

## AXIOS:

### GET

The screenshot shows the Visual Studio Code editor with the file explorer on the left displaying the project structure. The main editor window shows the `axios.js` file in the `misc` directory. The code defines an Axios instance and a GET endpoint. The terminal at the bottom shows the command `node axios.js` being executed, and the output displays a JSON array of two tennis player objects.

```
1 import axios from "axios";
2
3 //axios() es el equivalente a GET
4
5 const api = axios.create({
6   baseURL: "http://localhost:8080/",
7 })
8
9 //GET
```

```
PS C:\Users\vguar\Desktop\BackEnd\desafios\siarindetenis\src\misc> node axios.js
PS C:\Users\vguar\Desktop\BackEnd\desafios\siarindetenis\src\misc> node axios.js
modificao
PS C:\Users\vguar\Desktop\BackEnd\desafios\siarindetenis\src\misc> node axios.js
[
  {
    _id: '63d926bfbf21d64821dbaf16',
    nombreTenista: 'Mariano',
    apellidoTenista: 'Guardia',
    dniTenista: 34748519,
    categoria: '1ra seleccion',
    posicion: 8,
    __v: 0
  },
  {
    _id: '63d927b5de83818a61fe79d3',
    nombreTenista: 'test',
    apellidoTenista: 'Guardia',
    dniTenista: 1,
    categoria: '1ra seleccion',
    posicion: 8,
    __v: 0
  }
]
```

