



DOCUMENTO_PRUEBAS_POSTMAN

_AA5_EV04

Aprendiz: Mónica Ismelia Cañas Reyes

Programa: Tecnólogo en Análisis y Desarrollo de Software

Institución: Servicio Nacional de Aprendizaje – SENA

Centro: Centro Nacional de Asistencia Técnica a la Industria – ASTIN

Evidencia: GA7-220501096-AA5-EV04

Diciembre de 2025

Registrar Usuario

```
{  
  "nombre": "Anita Ruiz",  
  "correo": "anitar@example.com",  
  "contrasena": "{{password}}"  
}
```

The screenshot shows the Postman application interface. At the top, there is a search bar labeled 'Search Postman' and various navigation icons. Below the header, a URL bar displays 'http://localhost:8080/api/auth/register'. The main area is a POST request configuration. The method is set to 'POST' and the URL is 'http://localhost:8080/api/auth/register'. The 'Body' tab is selected, showing a JSON payload:

```
1 {  
2   "nombre": "Anita Ruiz",  
3   "correo": "anitar@example.com",  
4   "contrasena": "{{password}}"  
5 }
```

Below the body, the 'Headers' section shows 14 items. The response status is '200 OK' in a green box. The response body contains the message: 'Usuario registrado exitosamente: anitar@example.com'. At the bottom, there are buttons for 'Runner' and 'Start Proxy'.

Login Exitoso

The screenshot shows the Postman application interface. At the top, the URL `http://localhost:8080/api/auth/login` is entered into the address bar. Below it, a POST request is selected with the URL `http://localhost:8080/api/auth/login`. The 'Body' tab is active, showing raw JSON data:

```
1 {
2   "correo": "anitar@example.com",
3   "contrasena": "{{password}}"
4 }
```

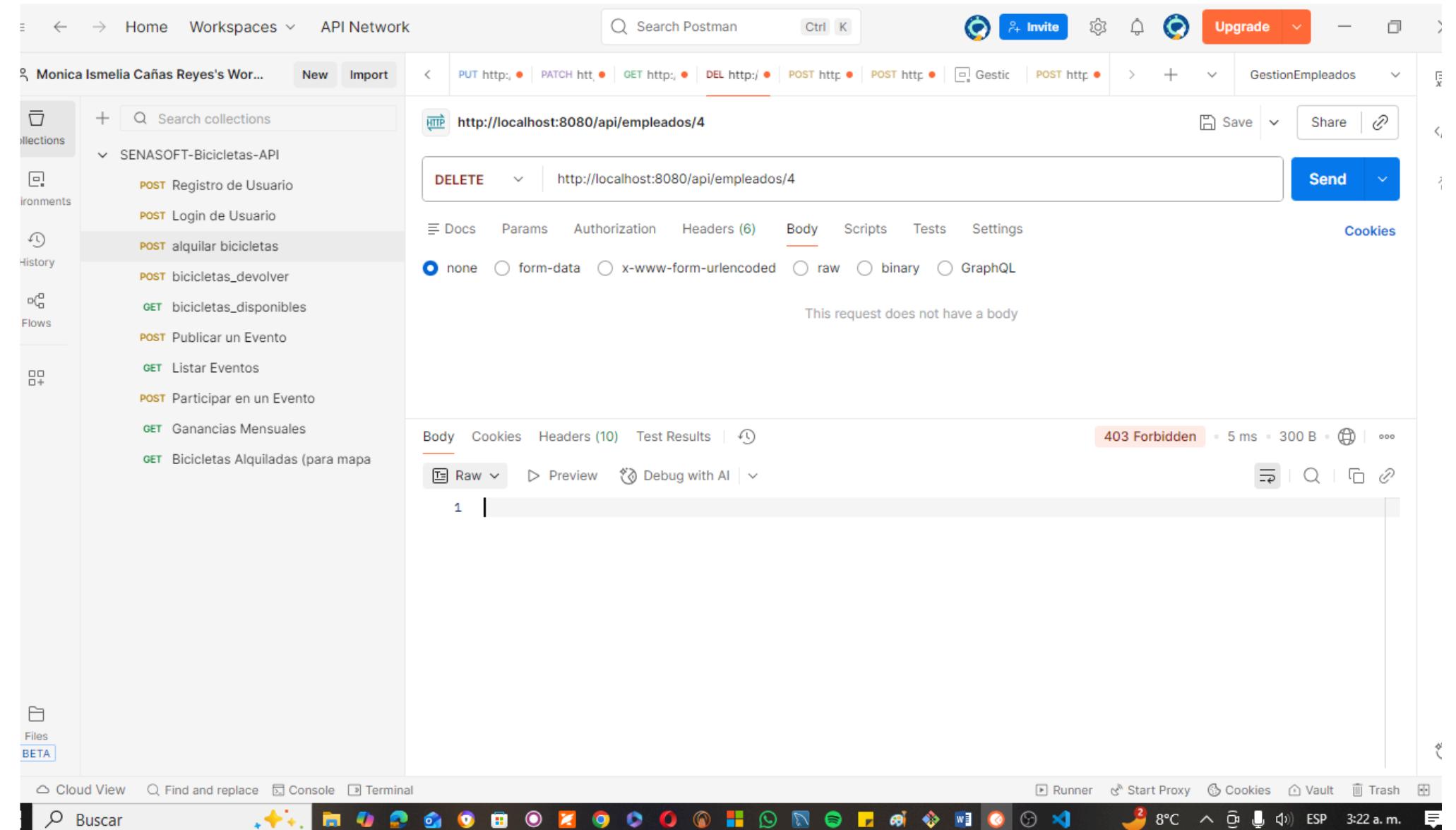
Below the body, the response status is `200 OK` with a response time of `261 ms` and a size of `583 B`. The response body is displayed in JSON format:

```
1 {
2   "token": "eyJhbGciOiJIUzI1NiJ9.eyJzdWIiOiJhbml0YXJAZXhbXBsZS5jb20iLCJpYXQiOjE3NjUyNjg0NjksImV4cCI6MTc2NTM1NDg2OX0.6UIa7Zbl_q9fKchPR399WWiRRFn5PcPekAKFooZFK"
3 }
```

```
{  
  "correo": "anitar@example.com",  
  "contrasena": "{{password}}"  
}
```

