Peticiones

DEV.F::
DESARROLLAMOS(PERSONAS);

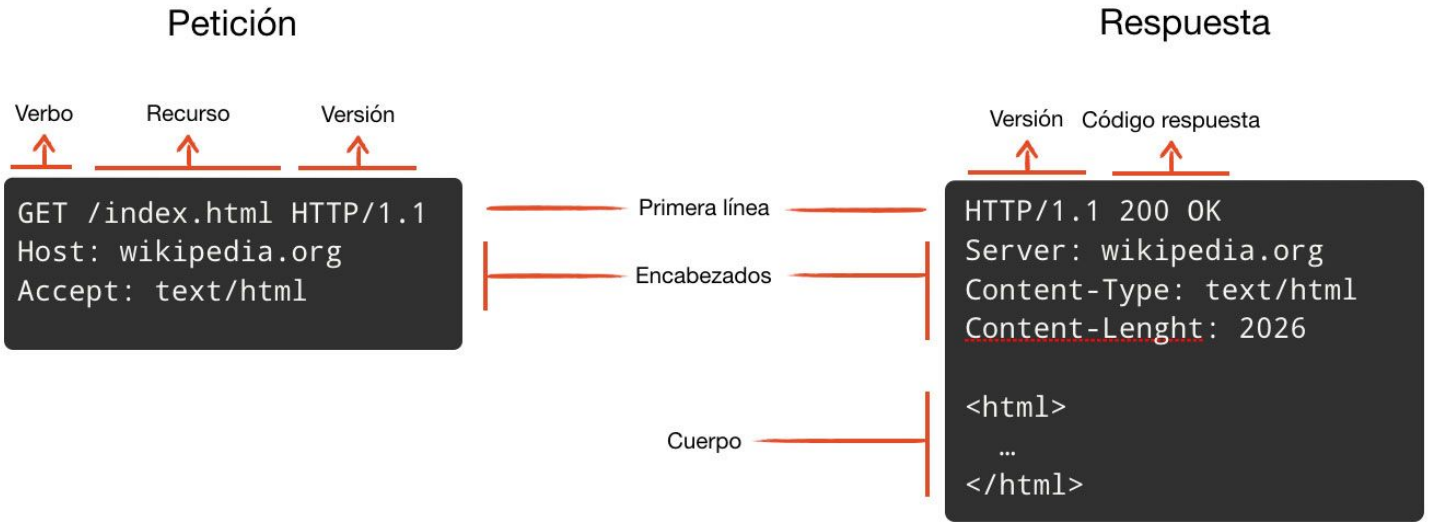
Mensajes HTTP

Los mensajes HTTP, son los medios por los cuales se intercambian datos entre servidores y clientes.

Hay dos tipos de mensajes: **peticiones**, enviadas por el cliente al servidor, para pedir el inicio de una acción; y **respuestas**, que son la respuesta del servidor.

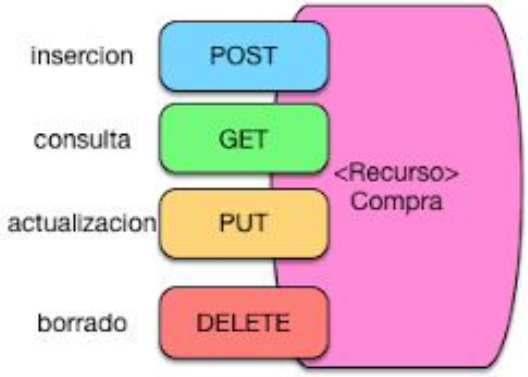
Mensajes HTTP

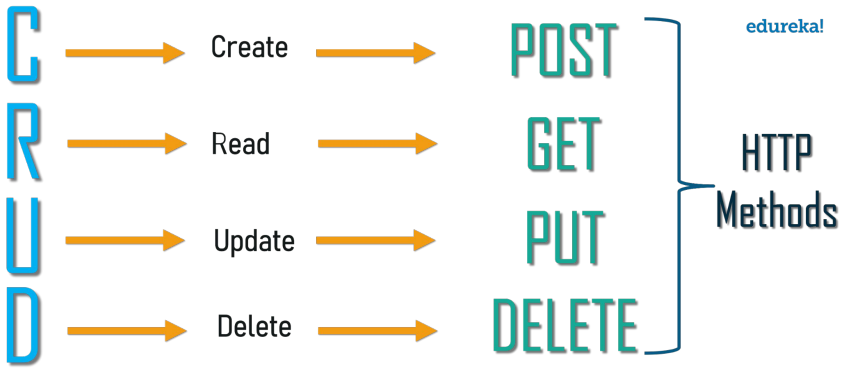
La línea de inicio y las cabeceras HTTP, del mensaje, son conocidas como la cabeza de la peticiones, mientras que su contenido en datos se conoce como el cuerpo del mensaje.



