

# A19. API REST CRUD con Node, Express y Mongoose

## Ejercicio 1

### Realizamos un POST

192.168.21.147:3000/accounts/

POST 192.168.21.147:3000/accounts/ Send

Params Authorization Headers (1) Body Pre-request Script Tests

none form-data x-www-form-urlencoded raw binary JSON (application/json)

```
1 {"name": "jesus", "balance": "10000"}
2
```

Body Cookies Headers (6) Test Results Status: 200 OK Time: 39 ms Size: 285 B

Pretty Raw Preview JSON

```
1 {
2   "_id": "5bfef4735b2a2e06d063f235",
3   "name": "jesus",
4   "balance": 10000,
5   "__v": 0
6 }
```

### Realizamos un GET a todas las cuentas

GET 192.168.21.147:3000/accounts/ Send

Pretty Raw Preview JSON

```
9   "_id": "5bfee4fe2c974404cb70608a",
10   "name": "Savings",
11   "balance": 2800,
12   "__v": 0
13 },
14 {
15   "_id": "5bfee5178a507f04d7104378",
16   "name": "Savings",
17   "balance": 121,
18   "__v": 0
19 },
20 {
21   "_id": "5bfee756c437bf05118d66c3",
22   "name": "Hola",
23   "balance": 121,
24   "__v": 0
25 },
26 {
27   "_id": "5bfef3187e727006ab250ea8",
28   "name": "andres",
29   "balance": 6000,
30   "__v": 0
31 },
32 {
33   "_id": "5bfef4735b2a2e06d063f235",
34   "name": "jesus",
35   "balance": 10000,
36   "__v": 0
37 }
38 ]
```

## Realizamos un GET según un ID

The screenshot shows a REST client interface. At the top, a GET request is configured to `192.168.21.147:3000/accounts/5`. Below the request bar, the response is displayed in the 'Body' tab, formatted as JSON. The response contains the following data:

```
{
  "_id": "5bfef4735b2a2e06d063f235",
  "name": "jesus",
  "balance": 10000,
  "__v": 0
}
```

## Realizamos un PUT

The screenshot shows a REST client interface with a PUT request configured to `192.168.21.147:3000/accounts/5bfef4735b2a2e06d063f235`. The request body is set to JSON and contains the following data:

```
{
  "name": "jesus arribi",
  "balance": "10000"
}
```

At the bottom of the interface, the status of the request is shown as `Status: 200 OK`, with a response time of `Time: 25 ms` and a size of `Size: 285 B`.

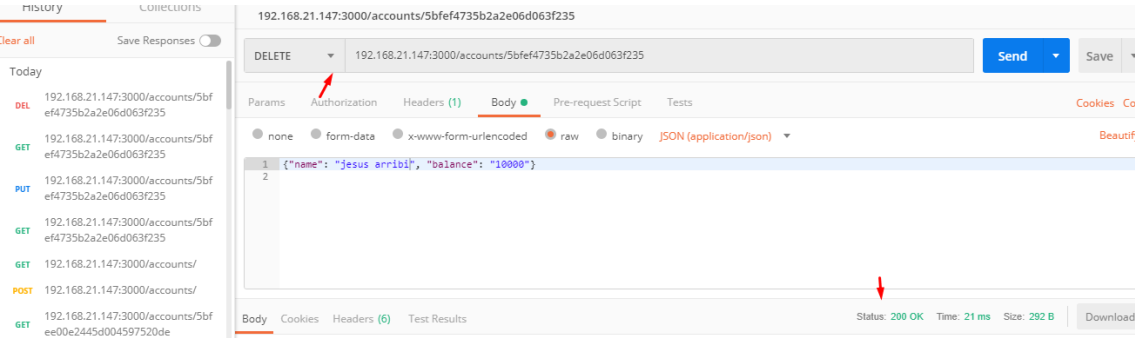
## Realizamos un GET al ID para ver los cambios

The screenshot shows a REST client interface with a GET request configured to `192.168.21.147:3000/accounts/5bfef4735b2a2e06d063f235`. The response is displayed in the 'Body' tab, formatted as JSON. The response contains the following data:

```
{
  "_id": "5bfef4735b2a2e06d063f235",
  "name": "jesus arribi",
  "balance": 10000,
  "__v": 0
}
```

A red arrow points to the `"name": "jesus arribi"` field in the response, indicating the change from the previous state. The status at the bottom is `Status: 200 OK`.

Hacemos un DELETE



Por último hacemos un GET al ID, vemos como ya no existe