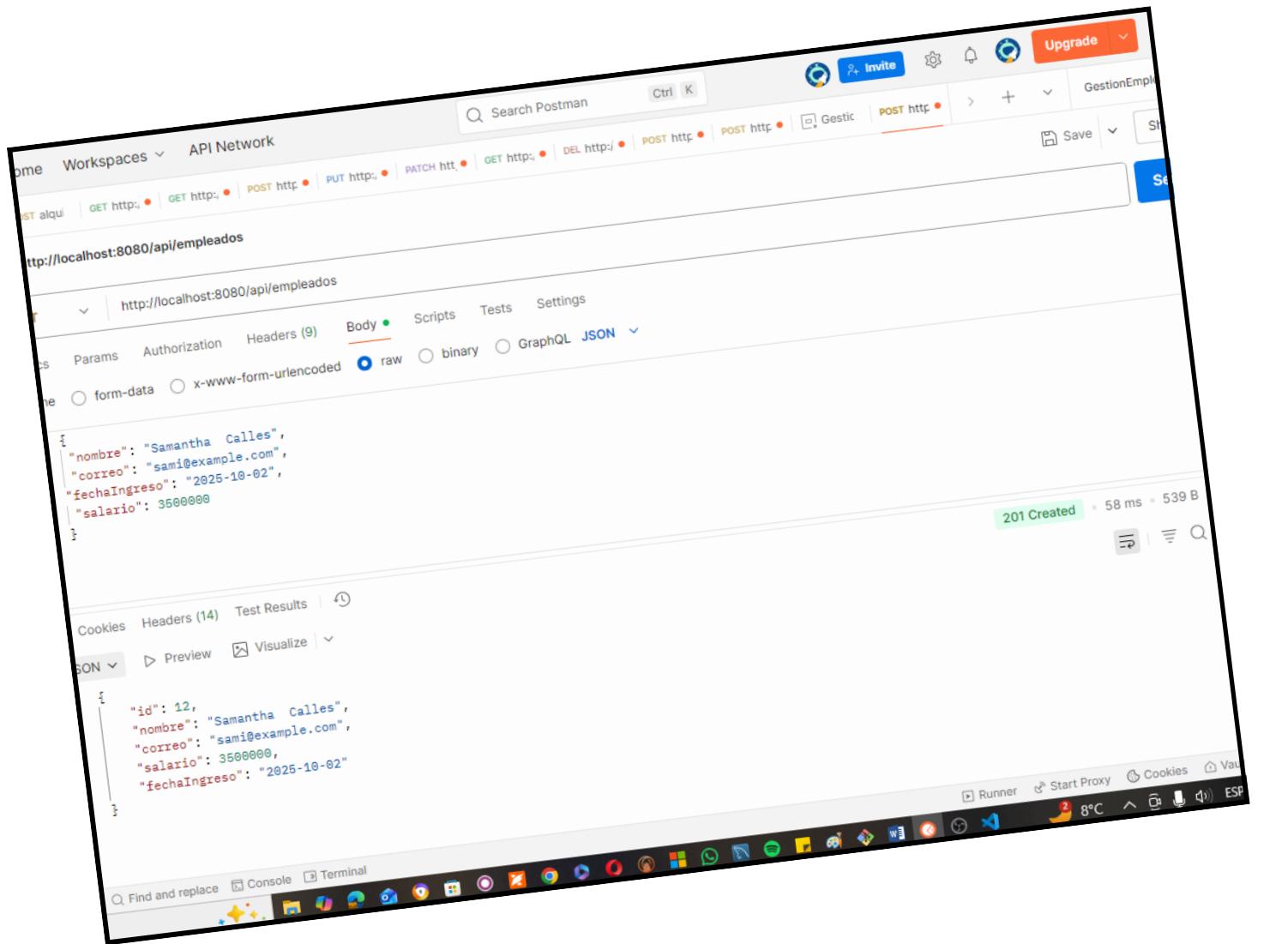
Manejo del Token

Error 403 Forbidden
Ingresar sin Token



Crear Empleado

```
{  
  "nombre": "Samantha Calles",  
  "correo": "sami@example.com",  
  "fechalngreso": "2025-10-02",  
  "salario": 3500000  
}
```



Obtención por ID

Buscar empleado por Id

HTTP <http://localhost:8080/api/empleados/7>

GET <http://localhost:8080/api/empleados/7>

Body Cookies Headers (14) Test Results

{ } JSON Preview Visualize

```
1 {  
2   "id": 7,  
3   "nombre": "Ana Pérez",  
4   "correo": "ana@example.com",  
5   "salario": 2500000.00,  
6   "fechaIngreso": "2025-12-01"  
7 }
```

HTTP <http://localhost:8080/api/empleados>

GET <http://localhost:8080/api/empleados>

Body Cookies Headers (14) Test Results

{ } JSON Preview Visualize

```
1 [  
2   {  
3     "id": 6,  
4     "nombre": "Andrea",  
5     "correo": "andrea@example.com",  
6     "salario": 1800000.00,  
7     "fechaIngreso": "2025-12-09"  
8   },  
9   {  
10    "id": 7,  
11    "nombre": "Ana Pérez",  
12    "correo": "ana@example.com",  
13    "salario": 2500000.00,  
14    "fechaIngreso": "2025-12-01"  
15  }]
```

