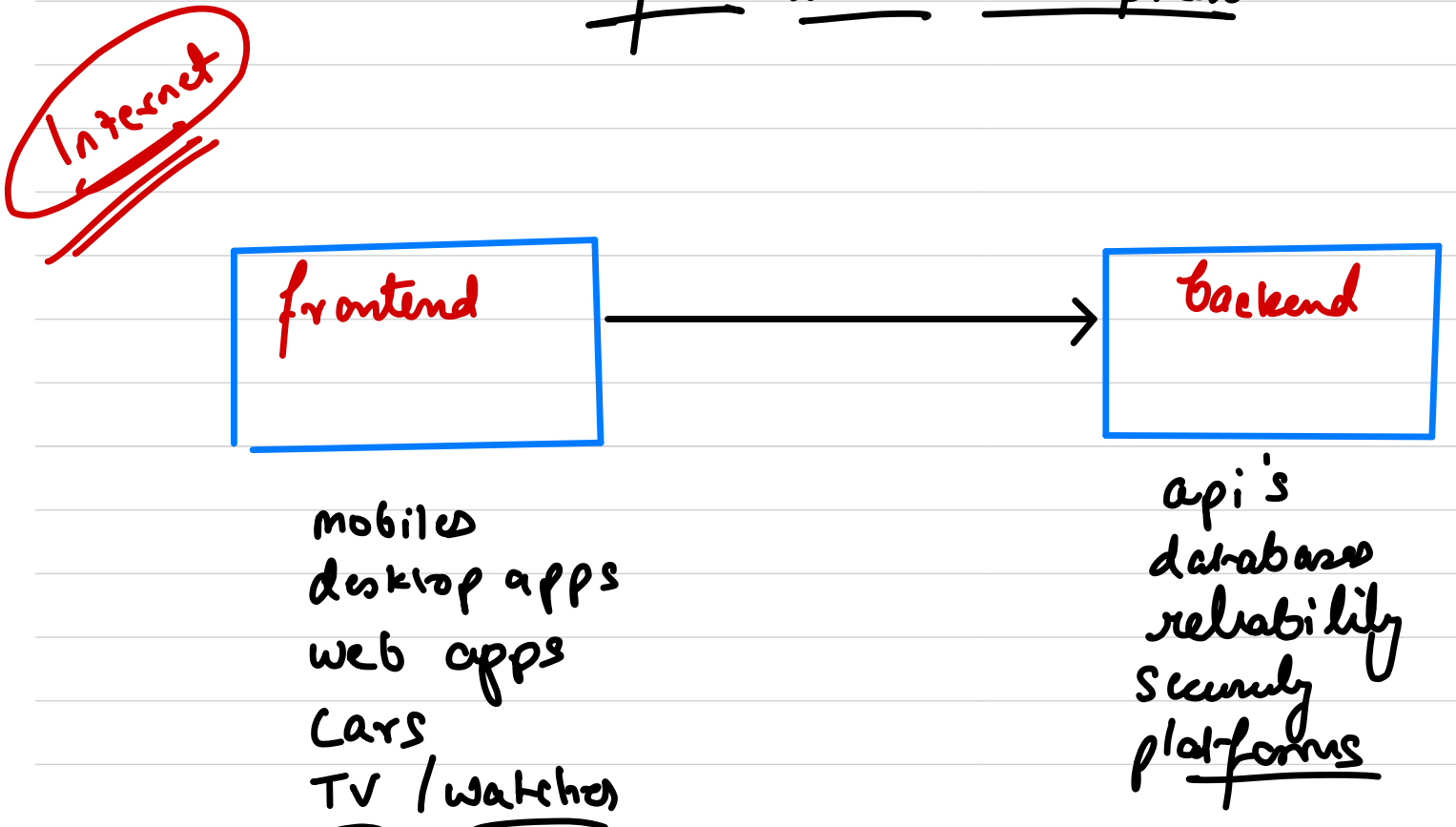
Full stack development



→ CLI → command line

linux/mac

→ GUI

x x

→ fast

→ server side

→ easier to automate

Shell

Command interpreter that helps us to interface with computer system

→ system calls → fork

Shell → Application → kernel

How Internet works ??

Internet is a system of globally interconnected devices.

2016-17

→ Approx 4.5 billion+ users.

