

Front End Software Development

Front End Technologies (weeks 7 - 12)

Week 06

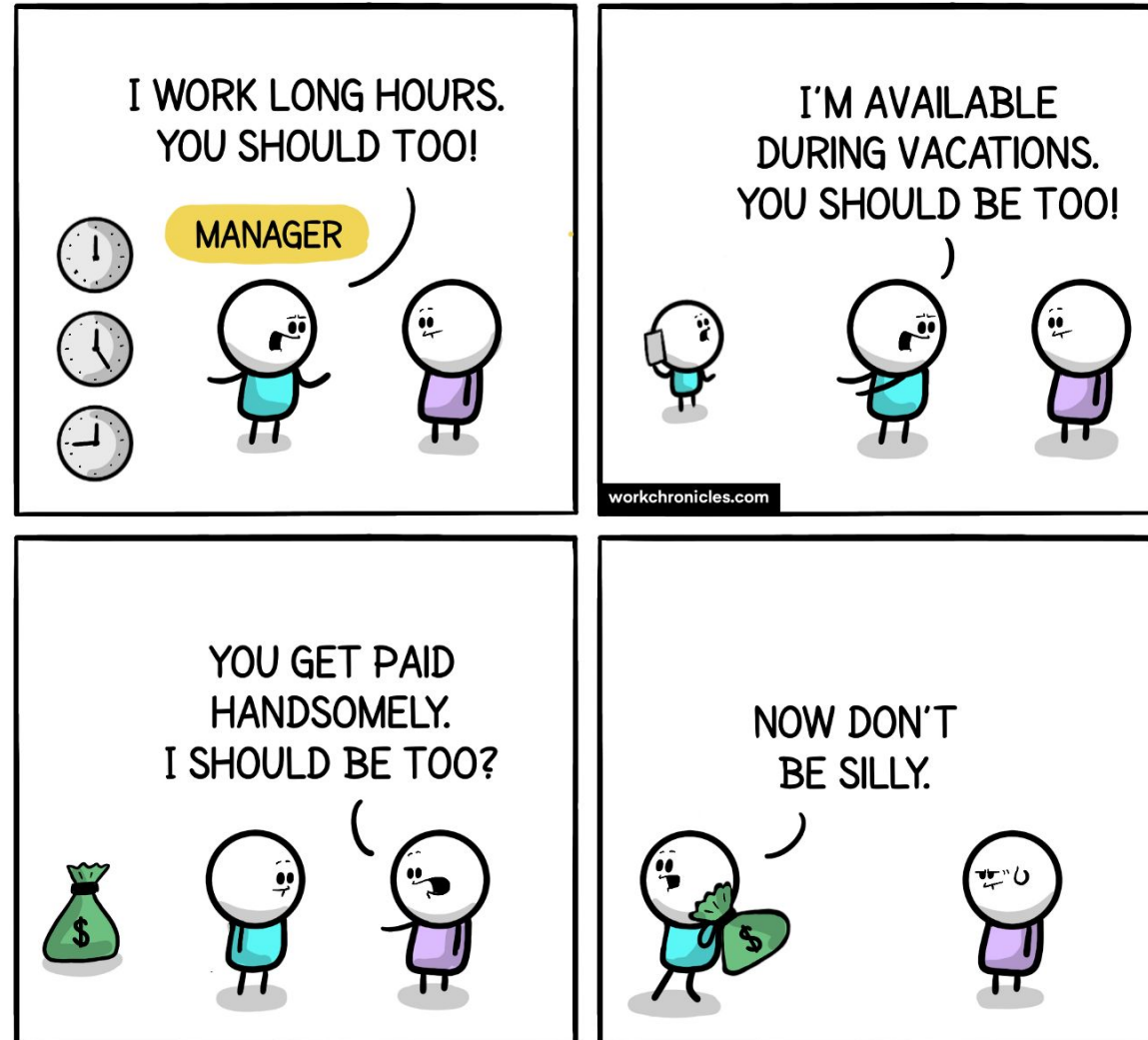


Agenda

- Questions
- Final Project
 - APIs
 - GET
 - PUT / POST
 - DELETE
 - Troubleshooting



Questions



Hello. I make comics about work.

