

re:coded

Workshop Agenda

Intro to Backend

Node and Express

Templating engine

REST API

Authintication

Intro to CRUD in MySQL

Sequelize and complex SQL

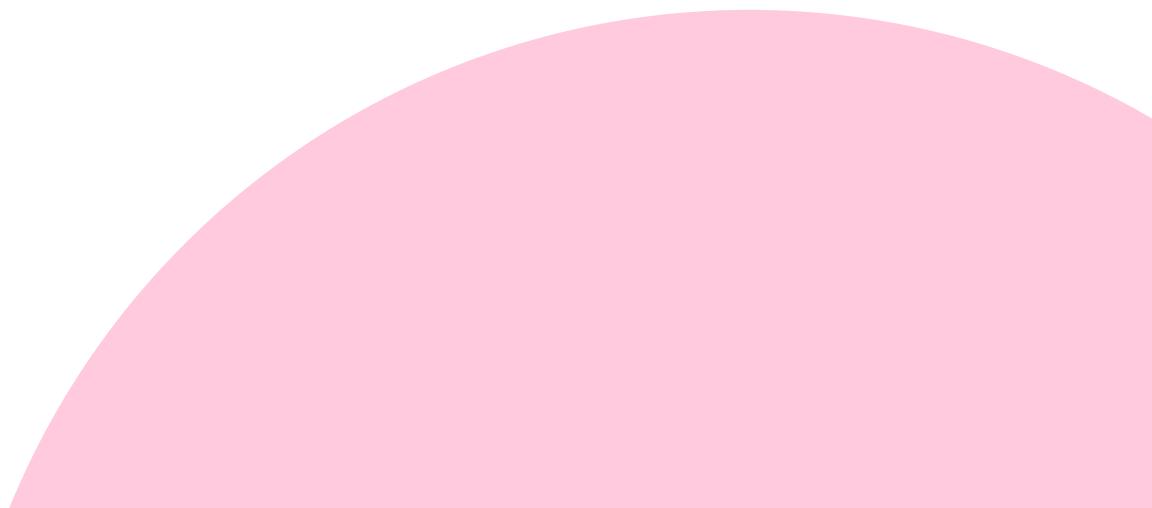
Today's Agenda

Intro to Backend

General web concept

Node JS and Express

Some ground rules