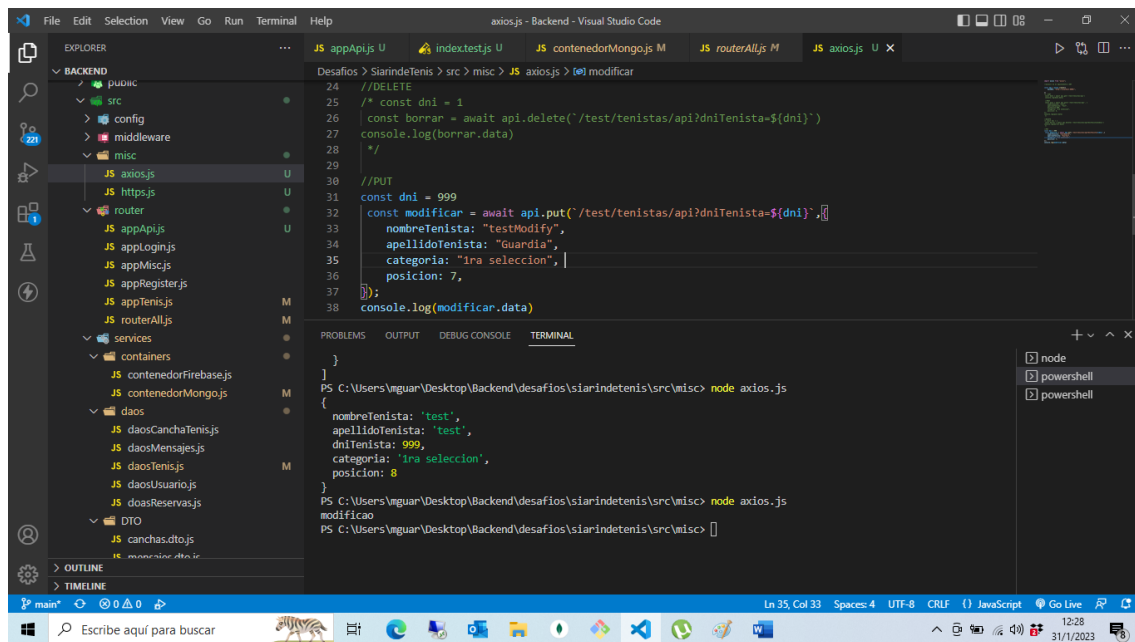
### POST

The screenshot shows the Visual Studio Code editor with the file explorer on the left. The main editor window shows the `axios.js` file with the POST method implementation. The code uses `await api.post()` to send a request with player data. The terminal shows the command `node axios.js` being executed, and the output displays the JSON object of the created player.

```
7
8
9 //GET
10 const data = await api.get('/test/tenistas/api')
11 console.log(data.data)
12
13 //POST
14 const post = await api.post('/test/tenistas/api', {
15   nombreTenista: "test",
16   apellidoTenista: "test",
17   dniTenista: 999,
18   categoria: "1ra seleccion",
19   posicion: 8,
20 });
21 console.log(post.data)
```

```
PS C:\Users\vguar\Desktop\BackEnd\desafios\siarindetenis\src\misc> node axios.js
{
  nombreTenista: 'test',
  apellidoTenista: 'test',
  dniTenista: 999,
  categoria: '1ra seleccion',
  posicion: 8
}
```

PUT

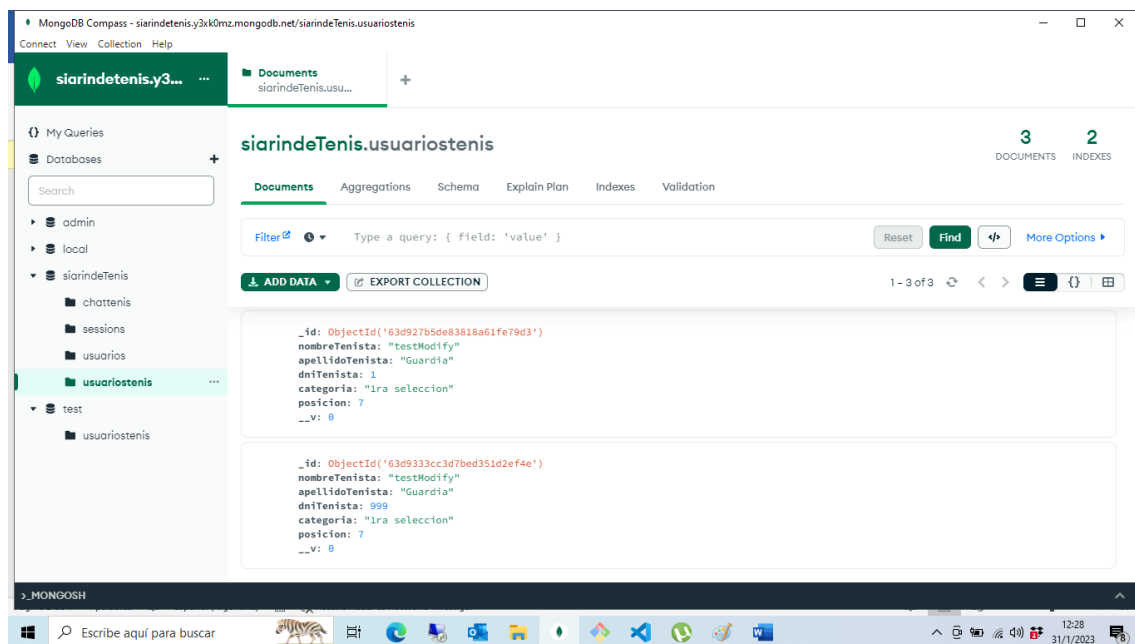


The screenshot shows the Visual Studio Code editor with a file explorer on the left displaying a project structure for 'Desafios > SiarindeTenis > src > misc'. The main editor shows the 'axios.js' file with a PUT endpoint implementation. The code defines a 'const dni = 999' and a 'const modificar' object with fields: 'nombreTenista: "testModify"', 'apellidoTenista: "Guardia"', 'dniTenista: 999', 'categoria: "Ira seleccion"', and 'posicion: 7'. The endpoint is implemented as 'const modificar = await api.put('/test/tenistas/api?dniTenista=\${dni}', { ...modificar })' followed by 'console.log(modificar.data)'. The terminal at the bottom shows the command 'node axios.js' being executed, and the output displays the modified user object.

