COMANDOS

DWEC

12/02/2025

Descargar el git en el windows

Creación del repositorio remoto.

Clonación del repositorio en local.

Meter los ejercicios

Ejercicio 1 -> de la nasa.

Con node

Ejercicio 1 -> Información del sistema.

Ejercicio 2 -> Read

13/02/2025

Ejercicio 3

En el mismo archivo está todo

Funcion info => para el mensaje de info

Funcion escribir => para escribir el mensaje en el archivo y sino existe crearlo

Funcion leer => Que se llama al acabar el escribir para leerlo

Ejercicio 4

Instalacion del chalk => npm i chalk

Su uso esta en la documentación del propio chalk.

Hacer la aplicación.

14/02/2025

Ejercicio 5 – REACT

Vite

Npm init –y -> Iniciar el proyecto, se crea el package.json

Npm create vite@latest

cd “nombre-Proyecto” -> Para entrar al proyecto

npm install -> Instalar las dependencias que están en el package.json

npm run dev -> Para desplegar

17/02/2025

Usando lo dado en la carpeta

App.js

El código proporciona una aplicación de lista de tareas (To-Do List) utilizando React. Permite agregar, eliminar y editar tareas de manera interactiva.

**Funciones principales**

1. **addTask()**:
   * Agrega una nueva tarea a la lista si el campo de entrada (inputValue) no está vacío. Después de agregarla, limpia el campo de entrada.
2. **removeTask(index)**:
   * Elimina una tarea de la lista usando el índice de la tarea seleccionada.
3. **editTask(index)**:
   * Permite iniciar la edición de una tarea. Al seleccionar la tarea para editar, el índice de la tarea se guarda en editingIndex y se carga el valor de la tarea en el campo de edición (editValue).
4. **saveEdit(index)**:
   * Guarda los cambios realizados en la tarea que se está editando, reemplazando la tarea anterior con el valor modificado en editValue. Luego, restablece editingIndex a null para salir del modo de edición.

**Elementos principales**

1. **Entrada de texto**:
   * Un campo para escribir nuevas tareas. Su valor se almacena en inputValue y se actualiza con el evento onInput.
2. **Botón "Agregar"**:
   * Un botón que, al hacer clic, llama a la función addTask para agregar la tarea a la lista.
3. **Lista de tareas (ul)**:
   * Se muestra la lista de tareas. Cada tarea puede ser editada o eliminada. Si una tarea está siendo editada, se muestra un campo de entrada con el valor de la tarea que puede ser modificado.
4. **Botones de edición y eliminación**:
   * Para cada tarea, existen dos botones:
     + **Editar**: Cambia el estado a modo de edición para esa tarea.
     + **Eliminar**: Elimina la tarea de la lista.

**Estado y Funciones de React**

* **useState**: Se utiliza para manejar el estado de las tareas, el valor del input de nuevas tareas, el índice de la tarea que se está editando y el valor de la tarea en edición.

**Comportamiento de la Aplicación**

* Cuando se agrega una tarea, esta aparece en la lista.
* Cuando se edita una tarea, el campo de texto se convierte en un campo de edición, donde se puede modificar el contenido.
* Los botones de "Guardar" y "Eliminar" están disponibles solo cuando una tarea está siendo editada.