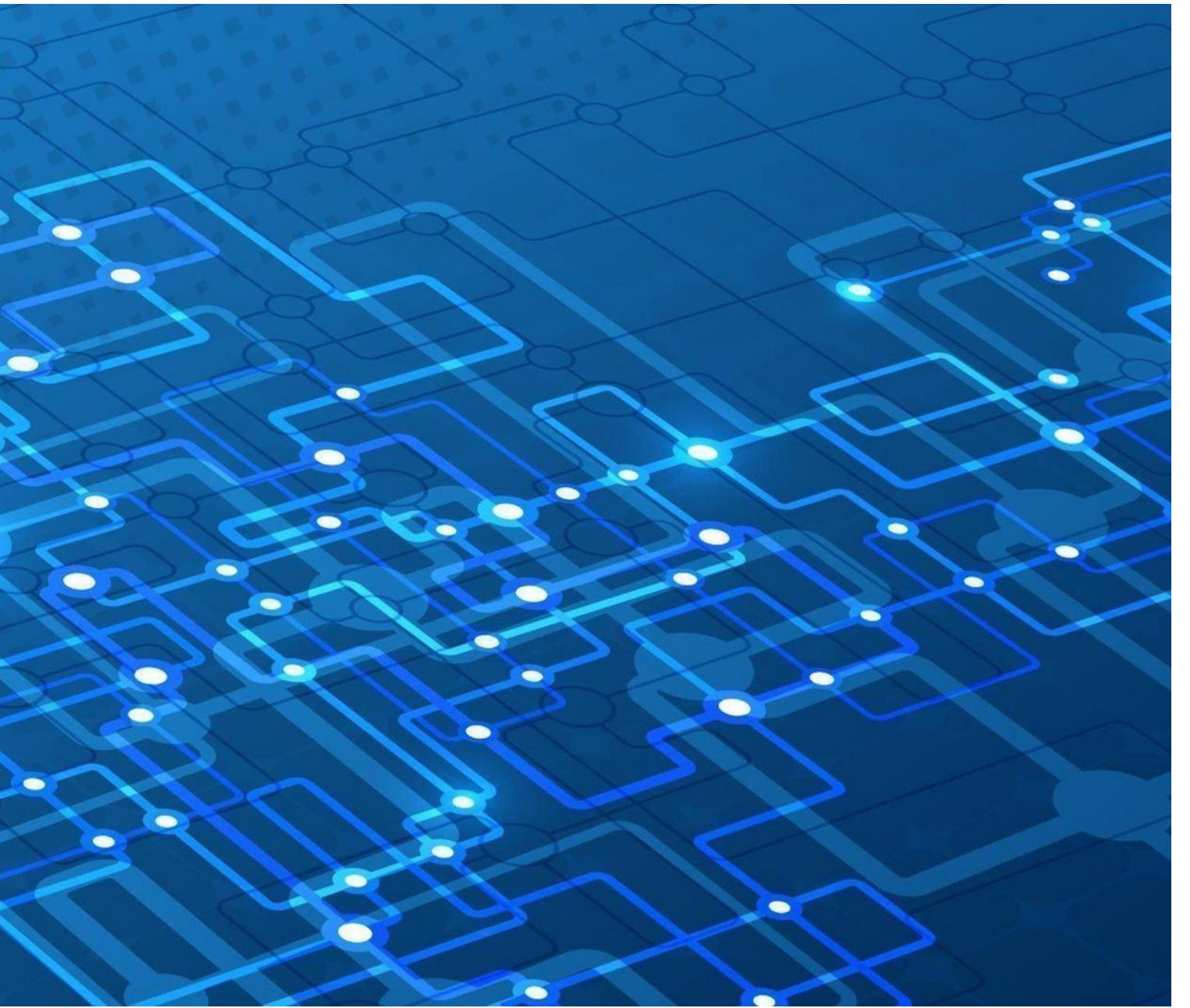


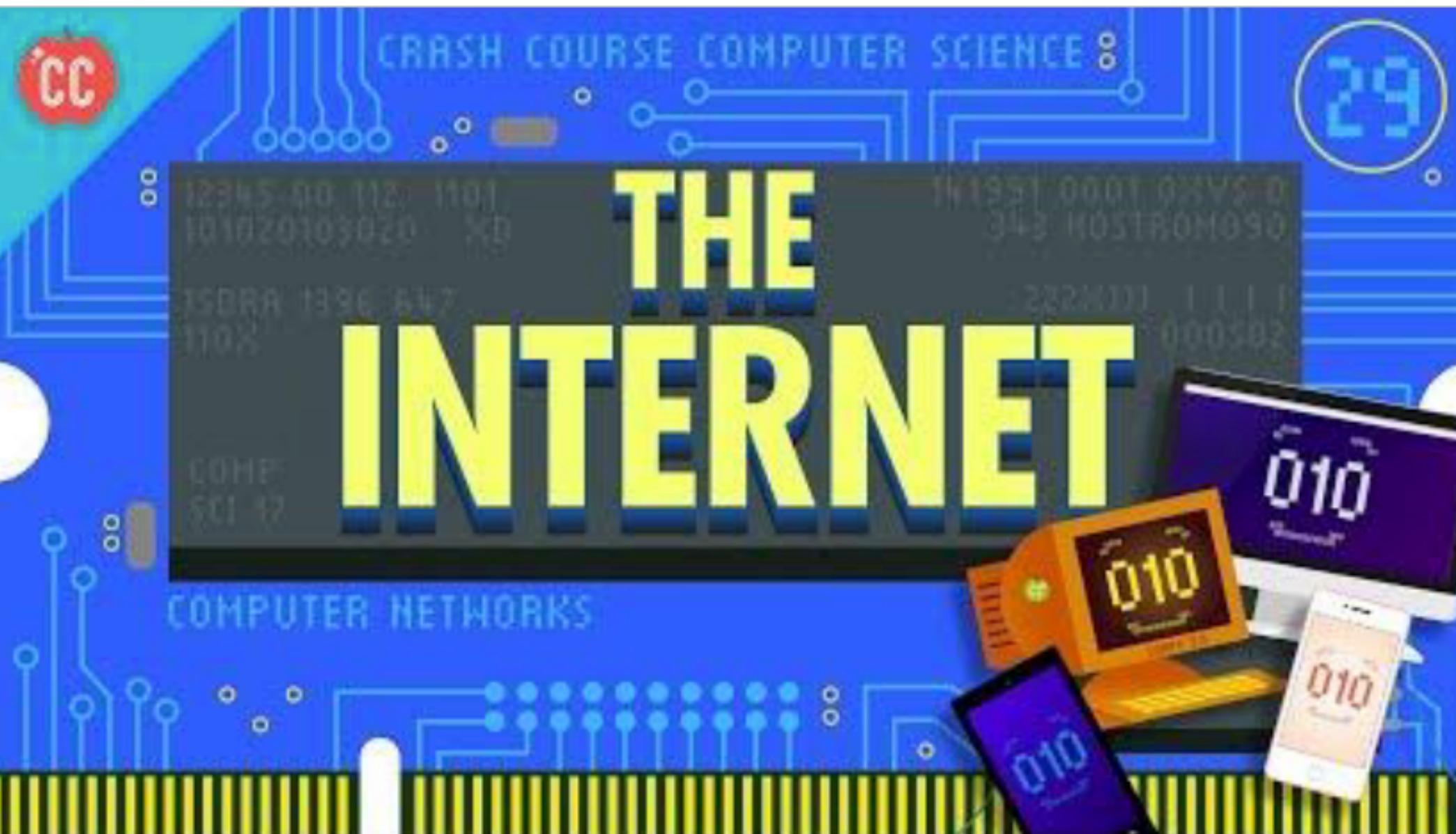
workshop

anatomy

:The internet

What happens when you click
a link?





[HTTPS://YOUTU.BE/AEAKRQ3SPW8](https://youtu.be/AEAKRQ3SPW8)

DOMAIN STRUCTURE

**TOP LEVEL
DOMAINS**

.org .gov .net .com .edu .uk ...