> 50% of the world population

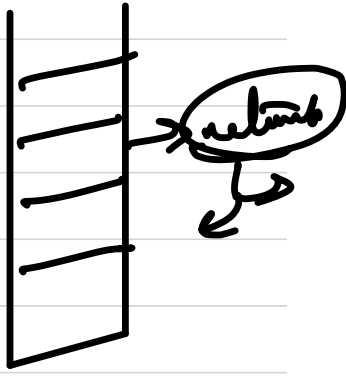
1.5 billion+ websites

10x increase in the internet usage from 1999 to 2013

→ [www.google.com] → ip address

DNS (Domain Name Server)

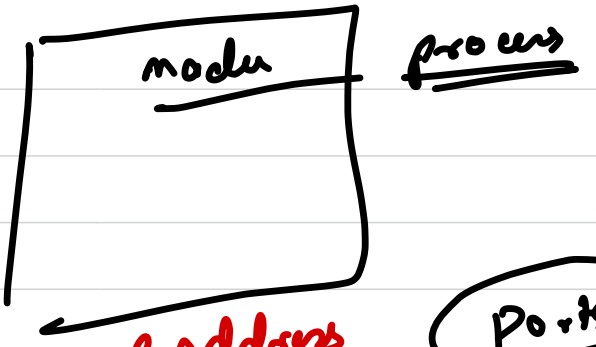
↳ phonebook of internet



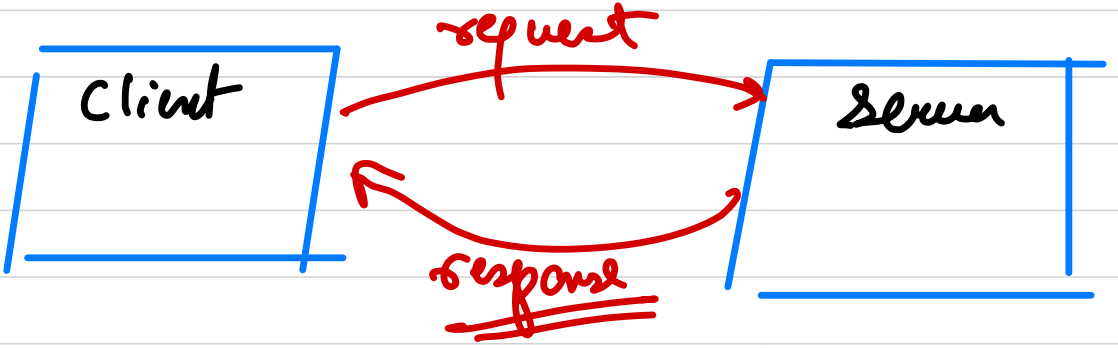
ISP (internet service provider)

