

API

O que vamos ver hoje?

- API
- REST
- JSON
- Thunder Client
- JSON Placeholder

O que é uma API?

O que é uma API?

- A sigla API deriva da expressão inglesa *Application Programming Interface*, que traduzida para o português significa: **interface de programação de aplicação**.
- Ela nada mais é do que um conjunto de normas entre ambos os sistemas para facilitar a comunicação entre o usuário e o software.
- Por meio delas é possível obter e enviar dados de uma maneira ágil, simples e segura.

O que é uma API?

- A API funciona como uma espécie de “mensageira”, que recebe requisições de um cliente e transmite até um servidor, onde será processada e devolvida.

Analogia do restaurante

- Quando vamos a um restaurante, escolhemos um prato no menu e fazemos o pedido ao garçom.
- Em seguida, o garçom encaminha esse pedido à cozinha, que o prepara. No fim, o garçom traz o prato pronto até a gente. Por fim, não sabemos como o prato foi preparado, apenas recebemos o que solicitamos.
- Ao receber o pedido, a API o encaminha ao sistema responsável e, em seguida, devolve o que solicitou.

API REST

API's REST

- Essas são as APIs mais populares e flexíveis encontradas na Web atualmente.
- O cliente envia solicitações ao servidor como dados.
- O servidor usa essa entrada do cliente para iniciar funções internas e retorna os dados de saída ao cliente.

API's REST

- REST significa Transferência Representacional de Estado.
- REST define um conjunto de funções como GET, POST, PUT, DELETE que os clientes podem usar para acessar os dados do servidor.
- Clientes e servidores trocam dados usando HTTP.

API's REST

- GET: Buscar dados.
- POST: Adicionar dados.
- PUT: Modificar dados.
- DELETE: Apagar dados.

JSON

JSON

- É o formato mais utilizado para o retorno de dados de uma API REST.
- JSON (*JavaScript Object Notation*) é uma forma de notação de objetos JavaScript, de modo que eles possam ser representados de uma forma comum a diversas linguagens.
- Se constitui basicamente de uma string (texto) com todos os dados necessários.

JSON



```
1 {  
2   "user": {  
3     "id": "1",  
4     "animal": "Bob",  
5     "idade": 2  
6   }  
7 }
```

Por que JSON é tão utilizado?

- Simplicidade
- Legibilidade
- Portabilidade

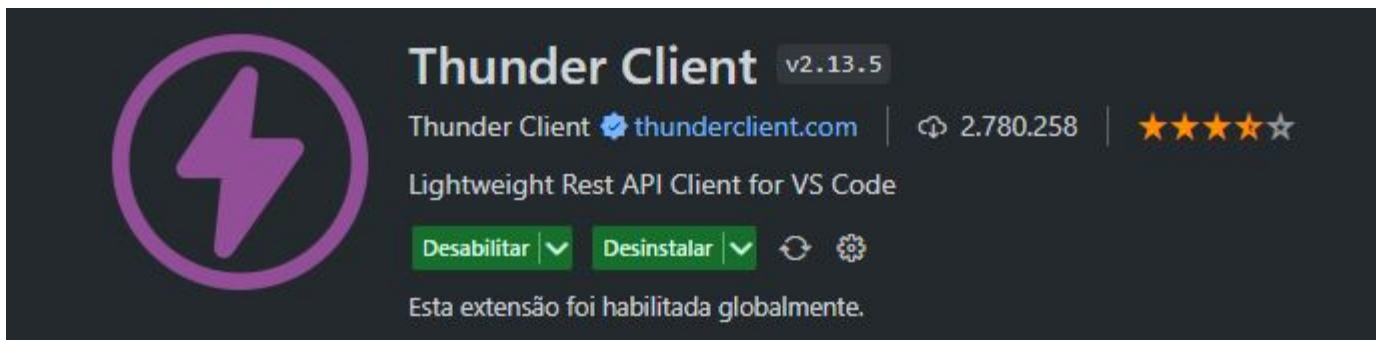
Regras JSON

- Não pode ter funções;
- Não pode ter comentários;
- Todo texto sempre tem aspas duplas;
- As propriedades sempre tem aspas duplas.

