

EXERCISES QUE TRABAJAREMOS EN LA CUE

- EXERCISE 1: UTILIZANDO POSTMAN CON UNA API REST PÚBLICA.

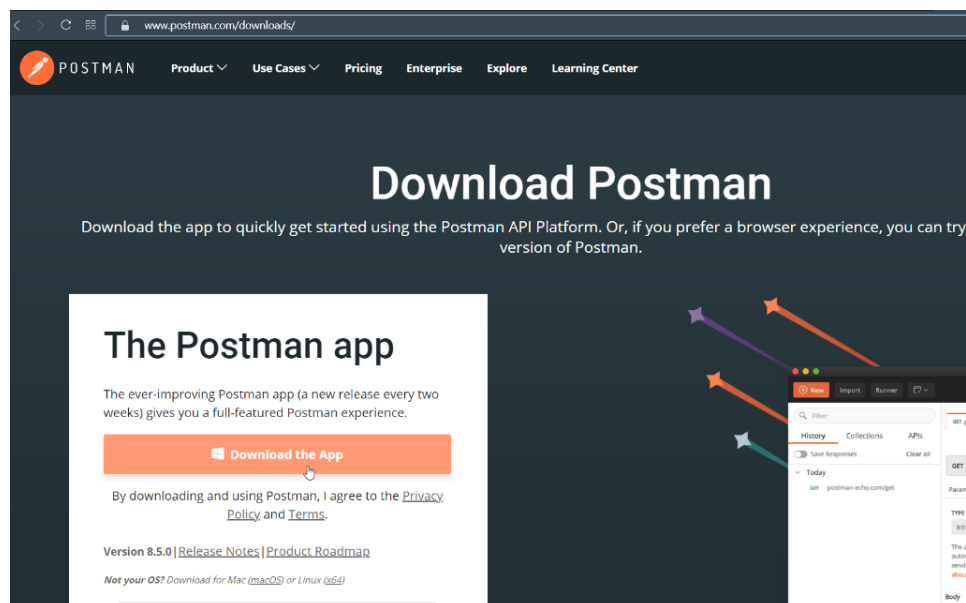
EXERCISE 1: UTILIZANDO POSTMAN CON UNA API REST PUBLICA

En este ejercicio, aprenderemos a consultar **API REST**, utilizando **Postman** y la **API** de Star Wars, que es de uso público y gratuito.

Postman es una herramienta para testear **APIs**. Permite realizar peticiones de manera sencilla, y se puede descargar desde su web oficial: <https://www.postman.com/downloads/>

NOTA: Recordemos que una **API** es un conjunto de definiciones y protocolos, que se utiliza para desarrollar e integrar el software de las aplicaciones. Suele considerarse como el contrato entre el proveedor de información y el usuario, donde se establece el contenido que se necesita del consumidor (la llamada), y el que requiere el productor (la respuesta).

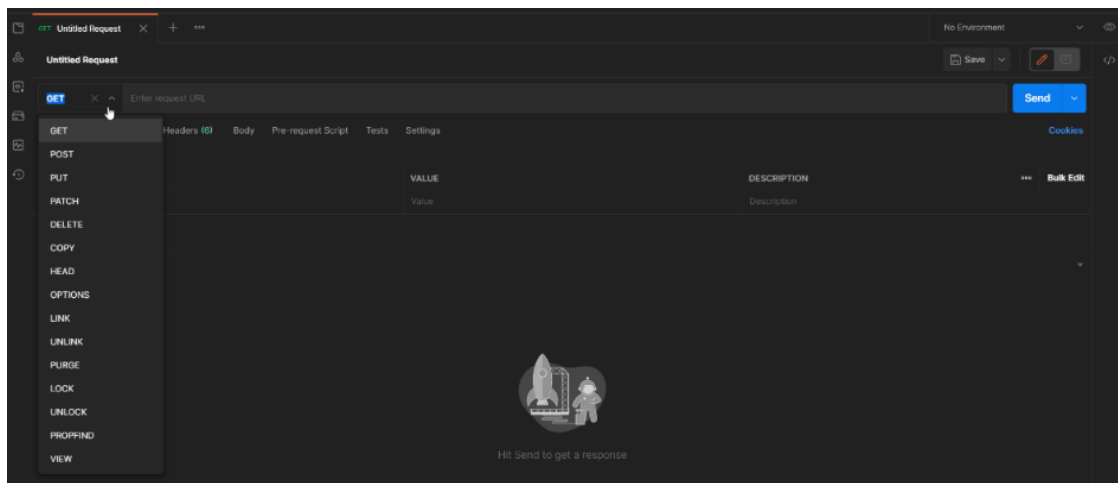
Como se puede observar en el sitio web de **Postman**, éste reconoce el sistema operativo del computador desde el que se está navegando, y descarga automáticamente la versión correspondiente.



