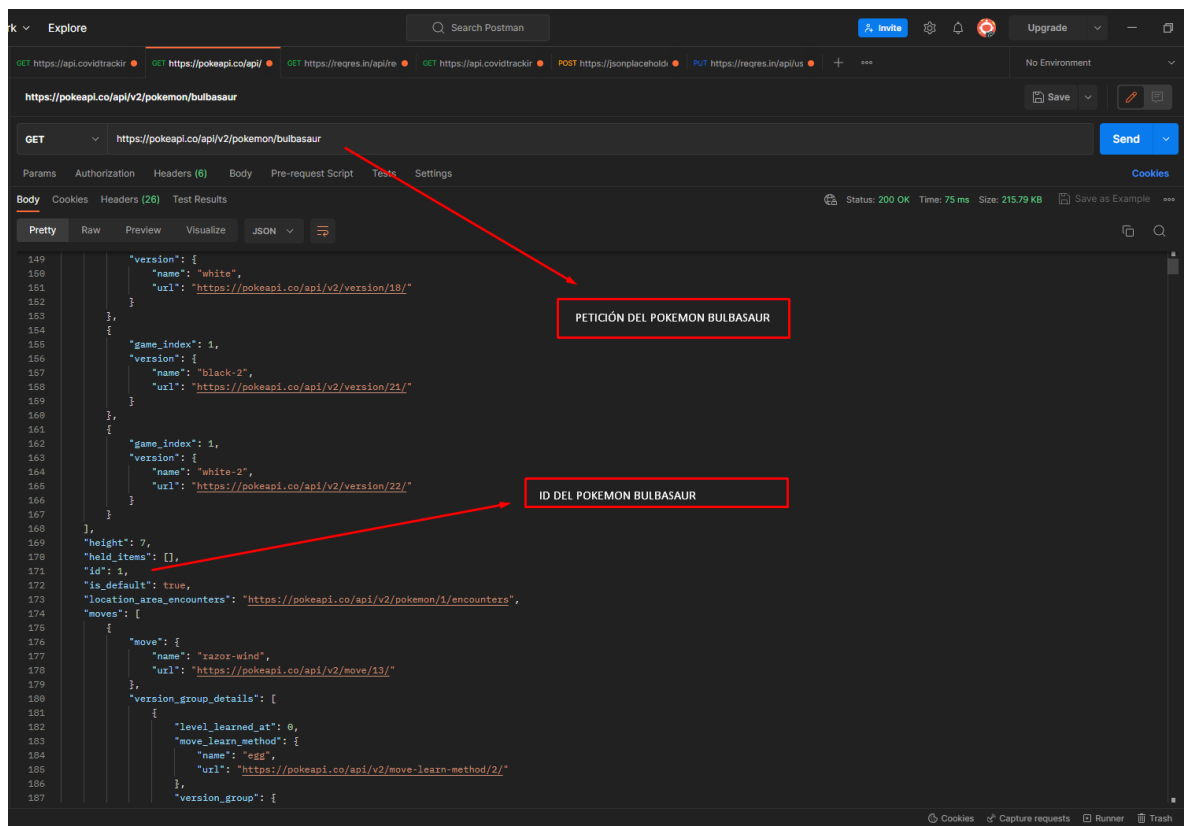
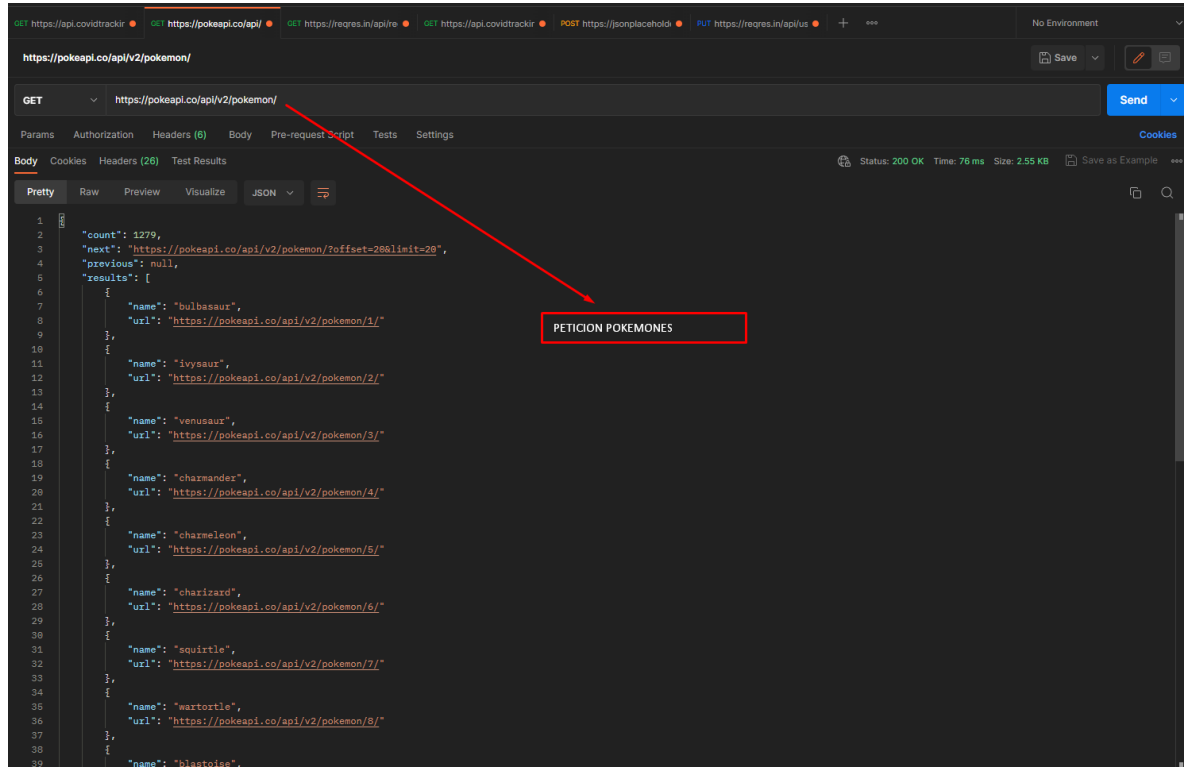


