frontend

**1-Table.jsx**

import axios from "axios";

import { useState, useEffect } from "react";

import { Link, useLocation } from "react-router-dom";

import Cookies from "js-cookie";

export default function Table({ api, styles }) {

const [data, setData] = useState([]);

const [loading, setLoading] = useState(true);

const [error, setError] = useState(null);

const isDoctorSubDepartment = api.includes("getDoctorSubDepartment");

const isCourseSubDepartment = api.includes("getCourseSubDepartment");

const isSubDepartment = api.includes("sub\_department")

const isMaterials = api.includes("materials");

const location = useLocation()

useEffect(() => {

async function getData() {

try {

const response = await axios.get(api);

if (response.status === 200 && Array.isArray(response.data)) {

setError(null);

setData(response.data);

} else {

if(isMaterials){

setError("There is no materials for this sub\_department")

} else {

setError("Unexpected response format or no data available.");

}

}

} catch (err) {

setError(err.response?.data.message || "Unable to fetch data.");

} finally {

setLoading(false);

}

}

if (api) getData();

}, [api]);

if (loading) return <div className="dark:text-white text-black">Loading...</div>;

if (error) return <div className="text-red-500">{error}</div>;

if (data.length === 0) return <div className="text-center text-gray-500">No data available.</div>;

const headers = Object.keys(data[0]);

const linkMappings = {

doctor\_name: (row) => `doctor/${row.doctor\_id}`,

course\_name: (row) => `course/${row.material\_id}`,

name: (row) => `${row.sub\_dep\_id}`

};

function handleClick() {

const currentValue = Cookies.get('check');

if (currentValue === 'false') {

Cookies.set('check', true);

}

else {

// Cookies.set('check', false);

return

}

}

const getLink = (header, row) => {

const link = linkMappings[header] ? linkMappings[header](row) : null;

if (location.pathname.match(/^\/materials\/\d+$/) && header === "name" && link !== null) {

const newId = link;

return location.pathname.replace(/\/\d+$/, `/${newId}`);

}

return link;

};

return (

<div className={`max-w-screen-2xl mt-8 ${styles}`}>

<div className="relative overflow-x-auto shadow-md sm:rounded-lg">

<table className="w-full text-sm text-left rtl:text-right text-gray-500 dark:text-gray-400">

<thead className="text-xs text-white uppercase bg-primary dark:bg-third dark:text-white">

<tr>

{headers.map((header, index) => (

<th

key={index}

scope="col"

className={`px-6 py-3 text-center ${

(isDoctorSubDepartment || isCourseSubDepartment || isSubDepartment || isMaterials) && index === 0 ? "hidden" : ""

}`}

>

{(isDoctorSubDepartment || isCourseSubDepartment || isSubDepartment || isMaterials) && index === 0 ? null : header}

</th>

))}

</tr>

</thead>

<tbody>

{data.map((row, rowIndex) => (

<tr

key={rowIndex}

className={`${

rowIndex % 2 === 0

? "bg-white dark:bg-gray-900"

: "bg-gray-50 dark:bg-gray-800"

} border-b dark:border-gray-700`}

>

{headers.map((header, colIndex) => (

<td

key={colIndex}

className={`px-6 py-4 text-center ${

(isDoctorSubDepartment || isCourseSubDepartment || isSubDepartment || isMaterials) && colIndex === 0

? "hidden"

: ""

}`}

>

{getLink(header, row) ? (

<Link

to={getLink(header, row)}

className="text-blue-500 hover:underline"

onClick = {() => handleClick()}

>

{row[header]}

</Link>

) : (

row[header]

)}

</td>

))}

</tr>

))}

</tbody>

</table>

</div>

</div>

);

}

**2-Materials.jsx**

import { LinkBox, Table, Navbar, Materials2 } from './index';

import React, { useState, useEffect } from 'react';

import useFetch from '../hooks/useFetch';

import Cookies from "js-cookie";

import axios from 'axios'

import { useNavigate, useParams } from 'react-router-dom'

export default function Materials() {

const [activeIndex, setActiveIndex] = useState(null);

const [subDepartmentId, setSubDepartmentId] = useState([]);

const { data, error, loading } = useFetch('/departments');

const navigate = useNavigate()

const params = useParams()

const isNumber = isNaN(params.id)

const getApi = (index) => {

const url = `http://localhost:4000/university/sub\_department/${data[index]?.dep\_id}`;

return url;

};

function handleShow(index) {

const cookieValue = Cookies.get('check');

if (!cookieValue) {

Cookies.set('check', 'true');

} else {

Cookies.set('check', cookieValue === 'true' ? 'false' : 'true');

}

navigate('/materials', { replace: true });

setActiveIndex((prev) => (prev === index ? null : index));

}

if (loading) return <div>Loading...</div>;

if (error) return <div>Error: {error}</div>;

if (!Array.isArray(data) || data.length === 0) return <div>No departments available.</div>;

return (

<div className="max-w-screen-2xl m-auto min-h-screen pt-8">

<Navbar />

<div className="pt-20">

<div className="flex flex-wrap justify-center gap-8">

{data.map((department, index) => (

<LinkBox

key={index}

styles="basis-2/5"

title={department.department\_name}

toggle={() => handleShow(index)}

desc={`View details for ${department.department\_name}`}

/>

))}

</div>

<div className="flex justify-center pb-8">

{activeIndex !== null && (

<Table api={getApi(activeIndex)} styles="basis-4/5" />

)}

</div>

{/\*{Cookies.get('check') && <Materials2 index={activeIndex !== null ? data[activeIndex]?.dep\_id : null} />}\*/}

{Cookies.get('check') && !isNumber &&<Materials2 />}

</div>

</div>

);

}