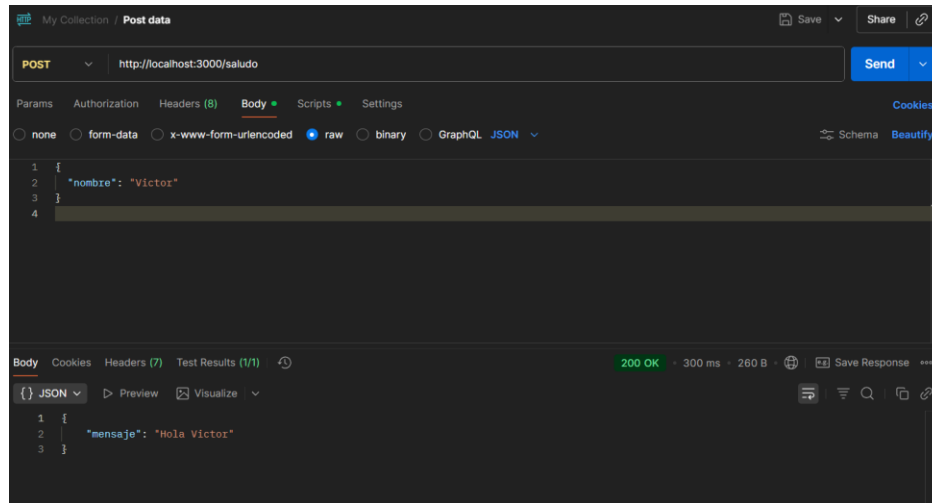
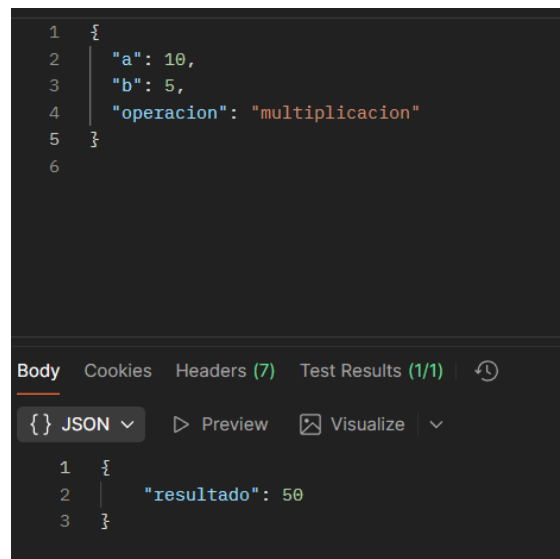