**SECOND LEVEL
DOMAINS**

google.com dftba.com

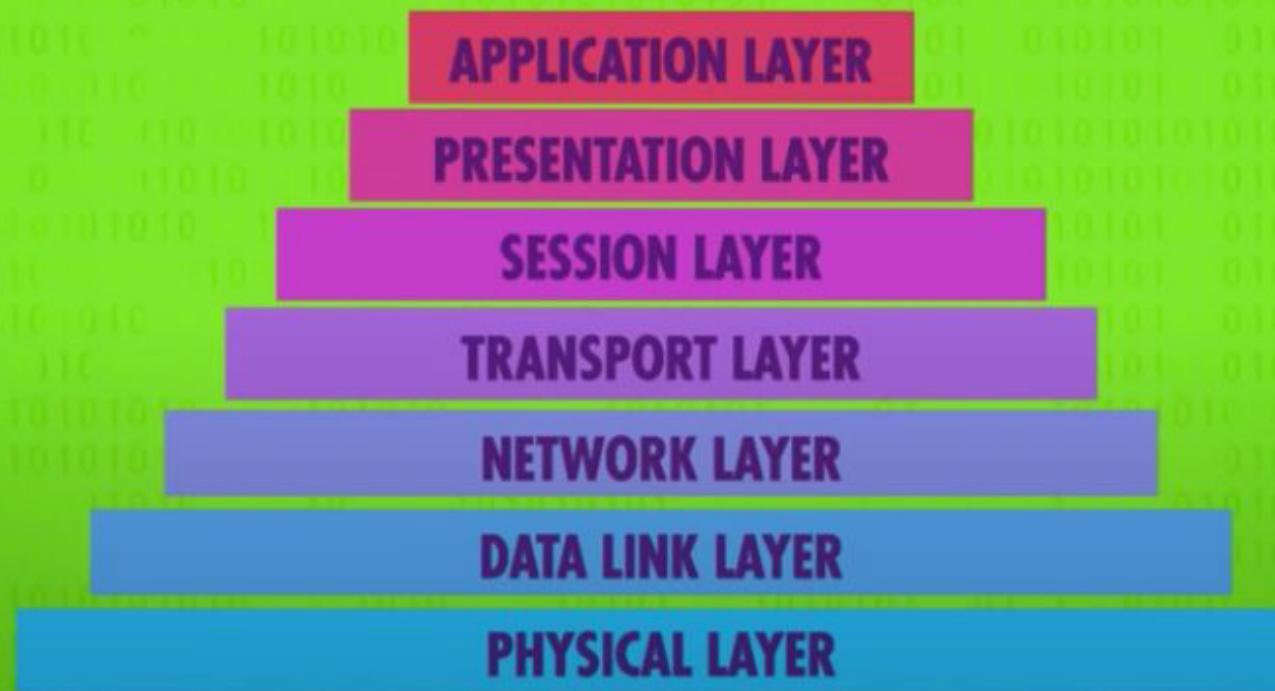
**SUB-DOMAIN
OF PARENT**

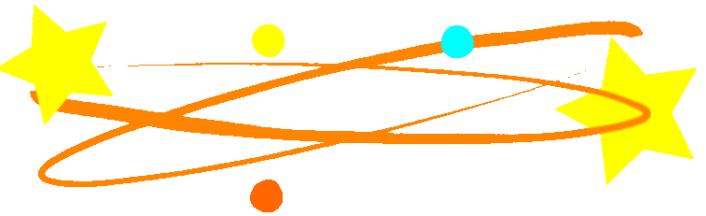
drive.google.com images.google.com store.dftba.com



LAYERS OF THE OSI MODEL

(OPEN SYSTEM INTERCONNECTION)

