

**GA7-220501096-AA5-EV03 Diseño y Desarrollo de servicios web**

**Integrante :**

**Rodney Zapata Palacio**

**Presentado al instructor:**

**Gustavo Adolfo Rodriguez**

**Servicio Nacional de aprendizaje SENA**

**Centro de Comercio y Servicios (Regional Cauca)**

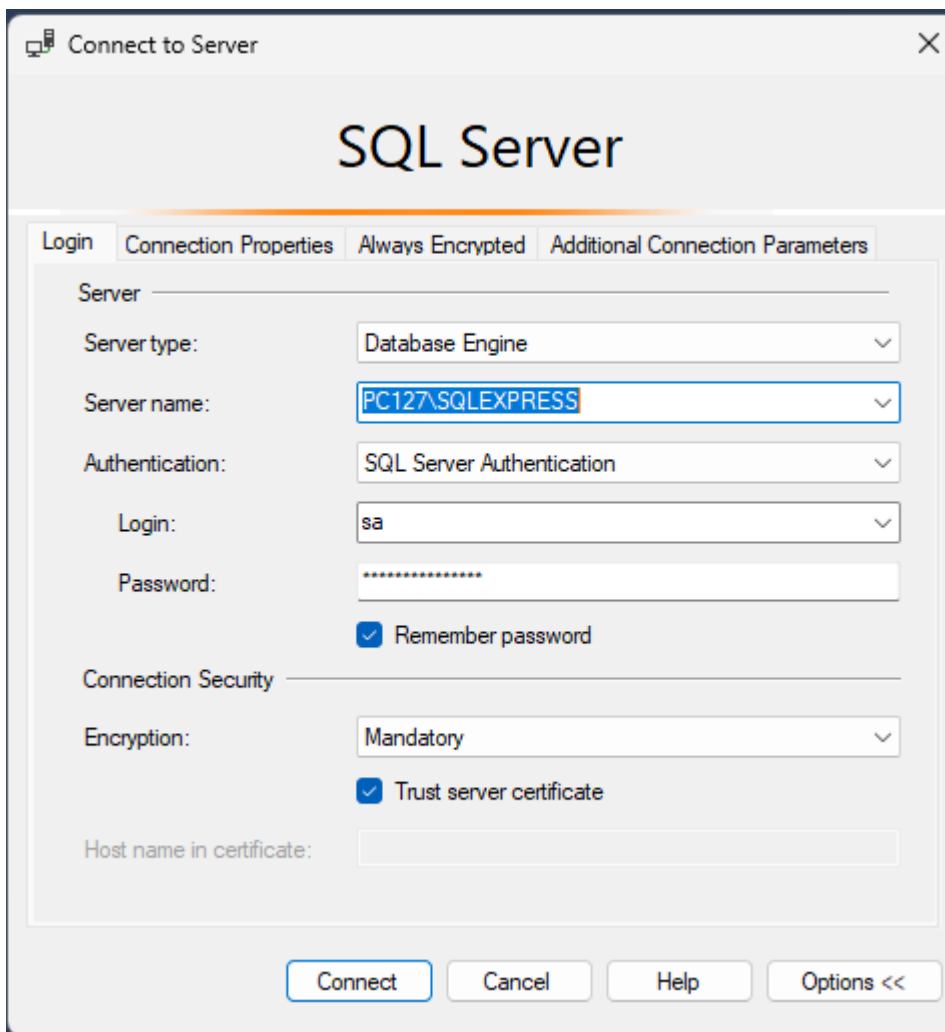
**Cauca - Popayán**

**Tecnólogo en Análisis y Desarrollo de Software**

**Ficha: 2675810**

**19 de Julio del 2024**

El servicio web que mostraremos a continuación consume una base de datos de SQLSERVER 2022 Express