Camargo Flores Victor Alonso

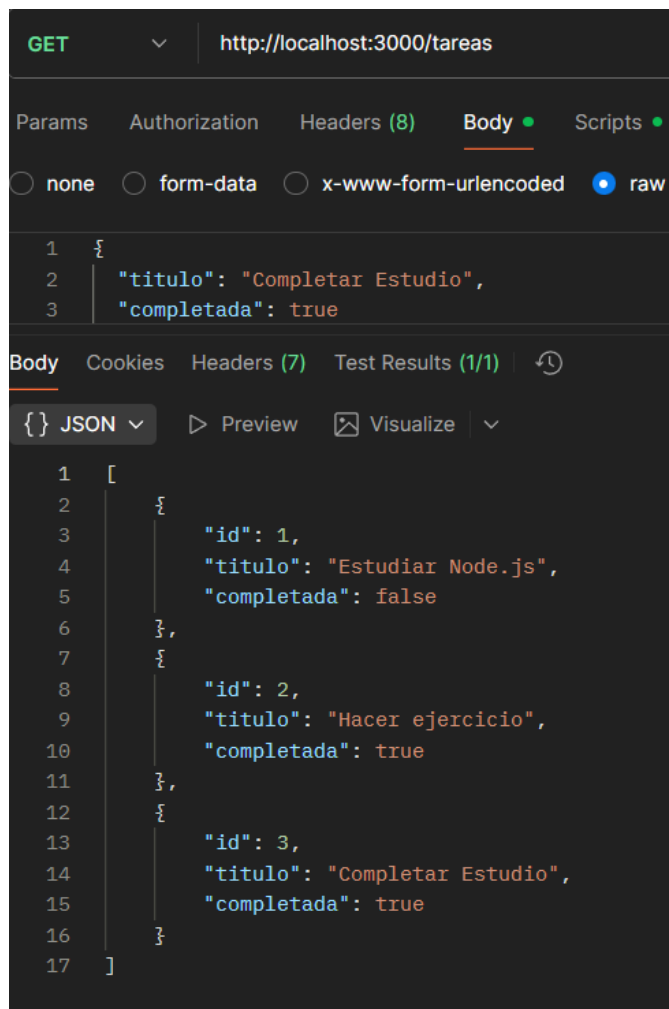
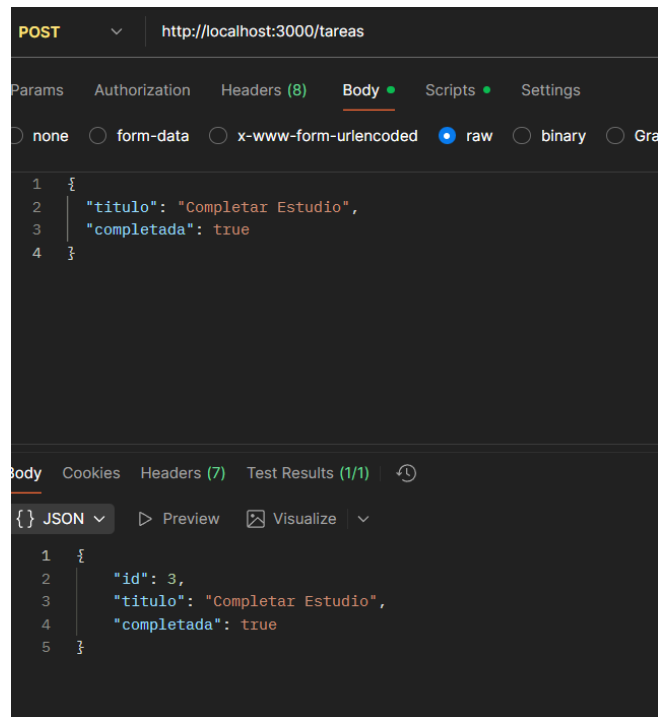
## Ejercicio No.1

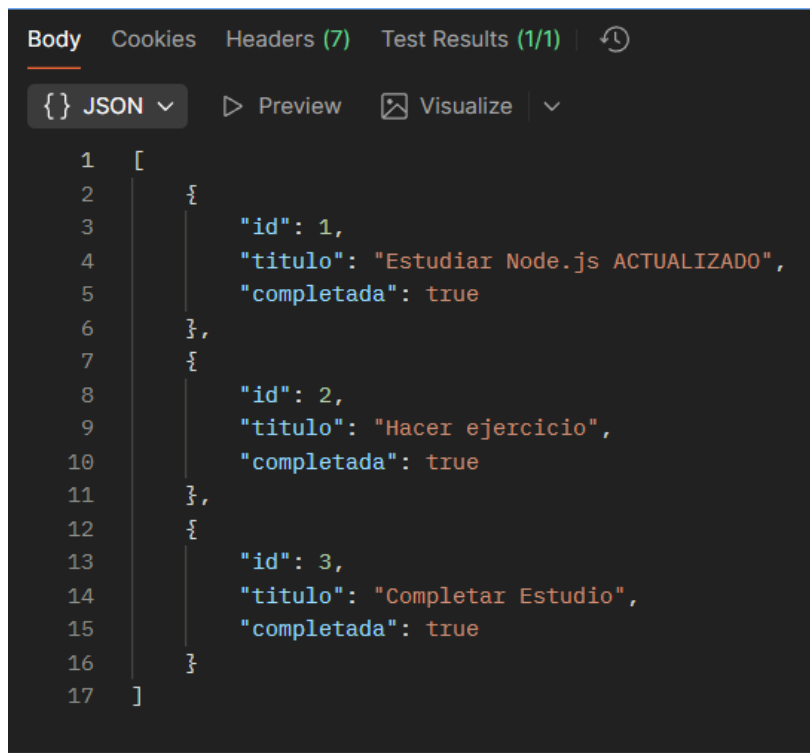
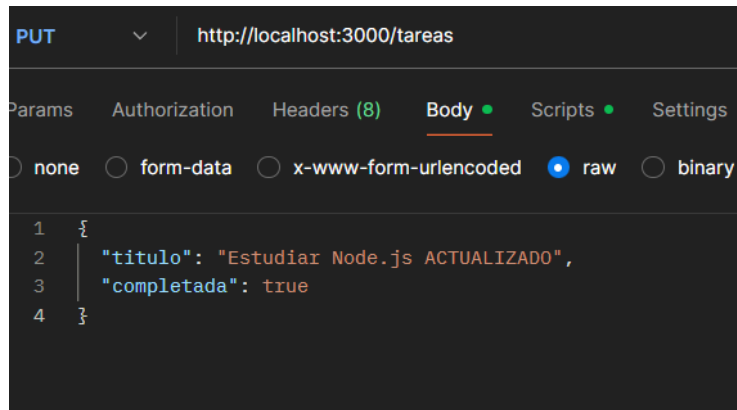


