

# 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

<b>100</b>	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
<b>101</b>	Switching Protocols — WebSocket または HTTP/2 へのアップグレード
<b>102</b>	Processing — サーバーがリクエストを受信し処理中 (WebDAV)
<b>103</b>	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

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

### REST API の使用例

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

<b>300</b>	Multiple Choices — 複数の表現が利用可能
<b>301</b>	Moved Permanently — リソースが移動した、ブックマークを更新すべき
<b>302</b>	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)
<b>303</b>	See Other — POST 後に GET でリダイレクト
<b>304</b>	Not Modified — キャッシュを使用 (ETag/If-Modified)
<b>307</b>	Temporary Redirect — 同一メソッドで一時的な場所へ
<b>308</b>	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

<b>301/308</b>	恒久的 — 検索エンジンはインデックスを更新する
<b>302/307</b>	一時的 — 元の URL が正規 URL のまま
<b>301/302</b>	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
<b>307/308</b>	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

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

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

## サーバーエラー 5xx

### 5xx コード

<b>500</b>	Internal Server Error — サーバーで未処理の例外が発生
<b>501</b>	Not Implemented — サーバーがそのメソッドをサポートしない
<b>502</b>	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
<b>503</b>	Service Unavailable — 過負荷またはメンテナンス中
<b>504</b>	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
<b>505</b>	HTTP Version Not Supported — バージョンが処理できない
<b>507</b>	Insufficient Storage — サーバーがリクエストを保存できない (WebDAV)
<b>511</b>	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

<b>500</b>	バックオフしてリトライ; 一時的な場合がある
<b>502/504</b>	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
<b>503</b>	リトライ前に Retry-After ヘッダーを確認
<b>501/505</b>	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)

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

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

### レスポンスヘッダー

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