Verbos HTTP

DEV.F::
DESARROLLAMOS(PERSONAS);





VERBOS HTTP

La primera línea de un mensaje de petición empieza con un verbo (también se le conoce como método). **Los verbos definen la acción que se quiere realizar sobre el recurso**.

Los verbos más comunes son:

- **GET:** Solicitar un recurso.
- **POST:** Publicar un recurso.
- **PUT:** Reemplazar un recurso.
- **DELETE:** Eliminar un recurso.
- **PATCH:** Actualizar un recurso

Nota: Cuando ingresas a una página desde un navegador, por debajo el navegador envía un mensaje GET, lo mismo cuando oprimes un vínculo a otra página.

DEV.F::

Códigos de Respuesta

DEV.F::
DESARROLLAMOS(PERSONAS);

Response Codes

La primera línea de un mensaje de respuesta tiene un código de 3 dígitos que le indica al cliente cómo interpretar la respuesta.

Los códigos de respuesta se dividen en cinco categorías dependiendo del dígito con el que inician:

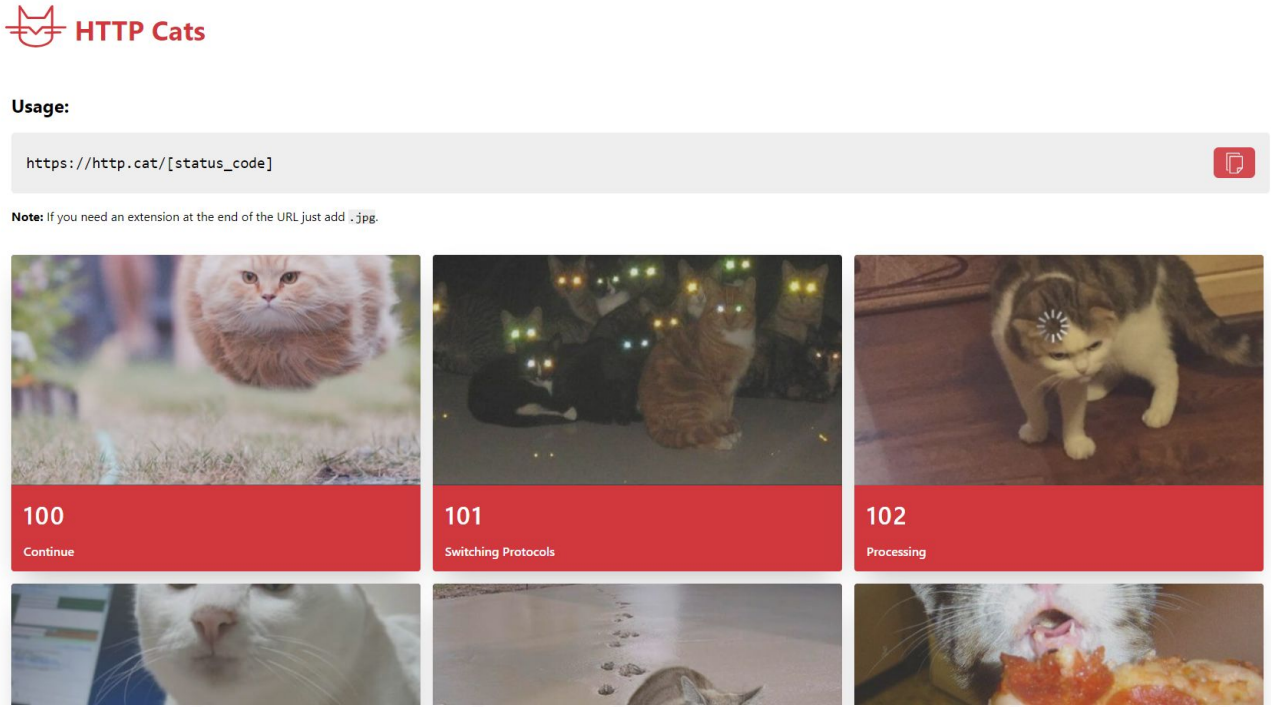
- 1XX: Información
- 2XX: Éxito
- 3XX: Redirección
- 4XX: Error en el cliente
- 5XX: Error en el servidor

¿Recuerdas el famoso error 404?