↳ router

Datacenter



packets

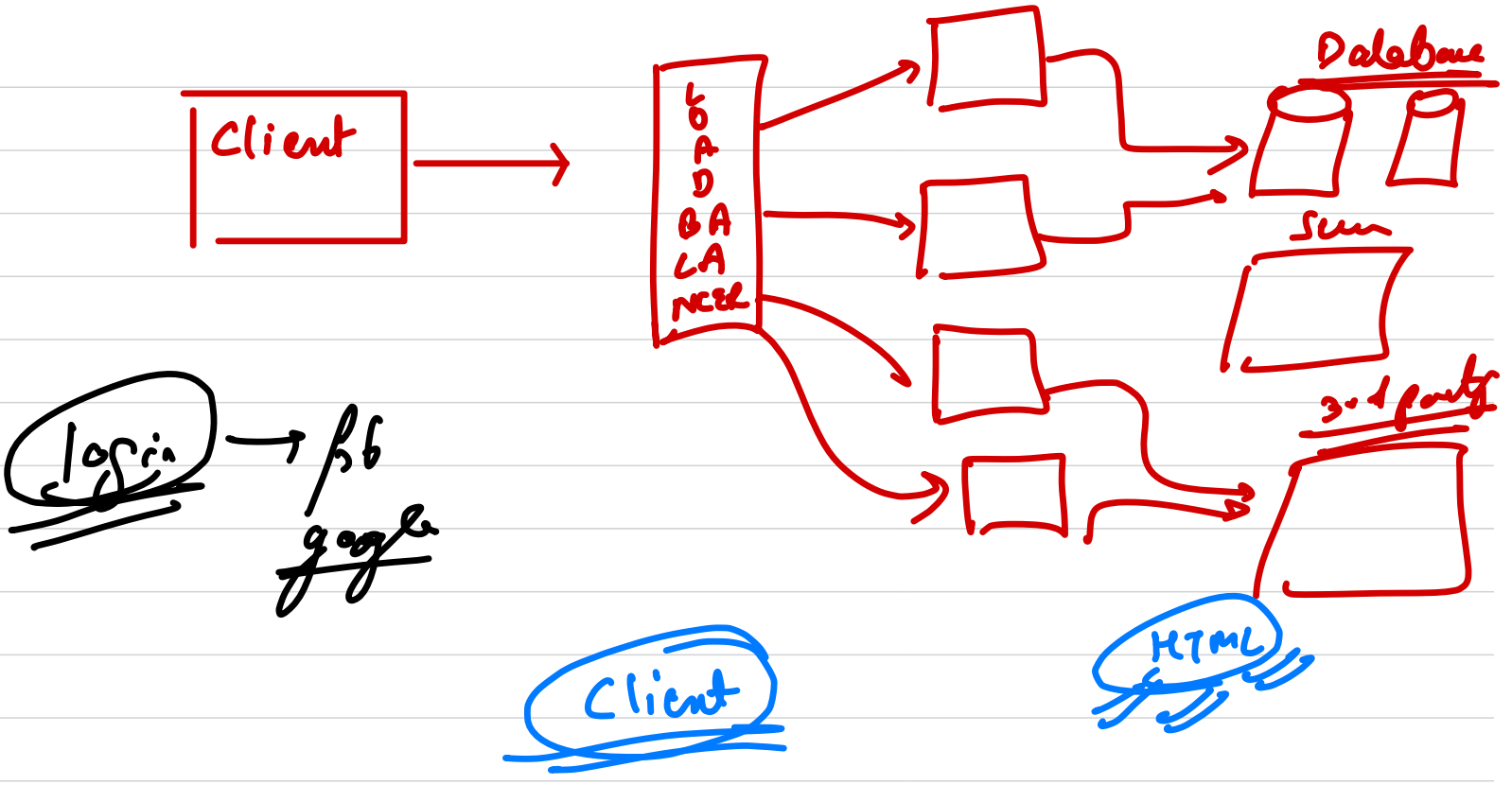


Protocols

HTTP
FTP
HTTPS
SMTP
etc....

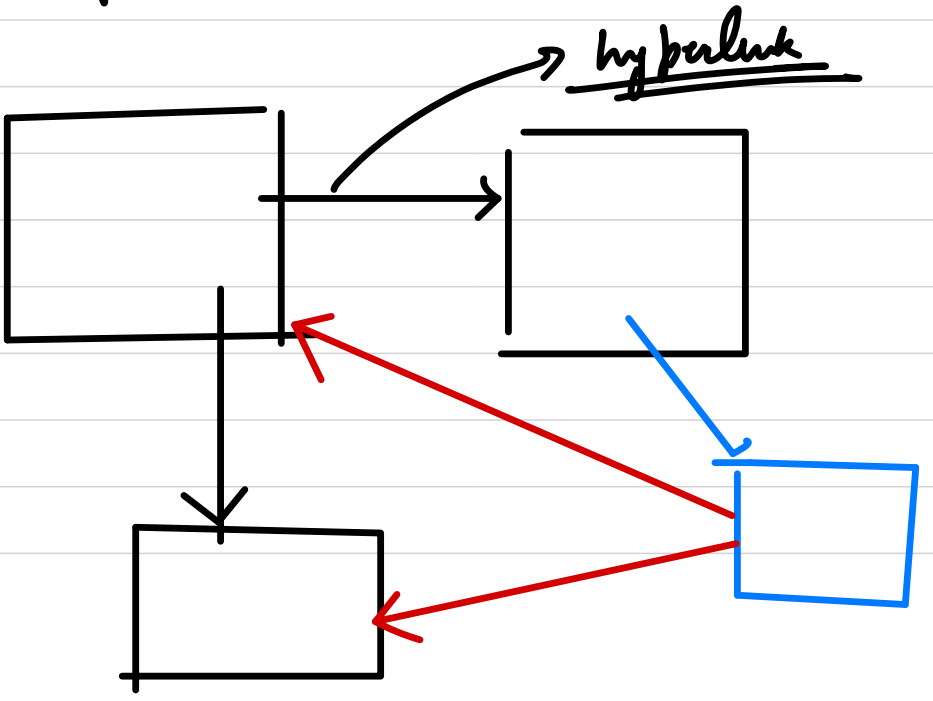
Components of Web

1) Web page → single document (HTML)
2) Web server → hardware or software that hosts website.
3) Search engine
4) Website.


Client
LOADBALANCER
Database
Server
3rd party
HTML
Client

[≡] web pages → HTML (Hyper Text Markup Lang)
↓
mobile
desktop

Hyper text is a special text that contains links to other text.