Join r/workchronicles or follow on Instagram/Twitter/FB

Work Chronicles
workchronicles.com

Final Project

Final Project Requirements

- Data interaction (**CRUD**)
 - If you can use an existing API, use **AJAX** to interact with it. However, you do not have to use an API.
 - If you do not use an API, store the entities you will create, read, update, and delete in an array.
- Application should have at least 1 entity per team member
- Use Bootstrap and CSS to style your project

When you delete a block of code
that you thought was useless



APIs

(Client Tools / Installation)

- Postman
 - <http://www.getpostman.com/>
- SoapUI
 - <https://www.soapui.org/tools/soapui/>
 - Also supports legacy SOAP web services
- Alternatives
 - CURL
 - Standard* on Linux, Mac, UNIX, Windows 1706+
 - Older versions of windows:
 - <https://curl.haxx.se/windows/>
 - Chrome / Firefox
 - JSONVue / JSONView

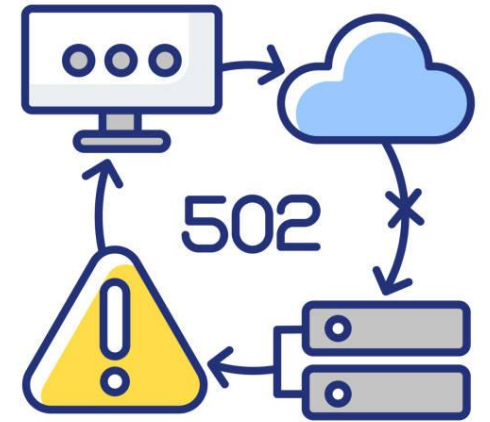


APIs

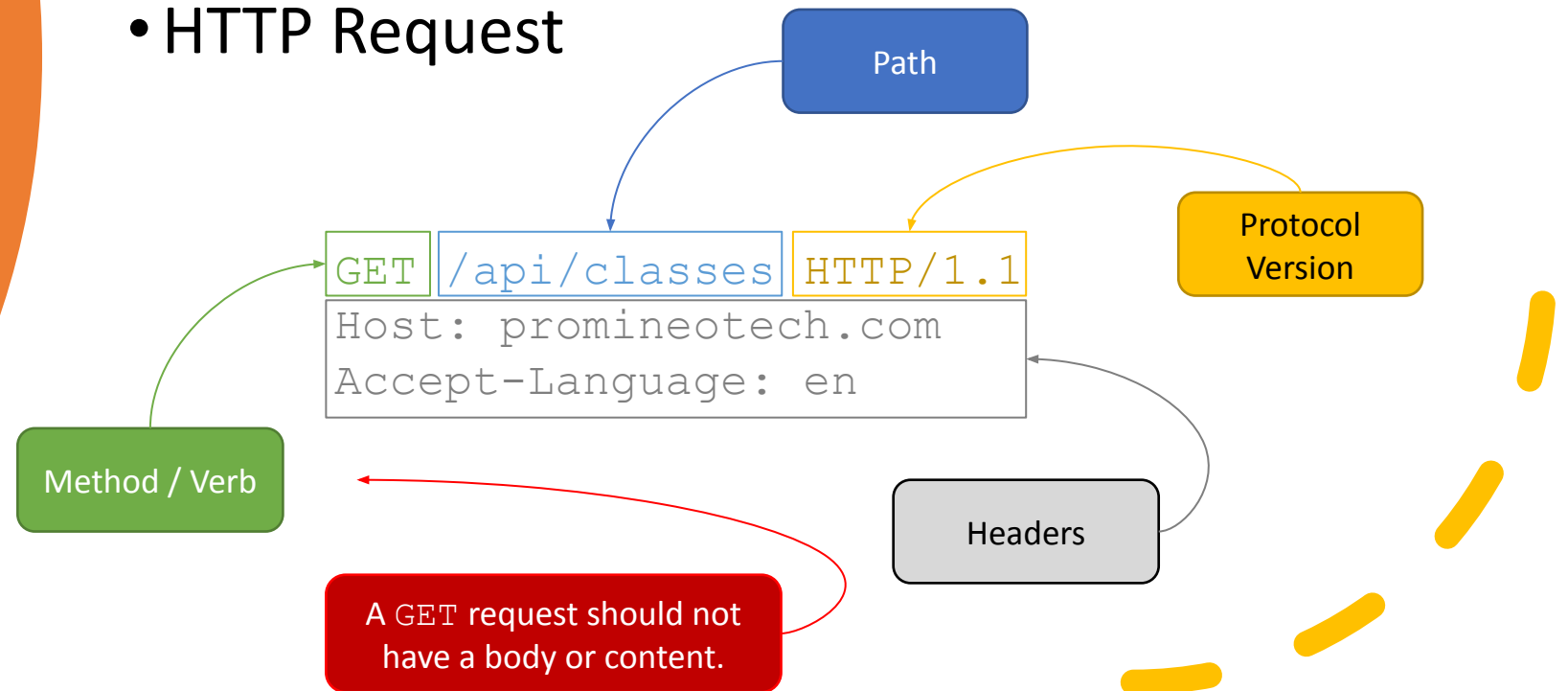
(GET)