## PETICIONES GET:



GET https://pokeapi.co/api/v2/pokemon/ivysaur

GET https://pokeapi.co/api/v2/pokemon/ivysaur

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

Body Cookies Headers (26) Test Results

Status: 200 OK Time: 71 ms Size: 189.2 KB Save as Example

Pretty Raw Preview Visualize JSON

```
151 {
152   "url": "https://pokeapi.co/api/v2/version/10/"
153 },
154 {
155   "game_index": 2,
156   "version": {
157     "name": "black-2",
158     "url": "https://pokeapi.co/api/v2/version/21/"
159   },
160 },
161 {
162   "game_index": 2,
163   "version": {
164     "name": "white-2",
165     "url": "https://pokeapi.co/api/v2/version/22/"
166   },
167 },
168 ],
169 "height": 10,
170 "held_items": [],
171 "id": 2,
172 "is_default": true,
173 "location_area_encounters": "https://pokeapi.co/api/v2/pokemon/2/encounters",
174 "moves": [
175   {
176     "move": {
177       "name": "swords-dance",
178       "url": "https://pokeapi.co/api/v2/move/14/"
179     },
180     "version_group_details": [
181       {
182         "level_learned_at": 0,
183         "move_learn_method": {
184           "name": "machine",
185           "url": "https://pokeapi.co/api/v2/move-learn-method/4/"
186         }
187       }
188     ]
189   }
190 ]
191 }
```

