

```
main.js
1 // use babel
2 import { CompositeDisposable } from 'atom';
3 import StructureView from './structure-view';
4 import metrics from '@ali/yin-metrics';
5
6 const logger = atom.yin.logger;
7 logger.info('[structure-view] Init started');
8 logger.profile('structureview init');
9
10 export default [
11   structureView: null,
12
13   activate(state) {
14     this.subscriptions = new CompositeDisposable();
15     this.subscriptions.add(atom.commands.on('atom-workspace', {
16       'yang-structure-view:toggle': () => this.toggle(),
17       'yang-structure-view:show': () => this.show(),
18       'yang-structure-view:hide': () => this.hide('self')
19     }));
20
21     logger.profile('structure-view Init');
22     logger.info('[structure-view] Init finished');
23   },
24
25   deactivate() {
26     this.subscriptions.dispose();
27     this.structureView.destroy();
28   },
29
30   serialize() {
31   }
32
33   switch (state) {
34     case 'edit':
35       logger.info('structure-view: edit view starts');
36       let editors = atom.workspace.getTextEditors();
37       if (editors.length < 1) {
38         editors.length === 1 && (editors[0].getPath());
39         atom.confirm({ message: 'WARN', detailedMessage: 'No file is opened!', buttons: [ 'Close' ] });
40         return;
41       }
42
43       if (!this.structureView) this.structureView = new StructureView();
44
45       const rightDock = atom.workspace.getRightDock();
46       try {
47         rightDock.insertItem(this.structureView);
48       } catch (err) {
49         logger.error(`Error occurred while trying to insert ${this.structureView} into the right dock`);
50       }
51     case 'read':
52       logger.info('structure-view: read view starts');
53       break;
54   }
55 }
```