- HTTP Methods / Verbs

- POST = Create
- GET = Read
- PUT = Update
- DELETE = Delete



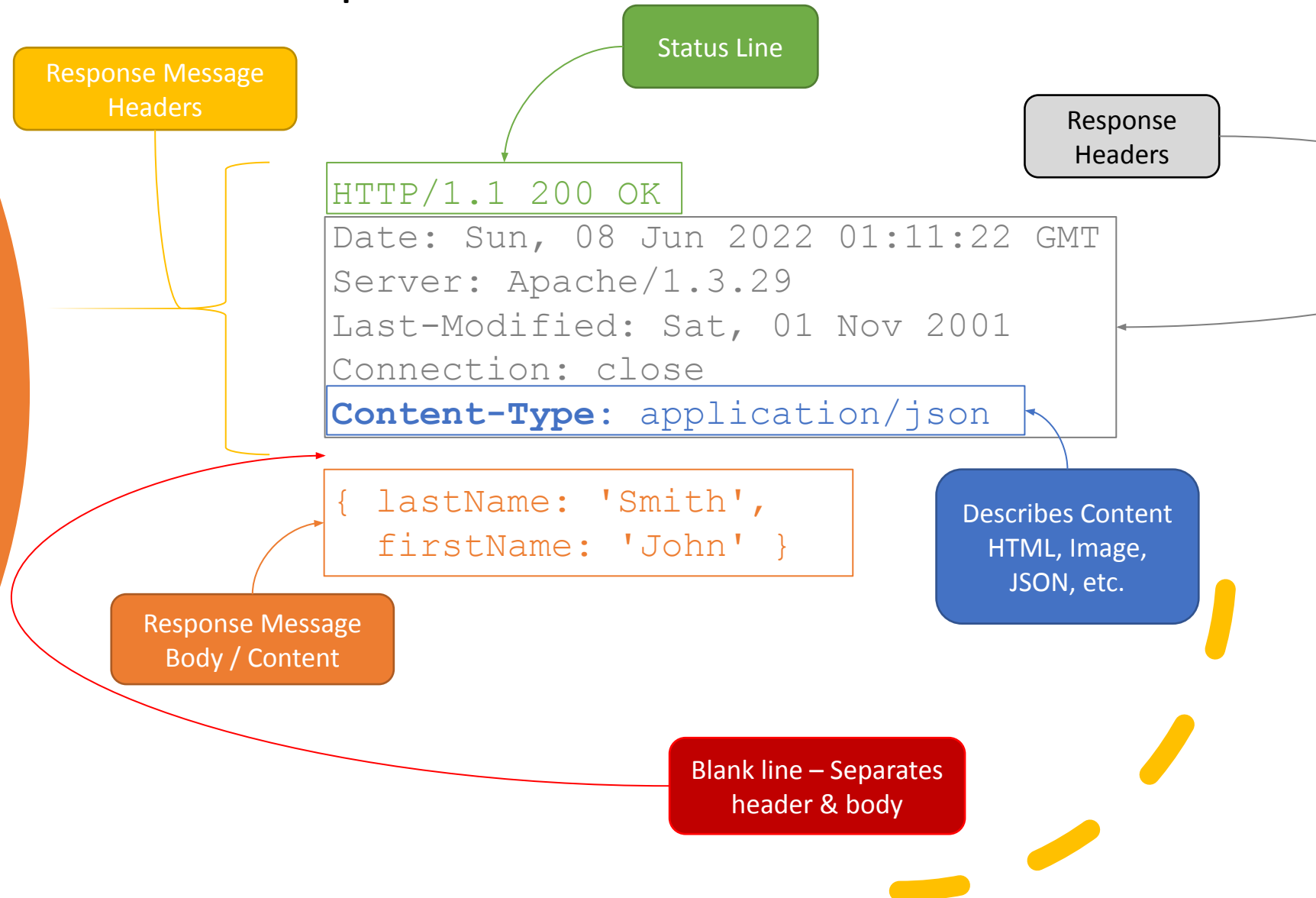
- HTTP Request



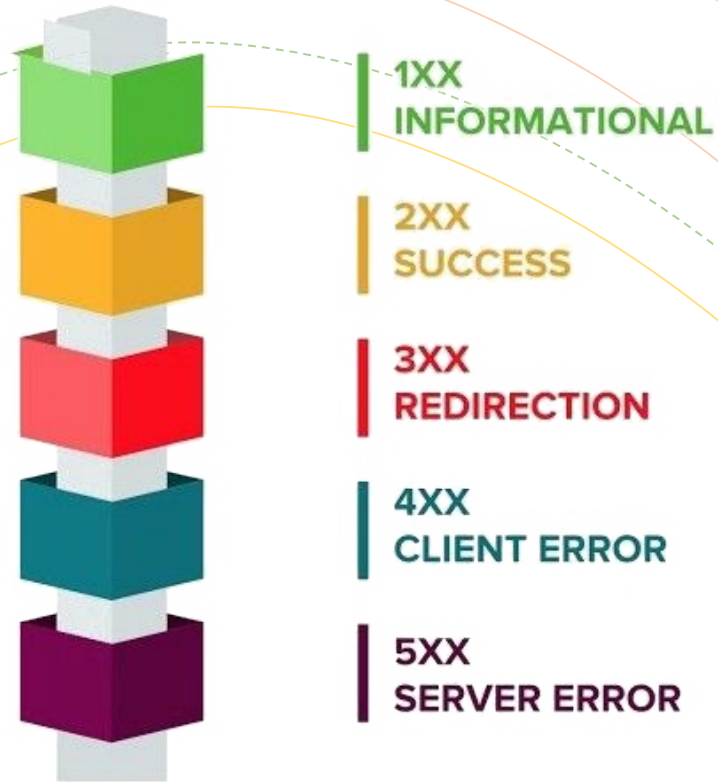
APIs

(HTTP Response)

• HTTP Response



APIs (HTTP Response / Status Codes)



Level 200 (Success)

- 200 : OK
- 201 : Created
- 203 : Non-Authoritative Information
- 204 : No Content

Level 400 (Client Error)

- 400 : Bad Request
- 401 : Unauthorized
- 403 : Forbidden
- 404 : Not Found
- 409 : Conflict

Level 500 (Server Error)