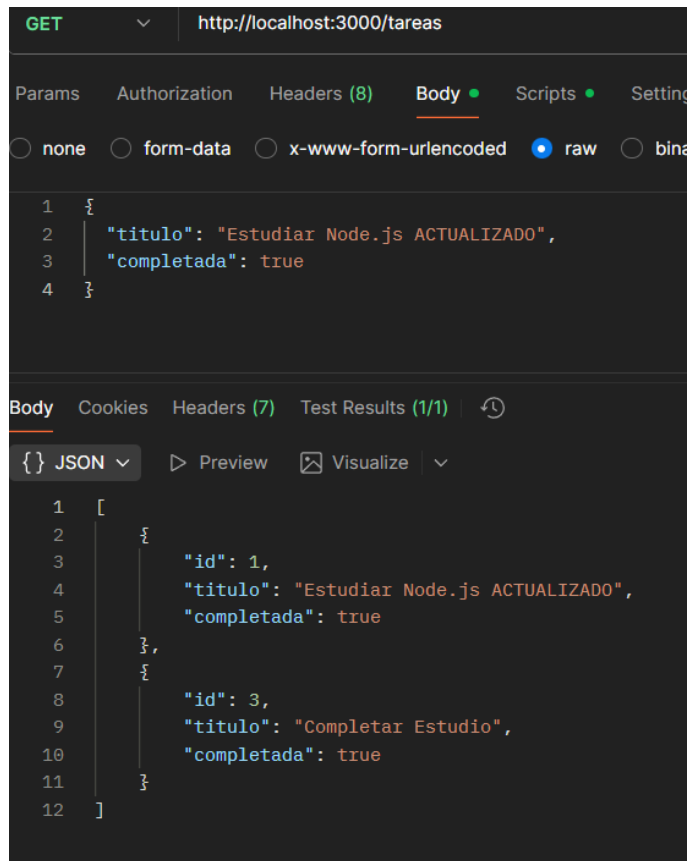
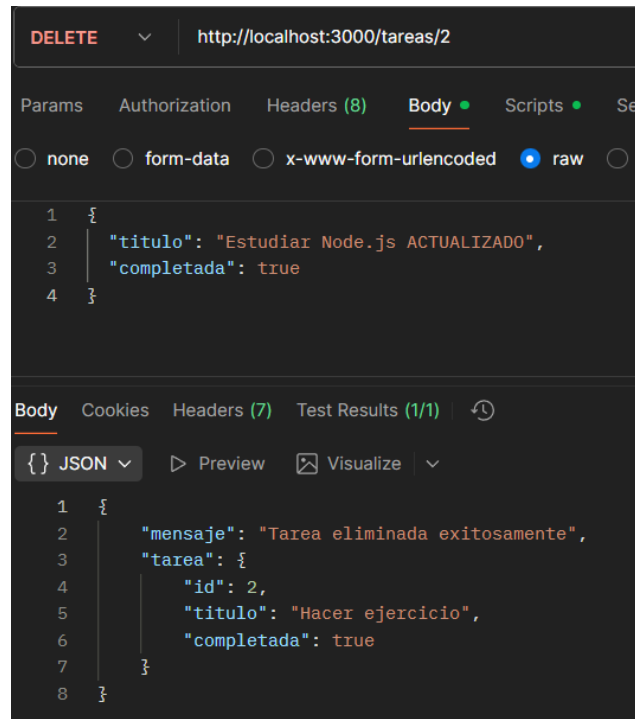
## Ejercicio No.2



