

Status Codes HTTP

Os **códigos de status HTTP** indicam o resultado de uma requisição.

Em APIs REST, usar o código correto torna a comunicação **clara, padronizada e previsível**.

◆ Dicas gerais de uso em APIs REST

-  Use o **código mais específico possível**; evite abusar de **200 OK** ou **500 Internal Server Error**.
-  Combine com **respostas JSON informativas**, exemplo:

```
POST /tasks
{
  "title": "Aprender status codes"
}
→
201 Created
{
  "id": 12,
  "message": "Tarefa criada com sucesso"
}
```

-  **REST tip:**
 - **201 Created** para novos recursos
 - **204 No Content** para DELETE ou PUT idempotentes
 - **4xx** para erros do cliente
 - **5xx** para erros do servidor
 - Redirecionamentos (**3xx**) raros em APIs REST; prefira **307** ou **308**.

◆ Classes de status codes

| Faixa | Categoria | Significado geral |
|-------|--|---|
| 1xx | Informacional | Pedido recebido; cliente deve continuar esperando. |
| 2xx |  Sucesso | Pedido processado com sucesso. |
| 3xx |  Redirecionamento | Cliente precisa de ação adicional (seguir URL, método etc.) |
| 4xx |  Erro do cliente | Requisição inválida ou não autorizada. |
| 5xx |  Erro do servidor | Servidor falhou ao processar requisição válida. |

 **REST tip:** 1xx raramente usado em APIs, geralmente apenas para handshakes ou long polling.

♦ Status Codes úteis para REST

Estes são os códigos que você **mais usará** em APIs REST, destacados com cores e ícones:

| Código | Categoria | Uso típico REST | Exemplo |
|--------|-----------|-------------------------------------|---------------------------|
| 200 | Sucesso | Requisição OK | GET /tasks |
| 201 | Sucesso | Recurso criado | POST /tasks |
| 204 | Sucesso | Sem corpo de resposta (idempotente) | DELETE /tasks/3 |
| 400 | Cliente | Requisição malformada | JSON inválido |
| 401 | Cliente | Falta autenticação | Token ausente |
| 403 | Cliente | Sem permissão | Usuário sem acesso admin |
| 404 | Cliente | Recurso não encontrado | /tasks/999 |
| 405 | Cliente | Método HTTP inválido | POST /users/1 |
| 409 | Cliente | Conflito de estado | Dados duplicados |
| 422 | Cliente | Validação falhou | Campo obrigatório ausente |
| 429 | Cliente | Límite de requisições | Rate limiting |
| 500 | Servidor | Erro genérico | Exceção não tratada |
| 503 | Servidor | Serviço indisponível | Manutenção/sobre carga |

♦ 1xx — Informacional

| Código | Nome | Significado | Exemplo |
|--------|---------------------|--|--------------------------|
| 100 | Continue | Cliente pode continuar enviando o corpo da requisição. | Uploads longos |
| 101 | Switching Protocols | Servidor trocando protocolo (HTTP → WebSocket). | Handshake WebSocket |
| 102 | Processing | Pedido recebido, ainda processando (WebDAV). | Operações assíncronas |
| 103 | Early Hints | Headers antecipados antes da resposta completa. | Pré-carregamento CSS/JSS |

♦ 2xx — Sucesso

| Código | Nome | Significado | Exemplo |
|--------|------|-------------------------|--|
| 200 | OK | Requisição bem-sucedida | GET /tasks retorna lista de tarefas |

| Código | Nome | Significado | Exemplo |
|--------|-------------------------------|---|----------------------------------|
| 201 | Created | Recurso criado com sucesso | POST /tasks → nova tarefa |
| 202 | Accepted | Pedido aceito, processamento ainda não concluído | Jobs assíncronos |
| 203 | Non-Authoritative Information | Resposta proveniente de fonte não autoritativa | Proxy modificando resposta |
| 204 | No Content | Pedido concluído sem corpo de resposta | DELETE /tasks/3 |
| 205 | Reset Content | Cliente deve resetar formulário/estado | Reset de formulário web |
| 206 | Partial Content | Responde parcialmente a requisição com Range : | Downloads segmentados |

(Demais códigos 2xx como WebDAV ou **IM Used** raramente aplicáveis em REST, mantidos apenas para referência)

◆ 3xx — Redirecionamento

| Código | Nome | Significado | Exemplo |
|--------|--------------------|---|--------------------------------------|
| 300 | Multiple Choices | Recurso possui múltiplas opções; cliente escolhe | /file → PDF ou DOCX |
| 301 | Moved Permanently | Recurso movido permanentemente | /old → /new |
| 302 | Found | Redirecionamento temporário | Logins |
| 303 | See Other | POST deve usar GET na URL indicada | Redirecionar após criação de recurso |
| 304 | Not Modified | Recurso não mudou; usar cache | Evita download duplicado |
| 307 | Temporary Redirect | Redirecionamento temporário, preserva método HTTP | APIs REST seguras |
| 308 | Permanent Redirect | Redirecionamento permanente, preserva método HTTP | Atualização de rota |

◆ 4xx — Erros do Cliente

| Código | Nome | Significado | Exemplo |
|--------|-------------|------------------------|---------------|
| 400 | Bad Request | Requisição malformada. | JSON inválido |

| Código | Nome | Significado | Exemplo |
|--------|----------------------|--------------------------------------|--------------------------------|
| 401 | Unauthorized | Requer autenticação. | Token ausente |
| 403 | Forbidden | Autenticado, sem permissão. | Acesso admin negado |
| 404 | Not Found | Recurso inexistente. | /tasks/999 |
| 405 | Method Not Allowed | Método HTTP não suportado na rota. | POST /users/1 |
| 406 | Not Acceptable | Tipo de mídia ou formato não aceito. | Accept: application/xml |
| 409 | Conflict | Conflito de estado. | Dados duplicados |
| 422 | Unprocessable Entity | Sintaxe ok, mas dados inválidos. | Campo obrigatório ausente |
| 429 | Too Many Requests | Límite de requisições excedido. | Rate limiting |

(Demais 4xx mantidos para referência ou casos especiais, raramente usados em REST)

◆ 5xx — Erros do Servidor

| Código | Nome | Significado | Exemplo |
|--------|-----------------------|---------------------------------|--------------------------|
| 500 | Internal Server Error | Erro genérico do servidor | Exceção não tratada |
| 501 | Not Implemented | Funcionalidade não suportada | Endpoint ausente |
| 502 | Bad Gateway | Proxy recebeu resposta inválida | API externa falhou |
| 503 | Service Unavailable | Servidor indisponível | Manutenção/sobre carga |
| 504 | Gateway Timeout | Proxy demorou para responder | Serviço downstream lento |

(Demais 5xx mantidos para referência ou WebDAV/uso especializado)

Leituras adicionais

-  [MDN Web Docs — HTTP Status Codes](#)

Guia completo e atualizado da Mozilla explicando cada código, categoria e casos de uso recomendados.

-  [RFC 9110 — HTTP Semantics \(IETF\)](#)

Especificação oficial dos códigos de status e definições formais segundo o padrão HTTP moderno.

-  [REST API Tutorial — HTTP Status Codes](#)

Explicações práticas e exemplos de uso de status codes em APIs REST.