

```
▼ [  
  ▼ {  
    "id": 1,  
    "nombre": "Henry",  
    "valor": "47"  
  },  
  ▼ {  
    "id": 2,  
    "nombre": "Alvin",  
    "valor": "02"  
  },  
  ▼ {  
    "id": 3,  
    "nombre": "Toto",  
    "valor": "14"  
  }  
]
```

servidor.js X

```
servidor.js > http.createServer() callback
1 // Importar módulos necesarios
2 import http from 'http';
3 import mysql from 'mysql2';
4
5 // Configuración de la conexión a MySQL
6 const connection = mysql.createConnection({
7   host: 'localhost',
8   user: 'root',
9   password: '',
10  database: 'base_de_datos_tarea_3'
11 });
12
13 // Conectar a MySQL
14 connection.connect(error => {
15   if(error){
16     return console.error('error:' + error.message);
17   }
18
19   console.log('Conectado a la base de datos MySQL');
20 });
21
22 const server = http.createServer((req, res) =>{
23   if(req.method === 'GET' && req.url === '/datos') {
24     // Realizar una consulta a la base de datos
25     connection.query('SELECT * FROM datos', (error, results) => {
26       if(error) {
27         console.log('Error en la consulta', error);
28         res.writeHead(500);
29         res.end('Error interno del servidor');
30         return;
31       }
32
33       // Enviamos los resultados como JSON
34       res.writeHead(200, {'Content-Type': 'application/json'});
35       res.end(JSON.stringify(results));
36     });
37   } else {
38     // Manejar otras rutas o métodos
39     res.writeHead(404);
40     res.end('No encontrado');
41   }
42 });
43
44 // Escuchar en un puerto específico
45 const PORT = 3000;
46 server.listen(PORT, () => {
47   console.log('Servidor corriendo en http://localhost:${PORT}');
48 });
49
```

JS servidor.js X

JS servidor.js > [?] server > http.createServer() callback

```
14 connection.connect(error => {
17   }
18
19   console.log('Conectado a la base de datos MySQL');
20 });
21
22 const server = http.createServer((req, res) =>{
23   if(req.method === 'GET' && req.url === '/datos') {
24     // Realizar una consulta a la base de datos
25     connection.query('SELECT * FROM datos', (error, results) => {
26       if(error) {
27         console.log('Error en la consulta', error);
28         res.writeHead(500);
29         res.end('Error interno del servidor');
30         return;
31       }

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\USUARIO\Documents\Tareas_2\Tarea3> npm run dev

```
> tarea3@1.0.0 dev
> nodemon servidor.js
```

```
[nodemon] 3.1.0
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,cjs,json
[nodemon] starting `node servidor.js`
Servidor corriendo en http://localhost:3000
Conectado a la base de datos MySQL
```