

Common errors

HTTP status code	Type/Description	Reason
201	Success	
400	Client error. Bad request	<p>Invalid URL parameter value ; missing body ; invalid body ; missing body field ; missing header ; invalid header value; request is malformed; required parameters is missing; unsupported parameter; repeats the same parameter multiple times in its request</p> <ul style="list-style-type: none"> <ul style="list-style-type: none"> HTTP/1.1 400 Bad Request Content-Type: application/problem+json Content-Language: en { "type": "https://myapi.com/validation-error", "title": "Validation error", "detail": "Your request parameters are not valid.", "invalid-params": [{ "name": "data", "reason": "cannot be blank." }] }
401	Client error. Unauthorized	<ul style="list-style-type: none"> the client request has not been completed because it lacks valid authentication credentials for the requested resource <ul style="list-style-type: none"> issue with your authentication credentials server received an unverified request An access token is missing. An access token is expired, revoked, malformed, or invalid for other reasons. HTTP/1.1 401 Unauthorized WWW-Authenticate: Bearer realm="example" HTTP/1.1 401 Unauthorized WWW-Authenticate: Bearer realm="example", error="invalid_token",

		<ul style="list-style-type: none"> ○ error_description="The access token expired" ○ ○ ○ ○ ○ Invalid or expired Token ; unknown clientId.
403	Client error. Forbidden access	<ul style="list-style-type: none"> • a client is forbidden from accessing a valid URL. The server understands the request, but it can't fulfil the request because of client-side issues. The caller isn't authorized to access an API that's using a Lambda authorizer • API tells the client that the credentials it provided (e.g., the access token) are valid, but it needs appropriate privileges to perform the requested action • Authorization role not allowed. Example:consumer has not the rights to create/read (permissions).
404	Client error. Not found	<ul style="list-style-type: none"> • client attempts to access a resource that it MUST NOT access at all, for example, because it belongs to another user. • URL not valid ; missing path parameter ; invalid object identifier.
405	Client error. Method not allowed	<ul style="list-style-type: none"> • the server knows the request method, but the target resource doesn't support this method • It's an HTTP response status code that indicates that the request method is known by the server but is not supported by the target resource. • a consumer will get a 405 Method Not Allowed response when sending a request with a method which is not explicitly mentioned in in the contract or documentation on the target resource. For instance, that means if the API's contract or documentation states the only method supported by a resource /resource is GET, a consumer sending a request on this resource using any other method, like DELETE /resource for instance, will get a 405 Method Not Allowed response. • Invalid method. Example: POST instead of GET.]
409	Client error. Conflict	<p>The HTTP Status Code 409 means that the client's request could not be processed because of conflict in the current state of the resource.</p> <p>Examples of conflicts that could result in an HTTP Status Code 409 include an edit conflict on a given resource as a result of multiple simultaneous updates.</p> <p>* * * * *</p> <ul style="list-style-type: none"> • <ul style="list-style-type: none"> ○ affect/update a file that does not exist (like setting the request URL to a path without a file name)

		<ul style="list-style-type: none"> ○ We are also getting the same error while we are trying to access a same resource with in milliseconds. Like if i try to <code>POST</code> some data to <code>/attachment</code> and with in milliseconds an other request will also go for the same resource i.e. <code>www.abc.com/attachment{ }</code> from the same user ○ when you name your files a reserved or blocked name. These could be names like <code>register</code>, <code>contact</code> ○ can also happen due to file size restrictions for <code>POST</code> (5mb) ○ wrong path provided. It may be that the url contains space and if the case url has to be properly constructed.<code>{ }</code> ○ Metadata don't match with file contents ○ Lifecycles statuses don't follow
500	Server error. Internal server error	Incident in the server, code error, ...
501	Server error. Not Implemented	Imagine now that standard contract you need to implement says consumer can use <code>GET</code> and <code>PUT</code> on <code>/resource</code> but you only implement <code>GET</code> in your version 1 for whatever reasonable reason. In that case, returning a <code>501 Not Implemented</code> would be the perfect answer as the “standard” contract says that <code>PUT /whatever-resource</code> is possible but it has not been implemented by the actual provider.