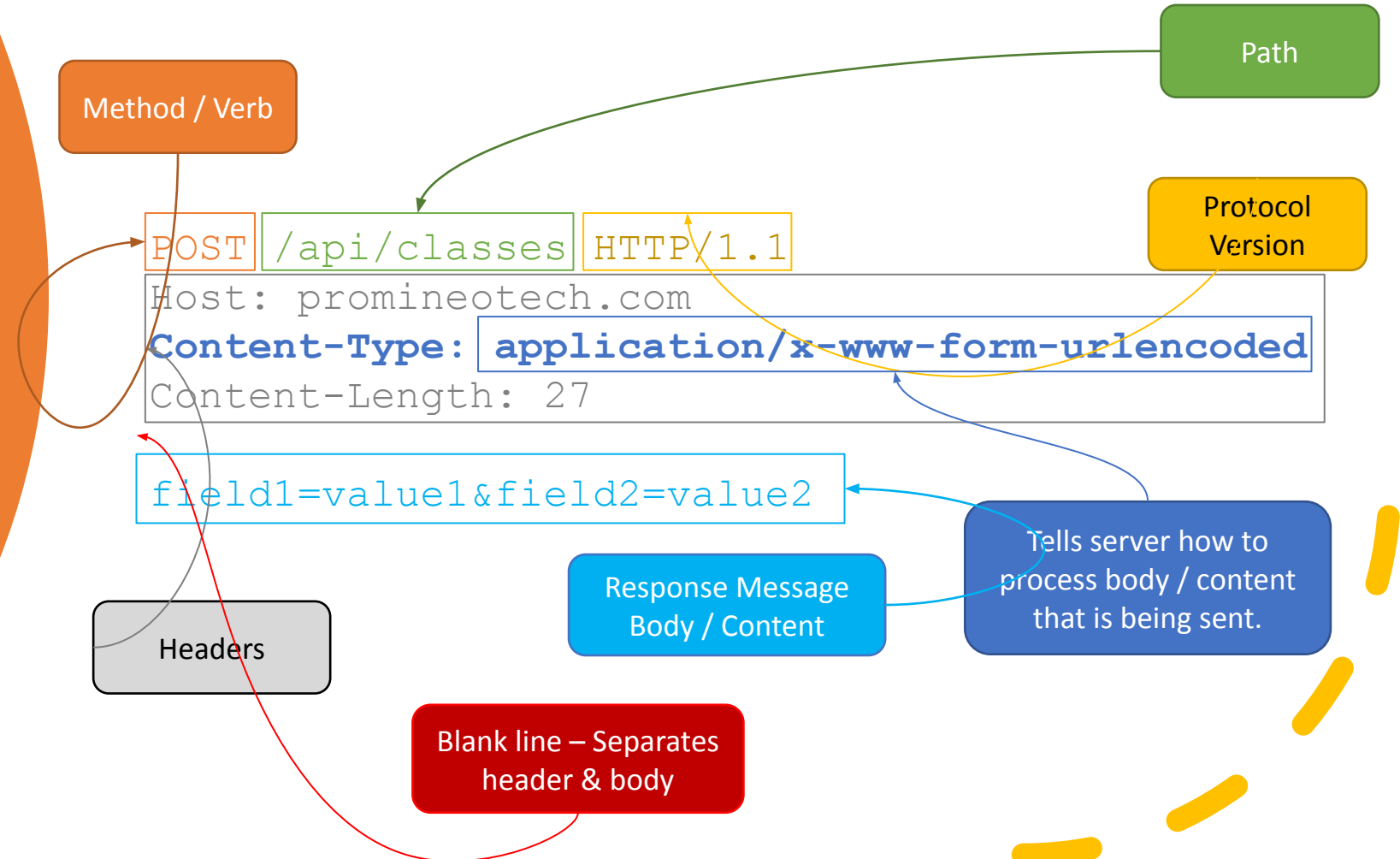
- 500 : Internal Server Error
- 501 : Not Implemented
- 502 : Bad Gateway
- 503 : Service Unavailable
- 504 : Gateway timeout

APIs

(PUT/POST)

