

JS


Build a Simple Express Server



Express

Setup Node project

Create a folder called **express-server**. Open it in your code editor. And run the following command in it.



```
# initialize a node project  
npm init -y
```

This will create a **package.json** file in your project. It holds your project meta-data. Without it, your project cannot work.

Create a server file

In the root directory, create an **index.js** file and import the express package. Afterward, create an express app.



```
const express = require('express');  
  
// create an express app  
const app = express();
```

Add middlewares

Let's use two middlewares. One to parse incoming requests with JSON payloads and the second to parse incoming requests with URL-encoded payloads.

```
index.js

const express = require('express');

// create an express server
const app = express();

// parse incoming requests to JSON payload
app.use(express.json());

// parse incoming requests with URL-encoded payloads
app.use(express.urlencoded({ extended: true }));
```


Create an API route

Create a root API route that will send JSON response when you hits `/` route.

```
index.js

const express = require('express');

// create an express server
const app = express();


// parse incoming requests to JSON payload
app.use(express.json());

// parse incoming requests with URL-encoded payloads
app.use(express.urlencoded({ extended: true }));

// '/' API route
app.get('/', (req, res) => {
  res.status(200).json({ message: 'API is running.' });
});
```

Start the server

Let's start the server using node. Open your terminal in your project directory and run this command:



```
# start the server  
node index.js
```


Listen for the server

Now let's listen for the server connection when it starts.

```
index.js

const express = require('express');

// create an express server
const app = express();

// parse incoming requests to JSON payload
app.use(express.json());

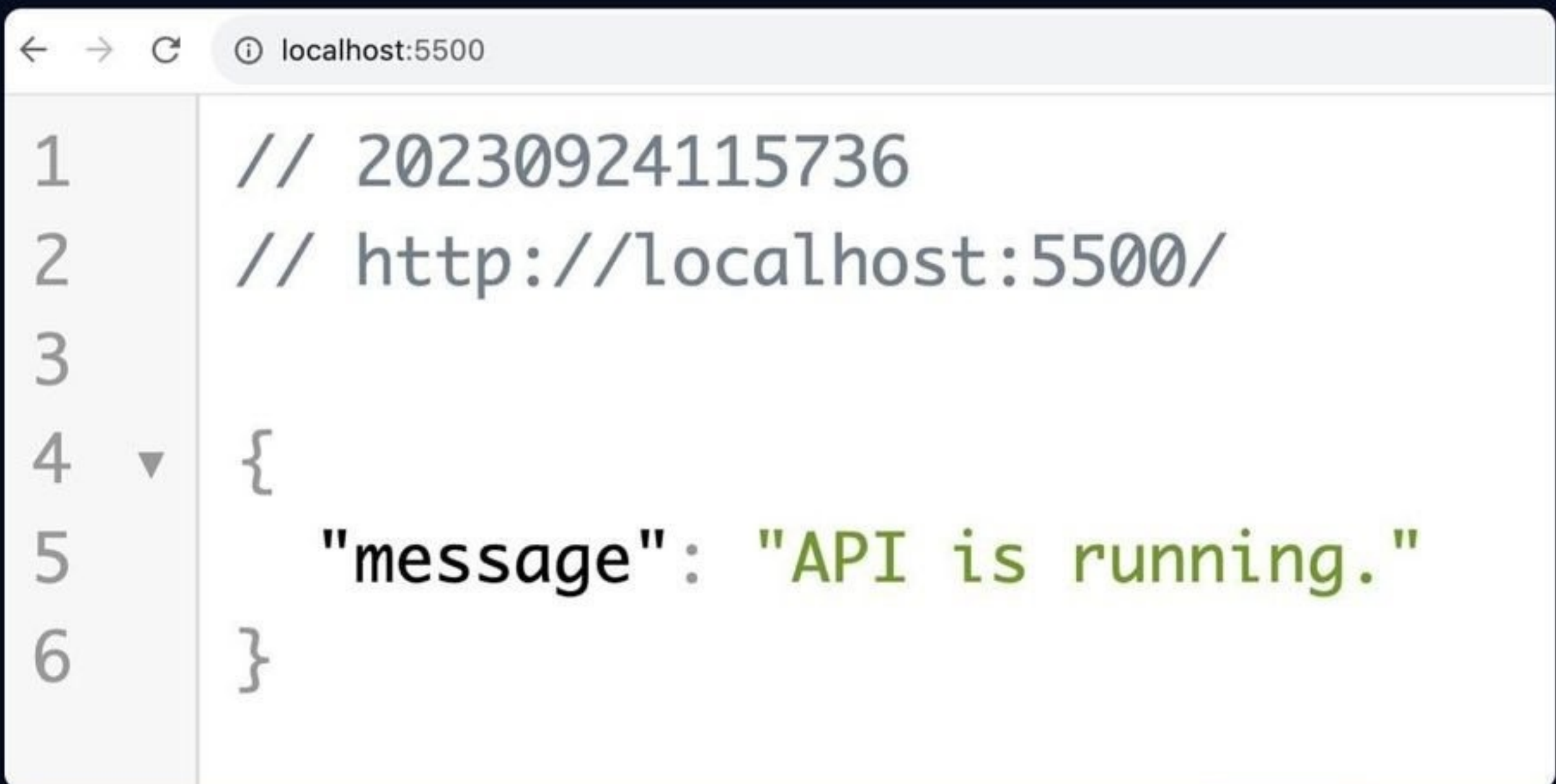
// parse incoming requests with URL-encoded payloads
app.use(express.urlencoded({ extended: true }));

// route
app.get('/', (req, res) => {
  res.status(200).json({ message: 'API is running.' });
});

// port
const port = process.env.PORT || 5500;
// listen for connection
app.listen(port, () => console.log(`Listening on Port: ${port}`));
```

Make the request

Visit localhost:5500 in your browser to see the API response.

A screenshot of a web browser window. The address bar shows 'localhost:5500'. The main content area displays a JSON response. The response is a single object with a 'message' property. The message is 'API is running.'.

```
1 // 20230924115736
2 // http://localhost:5500/
3
4 {
5   "message": "API is running."
6 }
```


Did you find it useful ?

Leave a **comment** !



K. RAKESH

@Web **Developer**

FOLLOW FOR MORE

Like



Comment



Repost