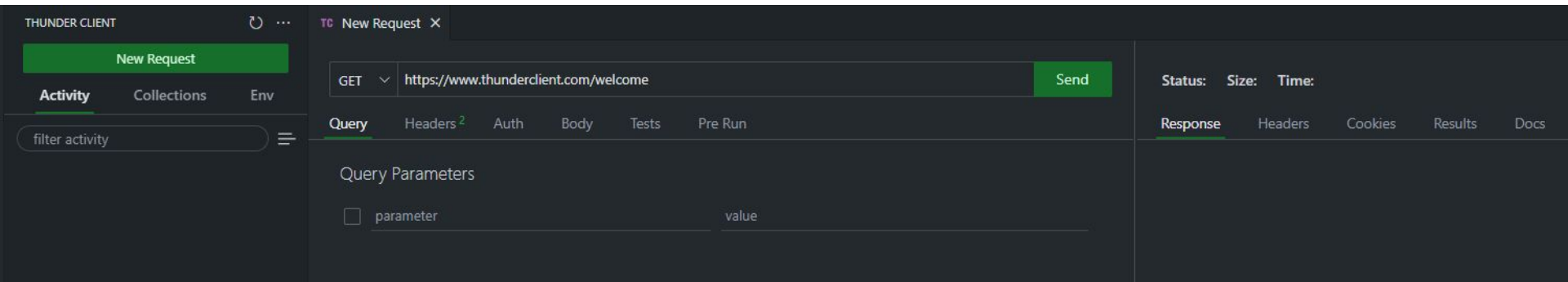
Testando o consumo de API's

Thunder Client

- Uma maneira de consumir uma api antes de ter a aplicação construída é utilizando a extensão do VSCode Thunder Client.



Thunder Client



Thunder Client

- Com ela conseguimos testar diferentes tipos de requisições e ver quais são os dados retornados.

Thunder Client

THUNDER CLIENT

New Request

ActivityCollectionsEnv

filter activity

GET api.github.com/users/rafaelkasper/rep...
just now

TO New Request X

GET https://api.github.com/users/rafaelkasper/reposSend

QueryHeadersAuthBodyTestsPre Run

Query Parameters

☐ parametervalue

Status: 200 OKSize: 127.9 KBTime: 422 ms

ResponseHeadersCookiesResultsDocs

```
1 [{
2   {
3     "id": 702139797,
4     "node_id": "R_kgDOKdnNlQ",
5     "name": "3000TalentosFrontend",
6     "full_name": "rafaelkasper/3000TalentosFrontend",
7     "private": false,
8     "owner": {
9       "login": "rafaelkasper",
10      "id": 42684330,
11      "node_id": "MDQ6VXNlcjQyNjg0MzMw",
12      "avatar_url": "https://avatars.githubusercontent.com/u/42684330?v=4",
13      "gravatar_id": "",
14      "url": "https://api.github.com/users/rafaelkasper",
15      "html_url": "https://github.com/rafaelkasper",
16      "followers_url": "https://api.github.com/users/rafaelkasper/followers",
17      "following_url": "https://api.github.com/users/rafaelkasper/following{/other_user}",
18      "gists_url": "https://api.github.com/users/rafaelkasper/gists{/gist_id}",
19      "starred_url": "https://api.github.com/users/rafaelkasper/starred{/owner}/{repo}",
20      "subscriptions_url": "https://api.github.com/users/rafaelkasper/subscriptions",
21      "organizations_url": "https://api.github.com/users/rafaelkasper/orgs",
22      "repos_url": "https://api.github.com/users/rafaelkasper/repos",
23      "events_url": "https://api.github.com/users/rafaelkasper/events{/privacy}",
24      "received_events_url": "https://api.github.com/users/rafaelkasper/received_events",
25      "type": "User",
26      "site_admin": false
27    },
28    "html_url": "https://github.com/rafaelkasper/3000TalentosFrontend",

```

JSON Placeholder

- <https://jsonplaceholder.typicode.com/>
- Fake API gratuita para estudos.

<u>/posts</u>	100 posts	GET	<u>/posts</u>
<u>/comments</u>	500 comments	GET	<u>/posts/1</u>
<u>/albums</u>	100 albums	GET	<u>/posts/1/comments</u>
<u>/photos</u>	5000 photos	GET	<u>/comments?postId=1</u>
<u>/todos</u>	200 todos	POST	/posts
<u>/users</u>	10 users	PUT	/posts/1
		PATCH	/posts/1
		DELETE	/posts/1

JSON Placeholder

TC New Request X

GET

https://jsonplaceholder.typicode.com/posts

Send

Query

Headers²

Auth

Body

Tests

Pre Run

Query Parameters

☐

parameter

value

Status: 200 OK Size: 26.88 KB Time: 412 ms

Response

Headers²⁴

Cookies

Results

Docs

```
1  [  
2    {  
3      "userId": 1,  
4      "id": 1,  
5      "title": "sunt aut facere repellat provident occaecati excepturi optio  
6      reprehenderit",  
7      "body": "quia et suscipit\nsuscipit recusandae consequuntur expedita et  
8      cum\nreprehenderit molestiae ut ut quas totam\nnostrum rerum est autem  
9      sunt rem eveniet architecto"  
10   },  
11   {  
12     "userId": 1,  
13     "id": 2,  
14     "title": "qui est esse",  
15     "body": "est rerum tempore vitae\nsequi sint nihil reprehenderit dolor beatae  
16     ea dolores neque\nfugiat blanditiis voluptate porro vel nihil molestiae ut  
17     reiciendis\nqui aperiam non debitis possimus qui neque nisi nulla"  
18   }  
19 ]
```

Dúvidas?



Tarefa

- Usar o ThunderClient e fazer requisições para o JSON Placeholder
 - Testar os endpoints disponíveis
 - get
 - post
 - put
 - delete