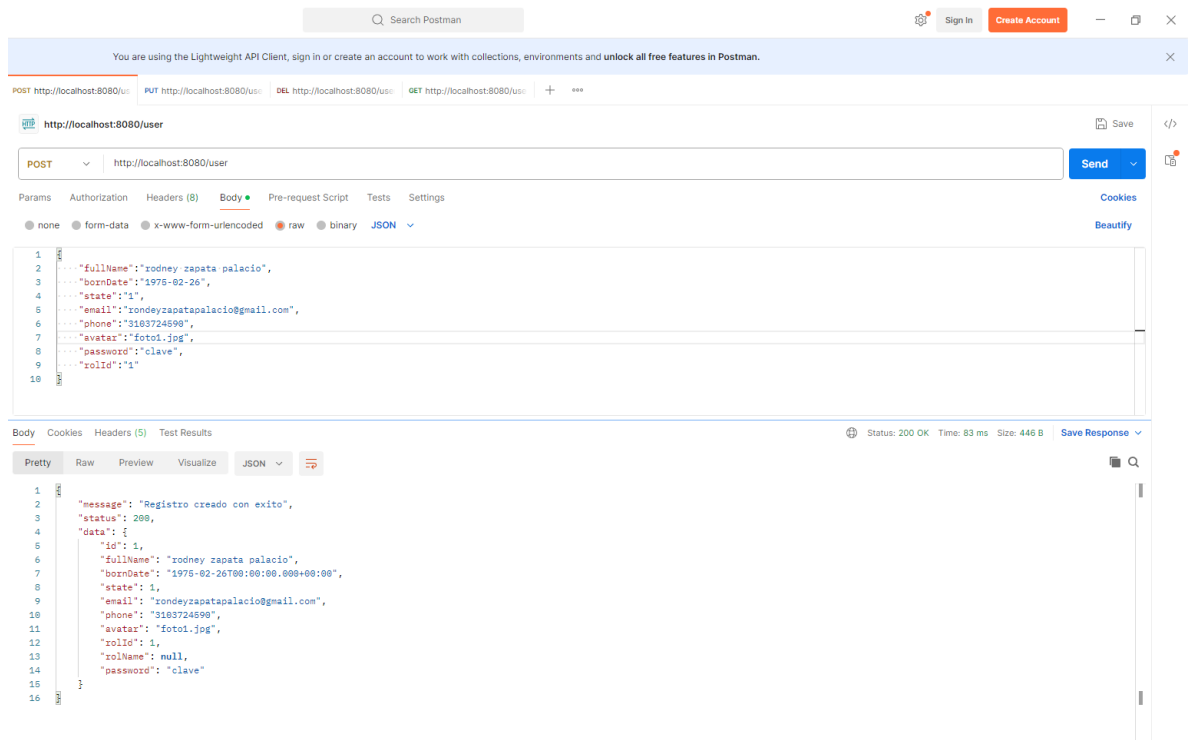


Nuestro cliente será Postman en el consumo del servicio web.

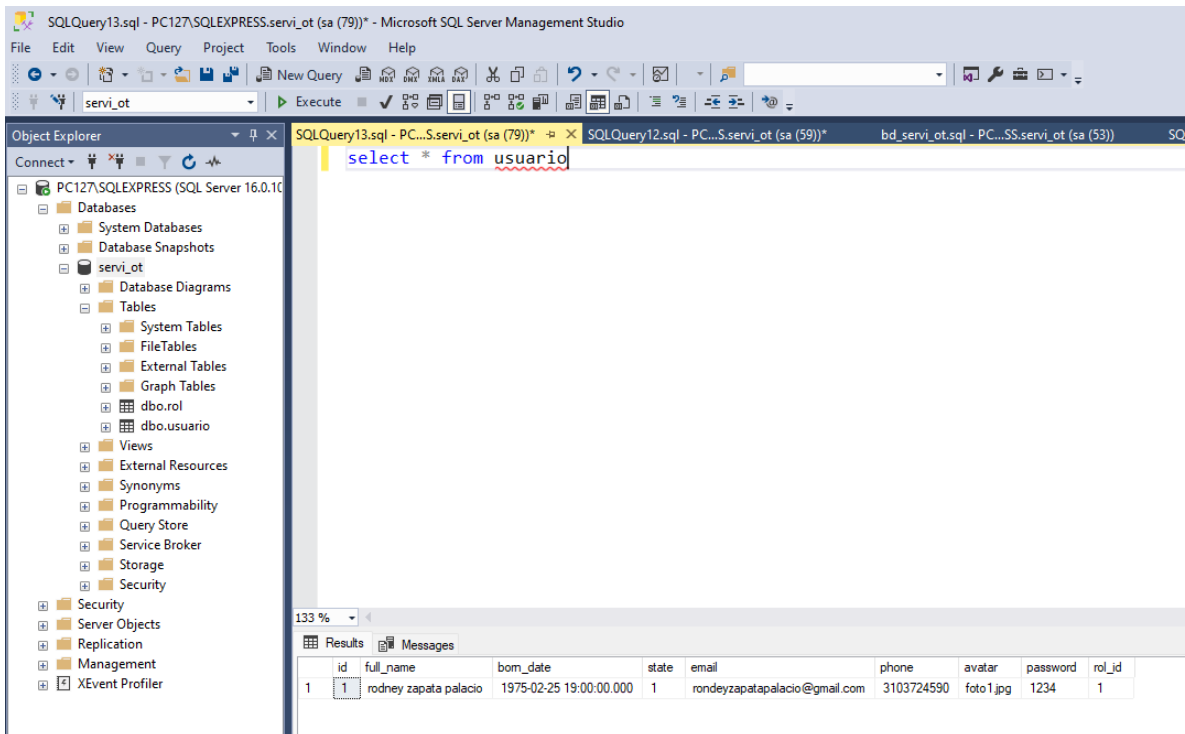
Utilizaremos la tabla de usuario, rol y Cliente