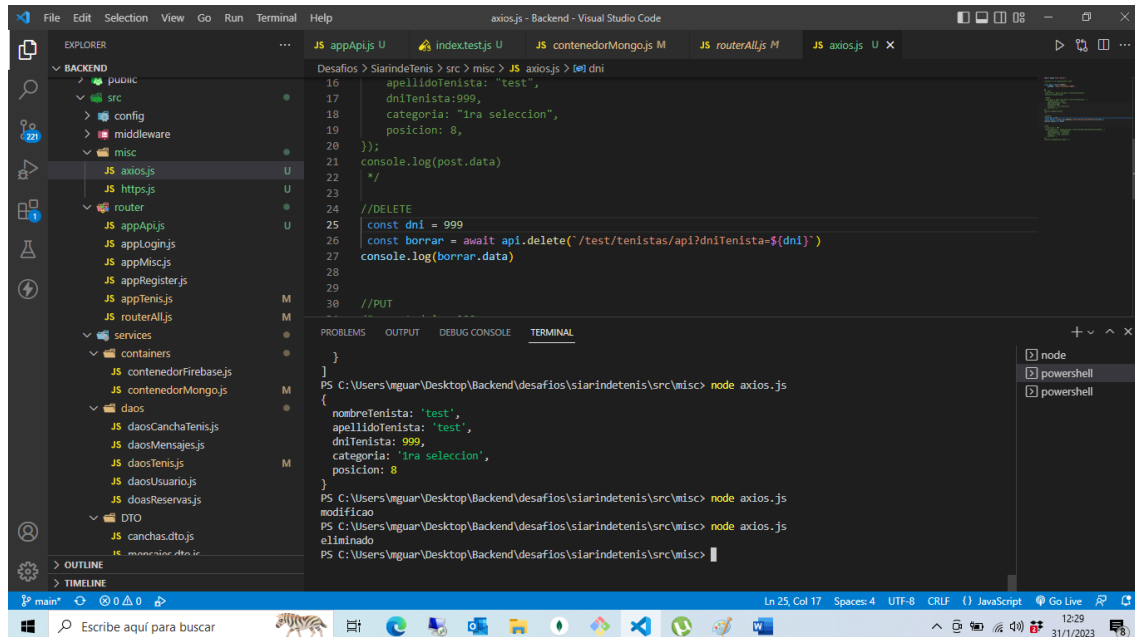
```
24 //DELETE
25 /* const dni = 1
26 const borrar = await api.delete('/test/tenistas/api?dniTenista=${dni}')
27 console.log(borrar.data)
28 */
29
30 //PUT
31 const dni = 999
32 const modificar = await api.put('/test/tenistas/api?dniTenista=${dni}', {
33   nombreTenista: "testModify",
34   apellidoTenista: "Guardia",
35   categoria: "Ira seleccion",
36   posicion: 7,
37 });
38 console.log(modificar.data)
```

```
PS C:\Users\mguar\Desktop\Backend\desafios\siarindetenis\src\misc> node axios.js
{
  nombreTenista: 'test',
  apellidoTenista: 'test',
  dniTenista: 999,
  categoria: 'Ira seleccion',
  posicion: 8
}
```

```
PS C:\Users\mguar\Desktop\Backend\desafios\siarindetenis\src\misc> node axios.js
modificao
PS C:\Users\mguar\Desktop\Backend\desafios\siarindetenis\src\misc>
```