Obtención de todos los empleados

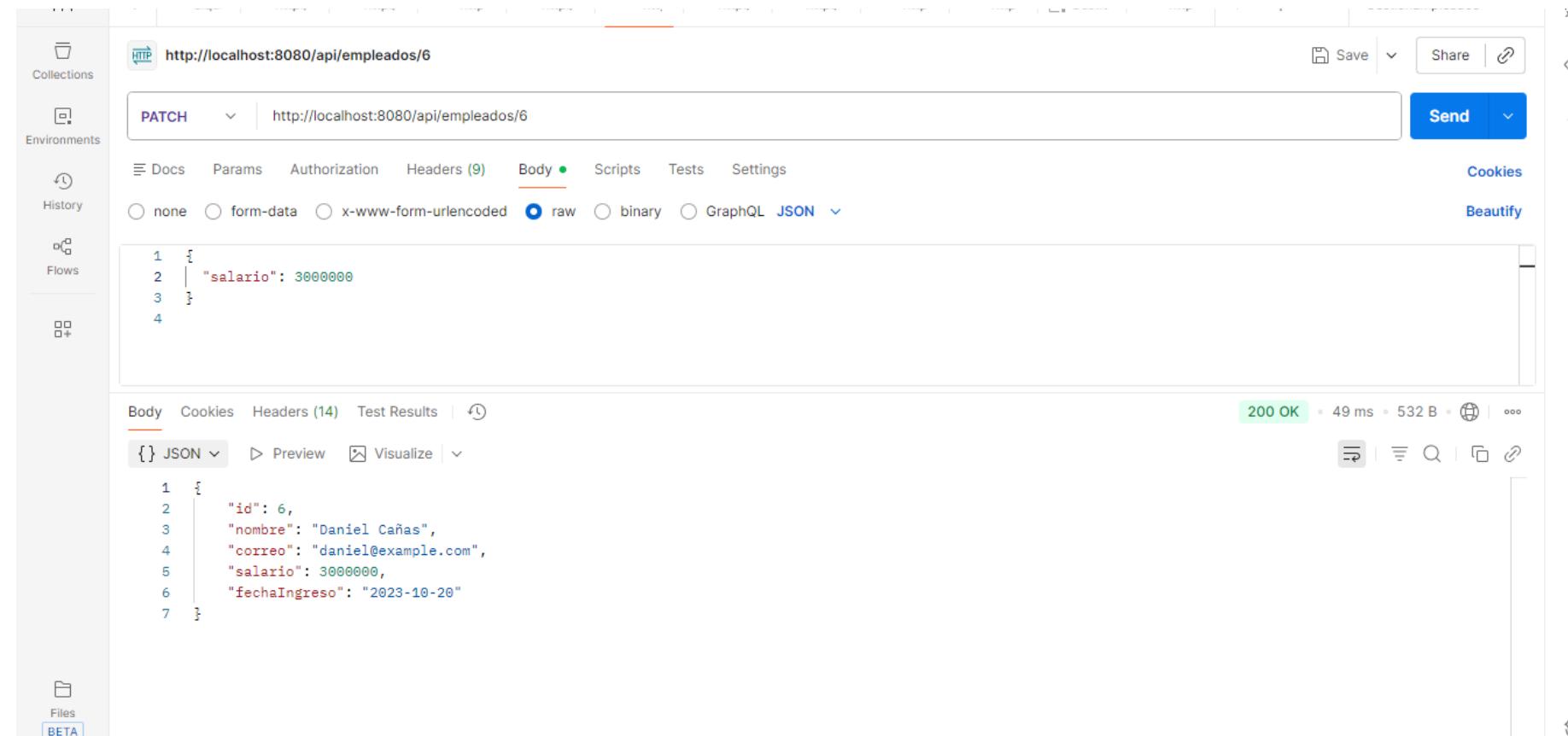
PUT

Actualización de todos los datos de un empleados

{

```
"nombre": "Daniel Cañas",
"correo": "daniel@example.com",
"salario": 4500000,
"fechaIngreso": "2023-10-20"
```