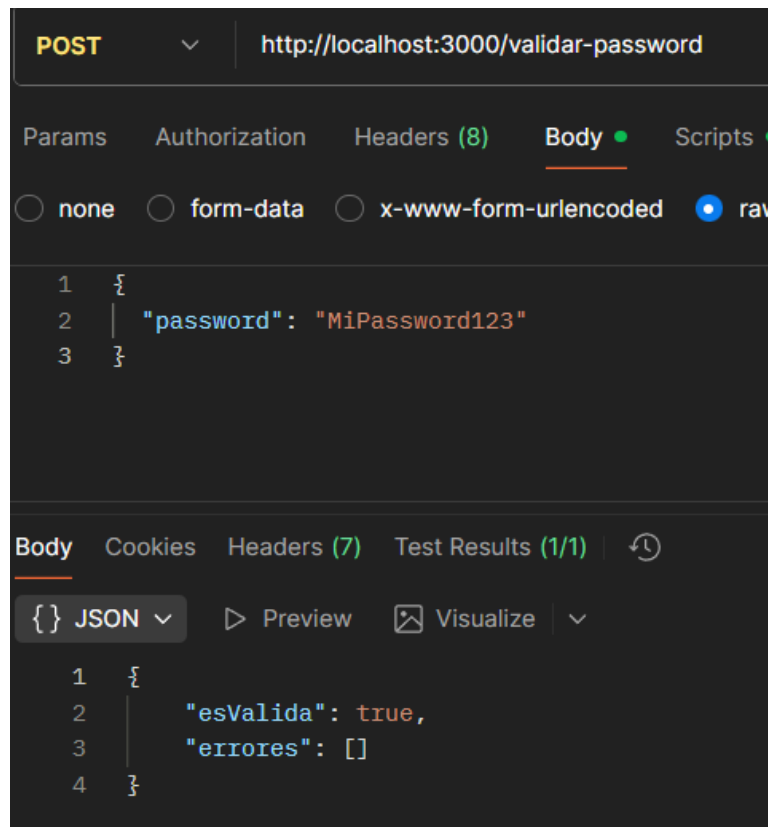
### Ejercicio No.3







#### Ejercicio No.4

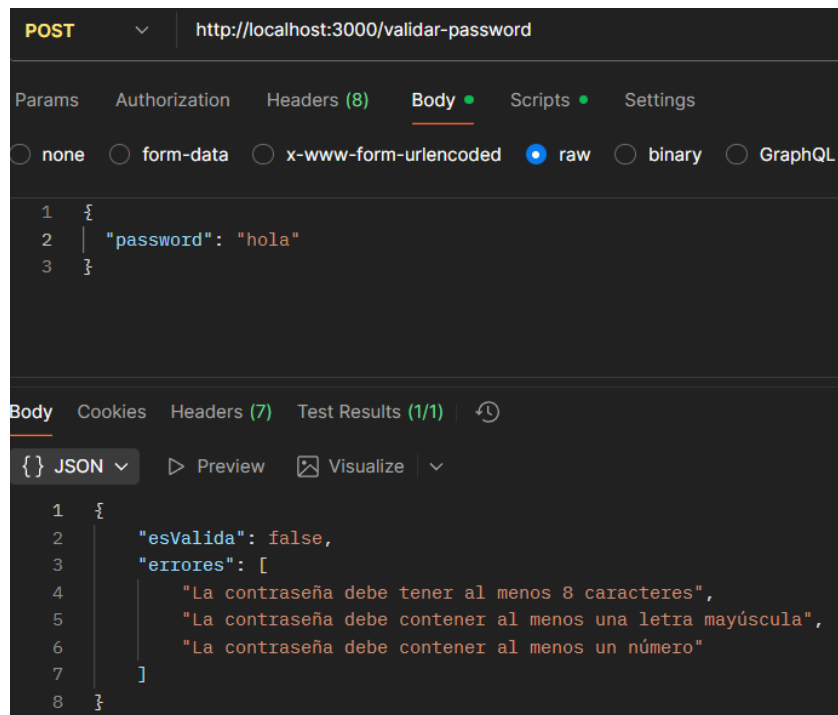


A screenshot of a REST client interface. The top bar shows a **POST** request to `http://localhost:3000/validar-password`. The **Body** tab is selected, showing a JSON payload: `{ "password": "MiPassword123" }`. The bottom section shows the response body as JSON: `{ "esValida": true, "errores": [] }`. The interface includes tabs for Params, Authorization, Headers (8), Body, and Scripts. The response section also has tabs for Body, Cookies, Headers (7), and Test Results (1/1).

```
POST http://localhost:3000/validar-password

{
  "password": "MiPassword123"
}
```

```
{
  "esValida": true,
  "errores": []
}
```



A screenshot of a REST client interface. The top bar shows a **POST** request to `http://localhost:3000/validar-password`. The **Body** tab is selected, showing a JSON payload: `{ "password": "hola" }`. The bottom section shows the response body as JSON: `{ "esValida": false, "errores": [ "La contraseña debe tener al menos 8 caracteres", "La contraseña debe contener al menos una letra mayúscula", "La contraseña debe contener al menos un número" ] }`. The interface includes tabs for Params, Authorization, Headers (8), Body, Scripts, and Settings. The response section also has tabs for Body, Cookies, Headers (7), and Test Results (1/1).