XML

XML → Xtensible Markup lang

↳ tag based langue

<age> 23 </age>

opening

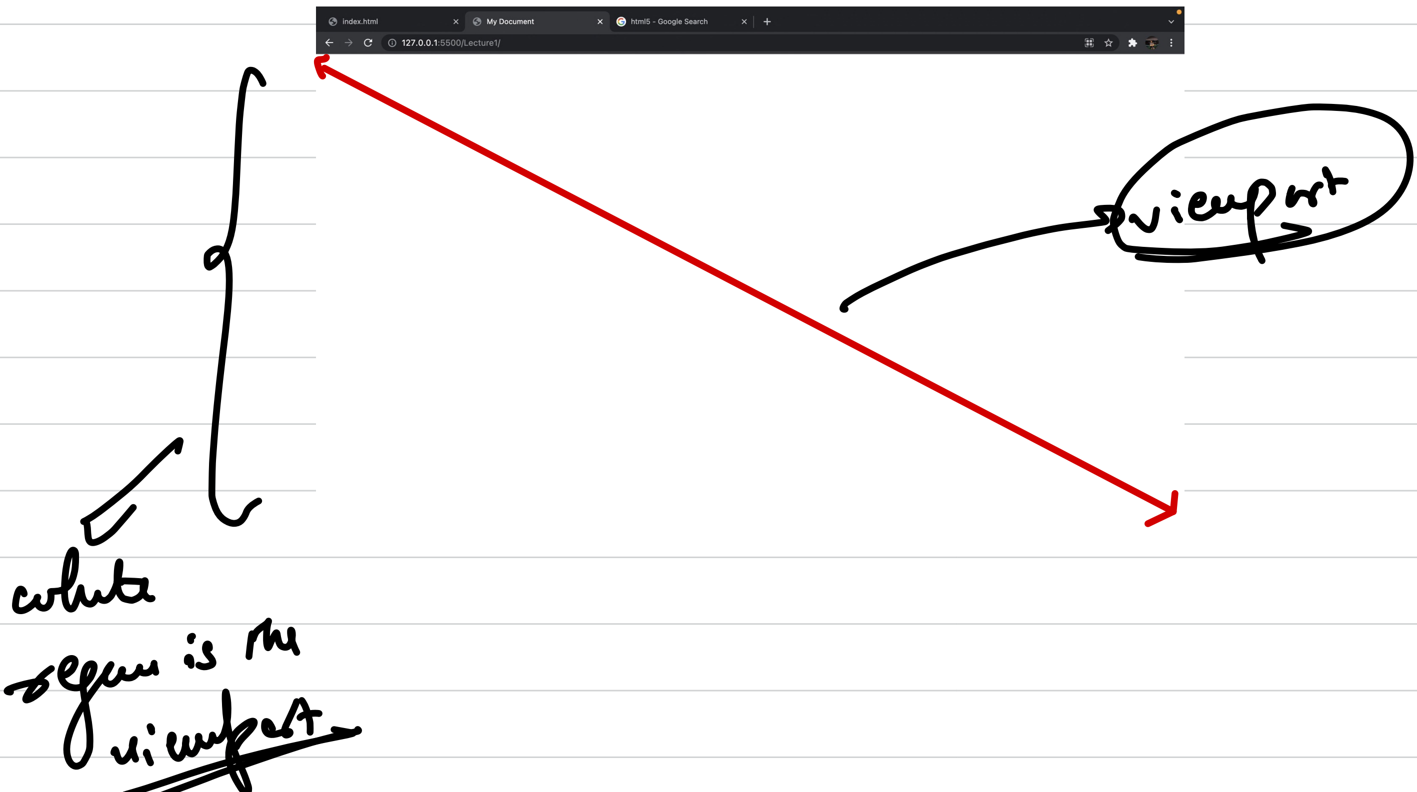
closing

one html tags → one html element

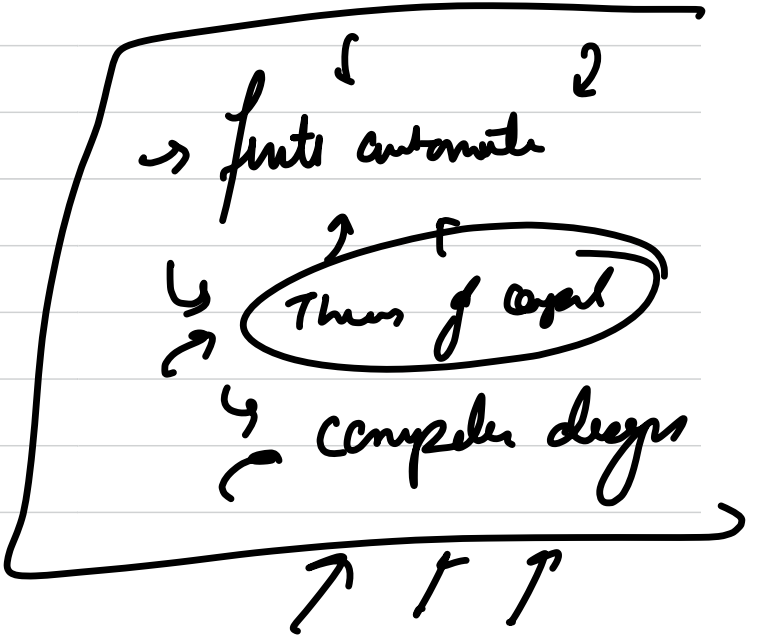
every element has properties → attributes

