node que se pide
   - Establecer el directorio de trabajo en **`/usr/src/app`** mediante la instrucción **`WORKDIR`**
   - Copiar **`packages.json`** al directorio anterior
   - Instalar las dependencias con **`npm install`**
   - Copiar el fichero **`app.js`** en **`/usr/src/app`**
   - Mediante **`npm`** se debe instalar **`express`**
   - Exponer el puerto 3000
   - Lanzar la aplicación **`app.js`** con el comando **`node`** mediante **`CMD`**

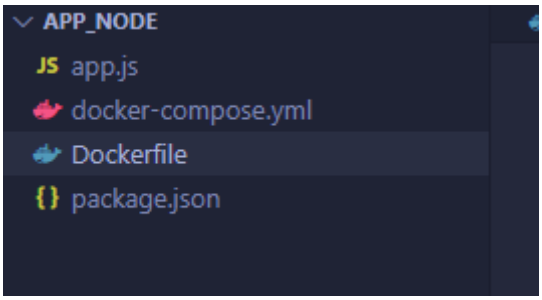
   La creación del contenedor se realizará mediante docker-compose.yml que permitirá vincular o mapear el puerto 3000 del contenedor con el puerto 3000 de la maquina Windows (este último puerto puede modificarse lógicamente). Este fichero creará también el servicio **app** y el contenedor de nombre **servidor_app**

   Se debe también mostrar una captura de pantalla del acceso a http://localhost:3000

   1.- Vamos a la carpeta app_node y ejecutamos el comando.

   ```
   C:\Users\guillermosr26\Documents\DAW\DAW2B-DAW-Guillermo-Sobrino\docker13\app_node>npm init -y
   ```

   2.- La estructura de carpetas será la siguiente.



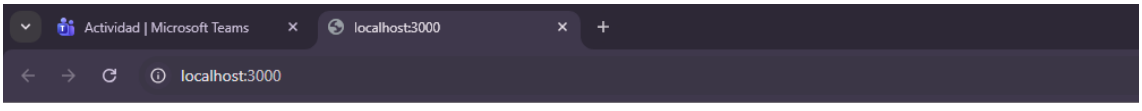
3.- Dockerfile:

```
Dockerfile > ...
1   FROM node:18-alpine
2   WORKDIR /usr/src/app
3   COPY package.json .
4   RUN npm install
5   COPY app.js .
6   RUN npm install express
7   EXPOSE 3000
8   CMD [ "node", "app.js" ]
```

4.- Docker-compose:

```
docker-compose.yml
    Run All Services
1   services:
      Run Service
2     app:
3       build: .
4       ports:
5         - 3000:3000
6   # -- build
```

5.- Resultado:



Contenedor node en Docker