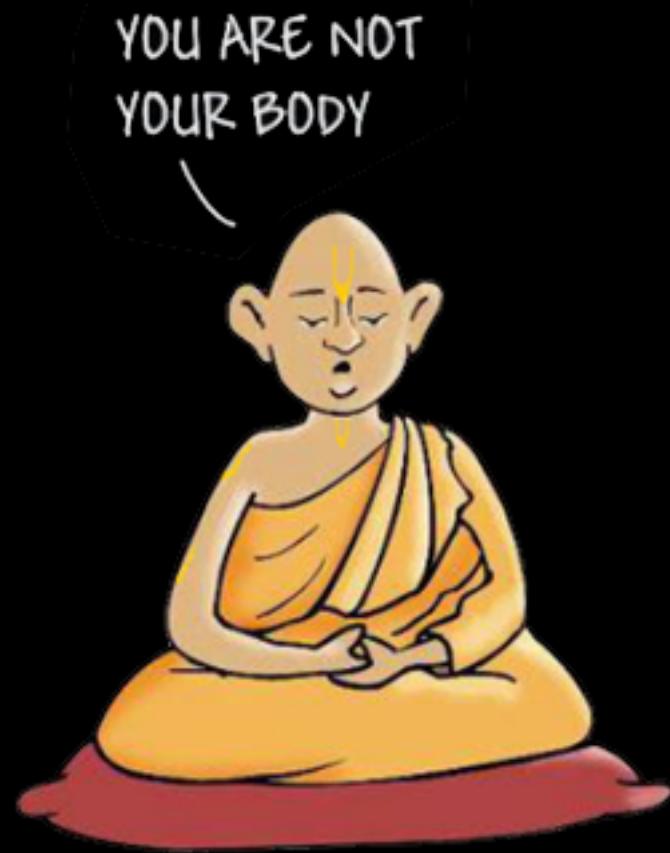


Web Programming

Yuan Wang
2025 Winter
Lecture 1

Who am I

- Name: Yuan Wang
- Email: ywang6@scu.edu
- Office: Heafey 205

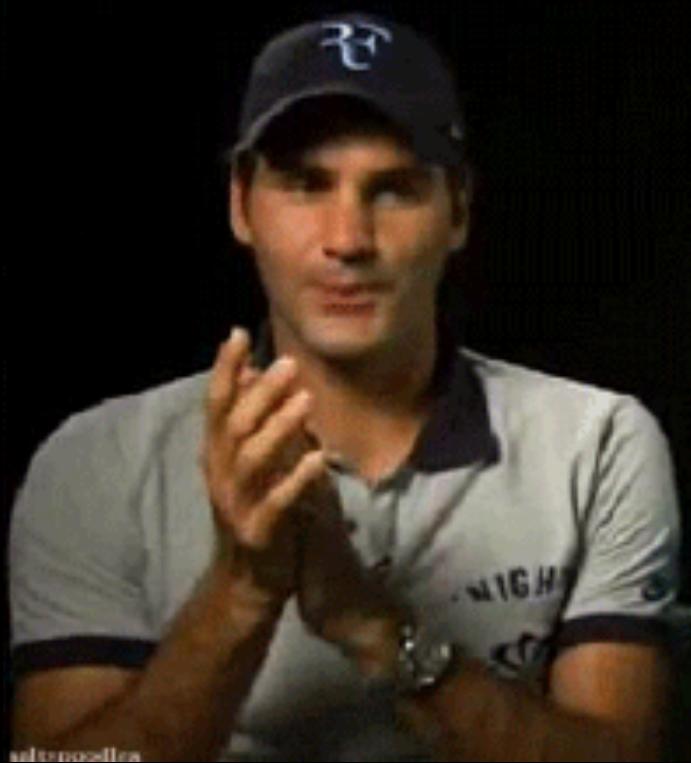


This course

- Office hour:
Thursday 12PM-2PM
- Lecture material and assignment:
Camino
- Grade:
homework + final
50% + 50% (approximately)
- No text book
resource: **internet**
chatGPT, DeepSeek :)



Get Started



Web overview and HTML



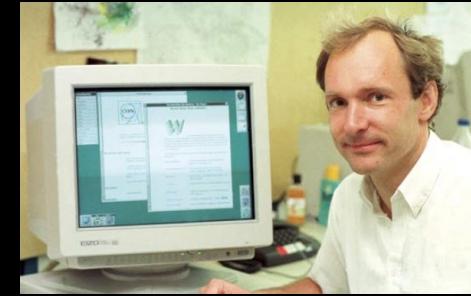
Internet

All because of **Tim Berners-Lee** ("TimBL")



1990

- Published the first website info.cern.ch
- Initiated the **HTTP protocol** to transfer **HyperText (HTML)** document on Internet with **URL**
- Build the **first Browser** to view document
(The first Browser was called **WorldWideWeb**)
- Write the **first Web Server** to send document
(**CERN httpd**)



Internet

- 1993:
Founded World Wide Web Consortium (W3C) (by Tim Berners-Lee)
Standards organization for World Wide Web



What is “Web”?

Stuff on the Internet
(through interface)

- Browser

- Mobile app



We are
working
With browser



Internet - Browsers

- 1993: Mosaic
- 1994: Netscape
- 1995: Internet Explorer
- 1996 **Browser war** – Netscape dead
- 1998: Firefox (result of Netscape going open)
- 2003: Safari (Apple)
- 2008: Chrome (google)

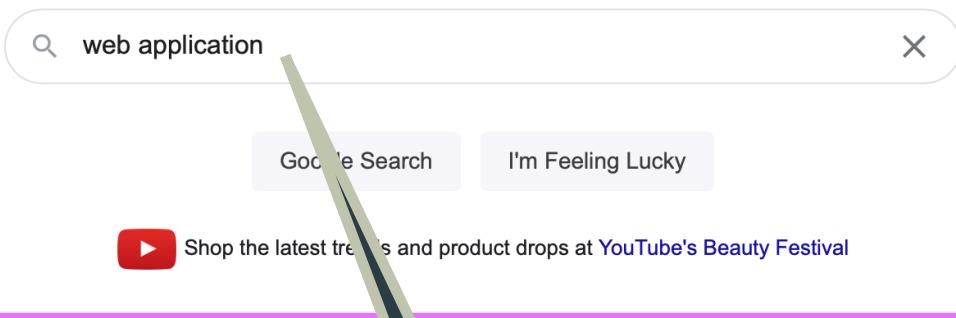


Internet - how to get what you want?



