

## A simple CRUD API in node and express to get us started with node/express business!

### Note:

CRUD stands for Create, Read, Update and Delete [more info](#)

## SET-UP NODE project

1- npm init -y

2-

```
npm install express
```

```
npm install --save-dev nodemon
```

3- Update package.json to use nodemon:

```
"scripts": {  
  "start": "nodemon index.js"  
}
```

---

## CREATE An Express SERVER

### *Content of index.js:*

```
const express = require('express')  
const app = express()  
const port = 3000  
  
//Middleware to parse JSON  
app.use(express.json())  
  
// Sample data  
let todos = [  
  { id: 1, task: "Learn Node.js", completed: false },  
  { id: 2, task: "Build a REST API", completed: false }  
];  
// GET /todos - Retrieve all to-do items  
// THIS IS THE Read part of CRUD  
app.get('/todos', (req, res) =>{
```

## A simple CRUD API in node and express to get us started with node/express business!

```
    res.json(todos)
  });

// Start the server
app.listen(port, () => {
  console.log(`Server is running on http://localhost:${port}`);
});
```

### Run your project now:

\$> npm start

On another terminal window run:

\$> curl http://localhost:3000/todos

What do you see?

Here's another way [easier!]:

Open your browser and navigate to <http://localhost:3000/todos>

**If you are comfortable with the above, let's add the Create part of CRUD on the next page**

---

After the GET part but before the app.listen add the following code:

```
// POST /todos - Add a new to-do item
app.post('/todos', (req, res) => {
  const newTodo = {
    id: todos.length + 1,
    task: req.body.task,
    completed: false
  };
  todos.push(newTodo);
  res.status(201).json(newTodo);
});
```

## A simple CRUD API in node and express to get us started with node/express business!

Question: What does code 201 in http mean?

Answer: <https://developer.mozilla.org/en-US/docs/Web/HTTP/Status/201>

Test it!

```
curl -X POST http://localhost:3000/todos -H "Content-Type: application/json" -d '{"task": "Review Express documentation"}'
```

---

### Let's do the Update of CRUD

```
// PUT /todos/:id - Update an existing to-do item
app.put('/todos/:id', (req, res) => {
  const id = parseInt(req.params.id);
  const todo = todos.find(t => t.id === id);
  if (!todo) {
    return res.status(404).send("To-Do item not found");
  }
  todo.task = req.body.task || todo.task;
  todo.completed = req.body.completed !== undefined ?
req.body.completed : todo.completed;
  res.json(todo);
});
```

Try it:

```
curl -X PUT http://localhost:3000/todos/1 -H "Content-Type: application/json" -d '{"task": "Update Express guide", "completed": true}'
```

---

## Your task is now to add the DELETE part of CRUD 😊

Basically, find the id parameter, if not found, send a 404 response BUT if it exists, delete it using any number of array methods then SEND a 204 status response!

**Everyone has 10 minutes to do it!**