User

**You**

using the asynch fetch with this api "https://jsonplaceholder.typicode.com/todos/" list all todos and for each todo render a checkbox which should hold the value of the completed state of the todo. and then if a user clicks on the checkbox it should toggle the states to update the api... when a user toggles the state it should update the backend. and add an input close to the label where it can be edited and a save button

todo.js

2nd opt

/\*document.addEventListener("DOMContentLoaded", () => {

  const todoList = document.getElementById("todoList");

  // Fetch todos from the API

  async function fetchTodos() {

    try {

      const response = await fetch(

        "https://jsonplaceholder.typicode.com/todos/"

      );

      const todos = await response.json();

      renderTodos(todos);

    } catch (error) {

      console.error("Error fetching todos:", error);

    }

  }

  // Render todos

  function renderTodos(todos) {

    todoList.innerHTML = "";

    todos.forEach((todo) => {

      const todoItem = createTodoItem(todo);

      todoList.appendChild(todoItem);

    });

  }

  // Create a todo item element

  function createTodoItem(todo) {

    const todoItem = document.createElement("div");

    todoItem.classList.add("todo");

    todoItem.innerHTML = `

      <input type="checkbox" ${todo.completed ? "checked" : ""}>

      <label>${todo.title}</label>

      <input type="text" class="edit-input" value="${todo.title}">

      <button class="edit-btn">Edit</button>

      <button class="save-btn">Save</button>

      <button class="delete-btn">Delete</button>

    `;

    const editBtn = todoItem.querySelector(".edit-btn");

    const saveBtn = todoItem.querySelector(".save-btn");

    const deleteBtn = todoItem.querySelector(".delete-btn");

    const label = todoItem.querySelector("label");

    const editInput = todoItem.querySelector(".edit-input");

    editBtn.addEventListener("click", () => {

      editBtn.style.display = "none";

      saveBtn.style.display = "inline-block";

      deleteBtn.style.display = "inline-block";

      editInput.style.display = "inline-block";

      label.style.display = "none";

    });

    saveBtn.addEventListener("click", () => {

      label.textContent = editInput.value;

      editBtn.style.display = "inline-block";

      saveBtn.style.display = "none";

      deleteBtn.style.display = "none";

      editInput.style.display = "none";

      label.style.display = "inline-block";

    });

    deleteBtn.addEventListener("click", () => {

      todoItem.remove();

    });

    return todoItem;

  }

  // Initial fetch of todos

  fetchTodos();

});

\*/