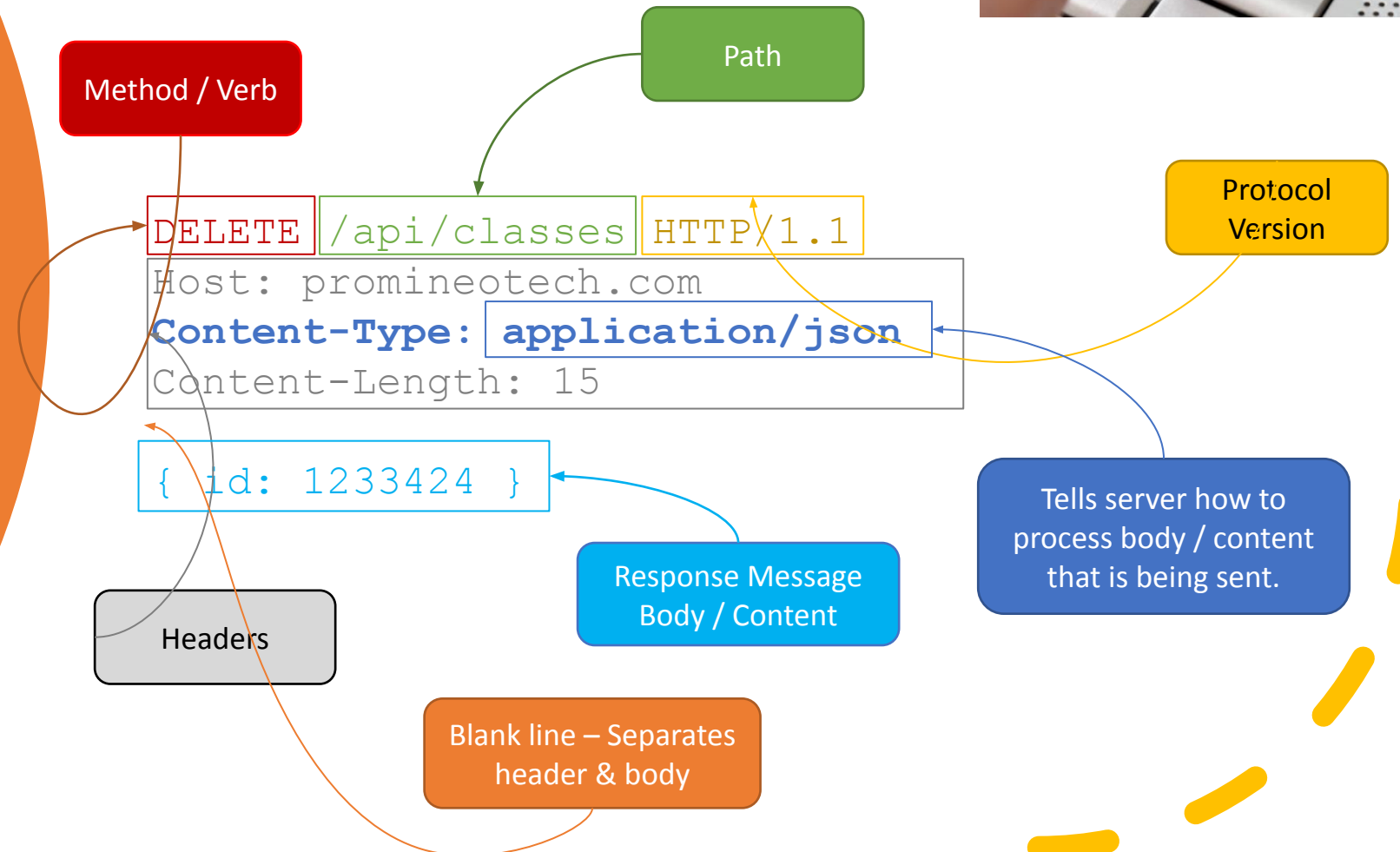
- POST = Create
- PUT = Update

TIP: Easy way to remember,
PUT vs POST, is "U" is in both
UPDATE and PUT.
A typical interview question...



