# Criando requisições em uma API

Requisições do tipo POST são usadas para enviar dados para a API. Por exemplo, criar um novo registro de usuário com: **nome, idade e endereço de e-mail**.

Diferente de requisições do tipo GET, você precisará passar um objeto das opções de configuração como um segundo argumento em requisições POST.

Quando se faz uma inserção na API , o que é enviado precisa estar em formato de string, utilizando o método JSON.stringfy

# Objetos de configuração de um post

* Method
* Headers
* Body

Para fazer uma solicitação GET simples com fetch, o único parâmetro obrigatório é a URL como argumento, por isso não havíamos declarado o método na função listaVideos(). Para uma solicitação POST, você precisará passar um objeto das opções de configuração como um segundo argumento. O objeto opcional pode receber vários parâmetros diferentes, mas o essencial é a declaração de que está usando o método POST. Você pode ler mais sobre as diferenças entre os dois tipos de requisição no artigo [Diferenças entre GET e POST](https://www.alura.com.br/artigos/diferencas-entre-get-e-post) do Yuri Matheus.

O método de requisição POST foi projetado para solicitar que o servidor web aceite os dados anexados no corpo da mensagem de requisição para armazenamento. Ele é normalmente usado quando se faz o upload de um arquivo ou envia-se um formulário web completo. Uma dica para você lembrar da função do POST é lembrar do termo que várias pessoas usam quando **enviam** fotos ou postagens em redes sociais: **postar.**

# Tratamento de erro com TRY e CATCH

Responsável por avisar quando algo dá errado

## Topicos

* Lidar com erros com try catch;
* Criar novos erros com throw new Error;
* Detectar erros através de condicionais;
* Verificar se uma lista está vazia pelo seu tamanho;
* Imprimir mensagens de erro.