

ACÀMICA

Agenda

Daily

Programo: Node JS - Variables de Entorno y NPM

Buenas prácticas

Break

Programamos: Moment JS

Programan: Nodemon

Cierre



NPM

Node Package Manager es un gestor de paquetes, el cual nos permitirá incluir librerías a nuestra aplicación mediante simples comandos de consola

Daily



Daily



Sincronizando...

Bitácora



¿Cómo te ha ido?
¿Obstáculos?
¿Cómo seguimos?

Challenge



¿Cómo te ha ido?
¿Obstáculos?
¿Cómo seguimos?

NPM



NPM

Node Package Manager es un gestor de paquetes desarrollado en su totalidad bajo el lenguaje JavaScript.



Node Package Manager

Al momento de utilizar librerías en nuestros proyectos, existen diferentes formas de importarlas.

1. Descargando los archivos
2. Directo desde la Web

Descargar los archivos tiene la ventaja de que es muy rápido ya que no depende de estar buscando archivos en otro sitio, pero tener que estar buscando los archivos en las páginas individuales de cada librería y descargarlos suena un poco tedioso.

Aquí es cuando brilla NPM (Node Package Manager).

Node Package Manager

Comandos básicos

npm -v > Imprime la version de npm usada

npm init > *Crea un proyecto*

npm search > *Búsqueda de paquetes*

npm install > *Instala paquetes*

npm uninstall > *Desinstala paquetes*

npm list > *Listado del árbol de dependencias*

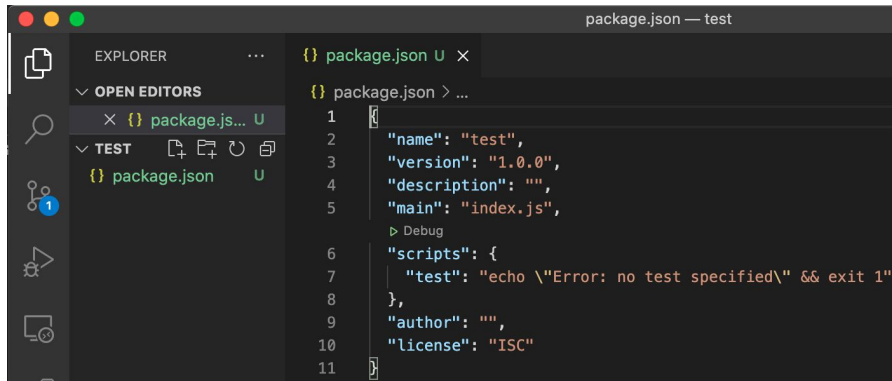
npm outdated > *Lista de paquetes para actualizar*

npm update > *Actualiza la versión del paquete*

npm help > *Ayuda*

Node Package Manager

npm init -y

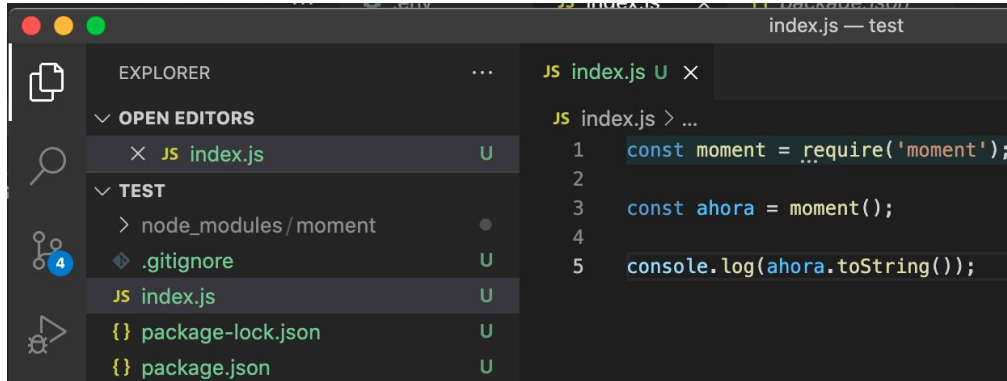


The screenshot shows the Visual Studio Code interface with a file explorer on the left and a code editor on the right. The file explorer shows a project named 'TEST' with a 'package.json' file. The code editor displays the content of 'package.json' with the following JSON structure:

```
1 {  
2   "name": "test",  
3   "version": "1.0.0",  
4   "description": "",  
5   "main": "index.js",  
6   "scripts": {  
7     "test": "echo \"Error: no test specified\" && exit 1"  
8   },  
9   "author": "",  
10  "license": "ISC"  
11 }
```