```
POST http://localhost:3000/validar-password

{
  "password": "hola"
}
```

```
{
  "esValida": false,
  "errores": [
    "La contraseña debe tener al menos 8 caracteres",
    "La contraseña debe contener al menos una letra mayúscula",
    "La contraseña debe contener al menos un número"
  ]
}
```

## Ejercicio No.5

The screenshot shows a REST client interface with a POST request to `http://localhost:3000/convertir-temperatura`. The request body is a JSON object with the following structure:

```
1 {
2   "valor": 50,
3   "desde": "C",
4   "hacia": "F"
5 }
```

The response body is also a JSON object, showing the converted values:

```
1 {
2   "valorOriginal": 50,
3   "valorConvertido": 122,
4   "escalaOriginal": "C",
5   "escalaConvertida": "F"
6 }
```

## Ejercicio No.6

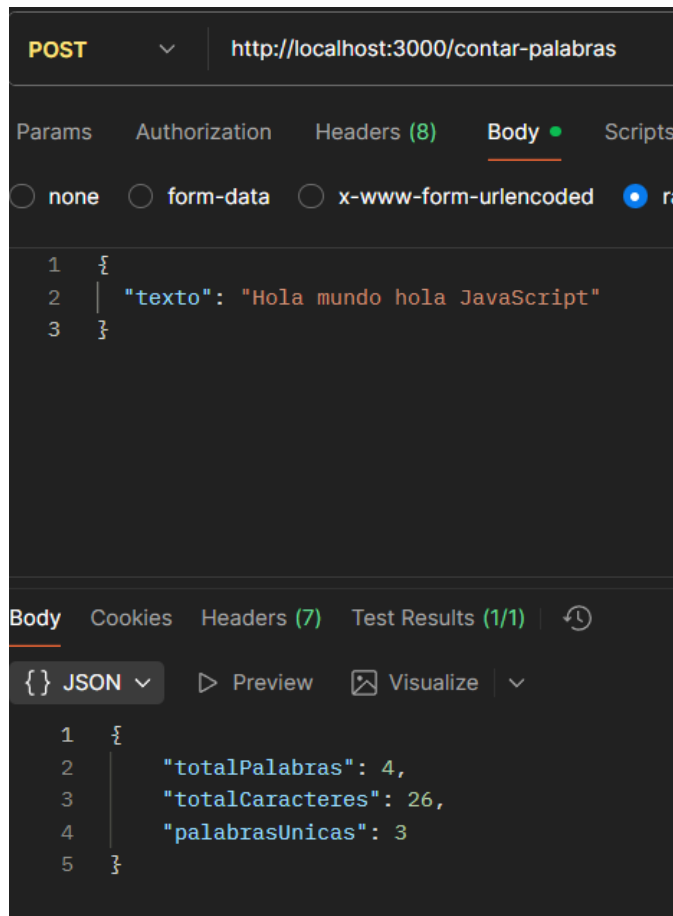
The screenshot shows a REST client interface with a POST request to `http://localhost:3000/buscar`. The request body is a JSON object with the following structure:

```
1 {
2   "array": [1, 2, 3, "hola", true, 5],
3   "elemento": "hola"
4 }
```

