

● 리액트에서 자주 쓰는 자바스크립트 형식

※알고있으면 유용한 자바스크립트 문법 (깃북 참고하기)

<https://learnjs.vlpt.us/useful/>

1. 비구조화 할당 (구조분해) 문법

1) 리액트

* 컴포넌트에 props를 받아올 때

```
const {todo, checkedTodo, deleteTodo} = props  
  
const TodoComp = ({todo}) => {
```

2) 라우터

* NavLink에서 함수를 통해서 isActive값을 받아올 때

```
<NavLink to="/articles"  
  className={({isActive})=>isActive?"link-style":undefined}  
>  
  Articles  
</NavLink>
```

● 사용하는 데이터 구조 작성하기

- 데이터를 만들때는 화면상에 있는 데이터와 그 데이터를 유일하게 알려주는 id값을 항상 함께 사용해야한다 (컴퓨터는 값을 구분하지 못함으로 id로 구분)

```
{  
  "id" : 1,  
  "title" : "첫번째 게시물입니다",  
  "content" : "내용을 작성하였습니다",  
  "date" : "2023-04-14",  
  "writer" : "green"  
},
```

● Context API를 이용한 리액트 공용 데이터 사용

- 공용 데이터 사용이 필요한 이유

react-router-dom을 쓰면서 데이터를 props를 통해서 옮기기 어려워짐.

> 각각의 페이지는 부모 자식 관계가 아니라 형제

app.js

```
<Routes>
  <Route path="/" element={<Layout/>}>
    <Route path="/" element={<Home/>}/>
    <Route path="/boardlist" element={<BoardList/>}/>
    <Route path="/boardlist/:id" element={<Board />}/>
  </Route>
</Routes>
```

부모(app.js)에서 자식(Route의 element)에 props값을 전달할 수 있지만 번거로우므로 Context를 이용하여 **부모-자식-자손으로 값을 전달**하는 것이 아닌 **부모에서 바로 자손으로 값을 전달**할 수 있다

* Context 사용법

1) Context 생성

```
import React from "react";

const ThemeContext = React.createContext('dark');

export default ThemeContext;
```

2) App.js에 생성한 Context import
데이터를 사용할 모든 컴포넌트를 감싼다

```
import ThemeContext from './context/ThemeContext';

function App() {
  return (
    <div className="App">
      <ThemeContext.Provider value='blue'>
        <ContextBox />
      </ThemeContext.Provider>
      .
      .
      .
    </div>
  );
}
```


- 3) 사용할 컴포넌트에 생성한 Context와 useContext import
Context의 value값 가져옴


```
import ThemeContext from '../context/ThemeContext'
import { useContext } from 'react'

function ContextText () {

  const text = useContext(ThemeContext)
  return <p>{text}</p>
}
```

위와 같은 방식으로

 DataContext.jsx

 ObjectContext.jsx

객체 전달하는 ObjectContext와

useState의 값을 분리해서 객체에 담아 전달하는DataContext를 구분해서 확인하세요

<https://github.com/hj-Seong/2023-react/tree/master/context-text>