

# 1. CREATE

POST http://127.0.0.1:8000/

http://127.0.0.1:8000/api/pokemons?nombre=Sandy&tipo=fuego&size=pequeño&peso=1.1

POST

http://127.0.0.1:8000/api/pokemons?nombre=Sandy&tipo=fuego&size=pequeño&peso=1.1

Params

Authorization

Headers (9)

Body

Pre-request Script

Tests

Settings

Query Params

	KEY	VALUE
<input checked="" type="checkbox"/>	nombre	Sandy
<input checked="" type="checkbox"/>	tipo	fuego
<input checked="" type="checkbox"/>	size	pequeño
<input checked="" type="checkbox"/>	peso	1.1
	Key	Value

Body

Cookies (2)

Headers (10)

Test Results

Pretty

Raw

Preview

Visualize

```
{"mensaje": "El pokemon ha sido creado de forma satisfactoria"}
```

## 2. SHOW

The screenshot shows a REST client interface with a GET request to `http://127.0.0.1:8000/api/pokemons`. The response is displayed in JSON format, showing a list of two Pokémon objects.

**Request:**

- Method: GET
- URL: `http://127.0.0.1:8000/api/pokemons`

**Response (JSON):**

```
1 {
2   "pokemons": [
3     {
4       "id": 1,
5       "nombre": "Sandra",
6       "tipo": "fuego",
7       "size": "pequeño",
8       "peso": 12.1,
9       "created_at": "2024-02-09T18:33:48.000000Z",
10      "updated_at": "2024-02-09T18:33:48.000000Z"
11    },
12    {
13      "id": 2,
14      "nombre": "Sandy",
15      "tipo": "fuego",
16      "size": "pequeño",
17      "peso": 1.1,
18      "created_at": "2024-02-09T18:34:02.000000Z",
19      "updated_at": "2024-02-09T18:34:02.000000Z"
20    }
21  ]
22 }
```

### 3.UPDATE

PUT http://127.0.0.1:8000/api

http://127.0.0.1:8000/api/pokemons/2?nombre=Sandy&tipo=volador&size=grande&peso=2.3

PUT

http://127.0.0.1:8000/api/pokemons/2?nombre=Sandy&tipo=volador&size=grande&peso=2.3

Params

Authorization

Headers (9)

Body

Pre-request Script

Tests

Settings

Query Params

	KEY	VALUE
<input checked="" type="checkbox"/>	nombre	Sandy
<input checked="" type="checkbox"/>	tipo	volador
<input checked="" type="checkbox"/>	size	grande
<input checked="" type="checkbox"/>	peso	2.3
	Key	Value

Body

Cookies (2)

Headers (10)

Test Results

Pretty

Raw

Preview

Visualize

JSON

1

2

3

{

"mensaje": "El pokemon ha sido editado de forma satisfactoria"

}

## 4.DELETE

The screenshot shows a REST client interface with a DELETE request to `http://127.0.0.1:8000/api/pokemons/2`. The response body is displayed in JSON format:

```
1 {  
2   "mensaje": "El pokemon ha eliminado de forma satisfactoria"  
3 }
```