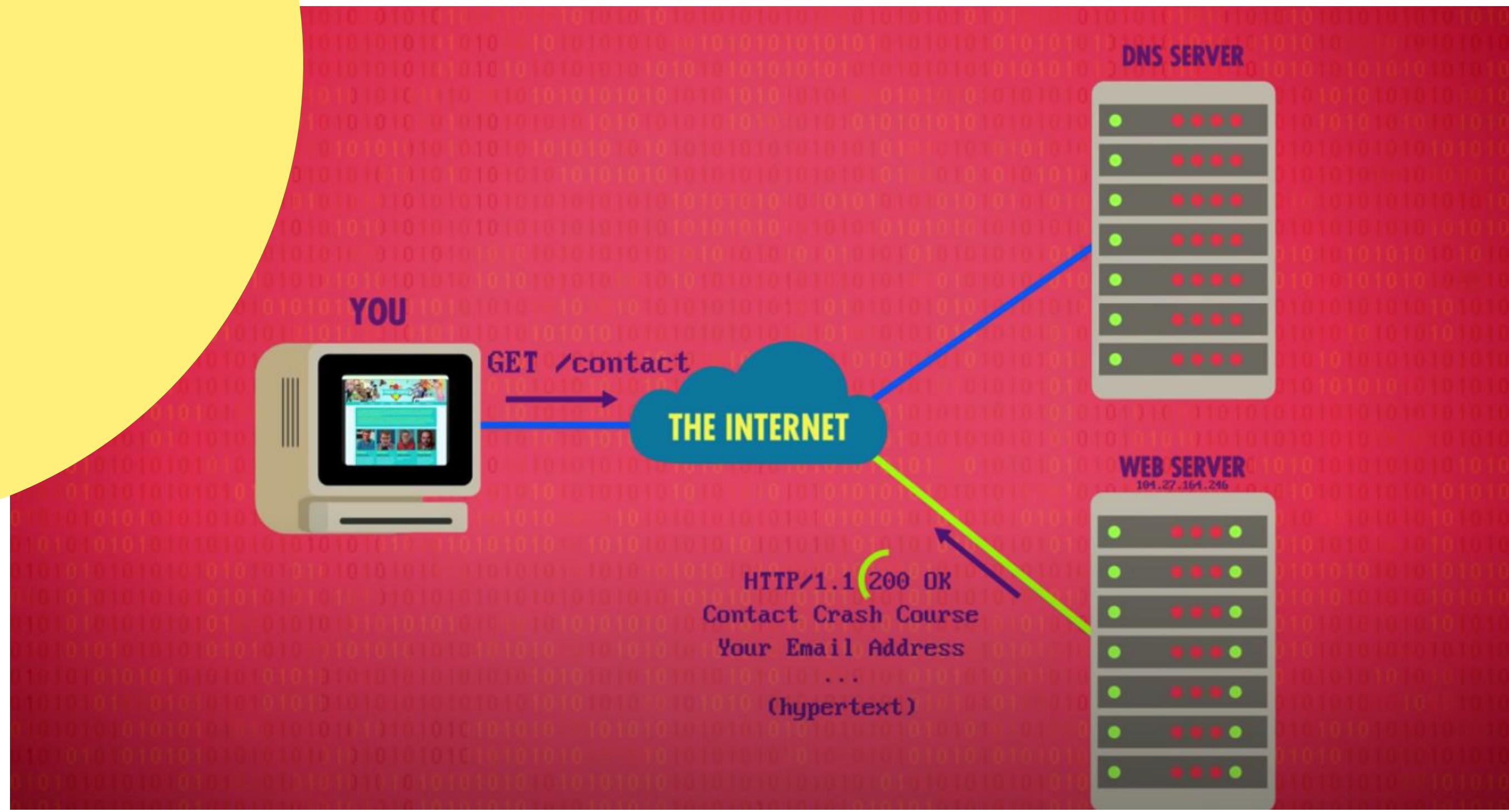




Common TCP/IP Ports

- 80: HTTP
- 443: HTTPS
- 53: DNS
- 25: SMTP
- 20/21: FTP





Anatomy of a Uniform Resource Locator

`https://www.google.com:443/feeling/lucky?q=easter+eggs&r=1#first`

Anatomy of a Uniform Resource Locator

`https://www.google.com:443/feeling/lucky?q=easter+eggs&r=1#first`

Scheme
Host
Port
Path
Query
Fragment

Anatomy of an HTTP Request

```
POST / HTTP/2
Content-Type: Application/JSON
{ "awesome": "yes" }
```

HTTP Verb
Request Path
HTTP Version
Header
Body

```
GET / HTTP/2
Host: www.google.com
User-Agent: curl/7.64.1
Accept: */*
```



```
POST /posts/upvotes HTTP/1.1
Host: localhost:3000
User-Agent: curl/7.64.1
