NODEMON (PRÁCTICA)

Tiempo estimado: 15min

El objeto de esta práctica es asentar y consolidar los conceptos presentados en la parte teórica de la lección.

Al finalizarla, el estudiante:

- Habrá instalado localmente nodemon.
- Habrá configurado el *script* de inicio para que la aplicación se ejecute mediante nodemon.
- Habrá disparado el reinicio automático de la aplicación, mediante la realización de cambios en archivos.
- Habrá realizado un reinicio manual.
- Habrá configurado el reinicio retardado.

Objetivos

En esta práctica, vamos a crear una aplicación Express que responda a peticiones HTTP remitidas por clientes. Realizaremos algunos cambios para observar cómo se comporta nodemon, nuestro monitor o supervisor de cambios, ante ellos.

Creación del proyecto

Para comenzar, vamos a crear el proyecto de la aplicación.

1. Abrir una consola.
2. Crear el directorio de la práctica.
3. Ir al directorio de la práctica.
4. Crear el archivo package.json con el siguiente contenido:

```
{
  "name": "express-app",
  "version": "0.1.0",
  "private": true,
  "scripts": {
    "start": "node ./app.js",
    "start-dev": "./node_modules/.bin/nodemon ./app.js"
  },
  "dependencies": {
    "express": "*"
  },
  "devDependencies": {
    "nodemon": "*"
  }
}
```

   Generalmente, nodemon sólo se usa durante el desarrollo. De ahí que lo ubiquemos entre las dependencias de desarrollo, no así de producción. Aunque para algunos proyectos de producción, nodemon podría ser también muy útil.

5. Crear el archivo index.html:

```
<!doctype html>

<html>
  <head>
    <title>Express app</title>
```



```
  </head>

  <body>
    <p>¡Hola Mundo!</p>
  </body>
</html>
```

6. Crear el archivo app.js con el siguiente contenido:

```
"use strict";

//imports
const express = require("express");
const http = require("http");

//app
const app = express();

//config app
app.get("*", function(req, res) {
  res.sendFile("index.html", {root: __dirname});
});

//listen
http.createServer(app).listen(8080, function() {
  console.log("Listening...");
});
```

7. Instalar dependencias:

```
> npm install
```

8. Iniciar la aplicación Express mediante nodemon:

```
> npm run start-dev
```

Recordemos, el *script* de inicio start se suele usar para el inicio de producción, mientras que start-dev para desarrollo.

9. Abrir el navegador.
10. Solicitar la página de inicio de la aplicación: localhost:8080.

REINICIO AUTOMÁTICO

Veamos cómo nodemon realiza reinicios automáticos al detectar cambios:

1. Solicitar localhost:8080/morrissey.

   Como se puede observar, devuelve la página de inicio.

2. Editar el archivo app.js.
3. Modificar app.get() como sigue:

```
app.get("*", function(req, res) {
  if (req.path == "/morrissey") {
    res.set("Content-Type", "application/json");
    res.send(JSON.stringify({
      name: "Steven Patrick Morrissey",
      origin: "Davyhulme, Lancashire, England"
    }));
  } else {
    res.sendFile("index.html", {root: __dirname});
  }
});
```

4. Guardar cambios.
5. Repetir la solicitud anterior localhost:8080/morrissey.

   Tras los cambios realizados, debería aparecer la información de Morrissey y no la de la página de inicio.

6. Ir a la consola donde tenemos iniciada la aplicación con nodemon.

7. Buscar una entrada como la siguiente:

   ```
   [nodemon] restarting due to changes…
   ```

   A diferencia de las prácticas anteriores donde teníamos que reiniciar la aplicación explícitamente, algo que resulta muy pesado, ahora es nodemon quien lo hace automáticamente por nosotros al detectar cambios en los archivos de la aplicación. Esto hace nuestro trabajo más fácil y aumenta nuestra productividad.

Monitorización parcial

A continuación, vamos a restringir los archivos que debe monitorizar nodemon:

1. Ir a la consola donde tenemos iniciada la aplicación con nodemon.
2. Buscar una entrada como la siguiente:

   ```
   [nodemon] watching: *.*
   ```

   Indica los archivos que monitoriza nodemon y cuyos cambios disparan el reinicio de la aplicación web. Realmente, nodemon *no* monitoriza *todos* los archivos, sólo aquellos cuya extensión es .js, .coffee, .litcoffee y .json. Más adelante en el curso, veremos cómo configurar más extensiones.
3. Modificar el *script* de inicio indicado en el archivo package.json para que sólo monitorice el archivo app.js:

   ```
   "scripts": {
     "start": "node ./app.js",
     "start-dev": "./node_modules/.bin/nodemon -w ./app.js ./app.js"
   }
   ```

   Recordemos que si es necesario indicar varios archivos y/o directorios, cada uno se debe especificar mediante su correspondiente opción -w o --watch.
4. Reiniciar la aplicación explícitamente para que nodemon arranque con las nuevas opciones.
5. Observar que ahora nodemon sólo monitoriza los cambios del archivo app.js:

   ```
   [nodemon] watching: ./app.js
   ```

Reinicio manual

En este punto, vamos a ver cómo realizar un reinicio explícito de la aplicación, sin necesidad de modificar ningún cambio:

1. Ir a la consola de nodemon.
2. Escribir rs y pulsar INTRO.

Reinicio retardado

De manera predeterminada, nodemon realiza el reinicio cuando detecta cambios. Vamos a configurarlo para que aplique un retardo de, digamos, 30 segundos y así no reinicie inmediatamente, lo que ayudará a realizar menos reinicios cuando se aplican varios cambios al proyecto de golpe:

1. Editar el archivo package.json.
2. Modificar el *script* de inicio de la aplicación:

   ```
   "scripts": {
     "start": "node ./app.js",
     "start-dev": "./node_modules/.bin/nodemon --delay 30s ./app.js"
   }
   ```
3. Reiniciar la aplicación explícitamente para que nodemon arranque con la nueva configuración.
4. Editar app.js y realizar cualquier cambio insignificante como, por ejemplo, añadir un comentario.
5. Guardar cambios.
6. Ir a la consola donde tenemos arrancada la aplicación con nodemon.
7. Contar hasta 30 mentalmente, el tiempo de retardo que debemos esperar a que nodemon reinicie la aplicación automáticamente.

Generalmente, 2 segundos suele ser un tiempo bastante bueno para que la aplicación no se reinicie automáticamente varias veces cuando se modifican varios archivos de una tacada.