1. Creacion de un usurio desde postman, con el método POST

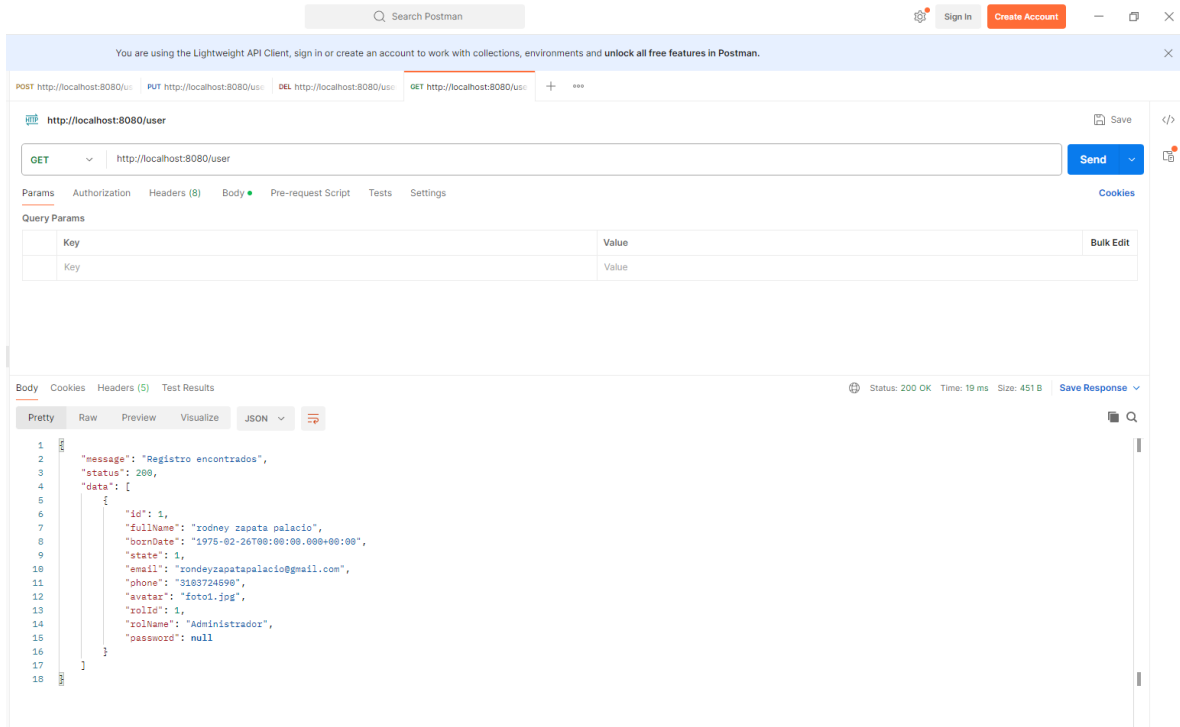
En el siguiente imagen podemos observar la inserción de nuestro primer usuario



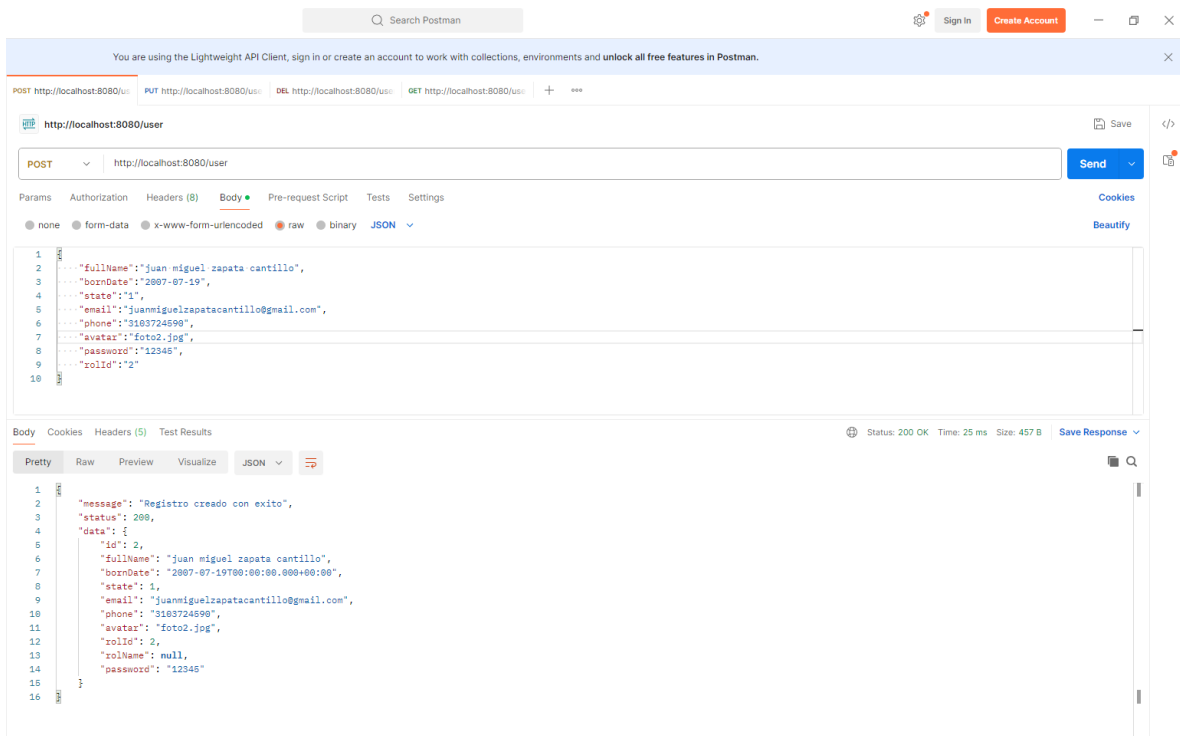
Una vez insertado podemos consultar el registro adicionado desde el motor de la base de datos



