

W12-P1: Insert a blog using local json array

The screenshot shows the Postman interface with a successful API call to `http://localhost:5000/api/blogs`. The response body is a JSON array containing two blog entries. The second blog entry is highlighted with a red box.

```
1 {
2   "blogList_xx": [
3     {
4       "id": 7,
5       "img": "/images/photo-7.jpg",
6       "remote_img": "",
7       "category": "travel",
8       "title": "sun rise from the mountain",
9       "desc": "Lorem ipsum dolor, sit amet consectetur adipisicing elit."
10    },
11    {
12      "id": 20,
13      "img": "/images/my-1.jpg",
14      "remote_img": "",
15      "category": "travel",
16      "title": "208410331 liang yu",
17      "desc": "Mid2-Exam"
18    }
19  ]
20 }
```

The screenshot shows a browser window with several tabs open. The active tab is a Postman collection named "w12-mid21_31". On the left, there's a code editor with two files: "server_xx.js" and "dataBlogs_xx.js". "server_xx.js" contains Node.js server logic for handling API requests like GET /api/blogs, POST /api/blogs, etc. On the right, the Postman interface displays a POST request to "http://localhost:5000/api/blogs". The "Body" tab is selected, showing a JSON payload:

```

1 {
2   "blog": {
3     "id": 20,
4     "img": "/images/my-1.jpg",
5     "remote_img": "",
6     "category": "travel",
7     "title": "208410331 liang yu",
8     "desc": "Mid2-Exam"
9   }
10 }

```

Below the JSON payload, the response status is shown as 200 OK with a response time of 7 ms and a size of 395 B.

f7bd345 liangyu9103 Wed May 8 19:53:25 2024 +0800 ### W12-P1: Insert a blog using

W12-P2: Update a blog with id=20

The screenshot shows a developer's workspace with several tabs open:

- Code Editor:** Shows the `server_xx.js` file containing Node.js code for a blog API. A specific section of the code is highlighted with a red box, showing the implementation of the `PATCH /api/blogs/:id` endpoint.
- Terminal:** Shows the command `node -server_xx` running in the background, with logs indicating successful requests for various endpoints like `/api/blogs`, `/api/blogs/20`, and `/api/blogs/clearAll`.
- Postman API Test:** An open test case titled "Update post id=20" is shown. It uses the `PATCH` method to update a blog entry with `id=20`. The request body is set to `JSON` and contains the following data:

```
1 {  
2   "img": "/images/my-1.jpg",  
3   "remote_img": "",  
4   "category": "lifestyle",  
5   "title": "208410331 liang yu",  
6   "desc": "w12 assignment"  
7 }
```
- Network Tab:** Shows the response from the `PATCH /api/blogs/20` request, which is a `200 OK` status with a response time of 7 ms and a size of 289 B. The response body is:

```
1 {  
2   "msg": "blog updated"  
3 }
```

da24083 liangyu9103

Wed May 8 20:27:57 2024 +0800

W12-P2: Update a blog with

W12-P3: Use React Query to read all blogs from Node server

The screenshot shows a browser window displaying a list of blogs and a code editor showing the corresponding React Query code.

Browser View: The browser shows a list of 10 blogs categorized into TRAVEL and LIFESTYLE. Each blog entry includes a thumbnail, title, category, description, and edit/delete links. A specific blog entry is highlighted with a red box, showing its details:

- Title:** 208410331 Liang Yu
- Category:** LIFESTYLE
- Description:** "w12 assignment"
- ID:** "20"
- Image:** "/images/my-1.jpg"
- Remote Image:** ""
- Title:** "208410331 liang yu"
- Length:** 7

Code Editor View: The code editor shows the `Mid2NodeBlogLocal_31.jsx` file. The `useQuery` hook is used to fetch data from the Node server. The `queryFn` is defined as an asynchronous function that uses `axios.get` to retrieve data from the URL `'http://localhost:5000/api/blogs'`. The retrieved data is stored in the `data.blogList_xx` variable.

```
JS server_xx.js | Mid2NodeBlogLocal_31.jsx | w12-mid21_31
client_xx > src > pages > mid1_xx > Mid2NodeBlogLocal_31.jsx > [e] Mid2NodeBlogLocal_31
10 import { LiaMugHotSolid, LiaTheRedYeti } from 'react-icons/lia';
11 import { CiGlobe } from 'react-icons/ci';
12
13 const Mid2NodeBlogLocal_31 = () => {
14   const [name, setName] = useState('Liang');
15   const [id, setId] = useState(123456789);
16
17   const queryClient = useQueryClient();
18
19   // Read
20   const { data, isLoading, isError } = useQuery({
21     queryKey: ['blogs_xx'],
22     queryFn: async () => {
23       const { data } = await axios.get('http://localhost:5000/api/blogs')
24       //console.log('blogs_31 data', data);
25       return data.blogList_xx;
26     },
27   });
28   console.log('blog data', data);
29
30   // Create
31   // const { mutate: addBlog } = useMutation();
32
33   // Update
34   // const { mutate: updateBlog } = useMutation();
35
36   // Delete
37   // const { mutate: deleteBlog } = useMutation();
38
39   // clearAll
40   // const { mutate: clearAll } = useMutation();
41
42   // Load All
43   // const { mutate: loadAll } = useMutation();
44
```

66a2272 liangyu9103

Wed May 8 21:01:12 2024 +0800 ### W12-P3: Use React Query to

W12-P4: Insert a blog using React Query

The screenshot shows a web application interface with a header containing tabs for 'tku', 'iClass', 'GitHub', 'SQL | Supabase', 'Update post id=20 - My Work', and 'Vite + React'. The main content area displays a list of blogs under the heading 'Mid2: Get Blogs Using React Query From Node'. Each blog item includes a thumbnail, title, category, and actions ('Read More', 'Edit', 'Del'). A specific blog entry is highlighted with a red box, showing details: 'Lifestyle' category, title '208410331 Liang', and description 'Mid2 Exam -- React'. Below the list are buttons for 'Add Blog', 'Clear All', and 'Load All'. To the right, a DevTools panel titled 'TANSTACK React Query v5' shows the state of the 'blogs_xx' query. It lists 7 items, with the 6th item highlighted by a red box. The item details are: id: 50, img: '/images/my-1.jpg', remote_img: '', category: 'lifestyle', title: '208410331 Liang', and desc: 'Mid2 Exam -- React'.

