

# Lista de códigos de status HTTP

A seguir está uma lista de HyperText Transfer Protocol (HTTP) códigos de status resposta. Isso inclui os códigos da IETF padrões de internet, outras especificações e alguns usados códigos adicionais. O primeiro dígito do código de status indica uma das cinco classes de resposta, o mínimo necessário para um cliente HTTP é que ele reconhece essas cinco classes. Microsoft IIS pode usar adicionais decimal sub-códigos específicos para fornecer mais informações, mas estes não estão listadas aqui. As frases utilizadas são os exemplos padrão, mas qualquer legível alternativa humana pode ser fornecido. Salvo disposição em contrário, o código de status é parte do padrão HTTP/1.1.

## 1xx Informativa

Solicitação recebida, dando continuidade ao processo. Essa classe de código de status indica uma resposta provisória, constituída apenas do Status-Line e cabeçalhos opcionais, e é encerrado por uma linha vazia. Desde HTTP/1.0 não define todos os códigos de status 1xx, os servidores não deve enviar uma resposta 1xx para um cliente HTTP/1.0, exceto sob condições experimentais.

**100 Continuar:** Isso significa que o servidor tenha recebido os cabeçalhos da solicitação, e que o cliente deve proceder para enviar o corpo do pedido (no caso de um pedido para que um corpo deve ser enviado, por exemplo, um POST pedido). Se o corpo é grande o pedido, enviando-os para um servidor, quando o pedido já foi rejeitada com base em cabeçalhos inadequado é ineficiente. Para ter um cheque do servidor se o pedido pode ser aceite com base no pedido de cabeçalhos sozinho, o cliente deve enviar Esperar: 100-continue como um cabeçalho no seu pedido inicial e verifique se a 100 Continuar código de status é recebido em resposta antes de permanente (ou receber 417 Falha na expectativa e não continuar).

### 101 Mudando protocolos:

Isso significa que o solicitante pediu ao servidor para mudar os protocolos e o servidor está reconhecendo que irá fazê-lo. 102 Processamento (WebDAV) (RFC 2518) Como uma solicitação WebDAV pode conter muitos sub-pedidos que envolvam operações de arquivo, pode demorar muito tempo para concluir o pedido. Este código indica que o servidor recebeu e está processando o pedido, mas nenhuma resposta ainda não está disponível. Isso impede que o cliente o tempo limite e supondo que o pedido foi perdido.

### 122 Pedido-URI muito longo:

Este é um padrão IE7 somente código não significa que o URI é mais do que um máximo de 2083 caracteres. (Ver código 414).

## 2xx Sucesso

Esta classe de códigos de status indica a ação solicitada pelo cliente foi recebida, compreendida, aceita e processada com êxito.

## **200 OK:**

Padrão de resposta para solicitações HTTP sucesso. A resposta real dependerá do método de solicitação usado. Em uma solicitação GET, a resposta conterá uma entidade que corresponde ao recurso solicitado. Em uma solicitação POST a resposta conterá a descrição de uma entidade, ou contendo o resultado da ação.

## **201 Criado:**

O pedido foi cumprido e resultou em um novo recurso que está sendo criado.

## **202 Aceito:**

O pedido foi aceite para processamento, mas o tratamento não foi concluído. O pedido poderá ou não vir a ser posta em prática, pois pode ser anulado quando o processamento ocorre realmente.

## **203 não-autorizado (desde HTTP/1.1):**

O servidor processou a solicitação com sucesso, mas está retornando informações que podem ser de outra fonte.

## **204 Conteúdo:**

O servidor processou a solicitação com sucesso, mas não está retornando nenhum conteúdo.

## **205 Reset:**

O servidor processou a solicitação com sucesso, mas não está retornando nenhum conteúdo. Ao contrário da 204, esta resposta exige que o solicitante redefinir a exibição de documento.

## **206 Conteúdo parcial:**

O servidor está entregando apenas parte do recurso devido a um cabeçalho intervalo enviados pelo cliente. O cabeçalho do intervalo é usado por ferramentas como wget para permitir retomada de downloads interrompidos, ou dividir um download em vários fluxos simultâneos.

## **207-Status Multi (WebDAV) (RFC 4918):**

O corpo da mensagem que se segue é um XML da mensagem e pode conter um número de códigos de resposta individual, dependendo de quantas sub-pedidos foram feitos.

# **3xx Redirecionamento**

O cliente deve tomar medidas adicionais para completar o pedido. Essa classe de código de status indica que a ação ainda precisa ser levado pelo agente do usuário, a fim de atender à solicitação. A ação necessária pode ser realizada pelo agente, sem interação com o usuário, se e somente se o método utilizado no segundo pedido é GET ou HEAD. Um agente do usuário não deve redirecionar automaticamente uma solicitação de mais de cinco vezes, desde que tais redirecionamentos geralmente indicam um loop infinito .

