

## TodoList 를 RestApi 로 구현하기

- 제출 파일 : **TodoRestApi.jsx**
- 참조파일 : RestGet11.jsx, RestPutDelete14.jsx, RestPost15.jsx, TodoComplete.jsx

1. 데이터 예 : db/todo.json 파일 모습

```
{
  "todos": [
    {
      "id": "1요리",
      "title": "요리",
      "isDone": false
    },
    {
      "id": "2청소",
      "title": "청소",
      "isDone": true
    },
    {
      "id": "3공부",
      "title": "공부",
      "isDone": true
    }
  ]
}
```

2. json-server 에 todos 를 요청한 결과 및 웹에 display 된 모습

The image displays two browser windows side-by-side. The left window, titled 'localhost:3001/todos/', shows a JSON array of three todo items: '1요리' (not done), '2청소' (done), and '3공부' (done). The right window, titled 'React App' and 'localhost:3000', shows a web interface titled 'Todo List'. It lists the same three items with checkboxes: '요리' is unchecked, while '청소' and '공부' are checked. Below the list is an input field, and 'Add' and 'Delete' buttons. At the bottom, it says 'Numbers : 3'.

### 3. 이 프로그램의 특성

- 기존 todoList 프로그램은 사용자로 부터 들어온 데이터들이 메모리에 저장되고, 브라우저의 새로 고침(refreshing)을 하면 모두 지워지는 특징이 있다.
- 이 프로그램에서는 데이터가 db/todo.json 에 저장되므로 **새로 고침을 해도 지워지지 않는다.**
- 수업시간에 소개한 REST api 를 이용하여 이를 구현한다.

### 4. 고려 사항

- CRUD (Create, Read, Update, Delete)에 해당하는 메소드는 각각 Post, Get, Put, Delete 이다.
- 이 중에서 가장 기본은 Get 이다. Fetch 함수에서 Get 은 어떻게 표현하는가? 또한, Get 은 어떤 hooks 함수와 함께 사용되는가?
- 화면이 rendering 되기 위해서는 db 로부터 읽어온 data 를 setTodos(data) 형태로 실행해야 한다. 이에 해당하는 코드는 어디에 위치해야 하는가?
- Create 에 해당하는 fetch-Post 는 언제 실행해야 하는가?
- Update 에 해당하는 fetch-Put 는 언제 실행해야 하는가?
- Delete 에 해당하는 fetch-Delete 는 언제 실행해야 하는가?
- Create/Update/Delete 에 해당하는 fetch 함수를 실행한 후, 이 내용대로 db/todo.json 파일의 "todos" 가 수정되었는 지를 확인하라. 수정된 db 내용은 웹 화면에 rendering 되어야 한다. 그런데, db 의 내용이 화면에 rendering 되는 상황은 이미 앞서 실행된 적이 있다. 따라서 Create/Update/Delete 에 의해서 내용이 수정되었다는 사실만 전달하면 rendering 자동적으로 해결된다. 이 상황을 어떻게 구현할 수 있는가? (힌트 : 각각의 경우에 대해서 **true/false 값을 갖는 상태 변수를 사용하고**, fetch 함수 실행 전과 후에 이 값을 바꿔주고, 이 변수의 값이 수정되었다는 사실을 알려 주면 됨)
- 각 todo 마다 id 를 내포하는데, 이 값은 지정되었다가 보다는 자율적으로 정하면 될 것 같다. 단순한 정수 1, 2, 3 등으로 하는 경우, "add"와 "delete" 에 의해서 todos 의 내용이 수정되므로, id 값을 일관성 있게 유지하기 어렵다.