HTML → Skeleton
CSS → Beauty
JS → motion

→ Live

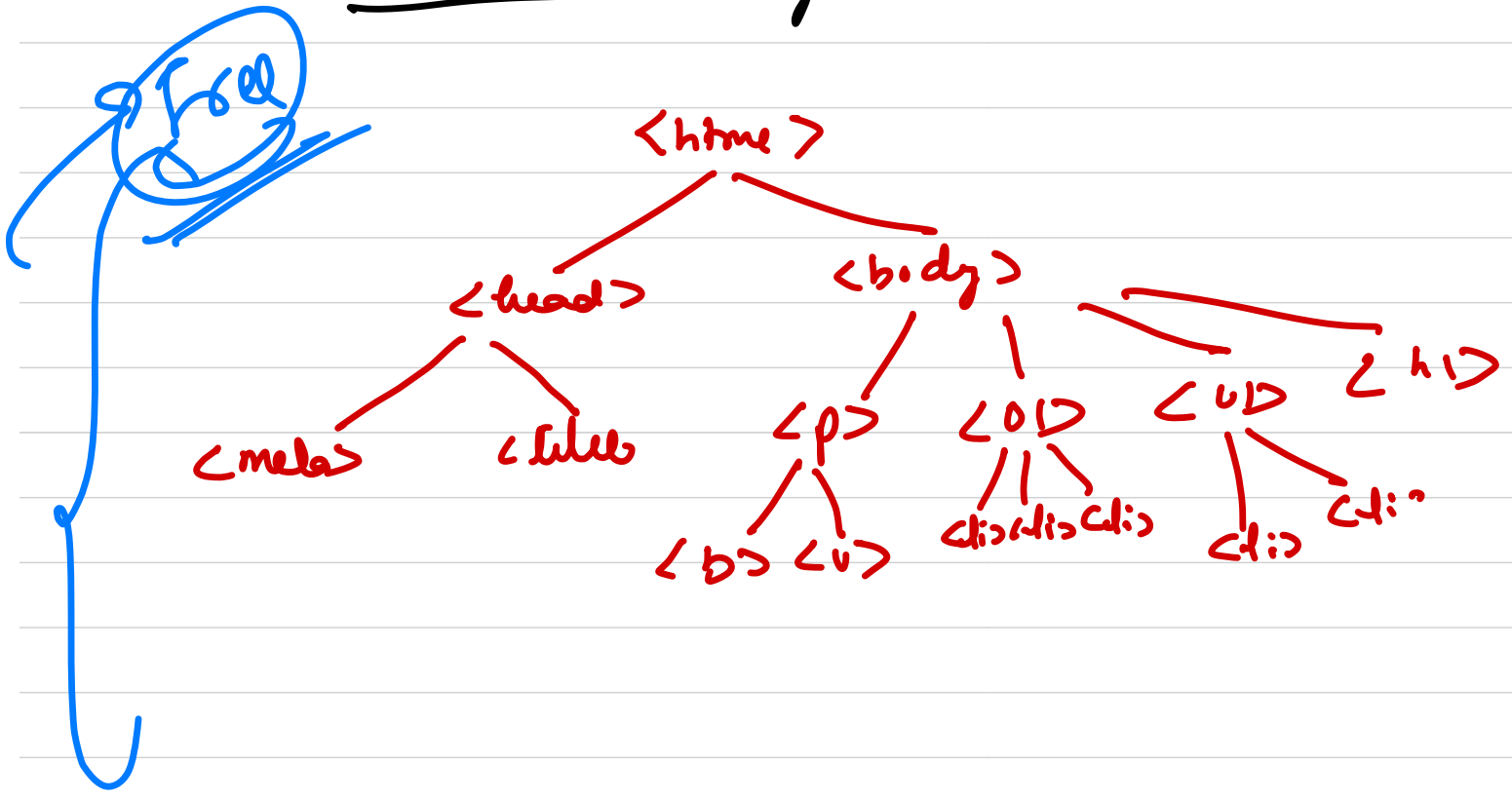

index.html
My Document
html5 - Google Search
127.0.0.1:5500/Lecture1/
viewport
white region is the viewport

even if CSS and JS is disabled, we should structure html such that we are able to depict all info properly.