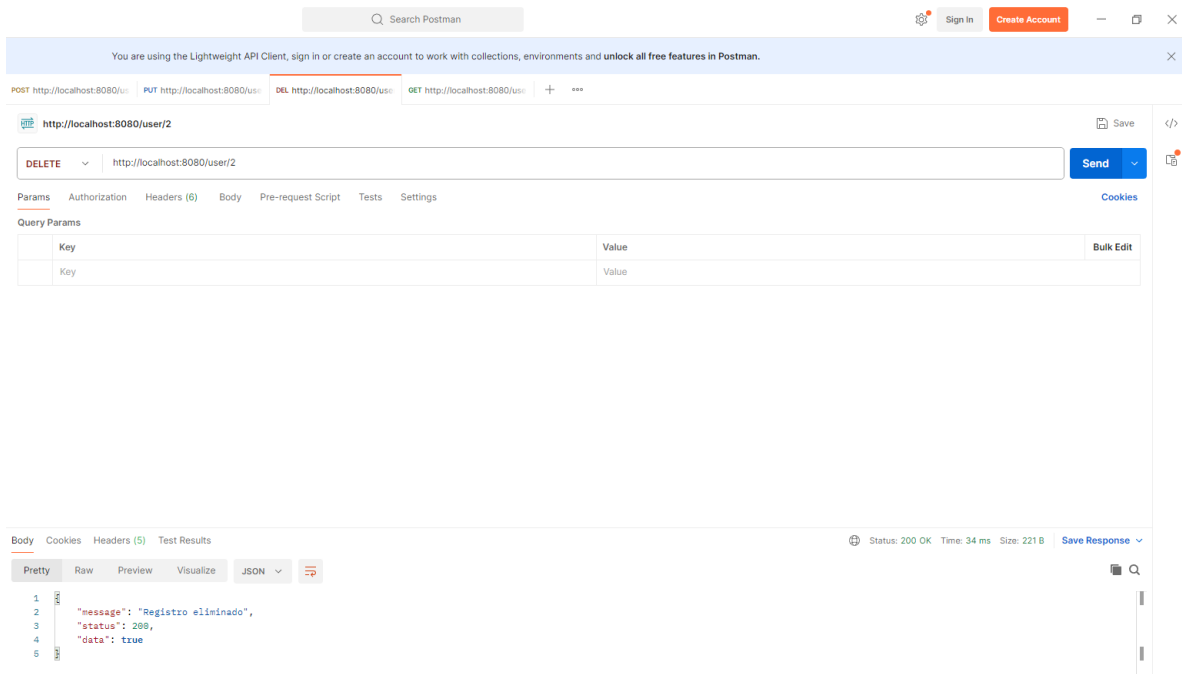
Tabien podemos consultarlo por el método GET desde postman



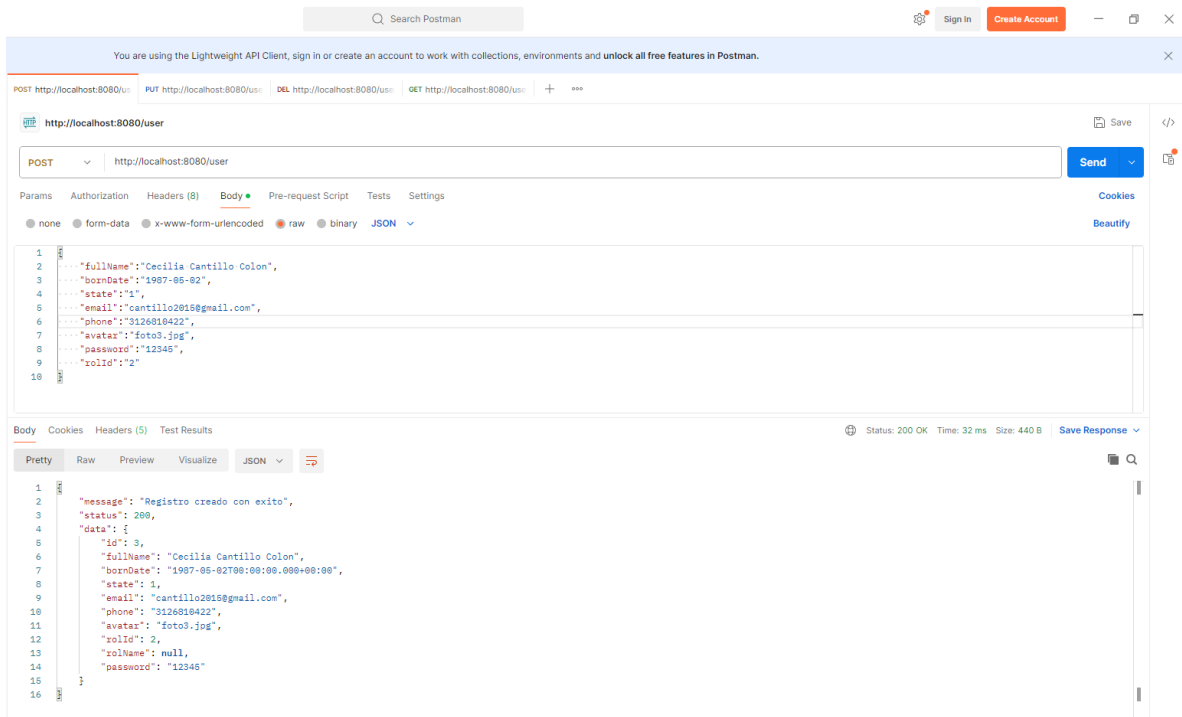
Insertamos segundo registro, según la imagen:



