

HTTP ステータスコード クイックリファレンス

ステータスコード、ヘッダー、よくあるレスポンスパターン

情報 1xx

1xx コード

100	Continue — サーバーがヘッダーを受信、クライアントはボディを送信すべき
101	Switching Protocols — WebSocket または HTTP/2 へのアップグレード
102	Processing — サーバーがリクエストを受信し処理中 (WebDAV)
103	Early Hints — 最終レスポンス前にリソースをプリロード

使用上の注意

```
# 100 Continue: client sends Expect header, waits for 100
curl -H "Expect: 100-continue" -d @large.json URL
# 101: upgrade to WebSocket
Connection: Upgrade / Upgrade: websocket
```

成功 2xx

2xx コード

200	OK — 標準的な成功レスポンス
201	Created — リソースの作成に成功 (POST/PUT)
202	Accepted — リクエストを受信し、非同期で処理中
203	Non-Authoritative Info — プロキシによって変換された情報
204	No Content — レスポンスボディなしで成功 (DELETE)
205	Reset Content — 成功、クライアントはフォームをリセットすべき
206	Partial Content — レンジリクエストが完了
207	Multi-Status — 複数のステータスコード (WebDAV)

REST API の使用例

GET → 200	ボディ付きでリソースを返す
POST → 201	リソース作成済み、Location ヘッダーを含める
PUT → 200/204	リソース更新 (ボディあり/なし)
DELETE → 204	削除済み、ボディなし
PATCH → 200	部分更新、変更後のリソースを返す

リダイレクト 3xx

3xx コード

300	Multiple Choices — 複数の表現が利用可能
301	Moved Permanently — リソースが移動した、ブックマークを更新すべき
302	Found — 一時的なリダイレクト (303 の代わりに誤用されることが多い)
303	See Other — POST 後に GET でリダイレクト
304	Not Modified — キャッシュを使用 (ETag/If-Modified)
307	Temporary Redirect — 同一メソッドで一時的な場所へ
308	Permanent Redirect — 同一メソッドで恒久的な場所へ

リダイレクトの動作

301/308	恒久的 — 検索エンジンはインデックスを更新する
302/307	一時的 — 元の URL が正規 URL のまま
301/302	リダイレクト時にメソッドが GET に変更される場合がある
307/308	元の HTTP メソッドを保持しなければならない

クライアントエラー 4xx

よくあるクライアントエラー

400	Bad Request — 構文が不正またはパラメーターが無効
401	Unauthorized — 認証が必要または失敗した
403	Forbidden — 認証済みだがアクセス権がない
404	Not Found — リソースが存在しない
405	Method Not Allowed — HTTP メソッドがサポートされていない
406	Not Acceptable — Accept ヘッダーを満たせない
408	Request Timeout — クライアントのリクエスト送信が遅すぎる
409	Conflict — リクエストが現在の状態と競合

その他のクライアントエラー

410	Gone — リソースが永久に削除された (単なる不在ではない)
411	Length Required — Content-Length ヘッダーがない
412	Precondition Failed — If-Match/If-Unmodified の条件が不成立
413	Content Too Large — リクエストボディが上限を超えた
414	URI Too Long — URL がサーバー上限を超えた
415	Unsupported Media Type — Content-Type が受け入れられない
422	Unprocessable Content — 構文は有効だが意味論的エラー
429	Too Many Requests — レート制限を超えた

サーバーエラー 5xx

5xx コード

500	Internal Server Error — サーバーで未処理の例外が発生
501	Not Implemented — サーバーがそのメソッドをサポートしない
502	Bad Gateway — アップストリームサーバーが無効なレスポンスを返した
503	Service Unavailable — 過負荷またはメンテナンス中
504	Gateway Timeout — アップストリームサーバーが時間内に応答しなかった
505	HTTP Version Not Supported — バージョンが処理できない
507	Insufficient Storage — サーバーがリクエストを保存できない (WebDAV)
511	Network Auth Required — キャプティブポータルへのログインが必要

リトライ戦略

500	バックオフしてリトライ; 一時的な場合がある
502/504	リトライ — アップストリームの問題が解消される可能性がある
503	リトライ前に Retry-After ヘッダーを確認
501/505	リトライ不可 — クライアントのリクエストを修正する

よく使うコード

最頻出コード (一覧)

200	OK — すべて正常
201	Created — 新しいリソースを作成
204	No Content — 成功、ボディなし
301	Moved Permanently — URL を更新
304	Not Modified — キャッシュを使用
400	Bad Request — リクエストを修正
401	Unauthorized — 先にログイン
403	Forbidden — 権限が不足
404	Not Found — URL が誤りまたは削除済み
422	Unprocessable — バリデーションエラー
429	Too Many Requests — 速度を落とす
500	Server Error — クライアントの問題ではない
502	Bad Gateway — プロキシ/アップストリームの障害
503	Unavailable — 後でリトライ

ヘッダーリファレンス

リクエストヘッダー

Accept	希望するレスポンスのメディアタイプ (例: application/json)
Authorization	クレデンシャル (Bearer トークン、Basic base64)
Content-Type	リクエストボディのメディアタイプ
If-None-Match	条件付き: キャッシュ検証用の ETag
If-Modified-Since	条件付き: キャッシュ検証用の日時
Cache-Control	キャッシュディレクティブ (no-cache、max-age)
User-Agent	クライアントの識別文字列

レスポンスヘッダー

Content-Type	レスポンスボディのメディアタイプ
Location	リダイレクト先または作成されたリソースの URL
ETag	キャッシュ検証用のエンティティタグ
Cache-Control	キャッシュディレクティブ (max-age、no-store)
Retry-After	リトライまでの待機時間 (429/503)
WWW-Authenticate	必要な認証スキーム (401 と共に送信)
Set-Cookie	クライアントにクッキーを設定

よくあるパターン

キャッシュフロー

```
# First request - server returns ETag
GET /api/data → 200, ETag: "abc123"
# Subsequent request - conditional
GET /api/data, If-None-Match: "abc123"
→ 304 Not Modified (use cache)
```

認証フロー

```
# Unauthenticated request
GET /api/secret → 401, WWW-Authenticate: Bearer
# With token
GET /api/secret, Authorization: Bearer <token>
→ 200 OK
```

レート制限

```
# Rate limited response
429 Too Many Requests
Retry-After: 60
X-RateLimit-Remaining: 0
X-RateLimit-Reset: 1700000000
```

コンテンツネゴシエーション

```
# Client prefers JSON, accepts XML
Accept: application/json, application/xml;q=0.9
# Server can't satisfy → 406 Not Acceptable
# Server returns best match → 200 + Content-Type
```