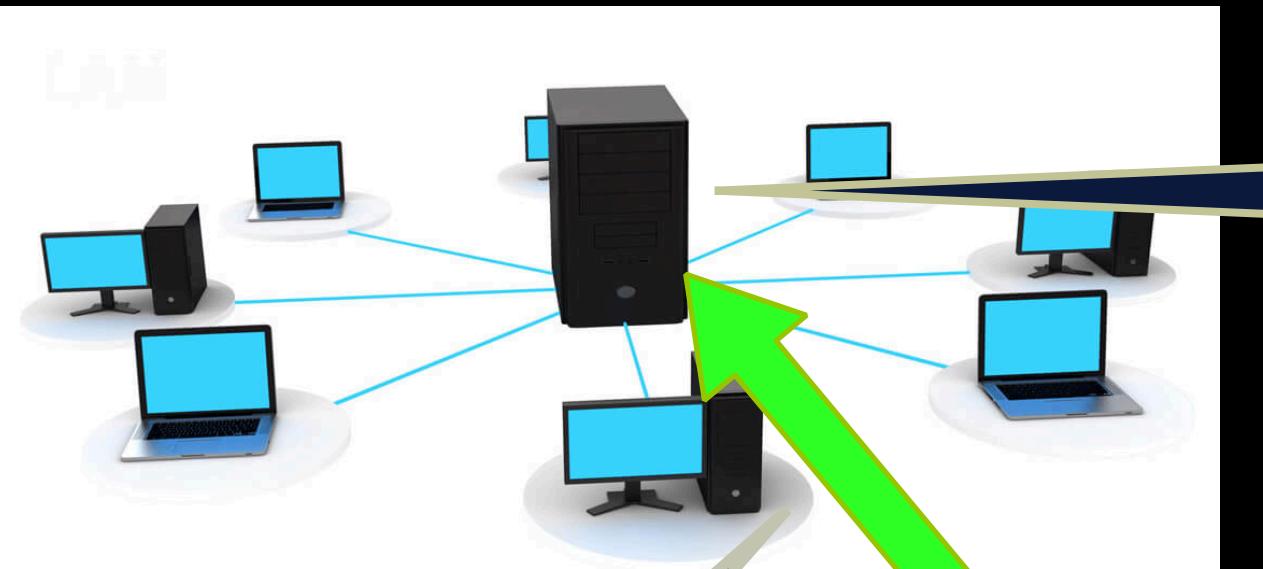
client

Google



what's going to happen
after you press return?





server
- document
- web program

request:

please return a
page that contains
“web application”

client

Google

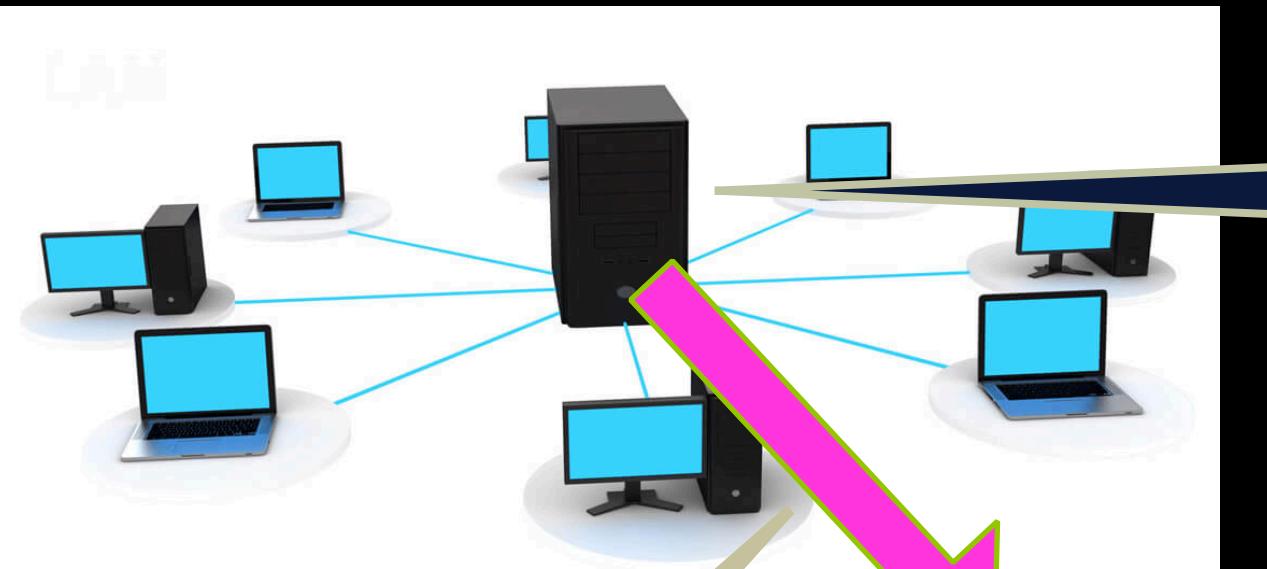
A search bar with the text "web application" inside. There is a magnifying glass icon at the beginning of the input field, and an "X" icon at the end.

Google Search

I'm Feeling Lucky

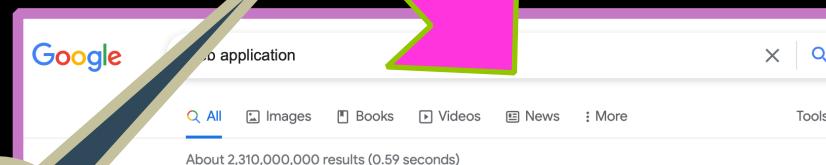


Shop the latest trends and product drops at YouTube's Beauty Festival



server

- document
- web program



client

response page

a page with
“web application”
related links

A Web application (Web app) is an **application program** that is stored on a remote server and delivered over the Internet through a browser interface. Web services are Web apps by definition and many, although not all, websites contain Web apps.

