

방미소 React 연습문제 #10

2022-05-17

App.js

```
import React from "react";
import styled from "styled-components";
import Spinner from "../components/Spinner";
import Table from "../components/Table";

import useAxios from "axios-hooks";
import useMountedRef from "../hooks/useMountedRef";

const SelectContainer = styled.div`
  position: sticky;
  top: 0;
  background-color: #fff;
  border-top: 1px solid #eee;
  border-bottom: 1px solid #eee;
  padding: 10px 0;
  margin: 0;

  select {
    margin-right: 15px;
    font-size: 16px;
    padding: 5px 10px;
  }
`;

const URL = "http://localhost:3001/traffic_acc";

const App = () => {
  const [{ data, loading, error }, refetch] = useAxios(URL);
  const [year, setYear] = React.useState("");
  const mountedRef = useMountedRef();

  const onSelectChange = React.useCallback((e) => {
    e.preventDefault();

    // 드롭다운의 입력값 취득
    const current = e.target;
    const value = current[current.selectedIndex].value;
    setYear(value);
  }, []);

  // state 상태값이 변경되었을 때 실행될 hook
  React.useEffect(() => {
    // 컴포넌트가 화면에 마운트 된 이후에만 동작하도록 한다.
    if (mountedRef.current) {
```

```

// 상태값중에서 빈값이 아닌 항목들을 옮겨담는다.
const params = {};
if (year) {
  params.year = parseInt(year);
}

// Ajax 재요청
refetch({
  params: params,
});
}
// hook 함수 안에서 다른 상태값을 사용할 경우 해당 상태값 모니터링해줘야
}, [mountedRef, refetch, year]);

if (error) {
  console.error(error);

  // 컴포넌트 자체가 함수이고, 함수가 실행도중 리턴을 하므로
  // 이 내용을 화면에 표시하고 컴포넌트의 실행은 중단한다.
  return (
    <div>
      <h1>{error.code} Error 오류</h1>
      <hr />
      <p>{error.message}</p>
    </div>
  );
}

// 메인 화면 구성
return (
  <div>
    <h1>Exam 10</h1>

    {/* 로딩바 */}
    <Spinner visible={loading} />

    {/* 검색 조건 드롭다운 박스 */}
    <SelectContainer>
      <select name="year" onChange={onSelectChange}>
        <option value="">-- 년도 선택 --</option>
        {[...new Array(2018 - 2005 + 1)].map((v, i) => {
          return (
            <option key={i} value={2005 + i}>
              {2005 + i}년도
            </option>
          );
        })}
      </select>
    </SelectContainer>

    {data && (
      <Table>
        <thead>
          <tr>

```

```

        <th>번호</th>
        <th>년도</th>
        <th>월</th>
        <th>교통사고 건수</th>
        <th>사망자 수</th>
        <th>부상자 수</th>
    </tr>
</thead>
<tbody>
    {data.map(({ id, year, month, accident, death, injury }, i) => {
        return (
            <tr key={i}>
                <td>{id}</td>
                <td>{year}년</td>
                <td>{month}월</td>
                <td>{accident.toLocaleString()}건</td>
                <td>{death.toLocaleString()}명</td>
                <td>{injury.toLocaleString()}명</td>
            </tr>
        );
    })}
</tbody>
<tfoot>
    <tr>
        <th colspan="3">합계</th>
        <th>
            {data
                .map((v, i) => v.accident)
                .reduce((p, c) => p + c, 0)
                .toLocaleString()}
            건
        </th>
        <th>
            {data
                .map((v, i) => v.death)
                .reduce((p, c) => p + c, 0)
                .toLocaleString()}
            명
        </th>
        <th>
            {data
                .map((v, i) => v.injury)
                .reduce((p, c) => p + c, 0)
                .toLocaleString()}
            명
        </th>
    </tr>
</tfoot>
</Table>
    )}
</div>
);
};