APIs (DELETE)

- DELETE = Delete
 - May / may not contain body



APIs

(SUMMARY)

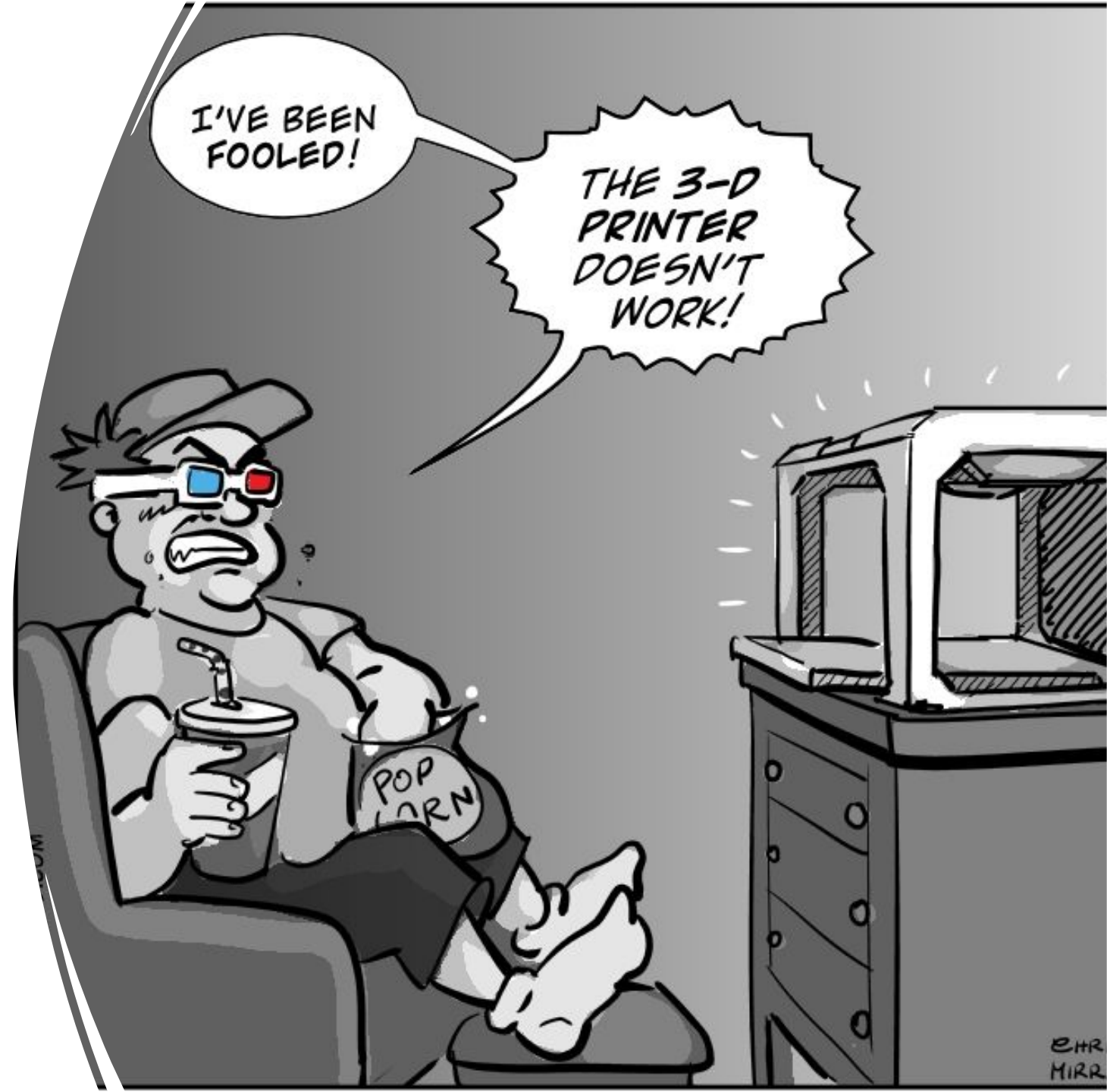
SAFE METHODS NO ACTION ON SERVER	{	GET	HTTP/1.1 MUST IMPLEMENT THIS METHOD
		HEAD	INSPECT RESOURCE HEADERS
MESSAGE WITH BODY SEND DATA TO SERVER	{	PUT	DEPOSIT DATA ON SERVER — INVERSE OF GET
		POST	SEND INPUT DATA FOR PROCESSING
		PATCH	PARTIALLY MODIFY A RESOURCE
		TRACE	ECHO BACK RECEIVED MESSAGE
		OPTIONS	SERVER CAPABILITIES
		DELETE	DELETE A RESOURCE — NOT GUARANTEED