The screenshot shows two side-by-side code editors in VS Code. The left editor is for 'server_xx.js' and the right is for 'Mid2NodeBlogLocal_31.jsx'. The 'server_xx.js' file contains a mutation for adding a blog, which is highlighted with a red box. The 'Mid2NodeBlogLocal_31.jsx' file contains the UI logic, including a button click handler for adding a blog, also highlighted with a red box. The terminal at the bottom shows the command 'node server_xx.js' running and a POST request to '/api/blogs' being successful with a response time of 0.868 ms.

```
client_xx > src > pages > mid1_xx > Mid2NodeBlogLocal_31.jsx > Mid2NodeBlogLocal_31
13 const Mid2NodeBlogLocal_31 = () => {
27   );
28   console.log('blog data', data);
29
30   // Create
31   const { mutate: addBlog } = useMutation({
32     mutationFn: async ({ id, img, remote_img, category, title, desc }) => {
33       try {
34         const { data } = await axios.post('http://localhost:5000/api/blogs', {
35           id,
36           img,
37           remote_img,
38           category,
39           title,
40           desc,
41         });
42         console.log('axios new data', data);
43       } catch (error) {
44         console.log(error);
45       }
46     },
47     onSuccess: () => {
48       queryClient.invalidateQueries({
49         queryKey: ['blogs_xx'],
50       });
51       toast.success('Blog add success');
52     },
53   });
54
55   // Update
56   // const { mutate: updateBlog } = useMutation({});
57
58   // Delete
59   // const { mutate: deleteBlog } = useMutation({});
60
61   // clearAll
62 }

[nodemon] starting 'node server_xx.js'
Server is running on port 5000
GET /api/logs 200 2.968 ms - 1007
title: 208410331 Liang
POST /api/blogs 200 0.868 ms - 136
GET /api/blogs 200 0.532 ms - 1135
```

```
client_xx > src > pages > mid1_xx > Mid2NodeBlogLocal_31.jsx > Mid2NodeBlogLocal_31
13 const Mid2NodeBlogLocal_31 = () => {
86   <div>
87     <Query 6 items</Query>
88     <option 8 items</option>
89     <gt;> 300000
90     <div>
91       <button>
92         <span>Add Blog</span>
93       </button>
94       <button>
95         <span>Clear All</span>
96       </button>
97       <button>
98         <span>Load All</span>
99       </button>
100      <div>
101        <table>
102          <thead>
103            <tr>
104              <th>Title</th>
105              <th>Category</th>
106              <th>Image</th>
107            </tr>
108          </thead>
109          <tbody>
110            <tr>
111              <td>208410331 Liang</td>
112              <td>lifestyle</td>
113              <td><img alt='my-1.jpg'></td>
114            </tr>
115            <tr>
116              <td>Sun Rise From The Mountain</td>
117              <td>TRAVEL</td>
118              <td><img alt='photo-7.jpg'></td>
119            </tr>
120            <tr>
121              <td>Serene Lake With Trees</td>
122              <td>TRAVEL</td>
123              <td><img alt='photo-8.jpg'></td>
124            </tr>
125            <tr>
126              <td>Rocks Of Queen Head In Yehliu, Taiwan</td>
127              <td>TRAVEL</td>
128              <td><img alt='photo-9.jpg'></td>
129            </tr>
130            <tr>
131              <td>Why Coffee Is Awesome</td>
132              <td>LIFESTYLE</td>
133              <td><img alt='photo-1.jpg'></td>
134            </tr>
135            <tr>
136              <td>Travel To Paris</td>
137              <td>TRAVEL</td>
138              <td><img alt='photo-2.jpg'></td>
139            </tr>
140            <tr>
141              <td>Coffee Brings Friendship</td>
142              <td>LIFESTYLE</td>
143              <td><img alt='photo-3.jpg'></td>
144            </tr>
145            <tr>
146              <td><img alt='my-1.jpg'></td>
147              <td><span>Lifestyle</span></td>
148              <td><span>208410331 Liang</span></td>
149            </tr>
150          </tbody>
151        </table>
152      </div>
153    </div>
154  </div>
155}

<button>
  <span>Add Blog</span>
</button>
<button>
  <span>Clear All</span>
</button>
<button>
  <span>Load All</span>
</button>
```

9a67b3a liangyu9103 Wed May 8 21:23:30 2024 +0800 ### W12-P4: Insert a blog using

W12-P5: Git logs of W12

git log --pretty=format:"%h%x09%an%x09%ad%x09%s" --after="2024-05-05"

9a67b3a liangyu9103	Wed May 8 21:23:30 2024 +0800	### W12-P4: Insert a blog using
66a2272 liangyu9103	Wed May 8 21:01:12 2024 +0800	### W12-P3: Use React Query to
da24083 liangyu9103	Wed May 8 20:27:57 2024 +0800	### W12-P2: Update a blog with
f7bd345 liangyu9103	Wed May 8 19:53:25 2024 +0800	### W12-P1: Insert a blog using