}



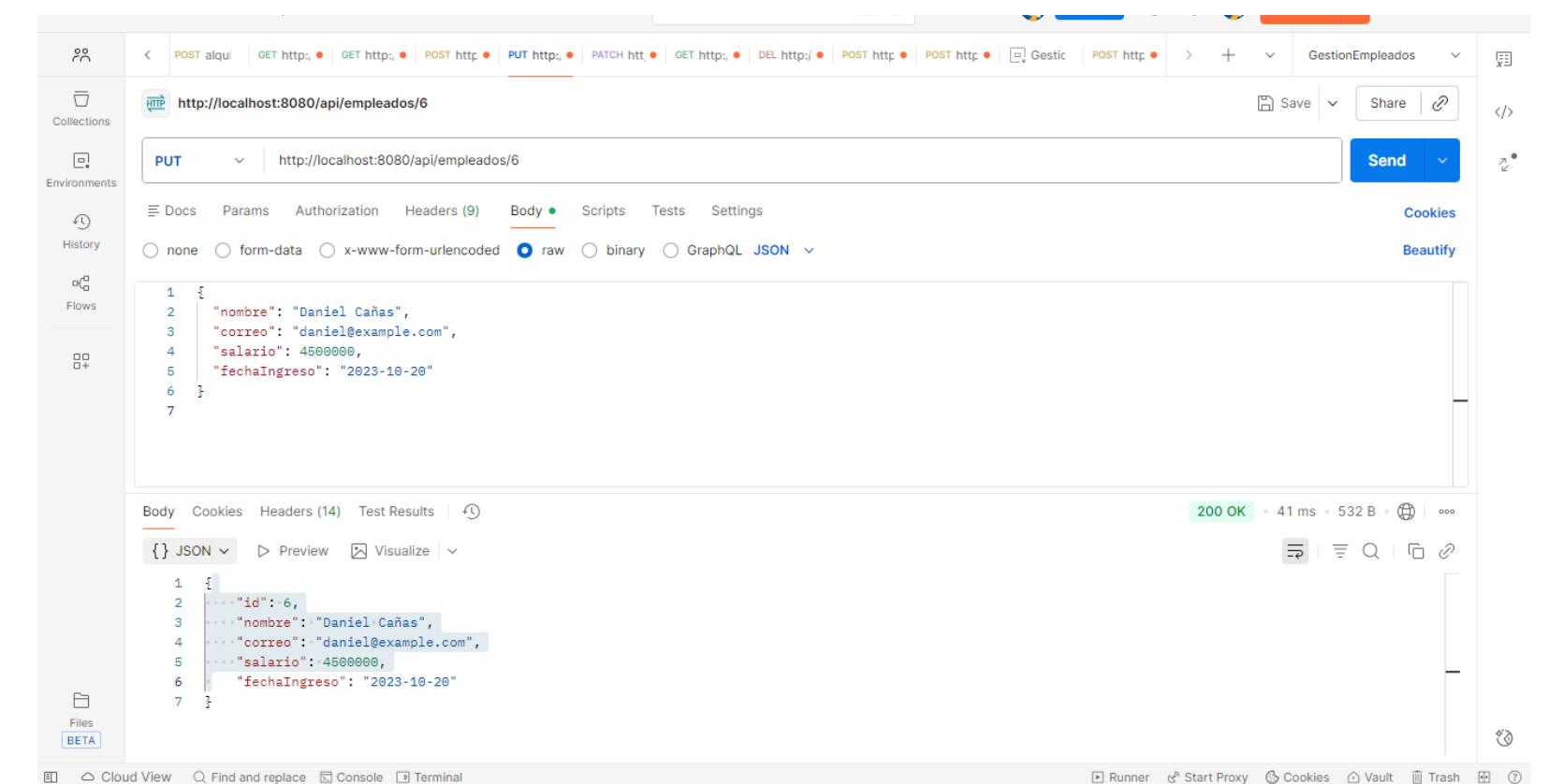
http://localhost:8080/api/empleados/6

PATCH

Body

```
1 {
2   "salario": 3000000
3 }
```

200 OK



http://localhost:8080/api/empleados/6

PUT

Body

```
1 {
2   "nombre": "Daniel Cañas",
3   "correo": "daniel@example.com",
4   "salario": 4500000,
5   "fechaIngreso": "2023-10-20"
6 }
```