APIs

(Resources / Optional)

Optional API resources:

- <https://mockapi.io/>
- <https://crudcrud.com/>
- <https://github.com/ripienaar/free-for-dev#apis-data-and-ml>
- <https://thecatapi.com/>
- <https://jsonplaceholder.typicode.com/>
- <https://learning.postman.com/docs/designing-and-developing-your-api/mocking-data/setting-up-mock/>

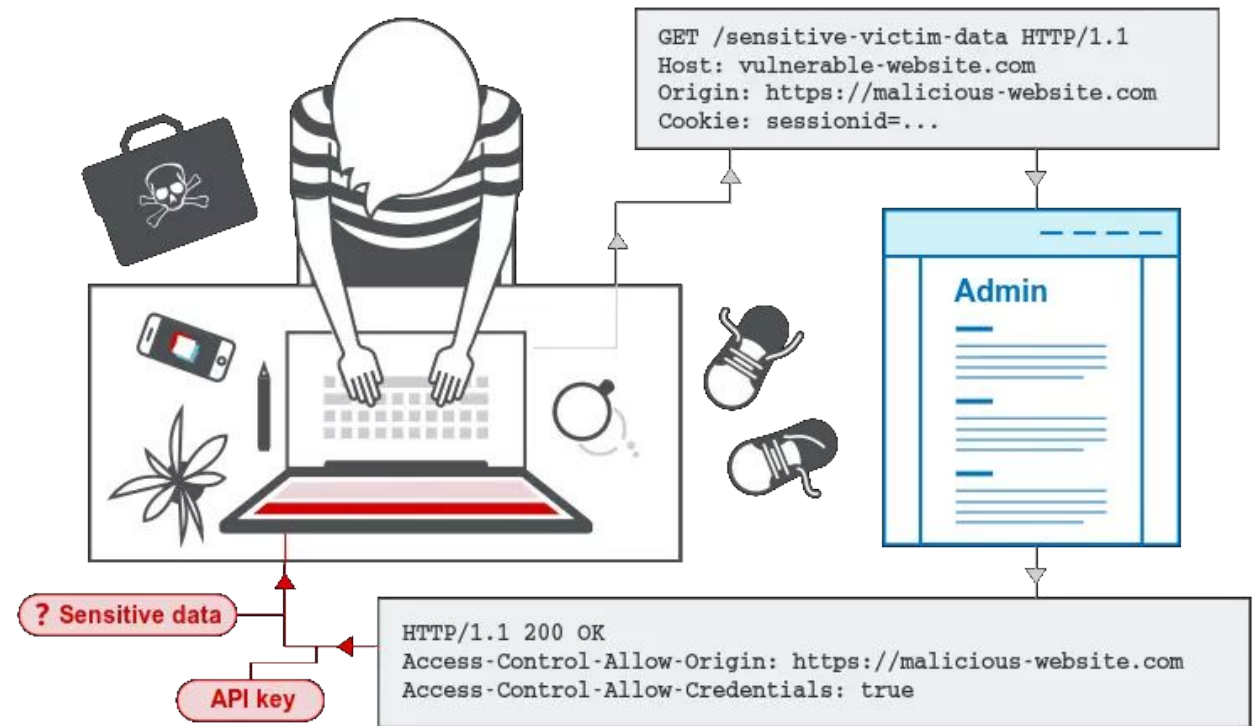


Troubleshooting

- CORS Policy Error
 - Try running your app on a live server.

```
npm install live-server -g
```
 - Navigate to your project directory via command line, run:

```
live-server
```
- *crudcrud.com*
 - On update, put unique id in **path/** instead of in body { } of request.





DEMO

Working with APIs