

HTTP and Web Architecture

Servlet is a java class that manage http request and responses.

HTTP methods (verbs) → Used to tell the server actually what kind of action we want to take.

- ① GET: you just want to read the informatⁿ.
- ② POST: You are sending new data to be processed.
- ③ PUT: Replacing something that already exists
- ④ DELETE: You are removing something.

Job of Servlet:

- ① Listen for request
- ② check for which method was used.
- ③ Run specific java code for that method.
- ④ Sends a response back to you.

Post contains data in form of packages, meaning data is hidden inside the body of the request.

Idempotency

if doing a action changes nothing it is then

Idempotent. → Turning off a already off button.
But if something does end up changing it is non idempotent. → Increasing the volume.

GET → Idempotent POST → non-Idempotent
PUT/PATCH → "
↓ ↘
updates the whole record changes the specific targetted data only. like: changing Age.

Understanding Status Codes (ANSWER)

Everytime servlet finishes its job it sends back a 3 digit response.

200 → OK (data found)

201 → created (User Saved)

400 → Bad request (Missing data)

404 → Not found (URL is wrong)

500 → Internal server error (Code threw an exception)

400: Bad request is found when user either leaves the field blank or sends in data of diff. data type. Like integer in place of string.

409: Conflict happens when the request is completely formatted but it clashes with what's already on the server.

Eg → Signing up with an email that has already been signed up with.

Request & Response headers

Headers are the metadata of the request.
↳ data about the data

In servlet these are key value pairs -

① Request headers (The client's "context")

These tell the servlet here is what I'm sending and here's what I can handle.

- Content-Type: Tells the servlet how to parse the body

Eg → application/json means the body is a JSON object.

- Authorization: The security badge

Eg → Bearer (JWT-Token)

Servlet filter reads this error, validates token, and decides if the user is allowed to enter.

- Accept: The "language" preference

Eg → text/html (Browser wants webpage)
application/json (Mobile app wants raw data)

② Response headers (The server's "Instructions")

These tell the browser: here's how you should treat what I just sent you.

- Content-Type: tells the browser how to render the data
- Location: Used for redirects
- Cache-Control: The "Expiration date".

→ HTTP is stateless, it doesn't remember anything after giving a response, then how does Servlet remember who you are, every time you log-in? We use headers to carry who we are and what we are sending.