DEV.F::

HTTP Cats

https://http.cat/



DEV.F::

PRO TIP

Los códigos de respuesta son una convención no una regla.

Cada desarrollador puede asignar el código de respuesta que desee a cada petición, pero se recomienda seguir las conveciones (lo más posible)

API Rest

DEV.F:
DESARROLLAMOS(PERSONAS);

¿Qué es Rest?

REST es una interfaz para conectar varios sistemas basados en el protocolo HTTP y nos sirve para obtener, generar datos y operaciones, devolviendo esos datos en formatos muy específicos, como XML y JSON.

- Protocolo cliente/servidor “SIN ESTADO”
- Utiliza verbos http
  - GET (leer)
  - POST (crear)
  - PUT (editar)
  - DELETE (borrar)
- Devuelve
  - JSON
  - XML

HTTP STATUS

200 – OK
201 – CREATED
403 – FORBIDDEN
404 – NOT FOUND

Ejemplos:

GET /rest/api/pedido/{id}
DELETE /rest/api/pedido/{id}
POST /rest/api/pedido
PUT /rest/api/pedido/{id}/producto/{idProducto}

Devuelve
JSON/XML

```
{
    “nombre”: “Pepito”,
    “coches” : [“Ford”, “Fiat”, “Audi”]
}
```

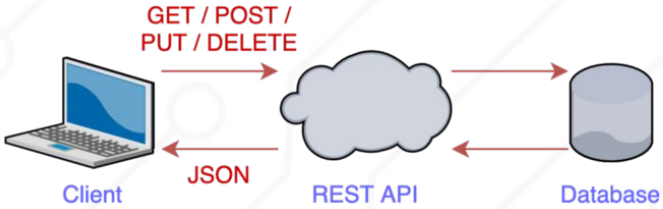
API + Rest = API Rest

API

API utiliza peticiones HTTP responsables de las operaciones básicas necesarias para la manipulación de datos (GET, POST, etc).

REST

Rest, es un conjunto de restricciones que se utilizan para que las solicitudes HTTP cumplan con las directrices definidas en la arquitectura: cliente servidor, sin estado, con cache, etc.



API REST

API Rest es el conjunto de buenas prácticas utilizadas en las peticiones HTTP realizadas por una API en una aplicación web.

Es decir, cuando se habla de API Rest, significa utilizar una API para acceder a aplicaciones back-end, de manera que esa comunicación se realice con los estándares definidos por el estilo de arquitectura Rest.

Nombrando Endpoints en API Rest

/users // lista todos los usuarios

/users/123 // lista a un usuario en específico

/users/123/orders // lista los pedidos de un usuario específico

/users/123/orders/0001 // lista una orden especifica de un usuario específico

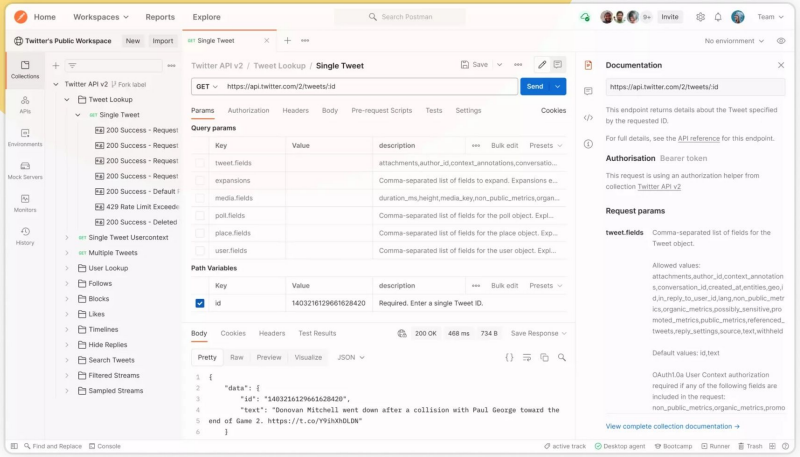
Endpoints y Verbos Http

Un mismo Endpoint, hara diferentes acciones dependiendo el verbo http usado para accederlo