<https://www.techtarget.com/definition/Web-application...>

What is Web Application (Web Apps) and its Benefits



Web application

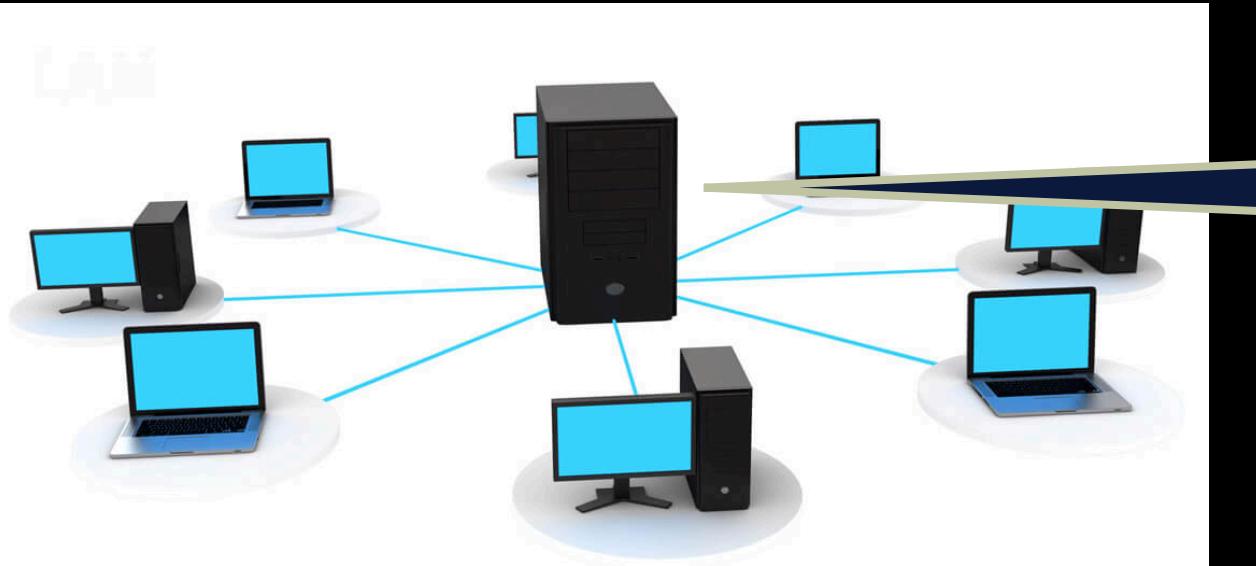
A web application is application software that runs on a web browser, unlike software programs that run locally and natively on the operating system of the device. Web applications are delivered on the World Wide Web to users with an active network connection.

[Wikipedia](#)

Advantages

Firewall

Layers



server
 - document
 - web program

client

what is going
to happen if you click
this link ?

Google search results for "web application":

- Ad · https://www.informationvine.com/results** : Web Application Development - Web Application Development

Search Web Application Development. Search & Find Now. Quick & Easy Answers. Find Relevant Information. Learn More. Types: Business, Health, Finance, Entertainment, Travel, Computers, Real Estate.

Relevant searches
 Explore now Find trusted content

Great Information
 Explore InformationVine.com Find what you need
- Ad · https://www.capterra.com/web-app/free-list** : Top 10 Web App Software - Free List of the Top Products

Easily Find The Web Application Software You're Looking For w/ Our Comparison Grid. Find...

A Web application (Web app) is **an application program** that is stored on a remote server and delivered over the Internet through a browser interface. Web services are Web apps by definition and many, although not all, websites contain Web apps.