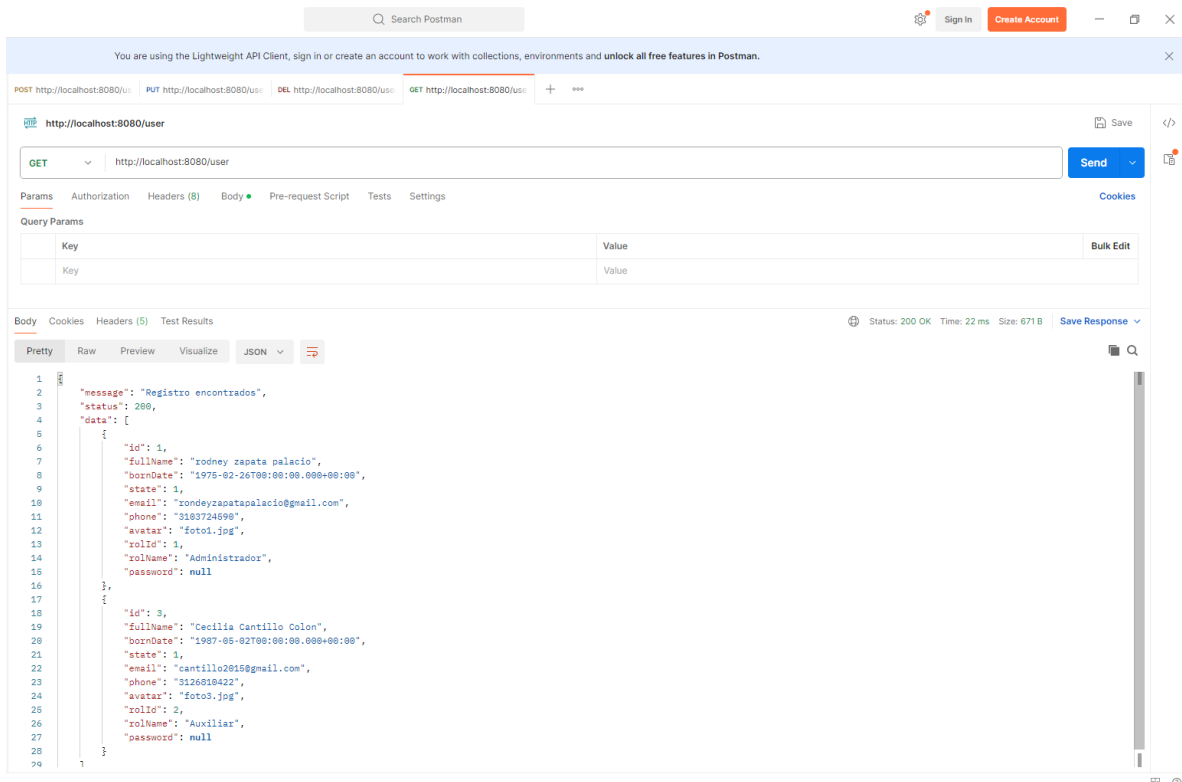
Eliminamos registro numero 2 según se muestra a continuación:



