

---

## NPM: Package.json

Gabriel Rodríguez Flores

October 21, 2021

- Package JSON
- NPM Scripting

## Contents

<b>1</b>	<b>Teoría</b>	<b>3</b>
1.1	Package.json . . . . .	3
<b>2</b>	<b>Ejemplos</b>	<b>3</b>
2.1	Automatizar distintos tipos de arranque . . . . .	3
2.2	Automatizar limpieza del proyecto . . . . .	4
2.3	Automatizar gestión de variables de entorno con .env . . . . .	4
<b>3</b>	<b>Ejercicios</b>	<b>4</b>
<b>4</b>	<b>Entregables</b>	<b>5</b>
4.1	En clase . . . . .	5
4.2	Tarea . . . . .	5
4.3	Trabajo . . . . .	5

# 1 Teoría

## 1.1 Package.json

- Documentación package.json
- Prestar especial atención a:
  - scripts
    - \* *Definición:* `"dev": "nodemon"`,
    - \* *Ejecución:* `npm run dev`
  - dependencies
  - devDependencies

Ejemplo package.json

```
1 {
2   "name": "test",
3   "version": "1.0.0",
4   "description": "",
5   "main": "index.js",
6   "scripts": {
7     "start": "node index.js",
8     "dev": "nodemon index.js",
9     "test": "echo \"Error: no test specified\" && exit 1"
10  },
11   "author": "",
12   "license": "ISC",
13   "dependencies": {
14     "chalk": "^1.0.0"
15  },
16   "devDependencies": {
17     "nodemon": "^1.0.0"
18  }
19 }
```

Cuando se ejecuta `npm install` sin indicar el módulo a instalar, se instalarán todos los módulos listados en el `package.json`. Esto agiliza la instalación de una aplicación cuando se quiere desplegar en otro servidor.

## 2 Ejemplos

### 2.1 Automatizar distintos tipos de arranque

```
1 {
2   ...,
3   "scripts": {
4     "start": "node index.js",
5     "dev": "nodemon index.js",
6   },
7   ...
8 }
```

## 2.2 Automatizar limpieza del proyecto

```
1 {
2   ...,
3   "scripts": {
4     "clear:modules": "rimraf **/node_modules",
5     "clear:locks": "rimraf **/package-lock.json",
6     "clear": "npm run clear:test && npm run clear:locks && npm run
      clear:modules",
7   },
8   ...
9 }
```

## 2.3 Automatizar gestión de variables de entorno con .env

```
1 {
2   ...,
3   "scripts": {
4     "clear:env": "rimraf **/.env",
5     "copy:env": "for d in workspaces/*; do cp ${d}/.env.template ${d}/.
      env; done",
6   },
7   ...
8 }
```

## 3 Ejercicios

1. Realizar un package.json con los script básicos de los ejemplos
2. Script llamado `date` que imprima la fecha actual (en bash o nodejs)
3. Crear una serie de scripts con el prefijo `crear` que cree ficheros y carpetas dentro de la carpeta `files`:
  - `crear` Crea la carpeta `files`

- `crear:js` se le pasa un argumento y crea un fichero `.js` con ese nombre
  - `crear:carpeta` se le pasa un argumento y crea una carpeta con ese nombre
  - `borrar` Borra completamente la carpeta `files`
  - `borrar:js` borra todos los ficheros `.js` de la carpeta `files`
4. Realizar un programa y los scripts correspondientes para imprimir un argumento pasado en colores:
- `imprime:azul` Imprime en azul
  - `imprime:rojo` Imprime en rojo
  - `imprime:verde` Imprime en verde

## 4 Entregables

### 4.1 En clase

Ejercicios 1 y 2

### 4.2 Tarea

Ejercicio 3 y 4

### 4.3 Trabajo

- Adaptar el proyecto de *gestor de notas* para que si le indicamos argumentos, acceda directamente a la opción seleccionada
- Añadir en el `package.json` unos scripts para atajar la ejecución de notas entrando directamente a las opciones.