Primer paso verificar que existe node.j

node --version

npm –version

abrir visual

1. Abrir terminal y colocar **npm init**
2. Instalar express desde la terminal **npm i express npm i nodemon -D**
3. En *package.json* agregar las siguientes líneas
4. "name": "web-avanzada-uris",
5. "version": "1.0.0",
6. "description": "mi primera uri",
7. "main": "index.js",
8. "type": "module",

Hay mismo cambiar el scripts

"scripts": {

    "dev": "nodemon  index.js"

  },

1. Crear el  **index.js**
2. import express from 'express';
3. import fs from "fs";
4. import bodyParser from "body-parser";
5. const app = express();
6. app.use(bodyParser.json());
7. const readData = () =>{
8. try {
9. const data = fs.readFileSync("./db.json");
10. return JSON.parse(data);
11. }catch(error){
12. console.log(error);
13. }
15. };
16. const writeData = (data) =>{
17. try {
18. fs.writeFileSync("./db.json", JSON.stringify(data));
19. }catch(error){
20. console.log(error);
21. }
23. };
24. app.get("/", (req,res) => {
25. res.send("WELCOME TO MY nodejs hghg");
26. });
27. app.get("/products", (req,res) => {
28. const data = readData();
29. res.json(data.products);
30. });
31. app.get("/products/:id", (req, res) => {
32. const data = readData(); // Obtén los datos de los productos
33. const id = parseInt(req.params.id); // Convierte el ID de la URL a un número entero
34. const product = data.products.find((product) => product.id === id); // Busca el producto por su ID
35. if (product) {
36. res.json(product); // Si se encuentra el producto, devuélvelo
37. } else {
38. res.status(404).json({ message: "Product not found" }); // Si no se encuentra el producto
39. }
40. });
41. app.post("/products", (req, res) => {
42. const data = readData(); // Obtén los datos de los productos
43. const body = req.body;
44. const newProduct = {
45. id: data.products.length + 1,
46. ...body,
47. };
48. data.products.push(newProduct);
49. writeData(data);
50. res.json(newProduct);
51. });
52. app.put("/products/:id", (req, res) => {
53. const data = readData(); // Obtén los datos de los productos
54. const body = req.body;
55. const id = parseInt(req.params.id); // Convierte el ID de la URL a un número entero
56. const productIndex = data.products.findIndex((product) => product.id === id);
57. data.products[productIndex] = {
58. ...data.products[productIndex],
59. ...body,
60. };
61. writeData(data);
62. res.json(data.products[productIndex]);
63. });
64. app.delete("/products/:id", (req, res) => {
65. const data = readData(); // Obtén los datos de los productos
66. const id = parseInt(req.params.id); // Convierte el ID de la URL a un número entero
67. const productIndex = data.products.findIndex((product) => product.id === id);
68. data.products.splice(productIndex,1);
70. writeData(data);
71. res.json({message: "product deleted "});
72. });
73. app.listen(3000, () =>{
74. console.log('Server listening on port 3000');
75. });

En consola poner y buscar en el navegador LocalHost:3000 (depende el puerto)

**npm run dev**

1. Crear la conexion a la base de datos y db.json

{

    "products": [

        {

            "id": 1,

            "issue\_date": "2025-01-23T00:00:00.000Z",

            "total\_amount": "10.00",

            "payment\_status": "Pendiente",

            "notification\_sent": 0

        },

        {

            "id": 2,

            "issue\_date": "2025-01-23T00:00:00.000Z",

            "total\_amount": "58.50",

            "payment\_status": "Pagado",

            "notification\_sent": 0

        },

        {

            "id": 4,

            "issue\_date": "2025-01-23T00:00:00.000Z",

            "total\_amount": "7.00",

            "payment\_status": "Pendiente",

            "notification\_sent": 0

        },

        {

            "id": 5,

            "issue\_date": "2025-01-23T00:00:00.000Z",

            "total\_amount": "8.00",

            "payment\_status": "Pendiente",

            "notification\_sent": 0

        },

        {

            "id": 6,

            "issue\_date": "2025-01-23T00:00:00.000Z",

            "total\_amount": "8.00",

            "payment\_status": "Pendiente",

            "notification\_sent": 0

        },

        {

            "id": 7,

            "issue\_date": "2025-01-23T00:00:00.000Z",

            "total\_amount": "8.00",

            "payment\_status": "Pendiente",

            "notification\_sent": 0

        }

    ]

}

1. descargar Rest Client como extensión y crear requests.http
2. GET http://localhost:3000/products HTTP/1.1
3. ###
4. //GET http://localhost:3000/products/3 HTTP/1.1
5. PUT http://localhost:3000/products/1 HTTP/1.1
6. Content-Type: application/json
7. {
8. "total\_amount": "10.00"
9. }
10. //DELETE http://localhost:3000/products/3 HTTP/1.1
11. //Content-Type: application/json
12. ###
13. POST http://localhost:3000/products
14. Content-Type: application/json
16. {
17. "issue\_date": "2025-01-23T00:00:00.000Z",
18. "total\_amount": "8.00",
19. "payment\_status": "Pendiente",
20. "notification\_sent": 0
21. }
22. ###



