

3.2 API routes and server validation

Name: _____ Class: _____ Date: _____

Total: 15 marks

KEY WORDS route • validate input • status code
• request method

Objective

Design predictable API routes and validate every request on the server.

You must be able to:

- distinguish GET from POST
- explain routes, input validation, and status codes
- recommend clear responses to invalid requests

1 Key knowledge and worked examples

■ Make the route match the action

GET /courses asks for a course list. POST /enrolments asks the server to create an enrolment. A **route** connects the URL and method to the code that handles it.

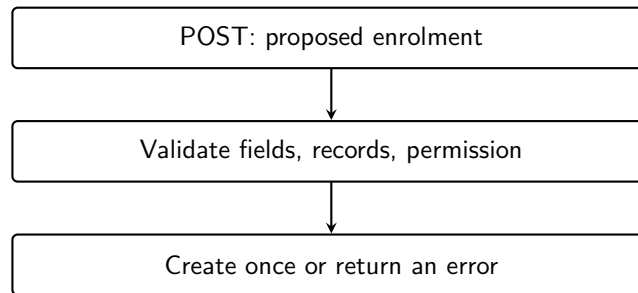
The server must **validate input** : check required values, types, and ranges. It can return 200 for success, 400 for a bad request, or 404 for something that does not exist.

■ Key knowledge: Validate a create request

Client checks help users, but server checks enforce the contract. Required fields, types, ranges and referenced records all need validation before writing data.

■ Worked example: Validate a create request

A POST request to /enrolments supplies a student id and course id. The server checks that each id has the expected type, that both records exist, and that the requester may enrol that student. Missing data produces a clear error rather than a partial database row. GET requests for a list read data; they should not secretly create an enrolment. A successful create response identifies the new resource.



2 Practice

2.1 State which method is normally used to retrieve a list of courses. [1]

2.2 State what status code 404 means. [1]

2.3 Explain why server validation must check that an age value is a number within an allowed range. [2]

3 Exam-style questions

3.1 A client sends `POST /enrolments` with no course identifier. Explain an appropriate server response. [4]

3.2 Explain why predictable route names and methods make an API easier for another developer to use. [4]

3.3 A create route accepts a missing course id and a student id containing arbitrary text. State two validation checks and explain when they must occur. [3]