The response body is a JSON object indicating the search results:

```
1 {
2   "encontrado": true,
3   "indice": 3,
4   "tipoElemento": "string"
5 }
```

## Ejercicio No.7



## Ejercicio No.8

The screenshot displays a REST client interface with a POST request to `http://localhost:3000/generar-perfil`. The request body is a JSON object with the following structure:

```
1 {  
2   "nombre": "Ana López",  
3   "edad": 15,  
4   "intereses": ["deportes", "lectura"]  
5 }
```

The response body is also in JSON format, showing the user profile data and additional metadata:

```
1 {  
2   "usuario": {  
3     "nombre": "Ana López",  
4     "edad": 15,  
5     "intereses": [  
6       "deportes",  
7       "lectura"  
8     ]  
9   },  
10  "id": "6c83c8a8-fffa-4de3-945d-c65f3cba1a71",  
11  "fechaCreacion": "2025-10-08T15:16:58.180Z",  
12  "categoria": "junior"  
13 }
```



## Ejercicio No.9

**POST** ⌵ http://localhost:3000/calcular-promedio

Params Authorization Headers (8) **Body** ● Scripts

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw

```
1 {
2   "calificaciones": [8, 7, 9, 6, 10]
3 }
```

Body Cookies Headers (7) Test Results (1/1) ↺

{} **JSON** ⌵ ▶ Preview 🖼️ Visualize ⌵

```
1 {
2   "promedio": 8,
3   "calificacionMasAlta": 10,
4   "calificacionMasBaja": 6,
5   "estado": "aprobado"
6 }
```

**POST** ⌵ http://localhost:3000/calcular-promedio

Params Authorization Headers (8) **Body** ● Scripts ●

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw

```
1 {
2   "calificaciones": [5, 4, 3, 6, 4]
3 }
```

Body Cookies Headers (7) Test Results (1/1) ↺

{} **JSON** ⌵ ▶ Preview 🖼️ Visualize ⌵

```
1 {
2   "promedio": 4.4,
3   "calificacionMasAlta": 6,
4   "calificacionMasBaja": 3,
5   "estado": "reprobado"
6 }
```

## Ejercicio No.10

The screenshot displays a REST client interface with a GET request to `http://localhost:3000/productos`. The 'Body' tab is selected, showing a raw JSON request body. Below the request, the 'Test Results' tab shows a successful response with a JSON body.

**Request:**

```
1 {
2   "nombre": "Foco LED",
3   "categoria": "iluminación",
4   "precio": 150
5 }
```

**Response:**

```
1 {
2   "total": 7,
3   "productos": [
4     {
5       "id": 1,
6       "nombre": "Laptop",
7       "categoria": "electrónica",
8       "precio": 15000
9     },
10    {
11      "id": 2,
12      "nombre": "Mouse",
13      "categoria": "electrónica",
14      "precio": 250
15    },
16    {
17      "id": 3,
18      "nombre": "Teclado",
19      "categoria": "electrónica"
20    }
21  ]
22 }
```