




 This page was translated from English by the community. [Learn more and join the MDN Web Docs community.](#)

HTTP

Hypertext Transfer Protocol (HTTP) é um protocolo de [camada de aplicação](#)  para transmissão de documentos hipermídia, como o HTML. Foi desenvolvido para comunicação entre navegadores web e servidores web, porém pode ser utilizado para outros propósitos também. Segue um [modelo cliente-servidor](#)  clássico, onde um cliente abre uma conexão, executa uma requisição e espera até receber uma resposta. É também um protocolo [sem estado](#)  ou [stateless protocol](#) , que significa que o servidor não mantém nenhum dado entre duas requisições (*state*). Apesar de ser frequentemente baseado em uma camada TCP/IP, pode ser utilizado em qualquer [camada de transporte](#)  confiável, ou seja, um protocolo que não perde mensagens silenciosamente como o UDP.

Tutoriais

Aprenda como usar HTTP com guias e tutoriais.


[Visão geral do HTTP](#)

Apresenta os recursos básicos do protocolo cliente-servidor: o que pode fazer e quais seus usos.

[Cache HTTP](#)

O Cache é muito importante para websites rápidos. Este artigo descreve diferentes métodos de cache e como utilizar os cabeçalhos HTTP para controlá-los.

[Cookies HTTP](#)

O funcionamento dos cookies é definido pela [RFC 6265](#) . Ao receber uma requisição HTTP, o servidor pode enviar um cabeçalho `Set-Cookie` com a resposta. Depois, o cliente retorna o valor do cookie a cada requisição ao mesmo servidor na forma de um cabeçalho `Cookie` HTTP. O cookie também pode ser configurado para expirar em um determinado tempo ou permanecer restrito a um domínio ou endereço específicos.

[Controle de Acesso HTTP \(CORS\)](#)

As requisições CORS são requisições HTTP de recursos de domínios diferentes do domínio que está fazendo a requisição. um exemplo é um site A (SiteA.com) realizando uma requisição de uma imagem por meio de um elemento *img* para um site B (SiteB.com/img.png). Hoje em dia encontramos o CORS em grande maioria das páginas webs, carregando scripts, imagens, folhas de estilos, etc.

[Dicas de cliente HTTP \(en-US\)](#)

Dicas do cliente são um conjunto de cabeçalhos de resposta que um servidor pode usar para solicitar proativamente informações de um cliente sobre o dispositivo, a rede, o usuário e as preferências específicas do agente do usuário. O servidor pode então determinar quais recursos enviar, com base nas informações que o cliente escolher fornecer.

[Evolução do HTTP \(en-US\)](#)

Uma breve descrição das mudanças que ocorreram no HTTP em versões mais antigas para o moderno HTTP/2 em diante.

[Diretrizes de segurança da Web da Mozilla](#)

uma coleção de dicas para ajudar as equipes operacionais na criação de aplicativos Web seguros.

[Mensagens HTTP](#)

Descreve os tipos e a estrutura das diferentes mensagens do HTTP/1.x e HTTP/2.

[Uma sessão típica em HTTP](#)

Mostra e explica o fluxo de uma sessão HTTP normal.

[Gerenciamento de conexões em HTTP/1.x](#)

Descreve os três modelos de conexão disponíveis no HTTP/1.x, seus pontos fortes e fracos.

[Controlando a pré-busca de DNS \(en-US\)](#)

O Firefox e outros navegadores atuais utilizam a **pré-busca de DNS**, ou seja, o navegador faz a busca do nome do domínio nos links contidos numa página antes que o recurso seja requisitado, como por exemplo imagens, scripts e folhas de estilo. Isto tudo ocorre em segundo plano, para que quando o recurso necessite ser carregado, o endereço DNS já esteja resolvido. Isto diminui a latência quando, por exemplo, um usuário clica em um link.

Referências

Navegue por uma detalhada documentação de referências em HTTP.

[Cabeçalhos HTTP](#)

As mensagens de cabeçalho HTTP são utilizadas para realizar a descrição de algum recurso ou comportamento do cliente ou servidor. Propriedades personalizadas nos cabeçalhos utilizam o prefixo 'X-' ; Outras no [registro IANA](#) [↗], cujo conteúdo original foi definido na [RFC 4229](#) [↗]. IANA também mantém o [registro de novas propostas para mensagens de cabeçalhos HTTP](#) [↗].

[Métodos de requisição HTTP](#)

As mais diferentes requisições podem ser utilizadas pelos métodos de requisições HTTP: [GET](#) e [POST](#) , mas também algumas requisições menos comuns pelos métodos [OPTIONS](#) , [DELETE](#) ou [TRACE](#) .

[Respostas de códigos de status em HTTP \(en-US\)](#)

Os códigos de status do HTTP indicam quando uma requisição foi completada. Códigos de status são agrupados em cinco classes: Informações, respostas, respostas de sucesso, redirecionamentos, erros de cliente e erros de servidor.

[Diretivas CSP](#)

Os campos de cabeçalho de resposta [Content-Security-Policy](#) permitem que os administradores do site controlem os recursos que o agente do usuário pode carregar para uma determinada página. Com algumas exceções, as políticas envolvem principalmente a especificação de origens de servidor e endpoints de script.

Ferramentas & recursos

Ferramentas que irão lhe ajudar a testar e debugar as suas requisições HTTP.

[Ferramentas de desenvolvedores Firefox](#) 

[Network monitor](#) 

[Observatório Mozilla](#) 

Um projeto desenvolvido para ajudar desenvolvedores, administradores de sistema e profissionais de segurança a configurar seus sites com segurança.

[RedBot](#) 

Uma ferramenta para visualizar seus Headers relacionados em cache.

[Como os navegadores funcionam \(2011\)](#) 

Um artigo bem abrangente sobre a parte interna dos navegadores e os fluxos de requisições via protocolo HTTP. Um artigo que todos os desenvolvedores web, com certeza, deveriam ler.

Last modified: 6 de nov. de 2022, [by MDN contributors](#)