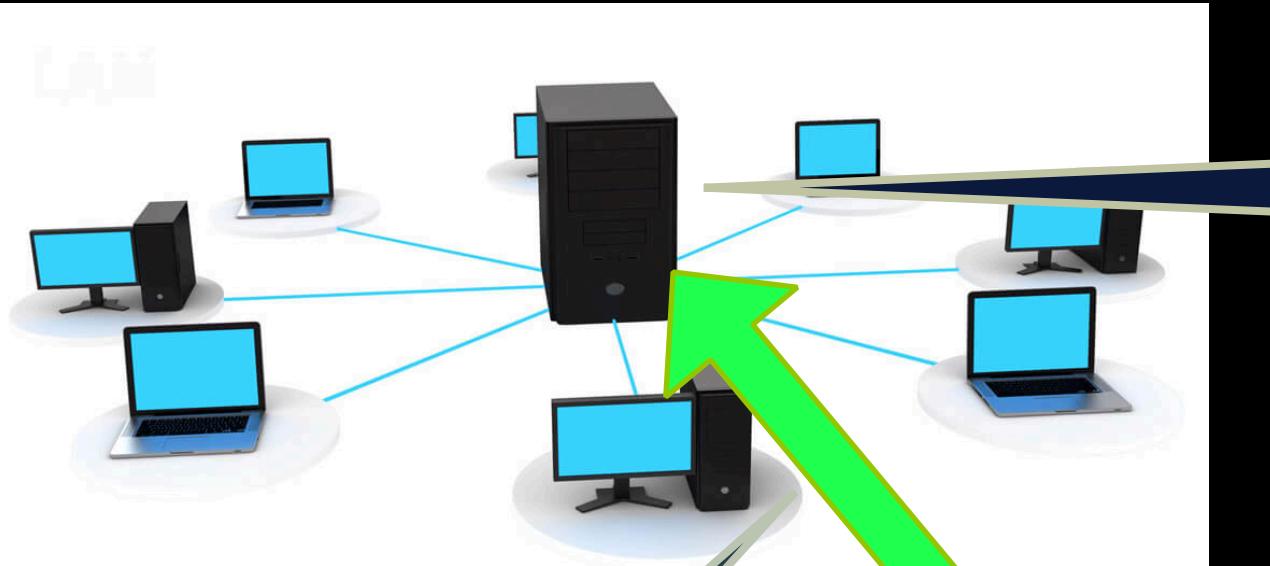
What's Web App 

[https://www.techtarget.com/definition/Web-application...](https://www.techtarget.com/definition/Web-application)

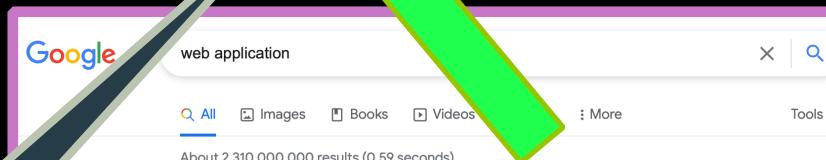
What is Web Application (Web Apps) and its Benefits

Web application details:

- Advantages**
- Firewall**
- Layers**



send a request:
please send me the
page of this
link



Web Application Development - Web Application Development

Search Web Application Development. Search & Find Now. Quick & Easy Answers. Find Relevant Information. Learn More. Types: Business, Health, Finance, Entertainment, Travel, Computers, Real Estate.

Relevant searches

Explore now Find trusted content

Great Information

Explore InformationVine.com Find what you need

Ad · https://www.capterra.com/web-app/free-list

Top 10 Web App Software - Free List of the Top Products

Easily Find The Web Application Software You're Looking For w/ Our Comparison Grid. Find...

A Web application (Web app) is an application program that is stored on a remote server and delivered over the Internet through a browser interface. Web services are Web apps by definition and many, although not all, websites contain Web apps.



https://www.techtarget.com › definition › Web-application...

What is Web Application (Web Apps) and its Benefits

server
- document
- web program

client



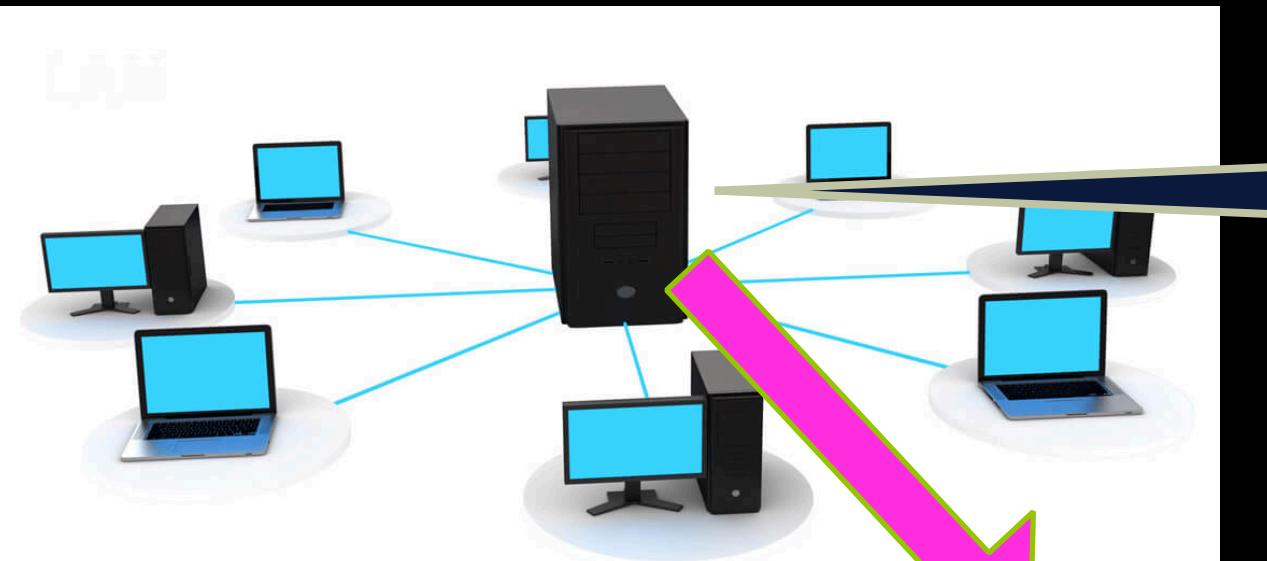
Web application

A web application is application software that runs on a web browser, unlike software programs that run locally and natively on the operating system of the device. Web applications are delivered on the World Wide Web to users with an active network connection. [Wikipedia](#)