```

```
export default App;
```

연습 코드

드롭다운을 map으로 구현하려던 중 무한 callback 문제가 발생하여 해결하지 못한 코드입니다.

```
import React, { useState, useEffect, useCallback } from "react";
import styled from "styled-components";
import useAxios from "axios-hooks";

import Spinner from "../src/components/Spinner";
import Table from "../src/components/Table";
import useMountedRef from "../src/hooks/useMountedRef";

const SelectContainer = styled.div`
  position: sticky;
  top: 0;
  background-color: #fff;
  border-top: 1px solid #eee;
  border-bottom: 1px solid #eee;
  padding: 10px 0;
  margin: 0;

  select {
    margin-right: 15px;
    font-size: 16px;
    padding: 5px 10px;
  }
`;

const URL = "http://localhost:3001/traffic_acc";

const Traffic = () => {
  /** 데이터 가져오기 */
  const [{ data, loading, error }, refetch] = useAxios(URL);
  const [state, setState] = useState("");
  const mountedRef = useMountedRef();
  console.log(data);

  if (error) {
    return (
      <h1>
        Error! {error.name} {error.message}
      </h1>
    );
  }

  /** 연도 배열 가져오기 */
  const onMakeList = () => {
    setInterval(() => {
```

```

    const arr = data.map((v) => v.year);
    const year = Array.from(new Set(arr));

    year.map((v, i) => {
      return (
        <option value={v} key={i}>
          {v}
        </option>
      );
    });
  }, 500);
};

/** 드롭다운 */
const onSelectChange = useCallback((e) => {
  e.preventDefault();

  // 드롭다운의 입력값 취득
  const current = e.target;
  const value = current[current.selectedIndex].value;

  // 상태값 갱신
  setState(value);
}, []);

/** 드롭다운 이벤트 */
useEffect(() => {
  if (mountedRef.current) {
    const params = {};
    for (const key in state) {
      if (state[key]) {
        params[key] = state[key];
      }
    }

    // Ajax 재요청
    refetch({
      params: params,
    });
  }
}, [state]);

return (
  <div>
    {/* 로딩바 */}
    <Spinner visible={loading} />
    <SelectContainer>
      <select name="year">
        <option value="">-- 연도 선택 --</option>
        {}
      </select>
    </SelectContainer>
    <Table>
      <thead>

```

```

        <tr>
          <th>번호</th>
          <th>년도</th>
          <th>월</th>
          <th>교통사고 건수</th>
          <th>사망자 수</th>
          <th>부상자 수</th>
        </tr>
      </thead>

      <tbody>
        <tr>
          <td></td>
        </tr>
      </tbody>
    </Table>
  </div>
);
};

export default Traffic;
```

Exam 10

2013년도 ▾

| 번호 | 년도 | 월 | 교통사고 건수 | 사망자 수 | 부상자 수 |
|-----|-------|-----|----------|--------|----------|
| 97 | 2013년 | 1월 | 16,016건 | 389명 | 25,094명 |
| 98 | 2013년 | 2월 | 14,187건 | 335명 | 22,255명 |
| 99 | 2013년 | 3월 | 17,465건 | 409명 | 26,461명 |
| 100 | 2013년 | 4월 | 18,031건 | 380명 | 27,795명 |
| 101 | 2013년 | 5월 | 19,686건 | 420명 | 29,676명 |
| 102 | 2013년 | 6월 | 18,329건 | 436명 | 27,673명 |
| 103 | 2013년 | 7월 | 18,441건 | 444명 | 28,387명 |
| 104 | 2013년 | 8월 | 18,691건 | 412명 | 29,190명 |
| 105 | 2013년 | 9월 | 17,806건 | 430명 | 26,938명 |
| 106 | 2013년 | 10월 | 19,797건 | 499명 | 29,507명 |
| 107 | 2013년 | 11월 | 19,396건 | 479명 | 28,986명 |
| 108 | 2013년 | 12월 | 17,509건 | 459명 | 26,749명 |
| 합계 | | | 215,354건 | 5,092명 | 328,711명 |