200 OK

PATCH

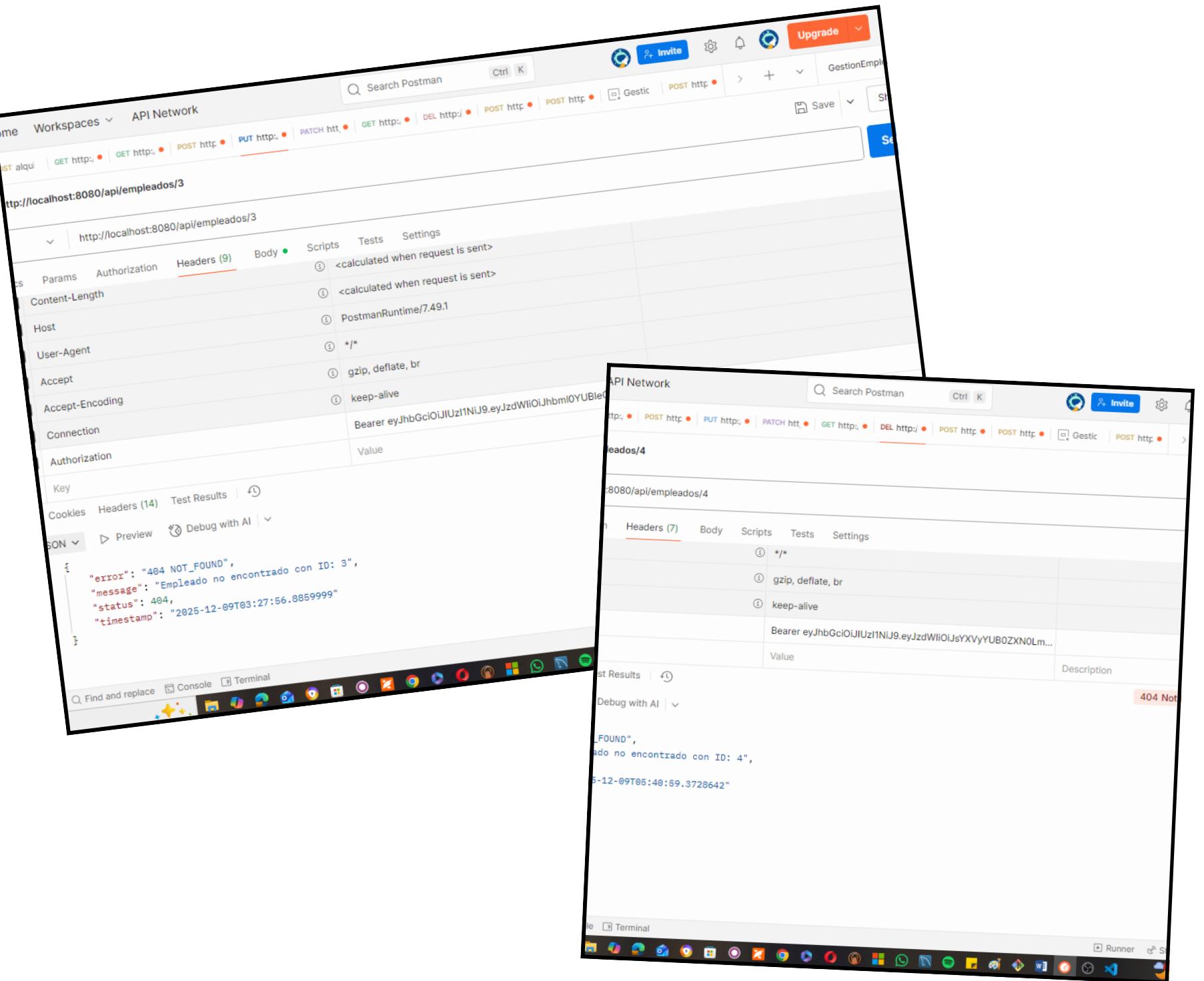