Advantages

Firewall

Layers



server
- document
- web program

client

The screenshot shows a web browser window with the following details:

- Header: TECHTARGET NETWORK, SEARCH, TIPS, WEBINARS, MORE CONTENT ▾, AGILE
- Left sidebar: TechTarget, Search Software Quality, TOPIC ▾ Software design and development
- Breadcrumbs: Home > Software design and development > Programming > Web application (Web app)
- Main content:
 - Section: DEFINITION
 - Title: **Web application (Web app)**
 - By: TechTarget Contributor
 - Share icons: f (Facebook), t (Twitter), in (LinkedIn), p (Pinterest), e (Email)
 - Description: A Web application (Web app) is an [application program](#) that is stored on a remote server and delivered over the Internet through a browser interface. [Web services](#) are Web apps by definition and many, although not all, websites contain Web apps. According to Web.AppStorm editor Jarel Remick, any website component that performs some function for the user qualifies as a Web app.

response page

return the
requested page

Web

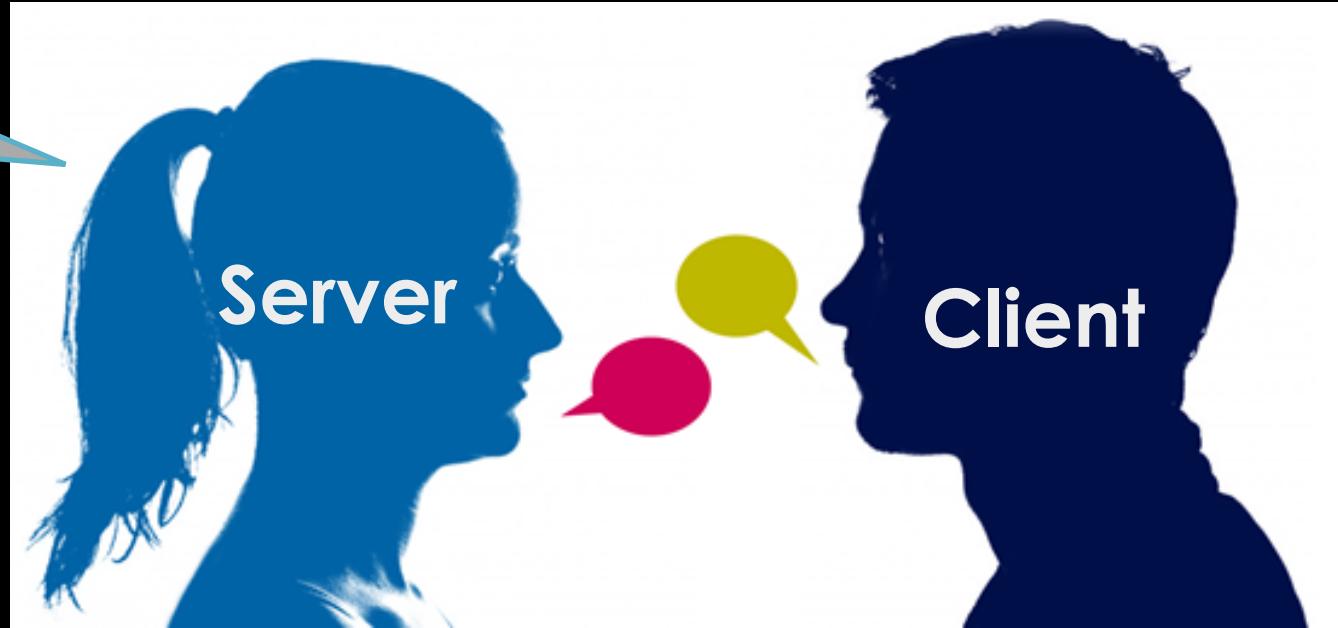
Stateless

Example:

Customer: can I have a bubble tea please
Water: ok, one bubble tea for you.

Customer: oh, no sugar please
Water: ????????????????