Accept: /*
Content-Type: application/json
Content-Length: 18

{"upvoted": "true"}
```



HTTP Status Codes

- 1xx informational response
- 2xx success
- 3xx redirection
- 4xx client errors
- 5xx server errors

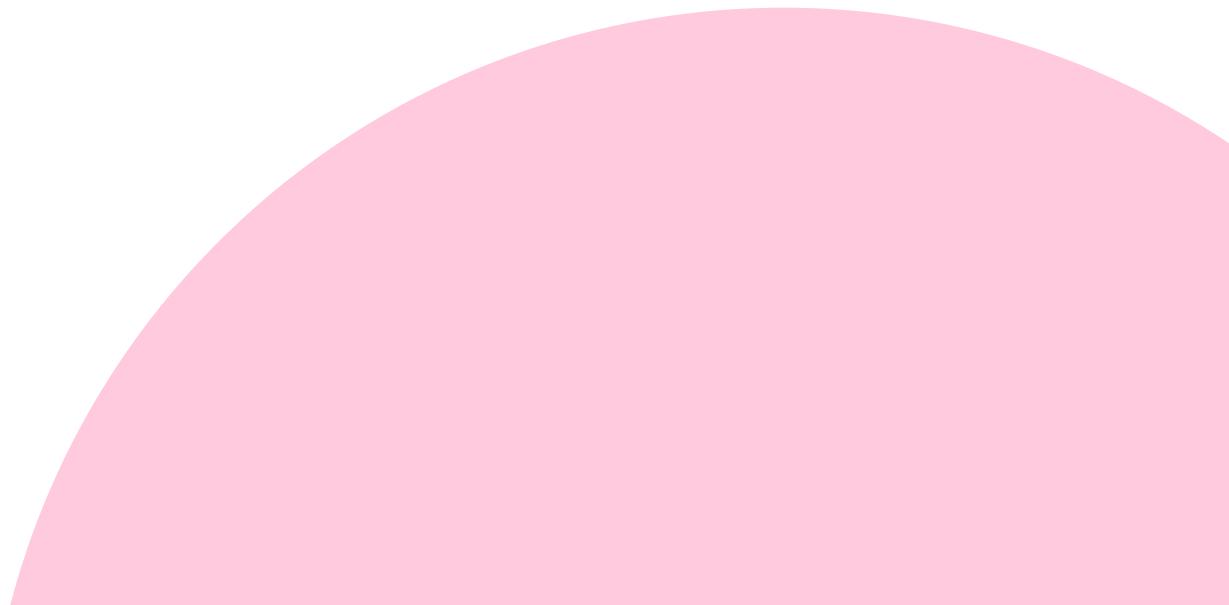
More details: https://en.wikipedia.org/wiki/List_of_HTTP_status_codes

Common HTTP Headers

- Accept
- Content-Type
- Content-Length
- Cookie
- Cache-Control

Common Content Types

- application/x-www-form-urlencoded
- multipart/form-data
- text/plain
- application/octet-stream
- text/html
- text/javascript
- image/jpeg



**Why do we need software to
run on servers? Why not do
everything in JS?**



See you tomorrow