

APIs

Introdução ao uso de APIs – Aula 1

APIS

O que é uma API

Application Programming Interface (Interface de Programação de Aplicações) é um conjunto de rotinas e padrões estabelecidos por um software para a utilização de suas funcionalidades por aplicativos que não pretendem envolver-se em detalhes da implementação do software, mas apenas usar seus serviços.



APIS

O que é uma API Web

No contexto de desenvolvimento Web, uma API é um conjunto definido de mensagens de requisição e resposta *HTTP*, geralmente expresso nos formatos *XML* ou *JSON*.

Alguns aplicativos populares como o *Twitter* e *Facebook* disponibilizam *APIs* para que você possa obter ou enviar dados para suas plataformas. No caso do *Twitter* é possível ler e publicar mensagens via *API*.

APIS

Como as APIs funcionam

Existem muitas *APIs* para muitos tipos de informação. Existem *APIs* que podem fornecer canções, artistas, álbuns e até informações sobre estilos musicais e artistas relacionados, por exemplo.

Um outro exemplo seria o *ESPN*, que fornece *APIs* com informações de atletas e placares de jogos.

O *Google* tem várias *APIs* em sua seção para Desenvolvedores. São *APIs* para tradução de idiomas, análises, geolocalização, etc.

APIS

Como as APIs funcionam


Você pode obter dados de uma *API* utilizando apenas um navegador de internet.

O site *freegeoip.net* disponibiliza uma *API* onde você fornece um endereço *IP* e recebe como resposta um *JSON* contendo informações de localização daquele endereço.

APIs

Como as APIs funcionam

Veja o *JSON* retornado para o *IP* 201.79.181.197.



A screenshot of a web browser window. The address bar shows the URL `freegeoip.net/json/201.79.181.197`. The page content displays a JSON object representing geolocation data for the specified IP address. The JSON object includes fields for IP, country code, country name, region code, region name, city, zip code, time zone, latitude, longitude, and metro code.

```
{  
  "ip": "201.79.181.197",  
  "country_code": "BR",  
  "country_name": "Brazil",  
  "region_code": "ES",  
  "region_name": "Espírito Santo",  
  "city": "Cachoeiro de Itapemirim",  
  "zip_code": "",  
  "time_zone": "America/Sao_Paulo",  
  "latitude": -20.8529,  
  "longitude": -41.1192,  
  "metro_code": 0  
}
```

APIS

Como as APIs funcionam

As *APIs* não são muito diferentes de um site comum pois funcionam via *HTTP*, o mesmo protocolo usado para buscar dados de sites, baixar um arquivo e fazer quase tudo mais na *Internet*. As únicas coisas que caracterizam uma *API* é a sintaxe extremamente correta que ela usa e o fato de que as *APIs* apresentam seus dados como *JSON* ou *XML*, em vez de *HTML*.

Convenções comuns

Quando fazemos *scraping*, normalmente não temos muitas regras a seguir, nem sempre conseguimos encontrar um padrão para obter as informações (vide aula de scraping de dados da Telelista).

As *APIs* seguem um conjunto de regras extremamente padronizado para produzir informações e também as produzem de uma maneira muito padronizada.

Portanto é fácil aprender algumas regras básicas simples que o ajudarão a lidar com qualquer *API*, contanto que ela tenha sido bem elaborada.

Convenções comuns

Vale ressaltar que algumas *APIs* não seguem essas regras fielmente, logo, é importante ler a documentação da *API* para levantar suas particularidades.

Convenções comuns - Métodos

Existem quatro maneiras de solicitar informações a um servidor Web via HTTP:

- GET
- POST
- PUT
- DELETE

Convenções comuns - Métodos

GET

Usado quando visitamos um site através da barra de endereços do navegador. É o método que usamos ao fazer uma chamada a

`http://freegeoip.net/json/201.79.181.197`

Usando o GET você está informando ao servidor Web que quer obter informações.

Convenções comuns - Métodos

POST

Usado quando preenchemos um formulário ou enviamos informações para o servidor.

Quando você faz *login* em um site está fazendo uma solicitação *POST* com seu nome de usuário e senha.

Convenções comuns - Métodos

PUT

Não é muito usado na interação com sites, mas algumas *APIs* utilizam. Uma solicitação *PUT* é realizada para atualizar um objeto ou uma informação. Uma *API* poderia requerer uma solicitação *POST* para criar um novo usuário e uma solicitação *PUT* para atualizar seus dados por exemplo.

Muitas *APIs* utilizam a solicitação *POST* em vez de *PUT*.