speaking
HTTP
protocol



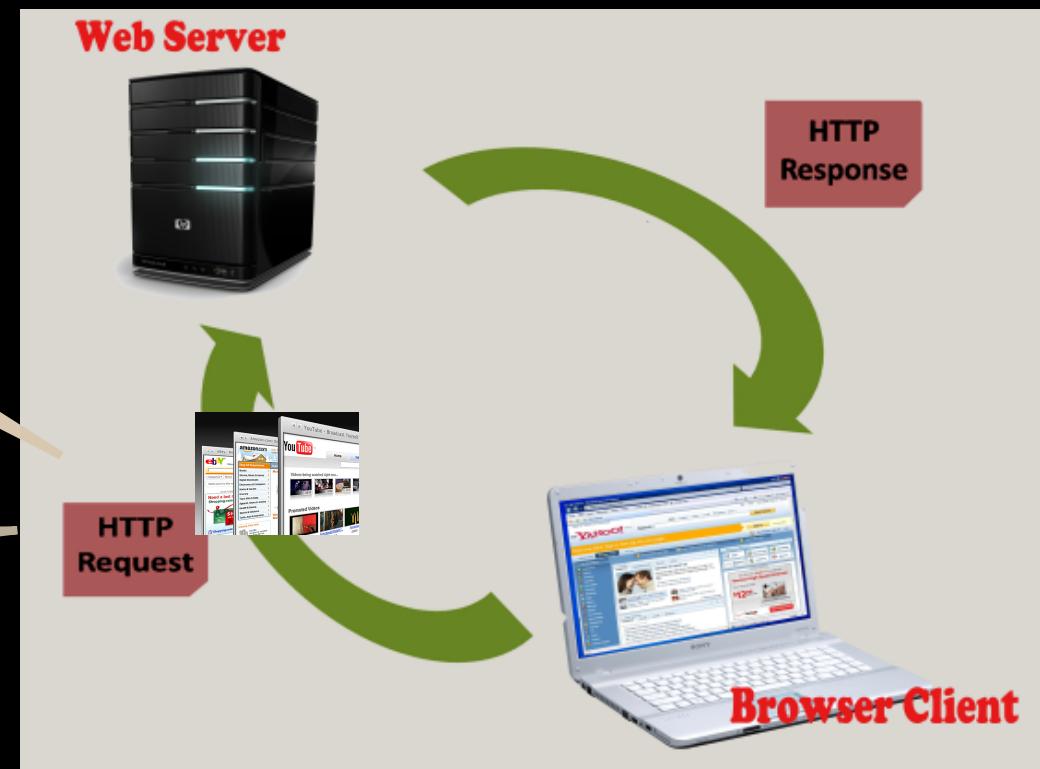
HTTP Protocol

Hyper Text Transfer Protocol

The language web application and web server
to talk to each other on the internet

browser
is responsible
to construct
request

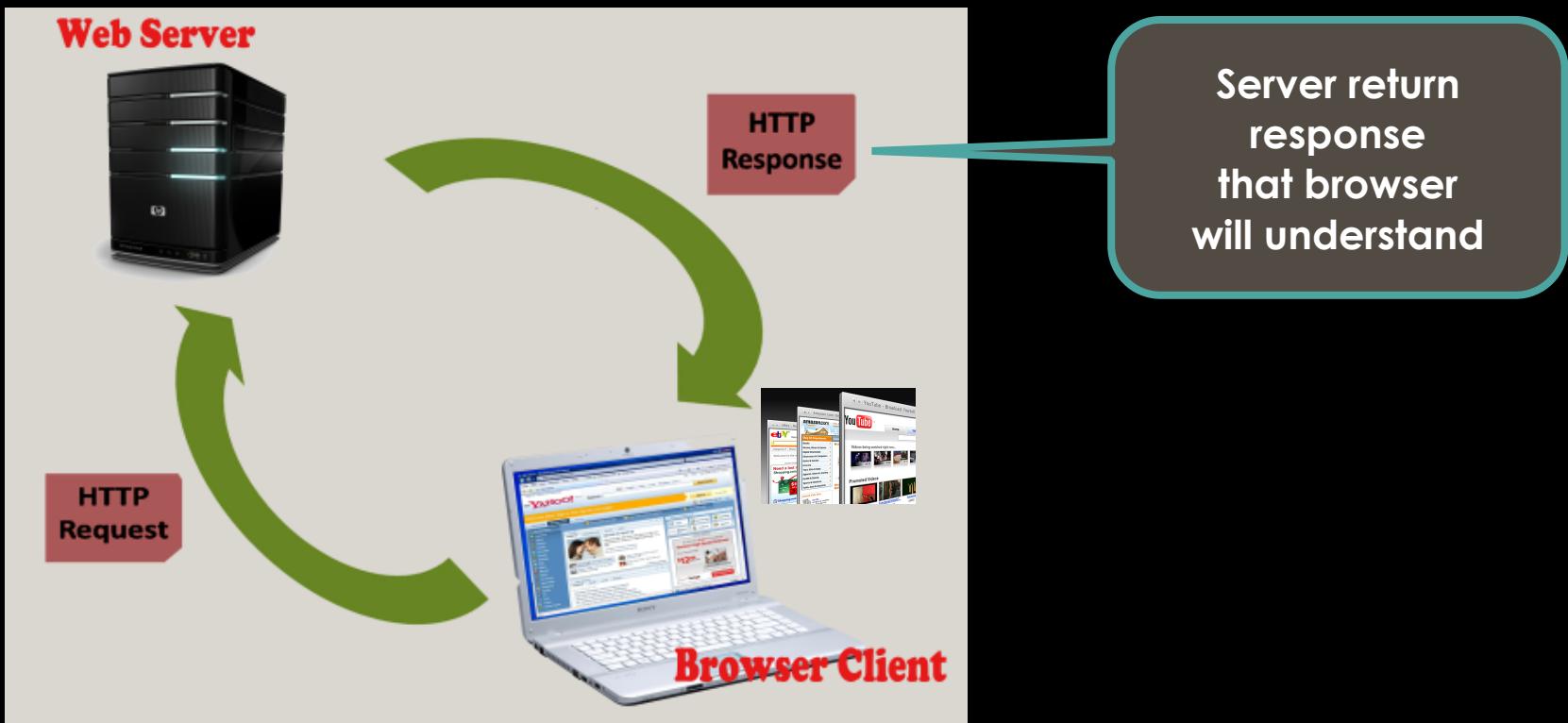
the request
will conform to
HTTP protocol



