

Les codes de statut

Le code de statut est la première chose qu'un client regarde. Un bon code évite au client de devoir lire le body pour savoir si ça a marché.

Une API qui répond toujours `200 OK` avec `{"error": true}` dans le body oblige chaque client à réinventer la gestion d'erreurs. Le code de statut est fait pour ça : utilisez-le.

Les exemples ci-dessous concernent l'API d'une bibliothèque en ligne : des livres, des membres, des emprunts.

2xx - Succès

Code	Quand l'utiliser	Exemple
<code>200 OK</code>	Succès avec un body	<code>GET /v1/livres</code>
<code>201 Created</code>	Ressource créée. Ajouter le header <code>Location</code>	<code>POST /v1/emprunts</code>
<code>204 No Content</code>	Succès sans body	<code>DELETE /v1/emprunts/42</code>

3xx - Redirection

Code	Quand l'utiliser
<code>301 Moved Permanently</code>	La ressource a changé d'adresse définitivement
<code>304 Not Modified</code>	La version en cache du client est toujours bonne

4xx - Erreur du client

Code	Quand l'utiliser	Exemple
<code>400 Bad Request</code>	Requête mal formée	Un body qui n'est pas du JSON valide

Code	Quand l'utiliser	Exemple
401 Unauthorized	Pas authentifié, ou authentification invalide	GET /v1/me sans token
403 Forbidden	Authentifié, mais pas autorisé	Un membre qui veut supprimer un livre du catalogue
404 Not Found	La ressource n'existe pas	GET /v1/livres/999999
405 Method Not Allowed	Méthode non supportée sur cette URI. Ajouter le header Allow	DELETE /v1/livres (supprimer tout le catalogue)
406 Not Acceptable	Format demandé dans Accept indisponible	Accept: image/png
409 Conflict	Conflit avec l'état actuel de la ressource	Emprunter un livre déjà emprunté
418 I'm a teapot	Le serveur est une théière (RFC 2324, poisson d'avril de 1998)	Demander un café à une théière
422 Unprocessable Content	Requête bien formée mais données invalides	Une date de retour antérieure à la date d'emprunt
429 Too Many Requests	Trop de requêtes. Ajouter le header Retry-After	Un script qui interroge l'API en boucle

5xx - Erreur du serveur

Code	Quand l'utiliser
500 Internal Server Error	Bug côté serveur. Ne jamais le renvoyer volontairement
502 Bad Gateway	Un intermédiaire n'a pas obtenu de réponse valide du serveur
503 Service Unavailable	Serveur indisponible (maintenance, surcharge)

Les pièges classiques

- **401 ou 403 ?** 401 : « je ne sais pas qui tu es ». 403 : « je sais qui tu es, et c'est non ».
- **400 ou 422 ?** 400 : je n'arrive pas à lire ta requête. 422 : je l'ai lue, mais son contenu n'est pas valide. FastAPI renvoie 422 par défaut pour les erreurs de validation.
- **404 ou 403 ?** Pour une ressource qui appartient à quelqu'un d'autre, répondre 404 évite de révéler qu'elle existe. On en reparle au chapitre 8.

La liste complète est sur MDN. Et pour les retenir : [http.cat](http://cat).

Revision #3

Created 2 October 2026 20:00:42 by Thibaud FRICHET

Updated 2 October 2026 21:01:01 by Thibaud FRICHET