CERTLABZ CERTIFICATION BOARD

Certified API Engineering Professional (CAEP)

Ten Sample Questions

These ten questions are written at the difficulty of the live exam. Six carry a figure, a data table, a configuration or a code listing; four are plain scenario questions, as on the live paper. Every one is a scenario a working API engineering professional would meet, and none of them appears in the live question bank.

The live exam is multistage adaptive: how you score on one stage selects the difficulty of the next, so no two candidates sit the same paper. It is a performance-based exam: each form can carry up to 5 to 12 performance-based questions alongside the multiple-choice questions, which means you build, sequence or complete something rather than only choosing from a list. Answers and full explanations follow each question here; the live exam does not disclose item-level answers.

Question 1 — Domain 1: API Foundations — DATA

Emma Wilson reviews how each HTTP method behaves when a call is retried.

Method	Idempotent?
GET	(to choose)
PUT	(to choose)
POST	(to choose)
DELETE	(to choose)

Table 1.1 HTTP methods to classify.

Which method is not idempotent, so a retry can create duplicates?

A. GET
B. PUT
C. POST
D. DELETE

Correct answer: C
GET, PUT and DELETE are idempotent because repeating the same call leaves the resource in the same state. POST is not idempotent, so a retried create can produce duplicate records. That is why safe retries and idempotency keys matter for POST.

Question 2 Domain 1: API Foundations CONFIG

James Carter inspects a create request and the response it returns.

```
POST /v1/orders HTTP/1.1
Content-Type: application/json

HTTP/1.1 2xx
Location: /v1/orders/1042
```

Listing 2.1 A create request and its response.

Which status code best fits the response for a new resource?

A 200 OK
B 201 Created
C 400 Bad Request
D 500 Server Error

Correct answer: B
A successful create returns 201 Created with a Location header pointing at the new resource. 200 OK does not signal creation, and 4xx or 5xx codes report client or server errors. The 201 code paired with Location is the correct pattern here.

Question 3 Domain 2: API Design DATA

Olivia Davis lists the query parameters offered on a collection endpoint.

Parameter	Purpose
limit	(to choose)
offset	(to choose)
sort	(to choose)
filter	(to choose)

Table 3.1 Query parameters to match.

Which parameter caps how many records a single page returns?

A limit
B offset
C sort

D filter

Correct answer: A
The limit parameter caps how many records a single page returns, while offset chooses where the page starts. Sort orders the results and filter narrows them, but neither controls page size. So limit is the parameter that bounds the page.

Question 4 Domain 3: Authentication, Authorization and Security CONFIG

David Brown configures a token request for a background job that runs with no user present.

```
POST /oauth/token HTTP/1.1
Content-Type: application/x-www-form-urlencoded

grant_type=?
scope=reports.read
```

Listing 4.1 The token request to complete.

Which grant type belongs in this server to server request?

A Authorization code
B Implicit grant
C Device code flow
D Client creds

Correct answer: D
A background job with no user present authenticates as itself, which is what the client credentials grant is for. Authorization code and implicit grants involve a user in a browser, and device code targets input-limited devices. So client credentials fits a server to server call.

Question 5 Domain 4: Beyond REST DATA

Ava Thompson compares ways to deliver events to a subscriber.

Approach	Delivers push?
REST polling	(to choose)
Webhooks	(to choose)
SOAP call	(to choose)
FTP pull	(to choose)

Table 5.1 Delivery approaches to compare.

Which approach pushes events to a subscriber without client polling?

A REST polling
B Webhooks

C SOAP call

D FTP pull

Correct answer: B
Webhooks push an event to a subscriber endpoint as soon as it happens, with no client polling. REST polling, SOAP calls and FTP pulls all require the client to ask repeatedly. So webhooks deliver server-driven events efficiently.

Question 6

Domain 5: Quality, Testing and Documentation

DATA

Sarah Miller sorts project artifacts by whether tools can read them.

Artifact	Machine-readable?
OpenAPI spec	(to choose)
Marketing page	(to choose)
Release blog	(to choose)
Changelog note	(to choose)

Table 6.1 Artifacts to classify.

Which artifact is the machine-readable contract clients can generate code from?

A Changelog note

B Marketing page

C OpenAPI spec

D Release blog

Correct answer: C
An OpenAPI spec is a machine-readable contract, so clients can generate code and tests from it directly. A marketing page, release blog or changelog note are written for people, not tools. So the OpenAPI spec is the artifact that defines the contract.

Question 7

Domain 3: Authentication, Authorization and Security

Liam Anderson needs somewhere to keep the API keys a service uses to call partners. What is the soundest practice?

A Store in a vault

B Hard-code in source

C Email to the team

D Log on every call

Correct answer: A
API keys and secrets belong in a managed secrets store with tight access control, not in source, email or logs. Hard-coding or logging a key exposes it to anyone who can read the code or the logs. So storing secrets in a vault is the soundest practice.

Question 8 Domain 6: Gateways, Performance and Operations

An API is overwhelmed by sudden bursts of traffic from a few clients. Which control best protects the service?

A. Add more endpoints today
B. Ignore the traffic
C. Cache every write
D. Rate limit clients

Correct answer: D
Rate limiting caps how many requests each client may send in a window, protecting the service during bursts. Removing limits or ignoring the load invites overload, and caching writes does not help write-heavy bursts. So rate limiting clients is the right control.

Question 9 Domain 4: Beyond REST

A trading dashboard needs prices the instant they change, without hammering the server with repeated requests. Which fits best?

A. Nightly batch file
B. WebSockets
C. Daily CSV export
D. Manual refresh

Correct answer: B
WebSockets keep an open connection so the server can push price updates the moment they change, with no polling. Batch files, CSV exports and manual refresh all add delay. So WebSockets fit a real-time update need.

Question 10 Domain 6: Gateways, Performance and Operations

The same read-only GET request is served many times a second and keeps regenerating an identical payload. What reduces the load?

A. Cache with ETags
B. Disable all caching
C. Slow every response
D. Drop the responses

Correct answer: A
ETags let a client revalidate and reuse a cached response, so repeated identical GET requests avoid regenerating the payload. Disabling caching or slowing responses raises load and latency, and dropping responses breaks the API. So caching with ETags is the fix.

CAEP-001. https://certlabz.com/certifications/certified-api-engineering-professional.html