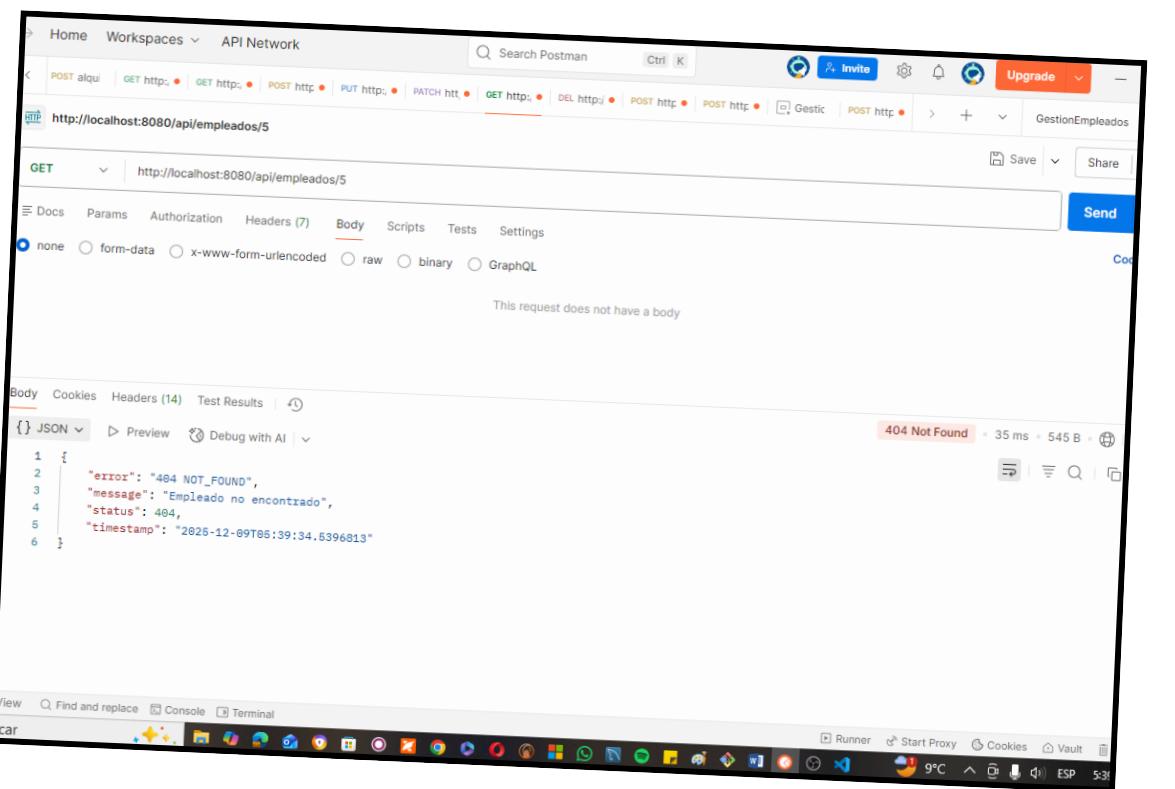
Actualización parcial de los datos de un empleados

