

## EJERCICIO 3:

Recomendamos previamente entender los conceptos de la sintaxis “json” antes de arrancar con los ejercicios.

Descargar el POSTMAN (aplicación para realizar request como cliente), adjuntando un screen de resolución para cada ítem:

1. Realizar un request GET a la URL: <https://reclutamiento-14cf7.firebaseio.com/personas.json>

The screenshot shows the Postman interface. At the top, a request is configured with the method 'GET' and the URL 'https://reclutamiento-14cf7.firebaseio.com/personas.json'. The 'Send' button is visible. Below the request configuration, the 'Params' tab is selected, showing a table with columns 'KEY', 'VALUE', and 'DESCRIPTION'. The table is currently empty. The 'Body' tab is also visible, showing the JSON response of the request. The response is a JSON array with two objects, each representing a person with fields 'apellido', 'dni', and 'nombre'.

KEY	VALUE	DESCRIPTION
Key	Value	Description

```
1 {
2   "-MN8WwtyElinuFe2rDXy": {
3     "apellido": "Hellberg",
4     "dni": 42296528,
5     "nombre": "Axel"
6   },
7   "-MN8Z8P38bq5MpysA13-": {
8     "apellido": "Maximiliano",
9     "dni": 39852478,
10    "nombre": "Adrian"
11  },
12 }
```

2. Realizar un request POST a la URL anterior, y con body:

```
{
  "nombre": "Tu nombre",
  "apellido": "Tu apellido",
  "dni": 11223322
}
```

Tip: (Marcar la opción “raw” como body)

Untitled Request

BUILD

✎

💬

POST

https://reclutamiento-14cf7.firebaseio.com/personas.json

Send

Save

- Params
- Authorization
- Headers (8)
- Body
- Pre-request Script
- Tests
- Settings
- Cookies
- Code
- none
- form-data
- x-www-form-urlencoded
- raw
- binary
- GraphQL
- Text

```
1 {
2   "nombre": "Gonzalo",
3   "apellido": "Castro",
4   "dni": 41639230
5 }
```

Body

Cookies

Headers (8)

Test Results

🌐

Status: 200 OK

Time: 424 ms

Size: 323 B

Save Response

Pretty

Raw

Preview

Visualize

JSON

🔍

```
1 {
2   "name": "-MRSf4P6orS4JmFck20A"
3 }
```

3. Realizar nuevamente un request GET a la URL: <https://reclutamiento-14cf7.firebaseio.com/personas.json> ¿Qué diferencias se observan?

The screenshot shows a REST client interface with a request bar at the top. The method is GET and the URL is `https://reclutamiento-14cf7.firebaseio.com/personas.json`. Below the request bar are tabs for Params, Authorization, Headers (6), Body, Pre-request Script, Tests, and Settings. The Params tab is active, showing a table with columns KEY, VALUE, and DESCRIPTION. The table contains one entry with KEY 'Key', VALUE 'Value', and DESCRIPTION 'Description'. Below the Params tab is a section for Query Params. The response section at the bottom shows the response body in JSON format, with a status of 200 OK, time of 412 ms, and size of 16.83 KB. The JSON response is a list of two objects, each representing a person with fields like 'apellido', 'dni', and 'nombre'.

Launchpad GET https://reclutamiento-14cf7.firebaseio.com/personas.json No Environment

Untitled Request BUILD Send Save

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

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
Key	Value	Description		

Body Cookies Headers (8) Test Results Status: 200 OK Time: 412 ms Size: 16.83 KB Save Response

Pretty Raw Preview Visualize JSON ...

```
1081 {
1082   "-MR1TUumTjf1-VdJ3gCX": {
1083     "apellido": "aa",
1084     "dni": 11,
1085     "nombre": "aa"
1086   },
1087   "-MRSf4P6orS4JmFck20A": {
1088     "apellido": "Castro",
1089     "dni": 41639230,
1090     "nombre": "Gonzalo"
1091   }
}
```

Se puede observar que por haber hecho el POST del punto 2, ahora al hacer el GET, aparecen los datos enviados anteriormente al final del archivo json recibido.