Node Package Manager

npm i moment



- node_modules
- node_modules/moment
- package-lock.json
- package.json (se modificó)

Variables de entorno





Variables de entorno

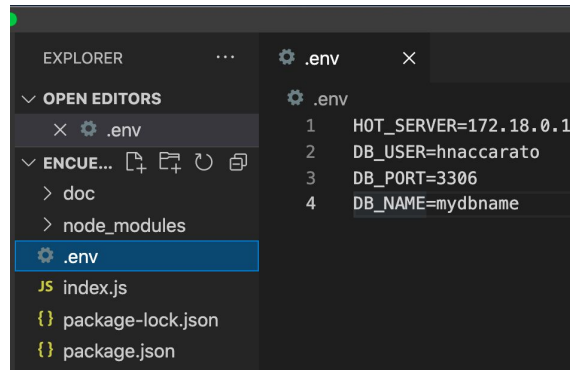
Las variables de entorno nos permiten obtener valores que varían según donde esté corriendo mi aplicación.

Por ejemplo, los valores de una variable de conexión a una base de datos **debe** cambiar si la aplicación se está ejecutando en nuestro entorno de desarrollo (nuestras computadoras) o el servidor live donde accede el usuario final.

[ver en windows](#) - [ver en linux](#) - [ver en macos](#)

Variables de entorno

```
{  
  "development": {  
    "SERVERURL": "http://localhost/",  
    "PORT": 3000  
  },  
  "production": {  
    "SERVERURL": "https://myapp.com/",  
    "PORT": 5000  
  }  
}
```



Programo

mentores/as



Variables de entorno & NPM

Veamos una demostración de cómo trabajar con las variables de entorno y el gestor de paquetes NPM.



Buenas prácticas



Variables de entorno

Utiliza variables de entorno para almacenar datos como:

- Credenciales de una base de datos
- Basepath de una API
- API Key
- API Client ID
- API Client Secret
- Definición de constantes

A close-up photograph of a white ceramic cup filled with a latte. The coffee has a thick layer of white foam with intricate brown latte art patterns. The cup is on a matching white saucer. In the background, a white napkin with a silver fork is visible, all set against a dark, textured surface.

¡BREAK!



Programamos

todos/as



Backend

Programamos

Programemos la siguiente estructura:

Instala Moment JS a través de NPM.

Crea un script para mostrar la siguiente información:

- Mostrar horario local
- Mostrar horario UTC
- Calcular diferencia horaria entre hora local y UTC
- Comparar 2 fechas y decir cual es mayor

Programan

estudiantes



Nodemon

Nodemon es una herramienta que permite reiniciar el Node JS cada vez que realizas un cambio en un archivo.

Instala [esta librería](#) de manera global y ejecuta cualquier script que tengas hasta el momento o crea uno nuevo y ejecútalo con **nodemon**.

Haz una modificación cualquiera y comprueba que reinicie el Node automáticamente.



Actividad

Imagen aleatoria

Instala la librería [cool-images](#), y crea un script para traer una imagen de manera aleatoria.
Ejecuta tu script con nodemon.



Más imágenes

Cool-images nos provee un método para obtener la cantidad de imágenes que deseemos.

Busca el método en la documentación y utilizarlo para obtener 10 imágenes.

Muestra la URL de cada imagen utilizando [foreach](#).



Log file

Cada ejecución del último script debe guardarse en un archivo llamado **log.txt**

El formato de log debe ser el siguiente:

Año/Mes/Día Hora:Minuto:segundo

Url 1

Url 2

Url 3

Año/Mes/Día Hora:Minuto:segundo

Url 1

Nodemon + Moment JS + FS + Tu lógica

Nota: implementar método [appendFile](#) de FS



Para la próxima

- 1) Termina el ejercicio del encuentro de hoy.
- 2) Lee la bitácora 41 y carga las dudas que tengas al Trello.
- 3) Resuelve el challenge.
- 4) Analiza el proyecto y comienza a trabajar en tu primera versión de los Endpoints con Swagger

En el encuentro que viene uno/a de ustedes será seleccionado para mostrar el ejercicio de hoy y otro/a mostrará cómo resolvió el challenge de la bitácora. De esta manera, ¡aprendemos todos/as de (y con) todos/as, así que vengan preparados/as!

ACÀMICA