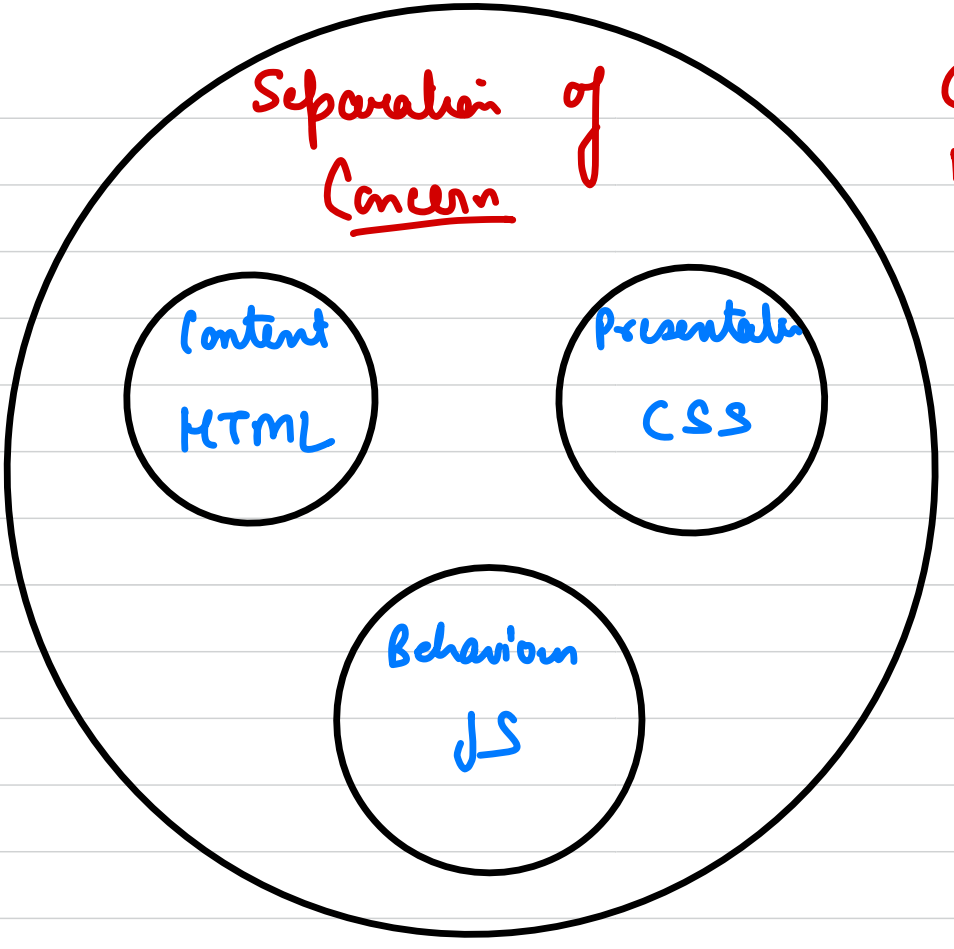



Block Tag → h1, p ....
Inline Tag, span

DOM → Document Object Model

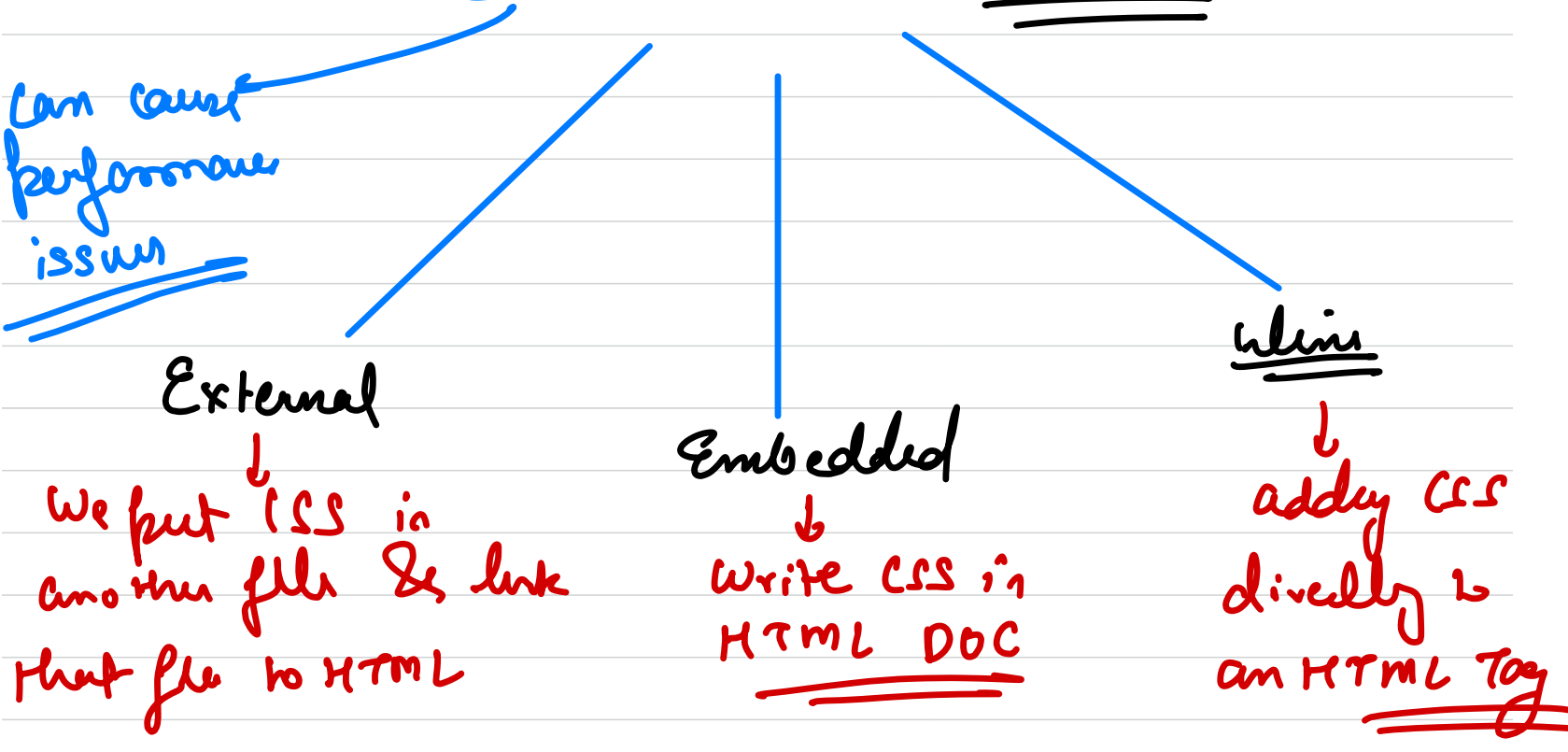


Cascading Style Sheet



CSS tells the browser how HTML structure is to be presented

How to add CSS in HTML DOC??