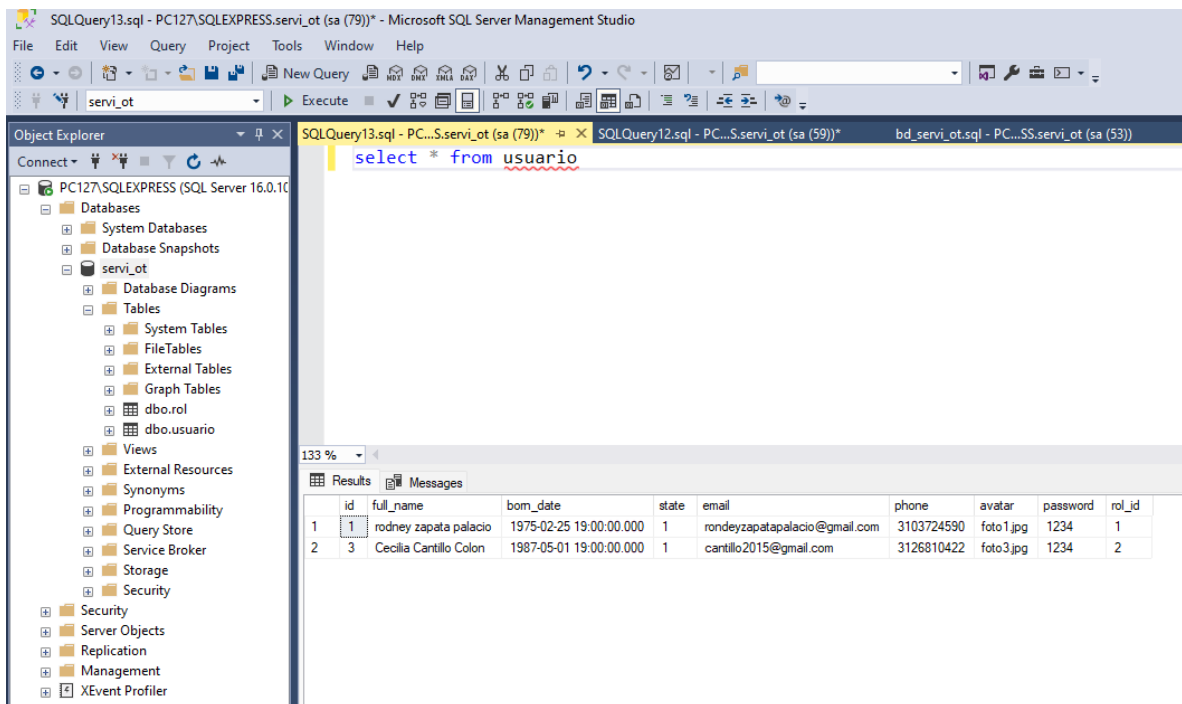
Insertamos un tercer registro según se muestra a continuación



Consultamos los usuarios después de eliminar el registro 2 desde postman



Y dentro del motor de la base de datos podemos notar que el registro con el id 2, ya no existe



Ahora consultaremos la información del registro 3 pasándole el Id desde postman

Search Postman

You are using the Lightweight API Client, sign in or create an account to work with collections, environments and unlock all free features in Postman.

POST http://localhost:8080/user/3 PUT http://localhost:8080/user/3 DEL http://localhost:8080/user/3 GET http://localhost:8080/user/3 GET http://localhost:8080/user/3 + ...

http://localhost:8080/user/3

GET http://localhost:8080/user/3 Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings

Query Params

| Key | Value | Bulk Edit |
|-----|-------|-----------|
| Key | Value |           |

Body Cookies Headers (5) Test Results

Status: 200 OK Time: 53 ms Size: 437 B Save Response

Pretty Raw Preview Visualize JSON

```
1  {
2    "message": "Registro encontrado",
3    "status": 200,
4    "data": {
5      "id": 3,
6      "fullName": "Cecilia Cantillo Colon",
7      "bornDate": "1987-06-02T00:00:00.000+00:00",
8      "state": 1,
9      "email": "cantillo2015@gmail.com",
10     "phone": "3126810422",
11     "avatar": "foto3.jpg",
12     "rolId": 2,
13     "rolName": "Auxiliar",
14     "password": null
15   }
16 }
```

## Modificación de usuarios

### Al usuario 3 le cambiamos la clave de 1234 por ceci123

Search Postman

You are using the Lightweight API Client, sign in or create an account to work with collections, environments and unlock all free features in Postman.

POST http://localhost:8080/user/3 PUT http://localhost:8080/user/3 DEL http://localhost:8080/user/3 GET http://localhost:8080/user/3 GET http://localhost:8080/user/3 + ...

http://localhost:8080/user/3

PUT http://localhost:8080/user/3 Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary JSON

Beautify

```
1  {
2    "fullName": "Cecilia Cantillo Colon",
3    "bornDate": "1987-06-02T00:00:00.000+00:00",
4    "state": 1,
5    "email": "cantillo2015@gmail.com",
6    "phone": "3126810422",
7    "avatar": "foto3.jpg",
8    "rolId": 2,
9    "password": "ceci123"
10 }
11
```

Body Cookies Headers (5) Test Results

Status: 200 OK Time: 310 ms Size: 437 B Save Response

Pretty Raw Preview Visualize JSON

```
1  {
2    "message": "Registro actualizado",
3    "status": 200,
4    "data": {
5      "id": 3,
6      "fullName": "Cecilia Cantillo Colon",
7      "bornDate": "1987-06-02T00:00:00.000+00:00",
8      "state": 1,
9      "email": "cantillo2015@gmail.com",
10     "phone": "3126810422",
11     "avatar": "foto3.jpg",
12     "rolId": 2,
13     "rolName": null,
14     "password": "ceci123"
15   }
16 }
```

Una vez actualizado podemos notar la nueva clave del registro 3 desde sqlserver

Microsoft SQL Server Management Studio

File Edit View Window Help

Execute

SQLQuery13.sql - PC...S.servi\_ot (sa (79))\* SQLQuery12.sql - PC...S.servi\_ot (sa (59))\* bd\_servi\_ot.sql - PC...SS.servi\_ot (sa (53)) SQLQue

```
select * from usuario
```