{

```
"salario": 5000000
```

}

Error de ID



Eliminar el registro de un empleado DELETE

Eliminar un empleado por ID

Nueva conexión

AS

Tables

Views

Stored Procedures

Functions

empleados

Result Grid

ID	Nombre	Correo	Salario	Fecha Ingreso
6	Daniel Cañas	daniel@example.com	3000000.00	2023-10-20
7	Ana Pérez	ana@example.com	2500000.00	2025-12-01
9	Ana Calles	anacalle@example.com	3500000.00	2025-10-02
10	Juan Pérez	juan@test.com	2500000.00	2025-12-07
11	Juan	jun@test.com	2000000.00	2023-01-01
12	Samantha Calles	sami@example.com	3500000.00	2025-10-02
13	Samy	samy@example.com	1800000.00	2025-12-09
*	NULL	NULL	NULL	NULL

Action Output

#	Time	Action	Message	Duration / Fetch
4	02:47:29	SELECT * FROM empresa.empleados LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000
5	04:55:09	SELECT * FROM empresa.empleados LIMIT 0, 1000	7 row(s) returned	0.000 sec / 0.000
6	05:25:59	SELECT * FROM empresa.empleados LIMIT 0, 1000	7 row(s) returned	0.000 sec / 0.000
7	05:42:17	SELECT * FROM empresa.empleados LIMIT 0, 1000	7 row(s) returned	0.000 sec / 0.000

http://localhost:8080/api/empleados/8

DELETE http://localhost:8080/api/empleados/8

Body

```
{ "mensaje": "Empleado eliminado" }
```

200 OK

Body Cookies Headers (14) Test Results

mensaje: Empleado eliminado

Validación en Base de Datos

GRACIAS