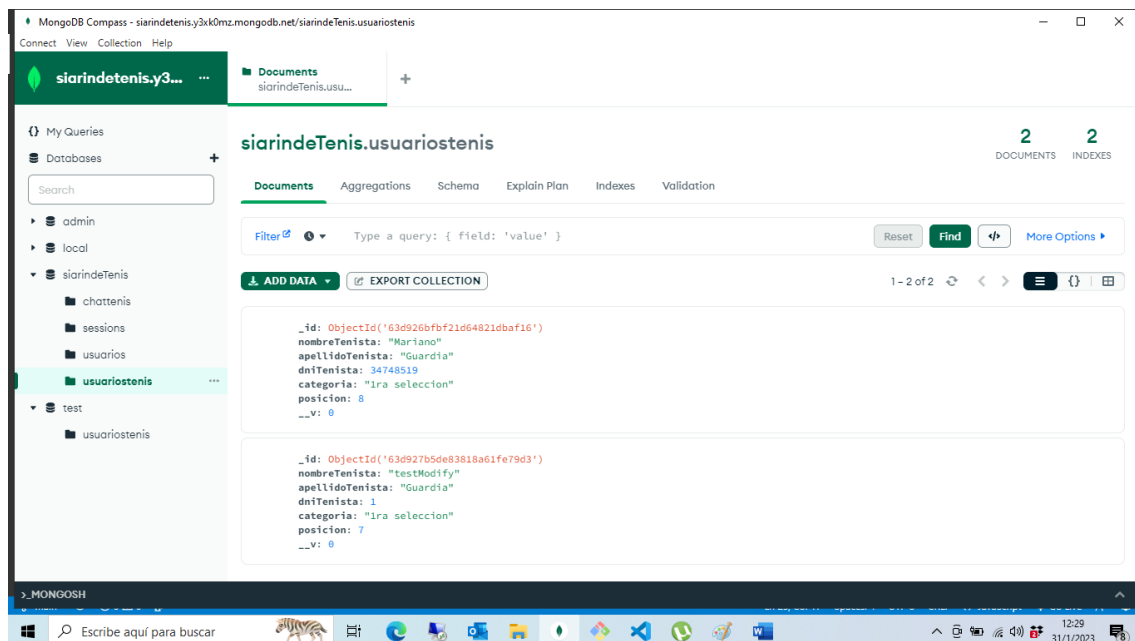
## DELETE



The screenshot shows the Visual Studio Code interface with a REST client open. The client is configured to send a DELETE request to the endpoint `/test/tenistas/api?dniTenista={dni}`. The request body is empty. The response shows a 200 status code and the deleted document details.

```
Desafios > SiarindeTennis > src > misc > JS axios.js > [X] dni
16  apellidoTenista: 'test',
17  dniTenista: 999,
18  categoria: 'Ira seleccion',
19  posicion: 8,
20  });
21  console.log(post.data)
22  */
23  //DELETE
24  const dni = 999
25  const borrar = await api.delete(`/test/tenistas/api?dniTenista=${dni}`)
26  console.log(borrar.data)
27  //PUT
28  }
29  }
30  }

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
PS C:\Users\vguar\Desktop\BackEnd\desafios\siarindeTennis\src\misc> node axios.js
{
  nombreTenista: 'test',
  apellidoTenista: 'test',
  dniTenista: 999,
  categoria: 'Ira seleccion',
  posicion: 8
}
PS C:\Users\vguar\Desktop\BackEnd\desafios\siarindeTennis\src\misc> node axios.js
modificao
PS C:\Users\vguar\Desktop\BackEnd\desafios\siarindeTennis\src\misc> node axios.js
eliminado
PS C:\Users\vguar\Desktop\BackEnd\desafios\siarindeTennis\src\misc>
```



# TEST

```
Desafios > SiarindeTennis > test > index.test.js > describe("test siarindeTennis") callback > it("should modify tennis player") callback > end() callback
```

```
32 |         if (err) throw err
33 |     })
34 |
35 | }
36 | it("should modify tennis player", () => {
37 |     request(app)
38 |       .put("/test/tenistas/api")
39 |       .expect(200)
40 |       .end((err, res) => {
41 |         if (err) throw err
42 |       })
43 |   })
44 |
45 | })
46 |
```

```
test siarindeTennis
  ✓ should list all tennis players
  ✓ should post tennis player
  ✓ should delete tennis player
  {}
  dni inválido
  ✓ should modify tennis player
  {}
  {}

4 passing (168ms)
```