

방미소 React 연습문제 #09

2022-05-16

App.js

```
import { useCallback, useEffect, useState } from "react";
import axios from "axios";

import Professor from "../components/Professor";
import Spinner from "../components/Spinner";
import Student from "../components/Student";

const App = () => {
  // 로딩 바
  const [loading, setLoading] = useState(false);
  // 데이터
  const [data, setData] = useState([]);
  // deptno 전달
  const [deptno, setDeptno] = useState(false);

  // 학과명 데이터 가져오기
  useEffect(() => {
    (async () => {
      setLoading(true);
      try {
        const res = await axios.get("http://localhost:3001/department");
        setData(res.data);
      } catch (e) {
        alert("axios 오류가 났어요");
      } finally {
        setLoading(false);
      }
    })();
  }, []);

  // 드롭다운 클릭
  const onDropdown = useCallback((e) => {
    const option = e.target.selectedIndex;
    const deptno = e.target[option].dataset.deptno;
    setDeptno(Number(deptno));
  }, []);

  return (
    <div>
      {/* 로딩 아이콘 */}
      <Spinner visible={loading} />
      <h1>Exam 09</h1>
      <hr />
    </div>
  );
};
```

```

    { /* 드롭다운 */ }
    <select onChange={onDropdown}>
      <option value="">-- 학과명 --</option>
      {data.map((v) => {
        return (
          <option key={v.id} value={v.dname} data-deptno={v.id}>
            {v.dname}
          </option>
        );
      })}
    </select>

    <Student deptno={deptno} />
    <Professor deptno={deptno} />
  </div>
);
};

export default App;

```

Student.js

```

import Spinner from "./Spinner";
import axios from "axios";
import React, { useEffect, useState } from "react";
import PropTypes from "prop-types";

const Student = ({ deptno }) => {
  console.log(deptno);
  const [loading, setLoading] = useState(false);
  const [student, setStudent] = useState([]);

  // deptno에 따라 목록 조회
  useEffect(() => {
    (async () => {
      setLoading(true);
      try {
        const res = await axios.get(`http://localhost:3001/student?deptno=${deptno}`);
        setStudent(res.data);
      } catch (e) {
        alert("axios 오류가 발생했습니다.");
      } finally {
        setLoading(false);
      }
      console.log(student);
    })();
  }, [deptno]);

```

```

return (
  <div>
    <Spinner visible={loading} />
    <h2>학생목록</h2>
    <table border="1">
      <thead>
        <tr>
          <th>학번</th>
          <th>이름</th>
          <th>아이디</th>
          <th>학년</th>
          <th>주민번호</th>
          <th>생년월일</th>
          <th>연락처</th>
          <th>키</th>
          <th>몸무게</th>
          <th>소속학과번호</th>
          <th>담당교수번호</th>
        </tr>
      </thead>
      <tbody>
        {student.map((v) => {
          return (
            <tr key={v.id}>
              <td>{v.id}</td>
              <td>{v.name}</td>
              <td>{v.userid}</td>
              <td>{v.grade}</td>
              <td>{v.idnum.substring(0, 6)}-*****</td>
              <td>{v.birthdate}</td>
              <td>{v.tel}</td>
              <td>{v.height}</td>
              <td>{v.weight}</td>
              <td>{v.deptno}</td>
              <td>{v.profno}</td>
            </tr>
          );
        })}
      </tbody>
    </table>
  </div>
);
};

Student.defaultProps = {
  deptno: -1,
};

Student.propTypes = {
  deptno: PropTypes.number,
};

```

```
export default Student;
```

Professor.js

```
import Spinner from "./Spinner";
import axios from "axios";
import React, { useEffect, useState } from "react";
import PropTypes from "prop-types";
import dayjs from "dayjs";

const Professor = ({ deptno }) => {
  console.log(deptno);
  const [loading, setLoading] = useState(false);
  const [professor, setProfessor] = useState([]);

  // deptno에 따라 목록 조회
  useEffect(() => {
    (async () => {
      setLoading(true);
      try {
        const res = await axios.get(`http://localhost:3001/professor?deptno=${deptno}`);
        setProfessor(res.data);
      } catch (e) {
        alert("axios 오류가 발생했습니다.");
      } finally {
        setLoading(false);
      }
      console.log(professor);
    })();
  }, [deptno]);

  return (
    <div>
      <Spinner visible={loading} />
      <h2>교수목록</h2>
      <table border="1">
        <thead>
          <tr>
            <th>교수번호</th>
            <th>이름</th>
            <th>아이디</th>
            <th>직급</th>
            <th>급여</th>
            <th>입사일</th>
            <th>보직수당</th>
            <th>소속학과번호</th>
          </tr>
```

```
    </thead>
    <tbody>
      {professor.map((v) => {
        return (
          <tr key={v.id}>
            <td>{v.id}</td>
            <td>{v.name}</td>
            <td>{v.userid}</td>
            <td>{v.position}</td>
            <td>{v.sal}</td>
            <td>{dayjs(v.hiredate).format("YYYY-MM-DD")}</td>
            <td>{v.comm}</td>
            <td>{v.deptno}</td>
          </tr>
        );
      })}
    </tbody>
  </table>
</div>
);
};

Professor.defaultProps = {
  deptno: -1,
};

Professor.propTypes = {
  deptno: PropTypes.number,
};

export default Professor;
```

Exam 09

-- 학과명 --

학생목록

학번	이름	아이디	학년	주민번호	생년월일	연락처	키	몸무게	소속학과번호	담당교수번호
----	----	-----	----	------	------	-----	---	-----	--------	--------

교수목록

교수번호	이름	아이디	직급	급여	입사일	보직수당	소속학과번호
------	----	-----	----	----	-----	------	--------

Exam 09

멀티미디어학과

학생목록

학번	이름	아이디	학년	주민번호	생년월일	연락처	키	몸무게	소속학과번호	담당교수번호
10201	김진영	simply	2	820606-*****	1982-06-05T15:00:00.000Z	055)419-6328	164	48	102	9905
10202	오유석	yousuk	4	770912-*****	1977-09-11T15:00:00.000Z	051)724-9618	177	92	102	9905
10203	하나리	hanal	1	850109-*****	1985-01-08T15:00:00.000Z	055)296-3784	160	68	102	
10204	윤진욱	samba7	3	790402-*****	1979-04-01T15:00:00.000Z	053)487-2698	171	70	102	9905

교수목록

교수번호	이름	아이디	직급	급여	입사일	보직수당	소속학과번호
9904	염일웅	blue77	전임강사	240	1998-10-11		102
9905	권혁일	refresh	교수	450	1986-02-11	25	102