

## NGINX

Lo primero es jugar con los comandos de docker como para averiguar en que carpeta estan los documentos a ser mostrados por el frontend

```
docker run --entrypoint=ls nginx /  
docker run --entrypoint=ls --name=02 nginx /
```

Vamos a dockerhub para ver donde esta la carpeta a servir por http

```
docker run --entrypoint=ls nginx /usr/share/nginx/html
```

Que pasa si no exportamos el puerto

Ahora exportar un puerto 8080

Pero para diferenciar del 80 hay que aplicarle otro contenido para esto, vamos a mapear una carpeta.

Mostrar como se ven los logs

Se puede realizar una ruta que no existe y verla en el log del container con un grep

## CONFIGURACION DE NGINX

Hacer un repaso de la configuración de nginx y poner un entry point para el front y un load balancer para los tres servidores de express.

Luego recién, vamos a realizar un dockerFile con el motor de node y con express, uuid, underscore, etc.

Esto hay que buscarlo en los cursos anteriores.

Nos tiramos a hacer el dockerfile con todo y ahi termina la clase, La siguiente vemos de hacer un middleware de fileupload o algo que tenga frontend y backend en la nube.

## ***CONFIGURANDO NGINX***

### Context & Directives

Directives son names values

Context es un scope basicamente

http context

server context

nginx.conf

```
events {}
http {
    types: {
        text/html html;
        text/css css;
    }
    server {
        listen 80;
        server_name ip_ff
        root
    }
}
```

nginx reload nginx.service o bien

nginx -t

dentro de server, configuramos los proxy\_pass

```
location = /api {  
  proxy_pass http://127.0.0.1:8080/  
}
```

```
location ^~ /api/ {  
  proxy_pass http://127.0.0.1:8080/api/  
}
```

Una vez establecida la configuracion vamos a realizar una imagen de docker con todo nuestro contenido.

## ***CREANDO UN DOCKERFILE***

Para construir una imagen de docker, primero debemos tener una imagen base de nodejs

```
FROM node  
LABEL author educacionit  
ENTRYPOINT ["node", "/clase6/server.js"]  
RUN mkdir -p clase6/lib  
COPY server.js /clase6/server.js  
COPY package.json /clase6/package.json  
ADD lib /clase6/lib  
WORKDIR /clase6  
RUN npm install  
EXPOSE 3000  
ENV USERNAME=laPepaPig
```

server.js

```
const express = require('express')
const http = require('http')
const genusuario = require('./lib/genusuario')

const app = express()
app.use(express.json())

const PORT = parseInt("3000")

app.get('/', (request, response) => {
  response.setHeader('Content-Type', 'application/json')

  let arr = []

  for (let x = 0; x < 5; x++) {
    arr.push(genusuario())
  }

  arr.push({'instancia':process.env['INST']})

  response
    .status(200)
    .end(JSON.stringify({arr}))
})

app.listen(PORT, (err) => {
  console.log(['escuchando el puerto', PORT].join(': '))
})
```

lib/genusuario.js

```
const faker = require('faker');
```

```
const uuid = require('uuid');
```

```
function crearObjetoFake() {
```

```
  return {
```

```
    id : uuid.v4(),
```

```
    firstName: faker.name.firstName(),
```

```
    lastName: faker.name.lastName(),
```

```
    city: faker.address.city(),
```

```
    streetName: faker.address.streetName(),
```

```
    country: faker.address.country(),
```

```
    accountName: faker.finance.accountName(),
```

```
    account: faker.finance.account(),
```

```
    amount: faker.finance.amount()
```

```
  }
```

```
}
```

```
module.exports = crearObjetoFake;
```