

# Task: File Explorer with Nested Folders

## Objective

Build a **React-based File Explorer** that allows users to **navigate, create, rename, delete, and organize files and folders** in a hierarchical structure.

---

## Features & Requirements

### Display a Nested File Structure

- Use a **tree-like structure** where **folders** can contain both **subfolders** and **files**.
- Display folders and files differently (e.g., folder icons, file icons).
- Clicking on a folder should **expand/collapse** its contents.

### User Actions on Files & Folders

#### Create a New Folder/File

- Users should be able to create a **new folder** inside any existing folder.
- Users should be able to create a **new file** inside any folder.

#### Rename a Folder/File

- Users should be able to rename a folder or file by clicking on it or using a right-click context menu.

#### Delete a Folder/File

- Users should be able to delete folders or files.
- Deleting a **folder** should delete all its contents.
- Show a **confirmation prompt** before deleting.

### Drag and Drop Functionality (Bonus)

- Users should be able to **move files and folders** using drag-and-drop.
- Moving a folder should **move all its subfolders and files** inside it.
- Ensure that users **cannot move a folder into itself**.

### Data Persistence

- Use **React State (useState or useReducer)** to manage the file structure.
- Optionally, store the file system in **Local Storage**, so that the structure persists after refresh.

### Search Functionality (Bonus)

- Implement a **search bar** that allows users to **search for files and folders by name**.

- Highlight the matched search results in the UI.
- 

## Example File Structure (JSON Representation)

Your file explorer should be able to manage a data structure like this:

```
{
  "name": "Root",
  "type": "folder",
  "children": [
    {
      "name": "Documents",
      "type": "folder",
      "children": [
        { "name": "Resume.pdf", "type": "file" },
        { "name": "Project.docx", "type": "file" }
      ]
    },
    {
      "name": "Pictures",
      "type": "folder",
      "children": [
        { "name": "Vacation.jpg", "type": "file" },
        { "name": "Profile.png", "type": "file" }
      ]
    },
    { "name": "Notes.txt", "type": "file" }
  ]
}
```

---

## Expected UI Behavior

**Folders can be expanded/collapsed** to show or hide their contents.

Clicking **"New Folder"** inside a folder should create a new folder inside it.

Clicking **"New File"** should create a new file.

Right-clicking should **open a context menu** with **Rename & Delete** options.

Users can **drag and drop files & folders** to organize them.

The file structure should be managed dynamically using **React State**.

---