

Tutorial 06 to do in class – Remember to upload the repo link to Teams.

Antes de iniciar:

- Terminar los tutoriales anteriores.
- Este tutorial muestra un ejemplo de creación de un proyecto con Express (Fast, unopinionated, minimalist web framework) y despliegue en GCP con Docker.

A. Node 20 installation

Install Node 20

Go to this link and download and install NodeJS 20 (express requires installing NodeJS).
<https://nodejs.org/es/>

B. Express Project Creation

Create a new Express project:

- Open your Terminal, and in a location of your choice (you can use the *xampp/htdocs/* location if you want), execute the following:

Execute in Terminal

```
mkdir randomquotes  
cd randomquotes
```

- It creates a new project folder “randomquotes” and moves to that folder. Express official documentation: <https://expressjs.com/>
- Execute the following command, and **press Enter in all the questions**.

Execute in Terminal

```
npm init
```

- Execute the following command to install the express package.

Execute in Terminal

```
npm install express
```

Index file

- Go to project root folder and create a file *index.js* with the next content.

Add Entire Code

```
const express = require('express')
const app = express()
const port = 3000

app.get('/', (req, res) => {
  res.send('Hello World!')
})

app.listen(port, () => {
  console.log(`Example app listening on port ${port}`)
})
```

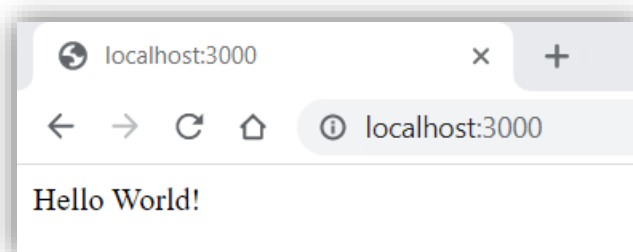
Run the application

In the Terminal, go to the project directory, and execute the following:

Execute in Terminal

```
node index.js
```

Go to <http://localhost:3000/>



C. RandomQuotes generator

Index modification

- Go to *index.js* and make the following changes in **bold**.

Modify Bold Code

```
const express = require('express')
const app = express()
const port = 3000

const phrases = [
  "Get ready to be inspired...",
  "See rejection as redirection.",
  "There is beauty in simplicity.",
  "You can't be late until you show up.",
  "Maybe life is testing you. Don't give up.",
  "Impossible is just an opinion.",
  "Alone or not you gonna walk forward.",
]

app.get('/', (req, res) => {
  const number = Math.floor(Math.random() * 7);
  res.send(phrases[number])
})

app.listen(port, () => {
  console.log(`Example app listening on port ${port}`)
})
```

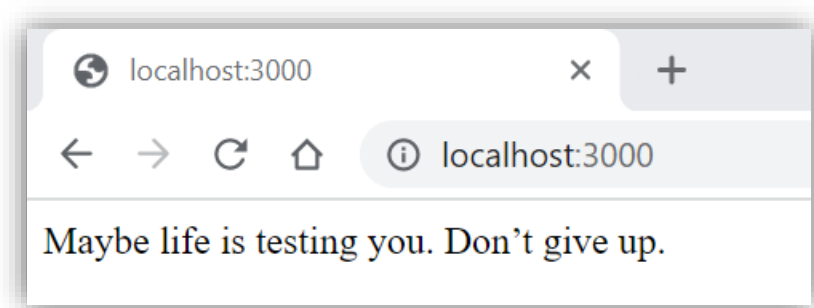
Run the application

In the Terminal, go to the project directory, and stop the server (if it is running -> with **ctrl + c**), and then, execute the following:

Execute in Terminal

```
node index.js
```

Go to <http://localhost:3000/>



D.Deployment in GCP with Docker

Dockerfile

- Go to the project root folder and create a file *Dockerfile* with the next content (**be careful, it should be a file called *Dockerfile* without any extension, please be sure it doesn't have .txt extension**).

Add Entire Code

```
FROM node:20
```

```
# Create app directory
```

```
WORKDIR /usr/src/app
```

```
# Install app dependencies
```

```
# A wildcard is used to ensure both package.json AND package-lock.json are copied
```

```
# where available (npm@5+)
```

```
COPY package*.json ./
```

```
RUN npm install
```

```
# If you are building your code for production
```

```
# RUN npm ci --only=production
```

```
# Bundle app source
```

```
COPY . .
```

```
EXPOSE 8080
```

```
CMD [ "node", "index.js" ]
```

Index modification

- Go to *index.js* and make the following changes in **bold**.

Modify Bold Code

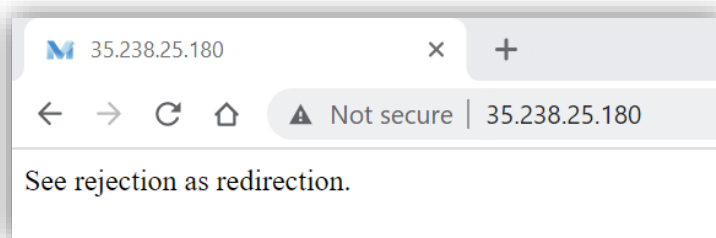
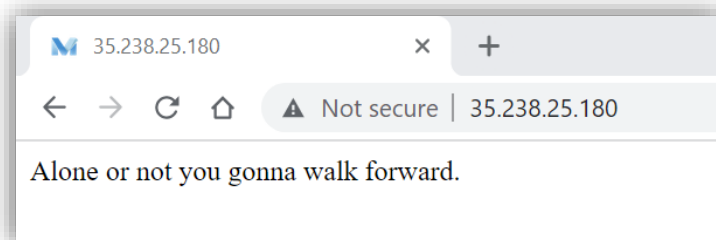
```
const express = require('express')
const app = express()
const port = 80
...
```

GCP with Docker

- Upload this Project to GitHub (in a new repo). Then, follow the steps A, B and C of the **“Presentación-14-Tutorial-laravel-docker”** to create a new virtual machine with Docker installed. It is very important that this process works for you. Because we need this virtual machine running for the next workshop and for the **“Taller 2”**.
- Finally, clone the new GitHub repo inside your new virtual machine, build the Docker image, and create the new container. Remember that this process was explained in **“Presentación-14-Tutorial-laravel-docker”**.

Run the application

- Go the virtual machine External IP. Refresh the website several times and you will see different **“Quotes”**. **Remember you should open that IP with HTTP, NOT USE HTTPS.**



Useful commands

```
sudo docker image build -t express-app .  
sudo docker container run -d --name express-docker -p 80:80 express-app  
git pull  
sudo docker container rm -f express-docker
```