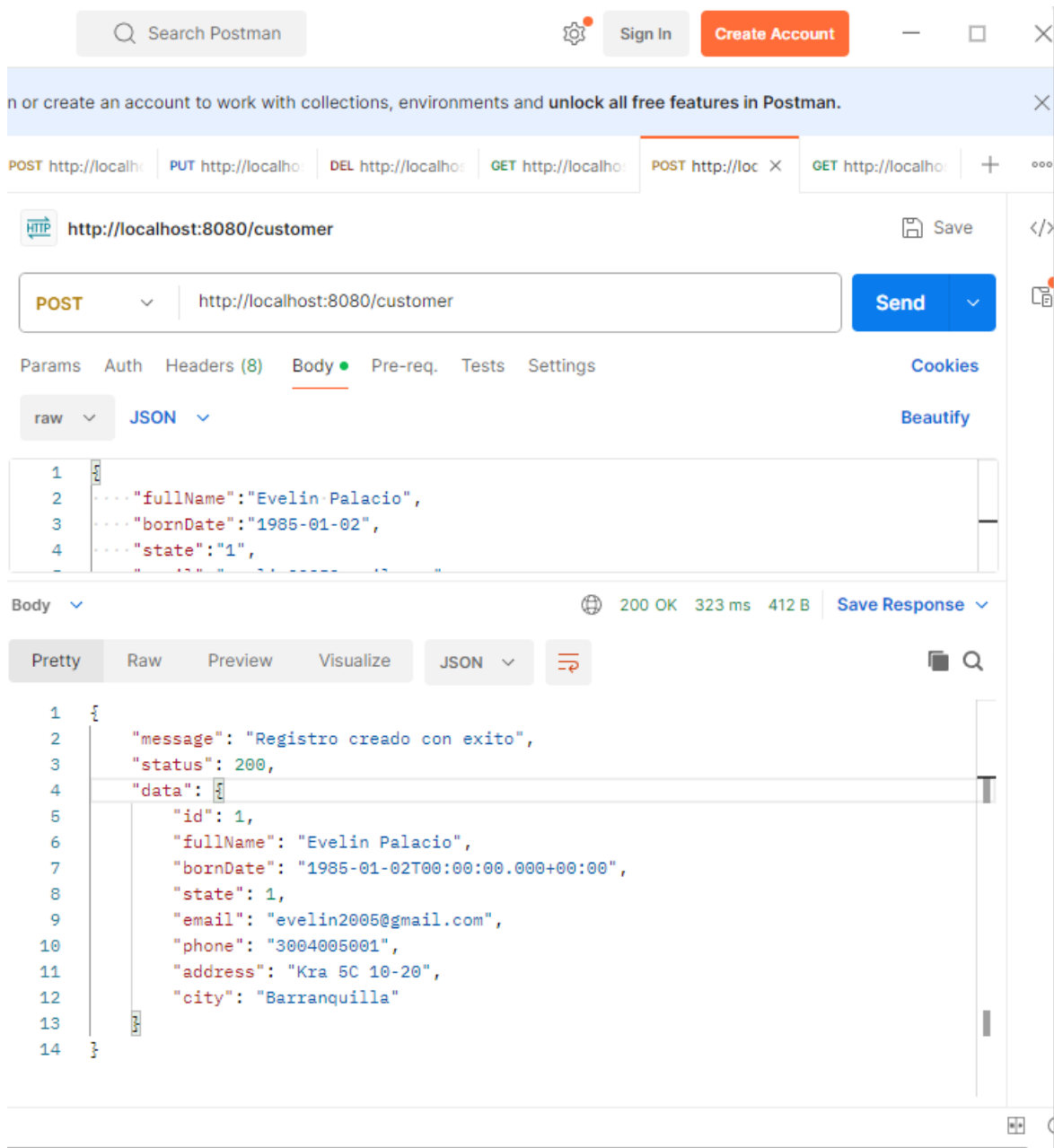
133 %

Results Messages

|   | id | full_name              | bom_date                | state | email                         | phone      | avatar    | password | rol_id |
|---|----|------------------------|-------------------------|-------|-------------------------------|------------|-----------|----------|--------|
| 1 | 1  | rodney zapata palacio  | 1975-02-25 19:00:00.000 | 1     | rondeyzapatapalacio@gmail.com | 3103724590 | foto1.jpg | 1234     | 1      |
| 2 | 3  | Cecilia Cantillo Colon | 1987-05-01 19:00:00.000 | 1     | cantillo2015@gmail.com        | 3126810422 | foto3.jpg | ceci123  | 2      |

Ahora insertamos un cliente nuevo desde Postman





Consultamos cliente ingresado desde e motor de la base de datos

PC127\SQLEXPRESS....ot - dbo.customer    SQLQuery13.sql - PC...S.servi\_ot (sa (79))\*    SQLQuery12.sql - PC...S.servi\_ot (sa (59))\*

```
select * from customer
```

133 %

Results    Messages

|   | id | full_name      | bom_date                | state | email                | phone      | address      | city         |
|---|----|----------------|-------------------------|-------|----------------------|------------|--------------|--------------|
| 1 | 1  | Evelin Palacio | 1985-01-01 19:00:00.000 | 1     | evelin2005@gmail.com | 3004005001 | Kra 5C 10-20 | Barranquilla |

Lo podemos consultar desde Postman

Search Postman

Sign In Create Account

in or create an account to work with collections, environments and unlock all free features in Postman.

