

# UT 08- Framework JavaScript

## MÓDULOS

### 1. Crear una web utilizando módulos: NASA

Modifica el ejercicio de la NASA de la unidad anterior (ej. 15), de modo que se divida en varios módulos. Necesitarás utilizar import y export.

## NODE.JS

### 2. Crear un programa en node.js utilizando módulos

Crea un programa en node.js utilizando varios módulos.

### 3. Utilizar librerías nativas de node.js

Crea un programa en **Node.js** que realice las siguientes tareas utilizando librerías nativas os y fs:

#### 1. Información del Sistema:

- Utilizando la librería os, el programa debe mostrar información del sistema operativo, como:
  - Nombre del sistema operativo.
  - Versión del sistema operativo.
  - Cantidad de memoria disponible.
  - Número de CPUs disponibles.

#### 2. Crear y Escribir en un Archivo:

- Utilizando la librería fs, el programa debe crear un archivo de texto llamado info\_sistema.txt en el que se guardará la información del sistema recolectada en el primer paso.
- El archivo debe contener las siguientes líneas:
  - El nombre del sistema operativo.
  - La versión del sistema operativo.
  - La cantidad de memoria disponible (en MB).
  - El número de CPUs disponibles.

#### 3. Leer y Mostrar el Contenido del Archivo:

- Después de haber creado el archivo info\_sistema.txt, el programa debe leerlo y mostrar su contenido por consola.

#### Requisitos:

- El programa debe ser ejecutado desde la línea de comandos.
- La información debe ser guardada en un archivo .txt dentro del mismo directorio donde se ejecuta el programa.
- El programa debe manejar cualquier error potencial, como problemas al escribir o leer el archivo.

#### Restricciones:

- Solo se pueden utilizar las librerías nativas de Node.js como os y fs.

- El archivo debe ser creado y gestionado de manera síncrona o asíncrona según se prefiera, pero debe cumplir con los requisitos de la tarea.

#### 4. Utilizar librerías de npm

Crear un programa en **Node.js** que utilice un paquete de npm sencillo para realizar las siguientes tareas:

1. **Instalación de Paquete:**
  - Utilizando el administrador de paquetes npm, instalar el paquete chalk (<https://www.npmjs.com/package/chalk>). Este paquete permite cambiar el color del texto en la consola.
2. **Interacción con el Usuario:**
  - El programa debe pedir al usuario que ingrese su nombre y su edad a través de la consola (utilizando la librería readline de Node.js para interactuar con el usuario).
3. **Mostrar Mensaje Personalizado:**
  - Utilizando el paquete chalk, el programa debe mostrar un mensaje de bienvenida que incluya el nombre y la edad del usuario, y que el mensaje se muestre en un color diferente dependiendo de la edad del usuario:
    - Si la edad es menor de 18 años, el mensaje debe aparecer en color rojo.
    - Si la edad está entre 18 y 65 años, el mensaje debe aparecer en color verde.
    - Si la edad es mayor de 65 años, el mensaje debe aparecer en color azul.