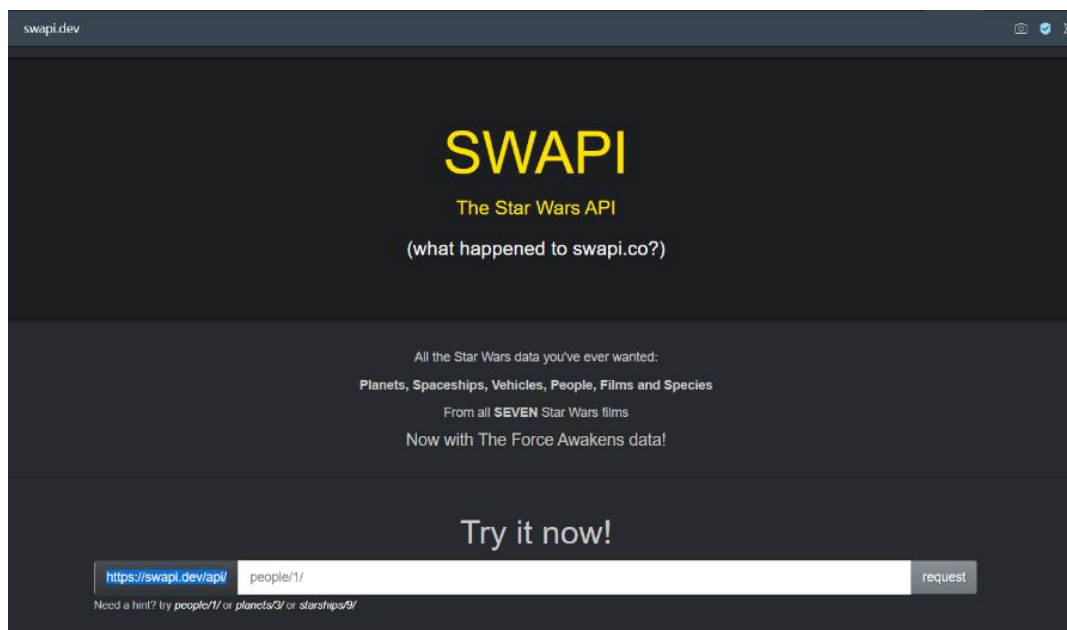


Una vez descargada la aplicación, te pedirá que te registres, pero también da opción de saltar este paso, e ingresar directamente.

Una vez adentro, solo debes agregar una nueva petición, e ingresar los datos necesarios. Te permite realizar peticiones de tipo: **GET, POST, PUT, PATCH, DELETE**, entre otras.

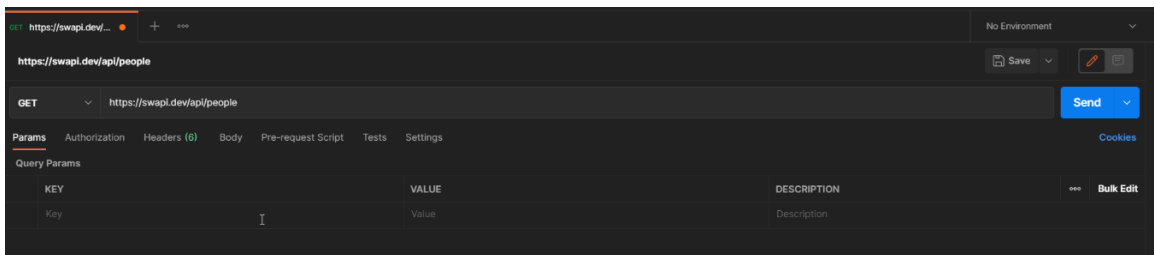


Para comenzar en swapi.dev, copiaremos la **URL** que nos aparece, que es la primera.

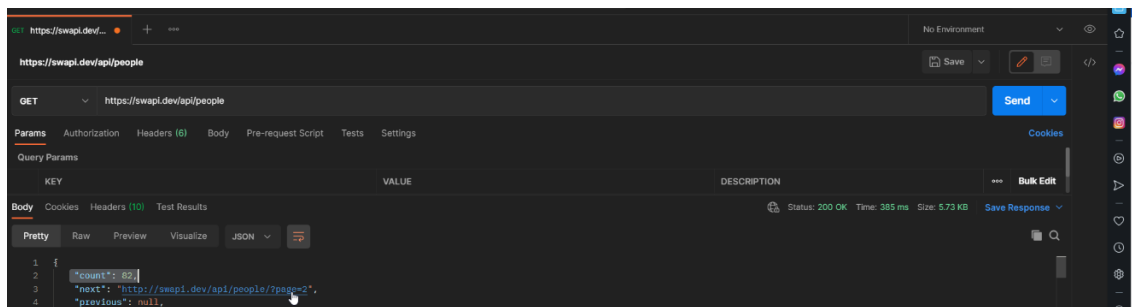




Nos dirigiremos a **Postman**, pegamos la **URL** (endpoint), y le agregaremos la palabra people; el tipo de petición será **GET**, y presionamos send.