Convenções comuns - Métodos

DELETE

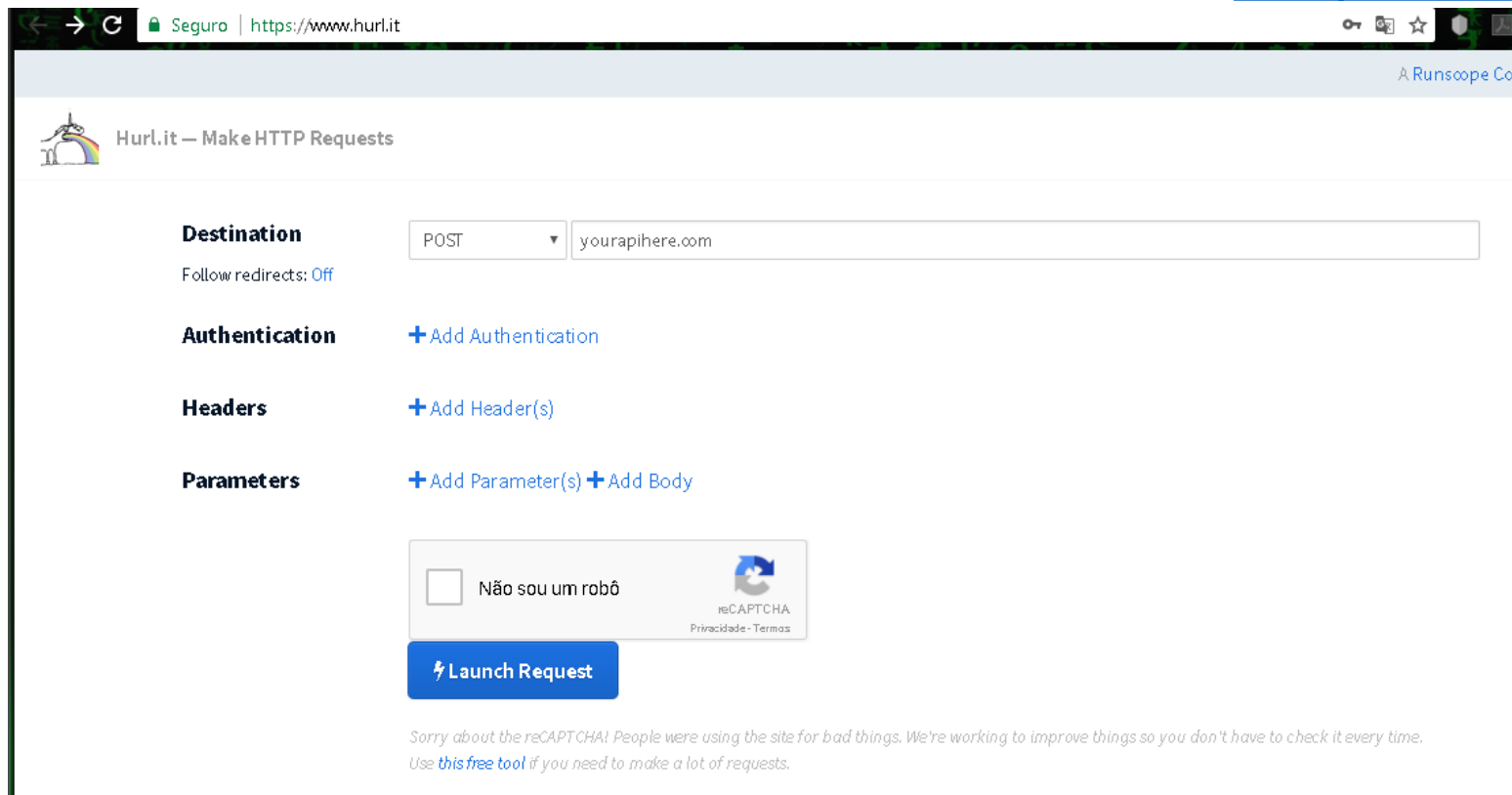
Usado para excluir um objeto. Por exemplo, ao enviarmos uma solicitação *DELETE* para `http://minhaapi.com/usuario/34`, será excluído o usuário com o id igual a 34.

Normalmente *APIs* não trabalham com *DELETE* porque na maioria das vezes, *APIs* fornecem informações e não são criadas para que usuários apaguem informações, porém, assim como o *PUT*, vale a pena entendermos.

APIS

Convenções comuns - Métodos

Podemos testar os métodos no site <https://www.hurl.it>



The screenshot shows the Hurl.it website interface. At the top, the browser address bar displays "Seguro | https://www.hurl.it". The page title is "Hurl.it — Make HTTP Requests". The main form includes a "Destination" section with a dropdown menu set to "POST" and a text input field containing "yourapihere.com". Below this, there are sections for "Authentication", "Headers", and "Parameters", each with a "+ Add" button. At the bottom, there is a reCAPTCHA widget with the text "Não sou um robô" and a "Launch Request" button. A small note at the bottom of the page reads: "Sorry about the reCAPTCHA! People were using the site for bad things. We're working to improve things so you don't have to check it every time. Use [this free tool](#) if you need to make a lot of requests."

Convenções comuns - Autenticação

Algumas APIs não utilizam autenticação, o que significa que suas informações estão abertas na internet para qualquer pessoa, porém, muitas APIs modernas requerem autenticação para que possam ser utilizadas.

Algumas empresas criam APIs e cobram por chamada ou até mesmo por assinatura mensal. Nestes casos é necessária a autenticação.

Tendo o usuário autenticado é possível limitar o número de chamadas por segundo, hora ou dia por exemplo, ou até mesmo restringir algumas funcionalidades. Algumas podem não ter restrições, mas registram seus usuários e as chamadas para fins de marketing.

Convenções comuns - Autenticação

Geralmente as APIs utilizam autenticação via token. Este token é passado para o servidor web a cada chamada feita. Esse token é fornecido ao usuário no ato do registro, sendo a sua identificação permanente para chamadas à API ou, podendo ser alterado com frequência, dependendo do nível de segurança da API.

APIS

Convenções comuns - Autenticação

Exemplo :

http://minhaapi.com/musicas?api_key=<MINHA_CHAVE>%20&cantor=cazuza

Estou fornecendo para a *API* o *token* recebido, representado aqui por <MINHA_CHAVE>. Neste caso, a API realiza a validação e retorna todas as letras das músicas do cantor informado (exemplo fictício).

Neste exemplo, o *token* foi passado via *URL*.

Convenções comuns - Autenticação

Os tokens também podem ser passados para o servidor por meio de um cookie no cabeçalho da solicitação. Podemos utilizar a biblioteca *urllib* para isso.

Exemplo:

```
token = "<MINHA_CHAVE>"  
webRequest =  
urllib.request.Request(http://myapi.com,  
headers={"token":token})  
html = urlopen(webRequest)
```

FIM