

AI 활용 빅데이터분석 풀스택웹서비스 SW 개발자 양성과정

React



jsonserver

빠르고 쉽게 RESTful API를 구축할 수 있는 개발 도구

- **빠른 설정**
별도의 서버 설정이나 복잡한 구성 없이 간단한 RESTful API 서버를 구축
- **모든 HTTP 메서드 지원**
GET, POST, PUT, PATCH, DELETE 등의 HTTP 메서드를 지원
이를 통해 CRUD (create, Read, update, Delete) 작업을 손쉽게 수행
- **자동화된 라우팅**
JSON 파일의 구조를 기반으로 자동으로 라우팅을 생성하여 데이터베이스 설정
없이 간단하게 데이터를 관리
- **필터링 및 페이징**
쿼리 매개변수를 사용하여 데이터를 필터링하거나 페이징
- **정렬 및 검색**
데이터를 정렬하거나 특정 키워드로 검색

- **설치**
`npm install -g json-server`
- **실행**
`npx json-server --watch db.json --port 3005`

```
{
  "posts": [ ...
  ],
  "comments": [ ...
  ],
  "todos": [
    {
      "id": "5935",
      "content": "HTML",
      "isDone": false
    },
    { ...
    }
  ]
}
```

입력(POST)

자바스크립

- ☐ HTML
- ☐ 리액트

입력

삭제

삭제

```
<div className="w-full flex justify-center items-center">
  <div className="w-4/5">
    <TailInput type = "text"
      |         |         |
      |         |         | inputRef = {txt1}
      |         |         | ph = "할일입력" />
    </div>
    <div className="w-1/5">
      <TailButton caption = "입력"
        |         |         |
        |         |         | color = "blue"
        |         |         | handleClick = {handlePost}
        |         |         | />
    </div>
  </div>
</div>
```

```
const handlePost = () => {
  const item = {content : txt1.current.value, isDone:false}
  fetch('http://127.0.0.1:3005/todos', {
    method: 'POST',
    headers: {
      'Content-Type': 'application/json'
    },
    body: JSON.stringify(item)
  })
  .then(resp => resp.json())
  .then(data => setTodos([data, ...todos]))
}
```

삭제(DELETE)

자바스크립트

입력

삭제

삭제

☐

HTML

☐

리액트

```
useEffect(()=>{
  if (!todos) return ;

  let tm = todos.map(item => <li key={item.id}
    className="flex w-full justify-center items-center">
    <span className="w-1/5">
      <input type="checkbox"
        checked={item.isDone}
        onChange={() => handlePut(item.id,item.isDone)} />
    </span>
    <span className="w-3/5">{item.content}</span>
    <span className="w-1/5">
      <TailButton caption = "삭제"
        color = "blue"
        handleClick = {(()) => handleDel(item.id)}
      />
    </span>
  </li>)

  setTags(tm);
}, [todos])

const handleDel = (id) => {
  fetch(`http://127.0.0.1:3005/todos/${id}`, {
    method: 'DELETE'
  })
  .then(() => setTodos(todos.filter(item => item.id !== id)))
}
```

수정(PUT)

자바스크립트

입력

☐

HTML

삭제

☐

리액트

삭제

```
useEffect(()=>{
  if (!todos) return ;

  let tm = todos.map(item => <li key={item.id}
    className="flex w-full justify-center items-center">
    <span className="w-1/5">
      <input type="checkbox"
        checked={item.isDone}
        onChange={() => handlePut(item.id,item.isDone)} />
    </span>
    <span className="w-3/5">{item.content}</span>
    <span className="w-1/5">
```

```
const handlePut = (id, isDone) => {
  const item = todos.filter(i => i.id === id)[0] ;
  item.isDone = !isDone ;
  console.log(item)
  fetch(`http://127.0.0.1:3005/todos/${id}`, {
    method: 'PUT',
    headers: {
      'Content-Type': 'application/json'
    },
    body: JSON.stringify(item)
  })
  .then(resp => resp.json())
  .then(data => setTodos(todos.map(i => i.id === id ? data : i)))
}
```