### **300 Múltipla escolha:**

Indica várias opções para o recurso que o cliente pode acompanhar. É, por exemplo, poderia ser usado para apresentar opções de formato diferente para o vídeo, arquivos de lista com diferentes extensões, ou desambiguação sentido da palavra.

### **301 Movido:**

Esta e todas as solicitações futuras devem ser direcionada para o URI .

### **302 Encontrado:**

Este é um exemplo de boas práticas industriais contradizendo a norma. especificação HTTP/1.0 (RFC 1945) exigiu o cliente para executar um redirecionamento temporário (o que descreve frase original era "Movido Temporariamente"), mas os browsers populares executadas 302 com a funcionalidade de um 303 Consulte Outros. Por isso, acrescentou HTTP/1.1 códigos de status 303 e 307 a distinguir entre os dois comportamentos. No entanto, a maioria das aplicações Web e os quadros ainda usam o código de status 302 como se fosse o 303.

### **304 Not Modified:**

Indica que o recurso não foi modificado desde o último pedido. Normalmente, o cliente fornece um cabeçalho HTTP como o Se-Modificado-Desde cabeçalho para proporcionar um tempo contra o qual para comparar. Usando este poupa largura de banda e de reprocessamento no servidor e cliente, uma vez que apenas os dados do cabeçalho devem ser enviados e recebidos em comparação com a totalidade da página que está sendo reprocessados pelo servidor, em seguida, enviado novamente utilizando mais largura de banda do servidor e cliente .

### **305 Use Proxy (desde HTTP/1.1):**

Muitos clientes HTTP (como o Mozilla e Internet Explorer) podem não tratar corretamente as respostas com este código de status, principalmente por razões de segurança.

### **306 Proxy Switch:**

Deixou de ser usado.

### **307 Redirecionamento temporário (desde HTTP/1.1):**

Nesta ocasião, o pedido deve ser repetido com outro URI, mas futuras solicitações ainda pode usar a URI original. Em contraste com a 303, o método de pedido não deve ser mudado quando a reedição do pedido original. Por exemplo, uma solicitação POST deve ser repetido com outro pedido POST.

## **4xx Erro de cliente**

A classe 4xx de código de status é destinado para os casos em que o cliente parece ter cometido um erro. Exceto quando estiver respondendo a uma solicitação HEAD, o servidor deve incluir uma entidade que contém uma explicação sobre a situação de erro, e se é uma condição temporária ou permanente. Esses códigos de status são aplicáveis a qualquer método de solicitação. Os agentes do usuário devem exibir qualquer entidade incluída para o usuário. Estes são tipicamente os códigos de

erro mais comuns encontrados durante online.

#### **400 Bad Request:**

O pedido não pode ser entregue devido à sintaxe incorreta.

#### **401 Unauthorized:**

Semelhante ao 403 Forbidden, mas especificamente para o uso quando a autenticação é possível, mas não conseguiu ou ainda não foram fornecidos. A resposta deve incluir um cabeçalho do campo Authenticate-WWW contendo um desafio aplicável ao recurso solicitado. Veja Basic autenticação de acesso e autenticação Digest acesso .

#### **402 Pagamento necessário:**

Reservado para uso futuro. A intenção original era que esse código pode ser usado como parte de alguma forma de dinheiro digital ou de micropagamento regime, mas isso não aconteceu, e esse código não é usado normalmente. Como exemplo de seu uso, no entanto, da Apple, o MobileMe serviço gera um erro 402 ("HttpStatusCode: 402" no Mac OS X log do console) se a conta MobileMe é delinquente.

#### **403 Forbidden:**

O pedido foi um pedido legal, mas o servidor está recusando a responder a ela. Ao contrário de um Unauthorized 401 resposta, autenticação não fará diferença.

#### **404 Not Found:**

O recurso requisitado não foi encontrado, mas pode ser disponibilizado novamente no futuro. As solicitações subsequentes pelo cliente são permitidas.

#### **405 Método não permitido:**

Foi feita uma solicitação de um recurso usando um método de pedido não é compatível com esse recurso, por exemplo, usando GET em um formulário, que exige que os dados a serem apresentados via POST, PUT ou usar em um recurso somente de leitura.

#### **406 Não Aceitável:**

O recurso solicitado é apenas capaz de gerar conteúdo não aceitáveis de acordo com os cabeçalhos Accept enviados na solicitação.

