

Great! Here is a **simple, clean, step-by-step tutorial** to build a full **React CRUD App** using **Axios** with only:

- ✓ id
- ✓ name
- ✓ email
- ✓ phone

Backend:

- <https://jsonplaceholder.typicode.com/users>

This version allows you to pass the entire `form` directly to the API.

□ **STEP-BY-STEP TUTORIAL**

□ **STEP 1 — Create React App**

```
npx create-react-app react-crud
cd react-crud
```

□ **STEP 2 — Install Axios**

```
npm install axios
```

□ **STEP 3 — Create the Component**

Create a file inside `src/`:

- `src/UserCRUD.js`
-

□ STEP 4 — Add Full CRUD Code

Copy & paste this:

```
import React, { useEffect, useState } from "react";
import axios from "axios";

const API = "https://jsonplaceholder.typicode.com/users";

function UserCRUD() {
  const [users, setUsers] = useState([]);
  const [form, setForm] = useState({
    id: "",
    name: "",
    email: "",
    phone: ""
  });

  // GET users
  useEffect(() => {
    axios.get(API)
      .then(res => setUsers(res.data))
      .catch(err => console.log(err));
  }, []);

  // INPUT HANDLER
  const handleChange = (e) => {
    setForm({ ...form, [e.target.name]: e.target.value });
  };

  // CREATE — DIRECT form submit
  const handleAdd = (e) => {
    e.preventDefault();

    axios.post(API, form)
      .then(res => {
        setUsers([...users, res.data]);
        setForm({ id: "", name: "", email: "", phone: "" });
      });
  };

  // EDIT (load form fields)
  const handleEdit = (user) => {
    setForm({
      id: user.id,
      name: user.name,
      email: user.email,
      phone: user.phone
    });
  };
};
```

```

// UPDATE - DIRECT form submit
const handleUpdate = () => {
  axios.put(`${API}/${form.id}`, form)
    .then(res => {
      setUsers(users.map(u => u.id === form.id ? res.data : u));
      setForm({ id: "", name: "", email: "", phone: "" });
    });
};

// DELETE
const handleDelete = (id) => {
  axios.delete(`${API}/${id}`)
    .then(() => {
      setUsers(users.filter(u => u.id !== id));
    });
};

return (
  <div style={{ padding: 20 }}>
    <h2>React CRUD with Axios (Simple)</h2>

    { /* FORM */ }
    <form onSubmit={handleAdd}>
      <input name="id" placeholder="ID" value={form.id}
onChange={handleChange} />
      <input name="name" placeholder="Name" value={form.name}
onChange={handleChange} />
      <input name="email" placeholder="Email" value={form.email}
onChange={handleChange} />
      <input name="phone" placeholder="Phone" value={form.phone}
onChange={handleChange} />

      <button type="submit">Add</button>
      <button type="button" onClick={handleUpdate}>Update</button>
    </form>

    <hr />

    { /* TABLE */ }
    <table border="1" width="100%">
      <thead>
        <tr>
          <th>ID</th><th>Name</th><th>Email</th><th>Phone</th><th>Actions</th>
        </tr>
      </thead>

      <tbody>
        {users.map(u => (
          <tr key={u.id}>
            <td>{u.id}</td>
            <td>{u.name}</td>
            <td>{u.email}</td>
            <td>{u.phone}</td>

            <td>
              <button onClick={() => handleEdit(u)}>Edit</button>
            </td>
          </tr>
        ))}
      </tbody>
    </table>
  </div>
);

```

```
        <button onClick={() => handleDelete(u.id)}>Delete</button>
      </td>
    </tr>
  </tbody>
</table>
</div>
  );
}

export default UserCRUD;
```

□ **STEP 5 — Use the Component in App.js**

Open **src/App.js** and replace everything with:

```
import UserCRUD from "../UserCRUD";

function App() {
  return <UserCRUD />;
}

export default App;
```

□ **STEP 6 — Run Your App**

```
npm start
```

□ **Your CRUD App is Ready!**

You now have:

- ✓ **GET users from API**
- ✓ **POST a new user**
- ✓ **PUT update a user**
- ✓ **DELETE a user**
- ✓ **With direct form → API submission**
- ✓ **Without creating extra objects or spreading**