

# Referência Rápida de Códigos de Status HTTP

Códigos de status, cabeçalhos e padrões comuns de resposta

## Informacional 1xx

### Códigos 1xx

- 100** Continue — servidor recebeu os cabeçalhos, cliente deve enviar o corpo
- 101** Switching Protocols — atualizando para WebSocket ou HTTP/2
- 102** Processing — servidor recebeu a requisição, ainda processando (WebDAV)
- 103** Early Hints — pré-carregamento de recursos antes da resposta final

### Nota de Uso

```
# 100 Continue: cliente envia cabeçalho Expect, aguarda 100
curl -H "Expect: 100-continue" -d @large.json URL
# 101: upgrade para WebSocket
Connection: Upgrade / Upgrade: websocket
```

## Sucesso 2xx

### Códigos 2xx

- 200** OK — resposta de sucesso padrão
- 201** Created — recurso criado com sucesso (POST/PUT)
- 202** Accepted — requisição recebida, processamento assíncrono
- 203** Non-Authoritative Info — transformado por proxy
- 204** No Content — sucesso sem corpo de resposta (DELETE)
- 205** Reset Content — sucesso, cliente deve redefinir o formulário
- 206** Partial Content — requisição de intervalo atendida
- 207** Multi-Status — múltiplos códigos de status (WebDAV)

### Uso em API REST

- GET → 200** Retorna recurso com corpo
- POST → 201** Recurso criado, inclui cabeçalho Location
- PUT → 200/204** Recurso atualizado (com/sem corpo)
- DELETE → 204** Excluído, sem corpo retornado
- PATCH → 200** Atualização parcial, retorna recurso modificado

## Redirecionamento 3xx

### Códigos 3xx

- 300** Multiple Choices — múltiplas representações disponíveis
- 301** Moved Permanently — recurso movido, atualize os favoritos
- 302** Found — redirecionamento temporário (frequentemente usado indevidamente como 303)
- 303** See Other — redirecione com GET após POST
- 304** Not Modified — use a versão em cache (ETag/If-Modified)
- 307** Temporary Redirect — mesmo método, localização temporária
- 308** Permanent Redirect — mesmo método, localização permanente

### Comportamento de Redirecionamento

- 301/308** Permanente — mecanismos de busca atualizam o índice
- 302/307** Temporário — URL original permanece canônica
- 301/302** Pode mudar o método para GET no redirecionamento
- 307/308** Deve preservar o método HTTP original

## Erro do Cliente 4xx

### Erros Comuns do Cliente

- 400** Bad Request — sintaxe malformada ou parâmetros inválidos
- 401** Unauthorized — autenticação necessária ou falhou
- 403** Forbidden — autenticado, mas sem permissão
- 404** Not Found — recurso não existe
- 405** Method Not Allowed — método HTTP não suportado
- 406** Not Acceptable — não é possível satisfazer o cabeçalho Accept
- 408** Request Timeout — cliente muito lento para enviar a requisição
- 409** Conflict — requisição conflita com o estado atual

## Mais Erros do Cliente

- 410** Gone — recurso permanentemente excluído (não apenas ausente)
- 411** Length Required — cabeçalho Content-Length ausente
- 412** Precondition Failed — If-Match/If-Unmodified falhou
- 413** Content Too Large — corpo da requisição excede o limite
- 414** URI Too Long — URL excede o limite do servidor
- 415** Unsupported Media Type — Content-Type não aceito
- 422** Unprocessable Content — sintaxe válida, erros semânticos
- 429** Too Many Requests — limite de taxa excedido

## Erro do Servidor 5xx

### Códigos 5xx

- 500** Internal Server Error — exceção não tratada no servidor
- 501** Not Implemented — servidor não suporta o método
- 502** Bad Gateway — servidor upstream enviou resposta inválida
- 503** Service Unavailable — sobrecarregado ou em manutenção
- 504** Gateway Timeout — servidor upstream não respondeu a tempo
- 505** HTTP Version Not Supported — versão não suportada
- 507** Insufficient Storage — servidor não pode armazenar a requisição (WebDAV)
- 511** Network Auth Required — login em portal cativo necessário

### Estratégia de Reenvio

- 500** Reenvie com backoff; pode ser transitório
- 502/504** Reenvie — problema upstream pode se resolver
- 503** Verifique o cabeçalho Retry-After antes de reenviar
- 501/505** Não reenvie — corrija a requisição do cliente

## Códigos Comuns

### Códigos Mais Usados (resumo)

- 200** OK — tudo funcionou
- 201** Created — novo recurso criado
- 204** No Content — sucesso, corpo vazio
- 301** Moved Permanently — atualize a URL
- 304** Not Modified — use o cache
- 400** Bad Request — corrija sua requisição
- 401** Unauthorized — faça login primeiro
- 403** Forbidden — permissões insuficientes
- 404** Not Found — URL errada ou excluída
- 422** Unprocessable — erros de validação
- 429** Too Many Requests — reduza a velocidade
- 500** Server Error — não é culpa sua
- 502** Bad Gateway — falha de proxy/upstream
- 503** Unavailable — tente novamente mais tarde

## Referência de Cabeçalhos

### Cabeçalhos de Requisição

- Accept** Tipos de mídia desejados na resposta (ex.: application/json)
- Authorization** Credenciais (Bearer token, Basic base64)
- Content-Type** Tipo de mídia do corpo da requisição
- If-None-Match** Condicional: ETag para validação de cache
- If-Modified-Since** Condicional: data para validação de cache
- Cache-Control** Diretivas de cache (no-cache, max-age)
- User-Agent** String de identificação do cliente

## Cabeçalhos de Resposta

- Content-Type** Tipo de mídia do corpo da resposta
- Location** URL de redirecionamento ou recurso criado
- ETag** Tag de entidade para validação de cache
- Cache-Control** Diretivas de cache (max-age, no-store)
- Retry-After** Tempo de espera antes de reenviar (429/503)
- WWW-Authenticate** Esquema de autenticação exigido (enviado com 401)
- Set-Cookie** Define cookie no cliente

## Padrões Comuns

### Fluxo de Cache

```
# Primeira requisição – servidor retorna ETag
GET /api/data → 200, ETag: "abc123"
# Requisição subsequente – condicional
GET /api/data, If-None-Match: "abc123"
→ 304 Not Modified (use cache)
```

### Fluxo de Autenticação

```
# Requisição não autenticada
GET /api/secret → 401, WWW-Authenticate: Bearer
# Com token
GET /api/secret, Authorization: Bearer <token>
→ 200 OK
```

### Limitação de Taxa

```
# Resposta com limite de taxa atingido
429 Too Many Requests
Retry-After: 60
X-RateLimit-Remaining: 0
X-RateLimit-Reset: 1700000000
```

### Negociação de Conteúdo

```
# Cliente prefere JSON, aceita XML
Accept: application/json, application/xml;q=0.9
# Servidor não pode satisfazer → 406 Not Acceptable
# Servidor retorna melhor correspondência → 200 + Content-Type
```