Recuros / Estado	POST	GET	PUT	DELETE
/casas	Crea una casa	Devuelve una lista de todos las casas	Modifica casas	Borra todas las casas
/casas/123	error	Devuelve los detalles de la casa **123**	Modifica la casa	Borra la casa

Accediendo a API Rest

DEV.F:
DESARROLLAMOS(PERSONAS);



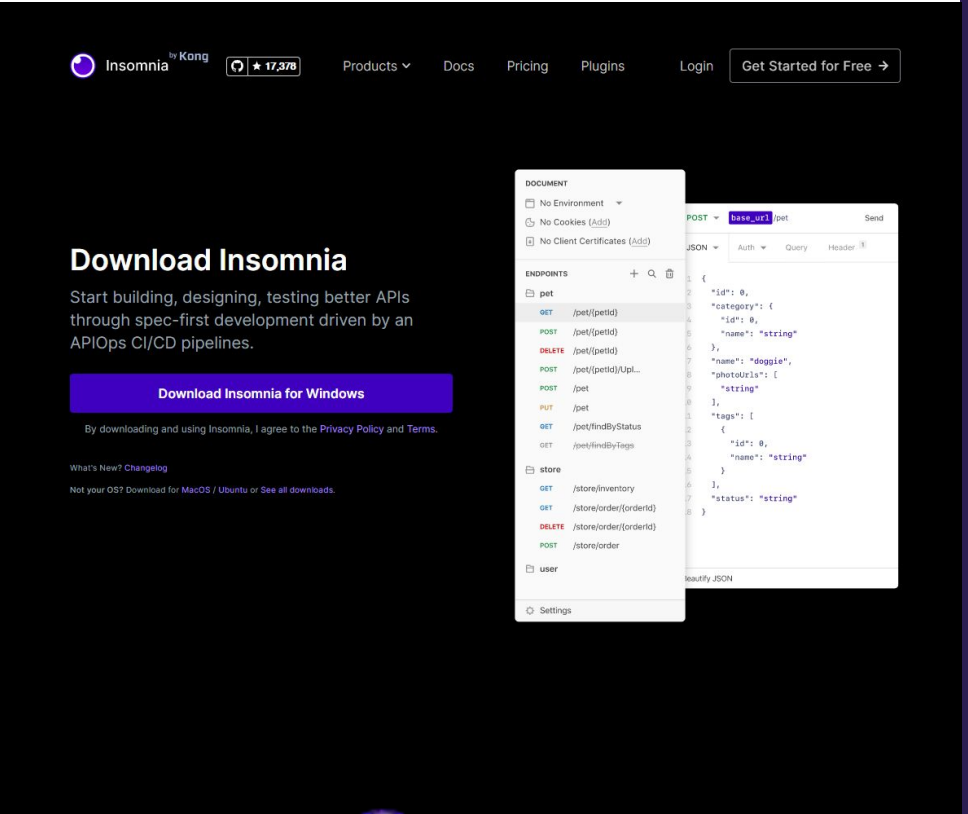
Postman

Postman nace como una herramienta que principalmente nos permite crear peticiones sobre APIs de una forma muy sencilla y poder, de esta manera, probar las APIs.

El interés fundamental de Postman es que lo utilicemos como una herramienta para hacer peticiones a APIs y generar colecciones de peticiones que nos permitan probarlas de una manera rápida y sencilla.