#### **407 Autenticação de proxy necessária**

#### **408 Timeout Pedido:**

O servidor sofreu timeout ao aguardar a solicitação. De acordo com as especificações HTTP W3: "O cliente não apresentar um pedido dentro do tempo que o servidor estava preparado para esperar o tempo. O cliente pode repetir o pedido, sem modificações na tarde qualquer. "

#### **409 Conflito:**

Indica que a solicitação não pôde ser processada por causa do conflito no pedido, como um conflito

de edição .

#### **410 Gone:**

Indica que o recurso solicitado não está mais disponível e não estará disponível novamente. Isto deve ser usado quando um recurso foi intencionalmente removido e os recursos devem ser removidos. Ao receber um código de estado 410, o cliente não deverá solicitar o recurso novamente no futuro. Clientes como motores de busca devem remover o recurso de seus índices. A maioria dos casos de uso não necessitam de clientes e motores de busca para purgar o recurso, e um "404 Not Found" pode ser utilizado.

#### **411 comprimento necessário:**

O pedido não especifica o comprimento do seu conteúdo, o que é exigido pelo recurso solicitado.

#### **412 Pré-condição falhou:**

O servidor não cumpre uma das condições que o solicitante coloca na solicitação.

#### **413 Entidade de solicitação muito grande:**

A solicitação é maior do que o servidor está disposto ou capaz de processar.

#### **414 Pedido-URI Too Long:**

O URI fornecido foi muito longo para ser processado pelo servidor.

#### **415 Tipo de mídia não suportado:**

A entidade tem um pedido tipo de mídia que o servidor ou o recurso não tem suporte. Por exemplo, o cliente carrega uma imagem como image / svg + xml, mas o servidor requer que imagens usar um formato diferente.

#### **416 Solicitada Não Faixa Satisfatório:**

O cliente solicitou uma parte do arquivo, mas o servidor não pode fornecer essa parte. Por exemplo, se o cliente pediu uma parte do arquivo que está para além do final do arquivo.

#### **417 Falha na expectativa:**

O servidor não pode cumprir as exigências do campo de cabeçalho Espere-pedido.

#### **418 Eu sou uma mula velha**

Este código foi definido em 1998 como um dos tradicionais IETF Fools 'brincadeiras de abril, na RFC 2324, Hyper Text Cafeteira Control Protocol, e não é esperado para ser implementado por servidores HTTP real.

#### **422 Entidade improcessável (WebDAV) (RFC 4918):**

O pedido foi bem formado, mas era incapaz de ser seguido devido a erros de semântica.

#### **423 fechado (WebDAV) (RFC 4918):**

O recurso que está sendo acessado está bloqueado.

#### **424 Falha de Dependência (WebDAV) (RFC 4918):**

A solicitação falhou devido à falha de uma solicitação anterior (por exemplo, um PROPPATCH).

#### **425 coleção não ordenada (RFC 3648):**

Definido em projectos de “WebDAV Avançada Coleções Protocolo”, mas não está presente no “Web Distributed Authoring and Versioning (WebDAV) Ordenados Coleções protocolo”.

#### **426 Upgrade Obrigatório (RFC 2817):**

O cliente deve mudar para um outro protocolo, como TLS/1.0 . Resposta n<sup>o</sup> 444 Um Nginx extensão do servidor HTTP. O servidor retorna nenhuma informação para o cliente e fecha a conexão (útil como um impedimento para malware). Com 449 Repetir Uma extensão de Microsoft. O pedido deve ser repetida após a realização da ação apropriada.

#### **450 bloqueados pelo Windows Controles dos Pais:**

Uma extensão de Microsoft. Este erro é dado quando Parental Controls do Windows estão ativadas e está bloqueando o acesso a determinada página da web.

#### **499 cliente fechou Pedido:**

Um Nginx extensão do servidor HTTP. Este código é introduzido para registrar o caso quando a conexão é fechada pelo cliente ao servidor HTTP é o processamento de seu pedido, fazendo com que servidor não consegue enviar o cabeçalho HTTP de volta.

## Referências

1. [http://pt.wikipedia.org/wiki/Anexo:Lista\\_de\\_c%C3%B3digos\\_de\\_status\\_HTTP](http://pt.wikipedia.org/wiki/Anexo:Lista_de_c%C3%B3digos_de_status_HTTP)

From:

<http://wiki.douglasqsantos.com.br/> - **DQS CONSULTORIA E TREINAMENTOS**

Permanent link:

[http://wiki.douglasqsantos.com.br/doku.php/lista\\_de\\_codigos\\_de\\_status\\_http\\_pt\\_br](http://wiki.douglasqsantos.com.br/doku.php/lista_de_codigos_de_status_http_pt_br)

Last update: **2017/09/05 12:18**