Postman nos entregó como respuesta un **JSON**, con la información solicitada, es decir, 82 personajes (línea 2). Además, notaremos que esta **API** tiene un sistema de paginación (línea 3), esto indica que debemos ir de una página a la siguiente, para poder ver a todos los personajes.



NOTA: Recordemos que **JSON** es un formato de texto, para representar datos estructurados en la sintaxis de objetos de **JavaScript**, y que permite los mismos tipos de datos que un objeto: cadenas, numero booleanos, arreglos y otros objetos.

Dentro del arreglo **results**, podemos ver un objeto con el nombre del personaje y sus atributos.

```
2  "count": 82,  
3  "next": "http://swapi.dev/api/people/?page=2",  
4  "previous": null,  
5  "results": [  
6    {  
7      "name": "Luke Skywalker",  
8      "height": "172",  
9      "mass": "77",  
10     "hair_color": "blond",  
11     "skin_color": "fair",  
12     "eye_color": "blue",  
13     "birth_year": "19BBY",  
14     "gender": "male",  
15     "homeworld": "http://swapi.dev/api/planets/1/",  
16     "films": [  
17       "http://swapi.dev/api/films/1/",  
18       "http://swapi.dev/api/films/2/",  
19       "http://swapi.dev/api/films/3/",  
20       "http://swapi.dev/api/films/6/"  
21     ]  
    }
```

Otro resultado que podemos obtener de esta **API**, son los planetas que existen en el universo de Star Wars. Para eso, cambiaremos la **URL** (endpoint), reemplazamos la palabra **people** por **planets** (<https://swapi.dev/api/planets>), presionamos send, y esperamos los resultados.

Aquí podemos ver que existen 60 planetas, almacenados con sus características: nombre, periodo de rotación, entre otros.

```
1  "results": [  
2    {  
3      "name": "Tatooine",  
4      "rotation_period": "23",  
5      "orbital_period": "304",  
6      "diameter": "10465",  
7      "climate": "arid",  
8      "gravity": "1 standard",  
9      "terrain": "desert",  
10     "surface_water": "1",  
11     "population": "200000",  
12     "residents": [  
13       "http://swapi.dev/api/people/1/",  
14       "http://swapi.dev/api/people/2/",  
15       "http://swapi.dev/api/people/4/",  
16       "http://swapi.dev/api/people/6/",  
17       "http://swapi.dev/api/people/7/",  
18       "http://swapi.dev/api/people/8/",  
19       "http://swapi.dev/api/people/9/",  
20       "http://swapi.dev/api/people/11/",  
21       "http://swapi.dev/api/people/43/",  
22       "http://swapi.dev/api/people/62/"  
23     ],  
24     "films": [  
25       "http://swapi.dev/api/films/1/",  
26       "http://swapi.dev/api/films/3/",  
27       "http://swapi.dev/api/films/4/",  
28       "http://swapi.dev/api/films/5/",  
29       "http://swapi.dev/api/films/6/"  
30     ],  
31     "created": "2014-12-09T13:50:49.641000Z",  
32     "edited": "2014-12-20T20:58:18.411000Z",  
33     "url": "http://swapi.dev/api/planets/1/"  
34   },  
  ]  
}
```

Ahora, para ser más específicos, vamos a obtener un personaje en particular. Lo haremos utilizando su **ID**, y el **endpoint** correspondiente. Volveremos a cambiar la **URL**, borraremos planets, agregamos people y el **ID**, que en este caso será 4 (<https://swapi.dev/api/people/4>), presionamos send; y como resultado, obtenemos a Darth Vader con las siguientes características: una altura de 202 cm, color de ojos amarillo, su color de piel, entre otros.

```
1 {
2   "name": "Darth Vader",
3   "height": "202",
4   "mass": "136",
5   "hair_color": "none",
6   "skin_color": "white",
7   "eye_color": "yellow",
8   "birth_year": "41.9BBY",
9   "gender": "male",
10  "homeworld": "http://swapi.dev/api/planets/1/",
11  "films": [
12    "http://swapi.dev/api/films/1/",
13    "http://swapi.dev/api/films/2/",
14    "http://swapi.dev/api/films/3/",
15    "http://swapi.dev/api/films/6/"
16  ],
17  "species": [],
18  "vehicles": [],
19  "starships": [
20    "http://swapi.dev/api/starships/13/"
21  ],
22  "created": "2014-12-10T15:18:20.704000Z",
23  "edited": "2014-12-20T21:17:50.313000Z",
24  "url": "http://swapi.dev/api/people/4/"
25 }
```