**Uso de Node.js y el Módulo File System (fs)**

**Objetivo**

El objetivo de este trabajo práctico es familiarizarse con el uso de la consola de Node.js y el módulo `fs` (File System) para leer, escribir y manipular archivos en el sistema.

**Ejercicio 1**

Escribir en el archivo un mensaje de inicio cada vez que el programa se ejecuta, con la fecha y hora actual en el formato:

[YYYY-MM-DD HH:MM:SS] - Inicio del programa

Simular la ejecución de una tarea que tarda 5 segundos. Mientras la tarea se ejecuta, escribir en log.txt:

[YYYY-MM-DD HH:MM:SS] - Ejecutando tarea...

Cuando la tarea finaliza, escribir en log.txt:

[YYYY-MM-DD HH:MM:SS] - Tarea completada

**Ejercicio 2**

Desarrollar un programa en Node.js que realice las siguientes operaciones utilizando el módulo `fs`:

1. Crear un archivo llamado `datos.txt` y escribir en él el siguiente contenido:

```

Nombre: [Tu Nombre]

Edad: [Tu Edad]

Carrera: [Tu Carrera]

```

2. Leer el archivo `datos.txt` e imprimir su contenido en la consola.

3. Agregar al final del archivo la fecha y hora actuales en el siguiente formato:

```

Fecha de modificación: [YYYY-MM-DD HH:MM:SS]

```

4. Renombrar el archivo `datos.txt` a `informacion.txt`.

5. Eliminar el archivo `informacion.txt` tras 10 segundos de haber sido renombrado.

**Ejercicio 3**

Instrucciones:

Crear un archivo llamado contactos.json con el siguiente contenido inicial:

json

Copiar

Editar

[

{

"nombre": "Juan Pérez",

"telefono": "123-456-7890",

"email": "juan@example.com"

}

]

Crear una función en Node.js que agregue un nuevo contacto al archivo contactos.json. El contacto debe ser un objeto con los siguientes campos:

json

Copiar

Editar

{

"nombre": "[Tu Nombre]",

"telefono": "[Tu Teléfono]",

"email": "[Tu Email]"

}

crear una funcion para Leer y mostrar en la consola todos los contactos almacenados en contactos.json.

Agregar una función que elimine un contacto dado su nombre.

// Código de prueba

agregarContacto('Carlos López', '987-654-3210', 'carlos@example.com');

mostrarContactos();

eliminarContacto('Juan Pérez');

mostrarContactos();

### **Ejercicio 4**

Desarrollar un programa en Node.js que:

1. Reciba como argumento el nombre de un archivo de texto.
2. Lea el contenido del archivo y cuente cuántas veces aparece una palabra específica (también pasada como argumento).
3. Imprima en consola el número de apariciones de la palabra.

#### **Ejemplo de uso:**

node contadorPalabras.js archivo.txt palabras

archivo.txt

Hola mundo, este es un ejemplo de conteo de palabras en un archivo.

Este archivo contiene palabras repetidas. Palabras, palabras y más palabras.

**Salida esperada (si la palabra aparece 5 veces en el archivo):**

La palabra "palabras" aparece 5 veces en el archivo "archivo.txt".

¡Buena suerte!

# Ejercicio 5

Crear un programa en Node.js que copie el contenido de un archivo a otro.  
  
1. Recibir dos parámetros desde la línea de comandos: el archivo de origen y el archivo de destino.  
2. Verificar si el archivo de origen existe. Si no existe, mostrar un mensaje de error.  
3. Copiar el contenido al archivo de destino y mostrar en consola un mensaje confirmando la copia.

# Ejercicio 6

Desarrollar un programa en Node.js que cree un directorio llamado `logs` si no existe y dentro de él cree un archivo `app.log`.  
  
1. Cada vez que el programa se ejecute, agregar al archivo `app.log` una línea con la fecha, hora y un mensaje de 'Ejecución exitosa'.  
2. Implementar una función que permita leer el archivo `app.log` y mostrar en consola las últimas 5 ejecuciones registradas.