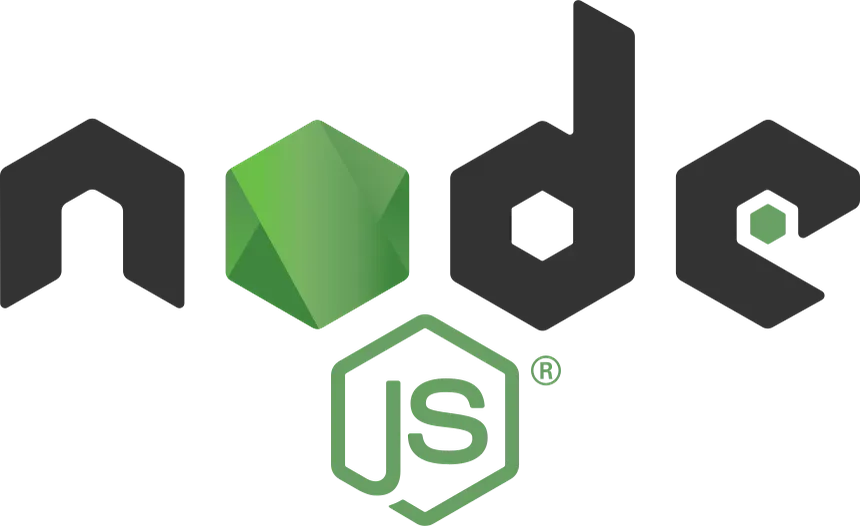
RODRIGO MARTÍNEZ DELGADO

Desarrollo web entorno cliente



# 20250207 – Instalación Node.js y concepto módulos

Instalación del entorno de ejecución Node.js en local y posterior explicación del sistema de módulos tanto con un ejemplo facilitado en la plataforma como la creación de un pequeño ejemplo.

* ejemplo-modulos-20250207
* ejemplo-modulos-avanzado-20250207

# 20250210 – Curso Node.js

Visualización de parte del curso del divulgador [midudev](https://www.youtube.com/watch?v=yB4n_K7dZV8) sobre que es Node.js, Window VS GlobalThis, NPM, Instalar PicoColors.

Posteriormente se ven algunos ejemplos de módulos:

* node:os
* node:path
* node:fs

Creación de nuestro propio proyecto usando el módulo picocolors:

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Nos crea un package.json con la propiedas de nuestro proyecto al cual agregamos "type: module" para evitar las extensiones .mjs

Instalamos el módulo [picocolors](https://www.npmjs.com/package/picocolors):

Texto

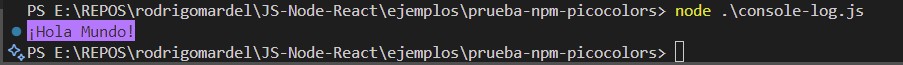
El contenido generado por IA puede ser incorrecto.

Captura de pantalla de computadora

El contenido generado por IA puede ser incorrecto.

Pequeña prueba dando color de fondo a un console.log

* prueba-npm-picocolors-20250210



# 20250212 - Git Visual Studio y Ejercicio1

Creación de repositorio en Github y clonación en local para trabajar desde Visual Studio Code. Se ha realizado sin problema a excepción de que VS no había conectado correctamente el repositorio.

Comandos utilizados:

winget install --id Git.Git -e --source winget

git clone https://github.com/rodrigomardel/JS-Node-React.git

git remote add pb github.com/rodrigomardel/JS-Node-React.git

Realización ejercicio1:

A la hora de retornar un valor en una función en el js exportado, no se puede recoger en el js que importo dicha función (revisar fetch y lógica).

# 20250213 – Ejercicio1

Repaso de los conceptos promesas, asincronía y funciones callback.

Ejercicio1 promesas:

El problema era el uso del return dentro de la promesa en el método then como retorno de los datos para la función principal.

Se usa como solución la llamada a otra función, dentro del mismo script, para el tratamiento de los datos. De esta forma solo se exporta la función principal.

# 20250214 – Ejercicio2 y Inicio React

Investigación de los módulos necesarios y su funcionalidad para crear una aplicación en Node.js para añadir y listar tareas.

El **módulo fs** de Node.js se utiliza para interactuar con el sistema de archivos, es decir, para leer, escribir, actualizar, eliminar archivos, etc. Su función principal es manipular archivos de forma sincrónica o asincrónica.

Visualización de parte del curso del divulgador [midudev](https://www.youtube.com/watch?v=7iobxzd_2wY&t=90s) sobre React (Cómo usar React sin dependencias). Explicación del ejemplo (ejemplos-react/react-sin-dependencias-20250214.html) y realización del ejercicio 5.

# 20250217 – Ejercicio 2 y inicio JSX

A la hora de crear la aplicación el planteamiento era erróneo, estaba importando módulos de Node para ser usados en el entorno de ejecución del navegador, esto no es posible. Hay que tener en cuenta que son entornos de ejecución distintos, con diferentes características.

Inicio a JSX con la visualización del curso del divulgador [midudev](https://www.youtube.com/watch?v=7iobxzd_2wY&t=90s) (JSX desde cero) explicación posterior probando el código en los dos compiladores principales [SWC](https://swc.rs/playground) y [Babel](https://babeljs.io/)

* ejemplos-react-con-JSX-20250217

Nuevo plantamiento del ejercicio 2, crear un contandor que guarda su valor en un archivo de texto (contador.txt). Puedes incrementar, decrementar o ver el valor actual del contador desde la terminal.