How to make custom modules in node js

Node JS Tutorial in Hindi #20 Modules and Routes | how to make custom ...

Node.js is great for creating modular server-side applications. In this post, we'll create a simple project using **modules**, **routing**, and **form handling**.

Project Structure

Create 3 files:

`root.js` (main server file)

`userForm.js` (HTML form)

`userDataSubmit.js` (handle form data)

root.js (Main Server File)

```
const http = require("http");
const userForm = require("./userForm");
const userDataSubmit = require("./userDataSubmit");

http
  .createServer((req, resp) => {
    resp.writeHead(200, { "Content-Type": "text/html" });

    if (req.url === "/") {
      userForm(req, resp);
    } else if (req.url === "/submit") {
      userDataSubmit(req, resp);
    }

    resp.end();
  })
  .listen(3200);
```

This file:

Creates the HTTP server

Routes / to form and `/submit` to form handler

Sends HTML response based on the route

userForm.js (HTML Form)

```
function userForm(req, resp) {
  resp.write(`
    <form action="/submit" method="POST">
      <input type="text" placeholder="Enter username" name="name" />
      <input type="text" placeholder="Enter email" name="email" />
      <button>Submit</button>
    </form>
  `);
```

```
  }

  module.exports = userForm;
```

This file:

Displays a form with two input fields

Sends data to `/submit` via POST method

userDataSubmit.js (Handle Form Data)

```
const querystring = require("querystring");

function userDataSubmit(req, resp) {
  let dataBody = [];

  req.on("data", (chunk) => {
    dataBody.push(chunk);
  });

  req.on("end", () => {
    let rawData = Buffer.concat(dataBody).toString();
    let readableData = querystring.parse(rawData);

    let dataString =
      "My name is " +
      readableData.name +
      " and my email is " +
      readableData.email;

    console.log(dataString);
  });

  resp.write(`<h1>You can get data from user form here</h1>`);
  }
```

```
module.exports = userDataSubmit;
```

This file:

Listens to data from the form

Collects data chunks using `Buffer`

Converts it to a readable object using `querystring`

Logs name and email in the console

Output Demo

```
node root.js
```

Open browser at:

`http://localhost:3200/`

© 2025 All Rights Reserved by **thecodingskills.com** and **Code step by step YouTube Channel**

Home About Contact