What is a REST API, and what do HTTP methods and status codes mean?

3 November 2026 · 5 min read · blogs.iictp.org/what-is-a-rest-api

A REST API lets one program talk to another over HTTP by acting on named resources, such as orders or users, with a small set of standard methods. A client sends a request to a URL, and the server replies with a status code and, usually, a body. The conventions make APIs predictable for anyone who has used one before.

TL;DR The short version

- A REST API lets programs talk over HTTP by acting on named resources with standard methods such as GET and POST.
- Status codes group by family: 2xx success, 3xx redirection, 4xx client errors and 5xx server errors; 401 means not authenticated and 403 means not allowed.
- Good APIs use consistent names, the right status codes, clear error bodies and versioning so changes do not break callers.

Read online, listen to it, or change the text size: https://blogs.iictp.org/what-is-a-rest-api

What are the main HTTP methods?

- **GET** reads a resource and changes nothing.
- **POST** creates a resource or triggers an action.
- **PUT** replaces a resource with the one you send.
- **PATCH** changes part of a resource.
- **DELETE** removes a resource.

What does safe and idempotent mean?

A safe method does not change server state, as with GET. An idempotent method gives the same result when repeated, as with PUT and DELETE, so a client can retry it after a timeout without causing duplicates. POST is not idempotent by default, which is why retried payments need extra protection. See what is idempotency.

What do the status code families mean?

- **2xx** means success, for example 200 OK, 201 Created or 204 No Content.
- **3xx** means redirection.

- **4xx** means the client made a mistake, for example 400 Bad Request, 401 Unauthorized, 403 Forbidden, 404 Not Found or 429 Too Many Requests.
- **5xx** means the server failed, for example 500 Internal Server Error or 503 Service Unavailable.

What is the difference between 401 and 403?

A 401 says the request lacks valid credentials, so the caller is not authenticated. A 403 says the caller is known but is not allowed to do this, so they are authenticated but not authorised. Mixing them up confuses clients and hides real access problems.

What makes an API easy to use?

Consistent names, the right status codes, clear error bodies that say which field was wrong and why, and versioning so that changes do not break existing callers. Never return a 200 with an error hidden in the body, and never expose a stack trace to the caller.

Related credential: Associate Technical Practitioner. Explore the exam facts and syllabus.

Sources and further reading

1. IETF. RFC 9110, HTTP Semantics (2022). rfc-editor.org.
2. IETF. RFC 6585, Additional HTTP Status Codes (2012), which defines 429. rfc-editor.org.
3. Fielding, R. T. (2000). Architectural Styles and the Design of Network-based Software Architectures, chapter 5. University of California, Irvine.

Published by IICTP, an independent technical credential body, on 3 November 2026. The current version of this post is at https://blogs.iictp.org/what-is-a-rest-api. More guides: blogs.iictp.org.