PETICION POKEMON IVYSAUR

ID POKEMON IVYSAUR

GET https://pokeapi.co/api/v2/pokemon/charmander

GET https://pokeapi.co/api/v2/pokemon/charmander

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

Body Cookies Headers (26) Test Results

Status: 200 OK Time: 72 ms Size: 249.54 KB Save as Example

Pretty Raw Preview Visualize JSON

```
157 {
158   "name": "black-2",
159   "url": "https://pokeapi.co/api/v2/version/21/"
160 },
161 {
162   "game_index": 4,
163   "version": {
164     "name": "white-2",
165     "url": "https://pokeapi.co/api/v2/version/22/"
166   },
167 },
168 ],
169 "height": 6,
170 "held_items": [],
171 "id": 4,
172 "is_default": true,
173 "location_area_encounters": "https://pokeapi.co/api/v2/pokemon/4/encounters",
174 "moves": [
175   {
176     "move": {
177       "name": "mega-punch",
178       "url": "https://pokeapi.co/api/v2/move/5/"
179     },
180     "version_group_details": [
181       {
182         "level_learned_at": 0,
183         "move_learn_method": {
184           "name": "machine",
185           "url": "https://pokeapi.co/api/v2/move-learn-method/4/"
186         },
187         "version_group": {
188           "name": "red-blue",
189           "url": "https://pokeapi.co/api/v2/version-group/1/"
190         }
191       }
192     ]
193   }
194 ]
195 }
```

PETICION POKEMON CHARMANDER

ID POKEMON CHARMANDER

GET https://pokeapi.co/api/v2/pokemon/solsagan

GET https://pokeapi.co/api/v2/pokemon/solsagan

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

Body Cookies Headers (21) Test Results

Status: 404 Not Found Time: 1496 ms Size: 1010 B Save as Example

Pretty Raw Preview Visualize Text

1. Not Found

PETICION CON NOMBRE INEXISTENTE

ERROR 404 PORQUE EL POKEMON NO EXISTE

## PETICIÓN POST:

POST <https://jsonplaceholder.typicode.com/posts/> Send

Params Authorization Headers (9) **Body** Pre-request Script Tests Settings Cookies Beautify

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

```
1 {
2   "userId": 1,
3   "title": "Cien años de soledad",
4   "body": "autori1"
5 }
```

**PETICION POST**

**JSON**

Body Cookies Headers (24) Test Results Status: 201 Created Time: 117 ms Size: 113 KB Save as Example

Pretty Raw Preview Visualize **JSON**

```
1 {
2   "userId": 1,
3   "title": "Cien años de soledad",
4   "body": "autori1",
5   "id": 101
6 }
```

**POST CREADO**

POST <https://jsonplaceholder.typicode.com/posts/> Send

Params Authorization Headers (9) **Body** Pre-request Script Tests Settings Cookies Beautify

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

```
1 {
2   "userId": 2,
3   "title": "La importancia de morir a tiempo",
4   "body": "autori2"
5 }
```

**PETICION POST**

**JSON**

Body Cookies Headers (24) Test Results Status: 201 Created Time: 128 ms Size: 114 KB Save as Example

Pretty Raw Preview Visualize **JSON**

```
1 {
2   "userId": 2,
3   "title": "La importancia de morir a tiempo",
4   "body": "autori2",
5   "id": 101
6 }
```

**POST CREADO**

POST <https://jsonplaceholder.typicode.com/posts/> Send

Params Authorization Headers (9) **Body** Pre-request Script Tests Settings Cookies Beautify

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

```
1 {
2   "userId": 3,
3   "title": "Lucifer",
4   "body": "autori3"
5 }
```

**PETICION POST**

**JSON**

Body Cookies Headers (24) Test Results Status: 201 Created Time: 127 ms Size: 113 KB Save as Example

Pretty Raw Preview Visualize **JSON**

```
1 {
2   "userId": 3,
3   "title": "Lucifer",
4   "body": "autori3",
5   "id": 101
6 }
```