POSTMAN

https://www.postman.com/



Insomnia

https://insomnia.rest/

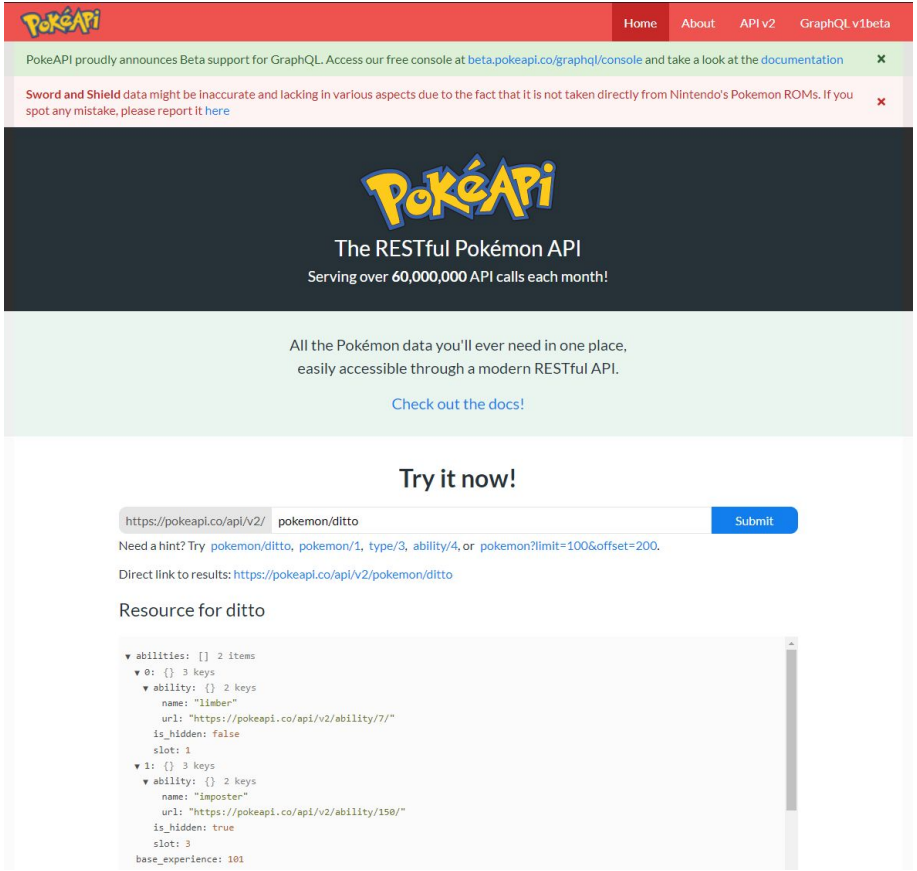
Insomnia

Es similar a Postman, con una interfaz un poco más sencilla.

DEV.F::

APIs para probar

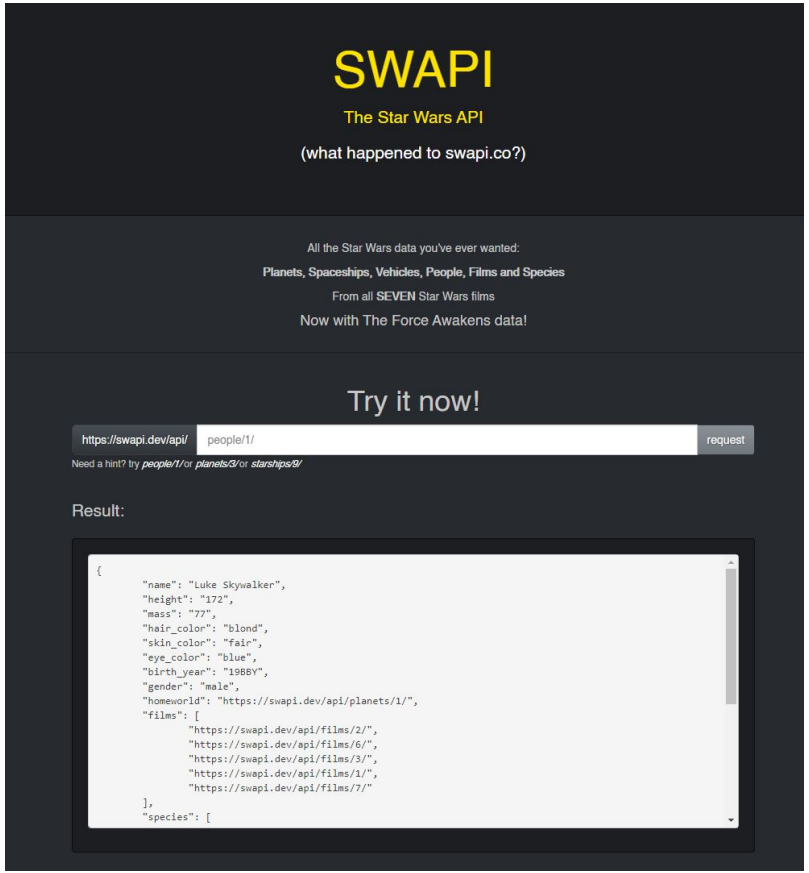
DEV.F::
DESARROLLAMOS(PERSONAS);



https://pokeapi.co/

PokéAPI

DEV.F::



https://swapi.dev/

SWAPI

(Star Wars)

DEV.F::