

08-simple-ajax

#01. 프로젝트 생성 및 초기화

1) 프로젝트 생성

프로젝트 이름은 영어 소문자만 사용 가능함.

```
yarn create react-app 07-hook-event
```

2) 프로젝트 생성 후 기초작업

프로젝트 초기화

대상	작업 내용
src폴더	App.css, App.test.js, index.css, logo.svg, setupTests.js, reportWebVitals.js 삭제
App.js	App.css와 logo.svg에 대한 참조(import) 구문 제거
index.js	index.css와 reportWebVitals.js에 대한 참조(import) 구문 제거 맨 마지막 행에 있는 <code>reportWebVitals()</code> 부분 삭제

3) 추가 패키지 설치

패키지 이름	설명
react-router-dom	SPA앱을 만들 때 사용. URL에 따라 실행할 Javascript를 분기한다.
react-helmet-async	<code><head></code> 태그 내의 SEO관련 태그를 설정한다.
sass	sass와 scss 컴파일 기능 제공
styled-components	styled component 지원
dayjs	날짜 처리 기능 제공 (리액트와 직접적인 연관은 없음)
axios	Ajax 요청 라이브러리
react-loader-spinner	로딩바(spinner) 컴포넌트

```
yarn add react-router-dom react-helmet-async sass styled-components dayjs  
axios react-loader-spinner
```

4) Router 적용

index.js

import 구문 추가

```
import { BrowserRouter } from 'react-router-dom';
```

index.js 파일에서 `<App />`을 `<BrowserRouter><App /></BrowserRouter>`로 변경

App.js

```
import { NavLink, Routes, Route } from "react-router-dom";
```

혹은

```
import { Link, Routes, Route } from "react-router-dom";
```

5) 프로젝트 실행

프로젝트를 VSCode로 열고, `Ctrl + ~`를 눌러 터미널 실행

```
yarn start
```

#02. 컴포넌트 성능 최적화 방법

1) `React.memo()` 사용.

컴포넌트 함수를 export 하기 전 `React.memo()`로 처리한다.

```
import React from 'react';

const News = () => {
  //... 생략 ...
};

// React.memo()를 사용하여 함수형 컴포넌트의 리렌더링 성능을 최적화
export default React.memo(NewsList);
```

2) `React.useCallback()` 사용

이벤트 핸들러 함수에 `React.useCallback()`을 적용한다.

```
const 함수이름 = React.useCallback(e => {
  //... 이벤트 처리 구현
}, []);
```

3) 함수형 상태값 업데이트

`useState()`로 상태값을 업데이트 할 때 상태값이 배열이나 JSON객체 타입일 경우 단순히 값을 전달하는 것이 아니고 함수형으로 처리한다.

적용전

```
const [target, setTarget] = React.useState([]);

// 상태값으로 적용할 배열 생성
const newArray = [...];
setTarget(newArray);
```

적용후

```
const [target, setTarget] = React.useState([]);

setTarget(function(target) {
  // target(원본)을 활용하여 상태값으로 활용할 새로운 배열 생성
  return 새로운배열;
});
```

적용후(축약형)

```
const [target, setTarget] = React.useState([]);
setTarget(target => 새로운배열);
```