**POST CREADO**

## PETICION DELETE:

DELETE `https://reqres.in/api/users/1` Send

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

Query Params

Key	Value	Description
Key	Value	Description

Body Cookies Headers (12) Test Results

Pretty Raw Preview Visualize Text `1`

Status: 204 No Content Time: 477 ms Size: 612 B Save as Example

DELETE `https://reqres.in/api/users/2` Send

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

Query Params

Key	Value	Description
Key	Value	Description

Body Cookies Headers (12) Test Results

Pretty Raw Preview Visualize Text `1`

Status: 204 No Content Time: 328 ms Size: 612 B Save as Example

DELETE `https://reqres.in/api/users/3` Send

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

Query Params

Key	Value	Description
Key	Value	Description

Body Cookies Headers (12) Test Results

Pretty Raw Preview Visualize Text `1`

Status: 204 No Content Time: 313 ms Size: 616 B Save as Example

## PETICIÓN PUT:

PUT <https://reqres.in/api/users/> Send

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

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

```
1 {"id": 1,
2  "firstName": "James",
3  "last_name": "Munoz",
4  "avatar": "QA"
5 }
```

PETICION POST

JSON

CODIGO RESPUESTA

Status: 200 OK Time: 327 ms Size: 815 B Save as Example

Body Cookies Headers (15) Test Results

Pretty Raw Preview Visualize JSON

```
1 {"id": 1,
2  "firstName": "James",
3  "last_name": "Munoz",
4  "avatar": "QA",
5  "updatedAt": "2023-03-29T19:45:14.552Z"
6 }
7
```

POST ACTUALIZADO

PUT <https://reqres.in/api/users/> Send

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

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

```
1 {"id": 2,
2  "firstName": "Juan",
3  "last_name": "Pineda",
4  "avatar": "Coach"
5 }
```

PETICION PUT

JSON

CODIGO RESPUESTA

Status: 200 OK Time: 317 ms Size: 814 B Save as Example

Body Cookies Headers (15) Test Results

Pretty Raw Preview Visualize JSON

```
1 {"id": 2,
2  "firstName": "Juan",
3  "last_name": "Pineda",
4  "avatar": "Coach",
5  "updatedAt": "2023-03-29T19:47:53.344Z"
6 }
7
```

POST ACTUALIZADO

PUT <https://reqres.in/api/users/> Send

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

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

```
1 {"id": 3,
2  "firstName": "Ivan",
3  "last_name": "Marquez",
4  "avatar": "Automatizador"
5 }
```

PETICION PUT

JSON

CODIGO RESPUESTA

Status: 200 OK Time: 301 ms Size: 829 B Save as Example

Body Cookies Headers (15) Test Results

Pretty Raw Preview Visualize JSON

```
1 {"id": 3,
2  "firstName": "Ivan",
3  "last_name": "Marquez",
4  "avatar": "Automatizador",
5  "updatedAt": "2023-03-29T19:58:17.732Z"
6 }
7
```

POST ACTUALIZADO

## CONCLUSIONES

Sabemos Serenity BDD y Karate son herramientas de automatización de pruebas que pueden utilizarse para probar servicios basados en API REST. Ambas herramientas proporcionan una serie de ventajas para el testing de API REST.

Una de las ventajas de Serenity BDD es su capacidad para generar informes detallados y visuales sobre el estado de las pruebas, lo que facilita la identificación de errores y la resolución de problemas. Además, Serenity BDD también es altamente personalizable y permite la integración con otras herramientas de automatización.

Por otro lado, Karate se destaca por su facilidad de uso. Karate también proporciona una sintaxis sencilla y clara para la escritura de pruebas, lo que lo hace ideal para equipos que no tienen mucha experiencia en programación.

Serenity BDD se enfoca más en la integración con herramientas de automatización de pruebas existentes y en la generación de informes detallados, mientras que Karate se centra más en la simplicidad de uso.