Content

style → HTML element attribute

Inline CSS

Code

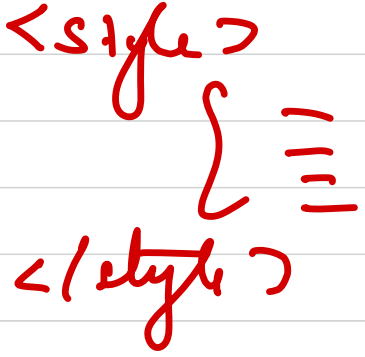
CSS file → .css

(link this file) → External CSS

<link href="./styles/1.css" rel="stylesheet" />

↳ one of the best ways

Embedded CSS



External Stylesheet →

→ reusable

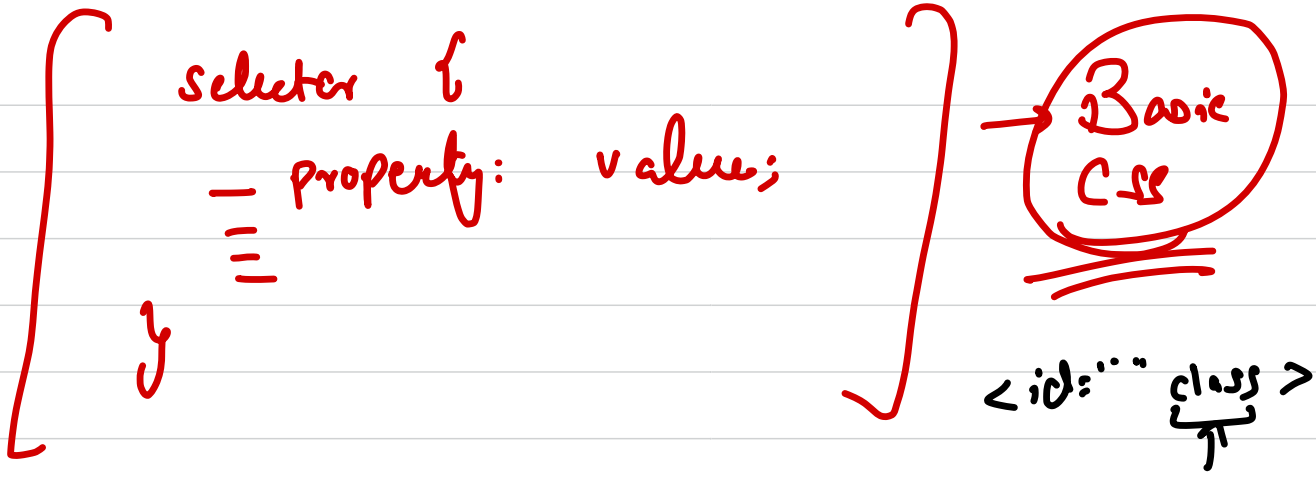
maintainable

Cachable

Code separation

Changes are site wide

* Selectors → these are used to refer to exact html elements. These selectors are used for referring individual & group of elements.



↳ There are 3 basic selectors, → CSS / JS

id selector → #myID {
}

class selector → .class {
}

tag selector → li {
}

Initially →

tag1 { }

Tag1, Tag2 { }

.class { }

#ID

:link

:active

latu

:first-child

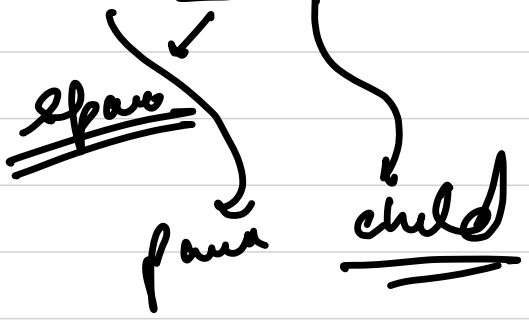
Specificity

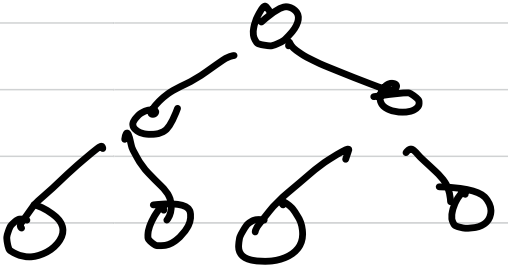
0 0 1 → tag

0 1 0 → class

1 0 0 → ID

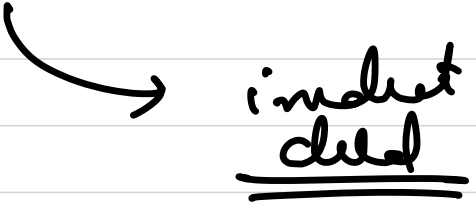
ul li → descendant selector





ul > li: C

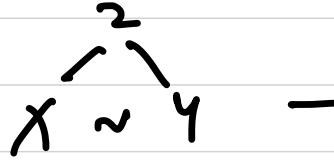
{



X + Y

↓

Y that immediately follows X.



→ Sets the CSS for all Y elements that are preceded by a X element with same parent.

```
{
    color: blue;
}
```

* Attribute selectors → they have same specificity as class

E [attr] → element E that has attribute attr with any value

E [attr = "val"] → element E that has attribute attr with value val. case sensitive

E [attr |= val] → Element E whose attribute attr has a value val or begins with "val-"

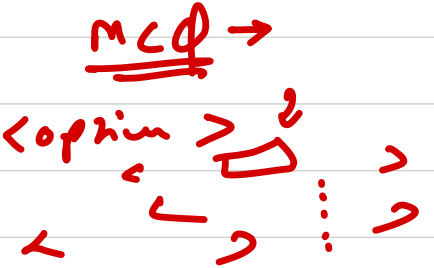
E [attr ^= val] → Element E whose attribute

Starts with val.

<a href="https://—" >

<a href="https://—" >

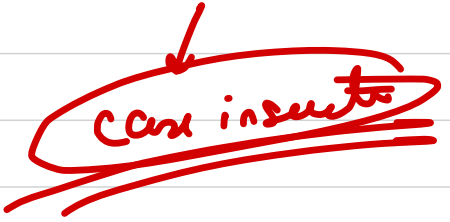
<a href="http://—" >



E [attr $= val] → Element E whose attribut
ends with val

E [attr *= val] → Element E whose allubuts
has a substring val.

$E[\text{attr} = \text{"} \quad \text{"} \quad i]$



UI pseudo class

:enabled

:disabled

:checked

structural selector

:root → It selects the root element.

:only-child → It targets an element that is only child of its parent. This means parent can have only one element.

:empty →

:nth-child

Counters in CSS

↳ We can change page appearance based on location on a doc:

→ property → counter-reset → (0 by default)