POST http://localho: PUT http://localho: DEL http://localho: GET http://localho: POST http://localho: GET http://loca X + ...

HTTP http://localhost:8080/customer Save </>

GET http://localhost:8080/customer Send

Params Auth Headers (6) Body Pre-req. Tests Settings Cookies

Query Params

| Key | Value | Bulk Edit |
|-----|-------|-----------|
| Key | Value |           |

Body 200 OK 29 ms 409 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "message": "Registro encontrados",
3   "status": 200,
4   "data": [
5     {
6       "id": 1,
7       "fullName": "Evelin Palacio",
8       "bornDate": "1985-01-02T00:00:00.000+00:00",
9       "state": 1,
10      "email": "evelin2005@gmail.com",
11      "phone": "3004005001",
12      "address": "Kra 5C 10-20",
13      "city": "Barranquilla"
14    }
15  ]
16 }
```

Insertamos un segundo cliente desde Postman

POST http://localhost:8080/ PUT http://localhost:8080/ DEL http://localhost:8080/ GET http://localhost:8080/ POST http://localhost:8080/ GET http://localhost:8080/ + ...

HTTP http://localhost:8080/customer Save </>

POST http://localhost:8080/customer Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies Beautify

none form-data x-www-form-urlencoded raw binary JSON

```
1 {
2   "fullName": "Tecnoglas",
3   "bornDate": "1985-01-02",
4   "state": "1",
5   "email": "samuel@tecno.com",
6   "phone": "3126810844",
7   "address": "via 40 calle 74",
8   "city": "Barranquilla"
9 }
```

Body Cookies Headers (5) Test Results 200 OK 24 ms 406 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "message": "Registro creado con exito",
3   "status": 200,
4   "data": {
5     "id": 2,
6     "fullName": "Tecnoglas",
7     "bornDate": "1985-01-02T00:00:00.000+00:00",
8     "state": 1,
9     "email": "samuel@tecno.com",
10    "phone": "3126810844",
11    "address": "via 40 calle 74",
12    "city": "Barranquilla"
13  }
14 }
```

Consultamos los dos clientes desde sqlserver

PC127\SQLEXPRESS....ot - dbo.customer SQLQuery13.sql - PC...S.servi\_ot (sa (79))\* X SQLQuery12.sql - PC...S.servi\_ot (

select \* from customer

133 %

Results Messages

|   | id | full_name      | bom_date                | state | email                | phone      | address         | city         |
|---|----|----------------|-------------------------|-------|----------------------|------------|-----------------|--------------|
| 1 | 1  | Evelin Palacio | 1985-01-01 19:00:00.000 | 1     | evelin2005@gmail.com | 3004005001 | Kra 5C 10-20    | Barranquilla |
| 2 | 2  | Tecnoglas      | 1985-01-01 19:00:00.000 | 1     | samuel@tecno.com     | 3126810844 | via 40 calle 74 | Barranquilla |

Ahora vamos a eliminar el cliente 1 desde postman

The screenshot shows the Postman interface with a DELETE request to `http://localhost:8080/customer/1`. The response is a JSON object indicating successful deletion.

DELETE `http://localhost:8080/customer/1` Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Query Params

| Key | Value | Bulk Edit |
|-----|-------|-----------|
| Key | Value |           |

Body Cookies Headers (5) Test Results 200 OK 19 ms 221 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "message": "Registro eliminado",
3   "status": 200,
4   "data": true
5 }
```

Consultamos los clientes y observamos el cliente 1 ya no existe

POST http://localhost:8080/PUT http://localhost:8080/DEL http://localhost:8080/GET http://localhost:8080/POST http://localhost:8080/GET http://localhost:8080/DEL http://localhost:8080/

HTTP http://localhost:8080/customer

GET http://localhost:8080/customerSend

ParamsAuthorizationHeaders (6)BodyPre-request ScriptTestsSettingsCookies

Query Params

| Key | Value | Bulk Edit |
|-----|-------|-----------|
| Key | Value |           |

BodyCookiesHeaders (5)Test Results200 OK 21 ms 403 BSave Response

PrettyRawPreviewVisualizeJSON

```
1  {
2    "message": "Registro encontrados",
3    "status": 200,
4    "data": [
5      {
6        "id": 2,
7        "fullName": "Tecnoglas",
8        "bornDate": "1985-01-02T00:00:00.000+00:00",
9        "state": 1,
10       "email": "samuel@tecno.com",
11       "phone": "3126810844",
12       "address": "via 40 calle 74",
13       "city": "Barranquilla"
14     }
15   ]
16 }
```