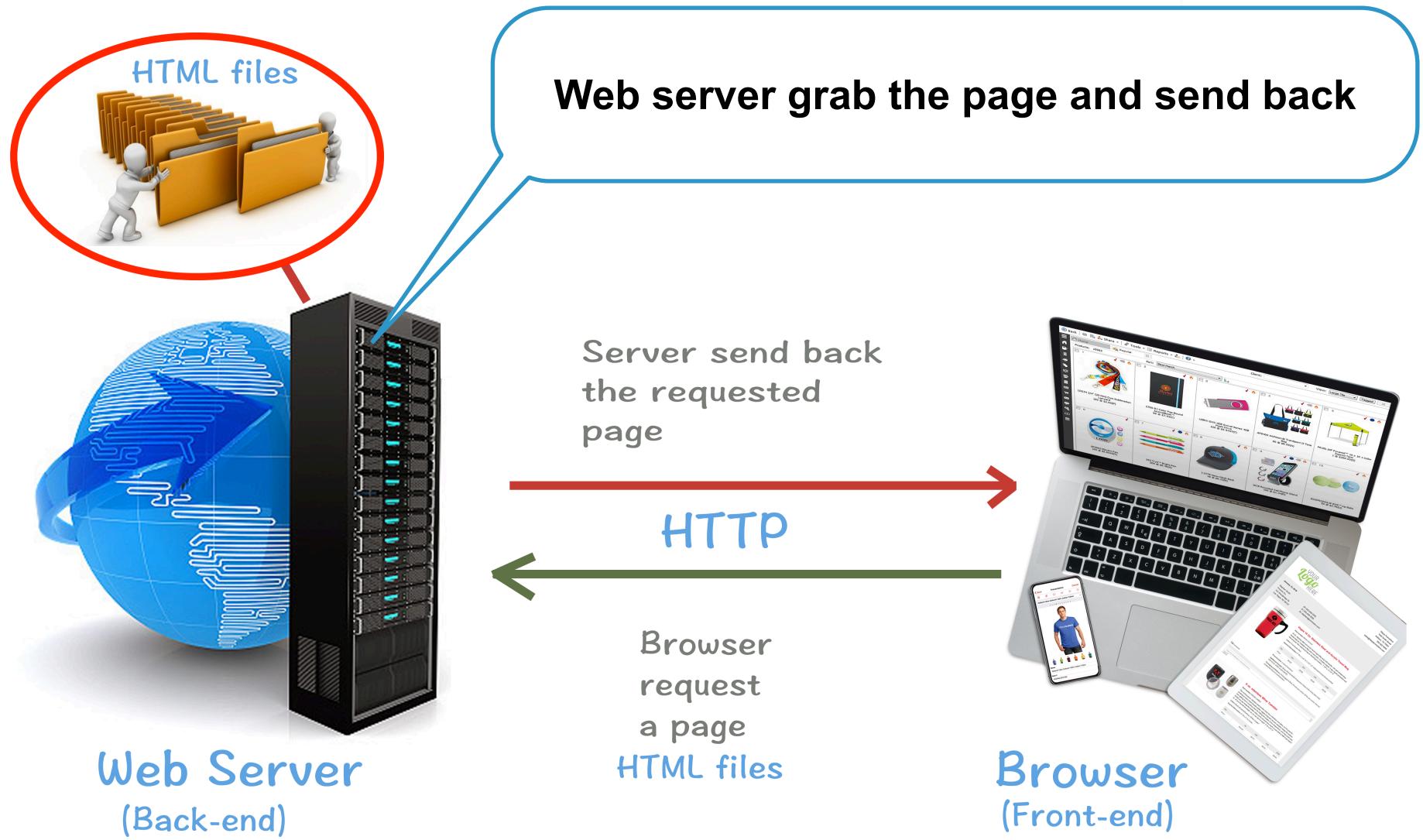
HTTP Protocol

Hyper Text Transfer Protocol

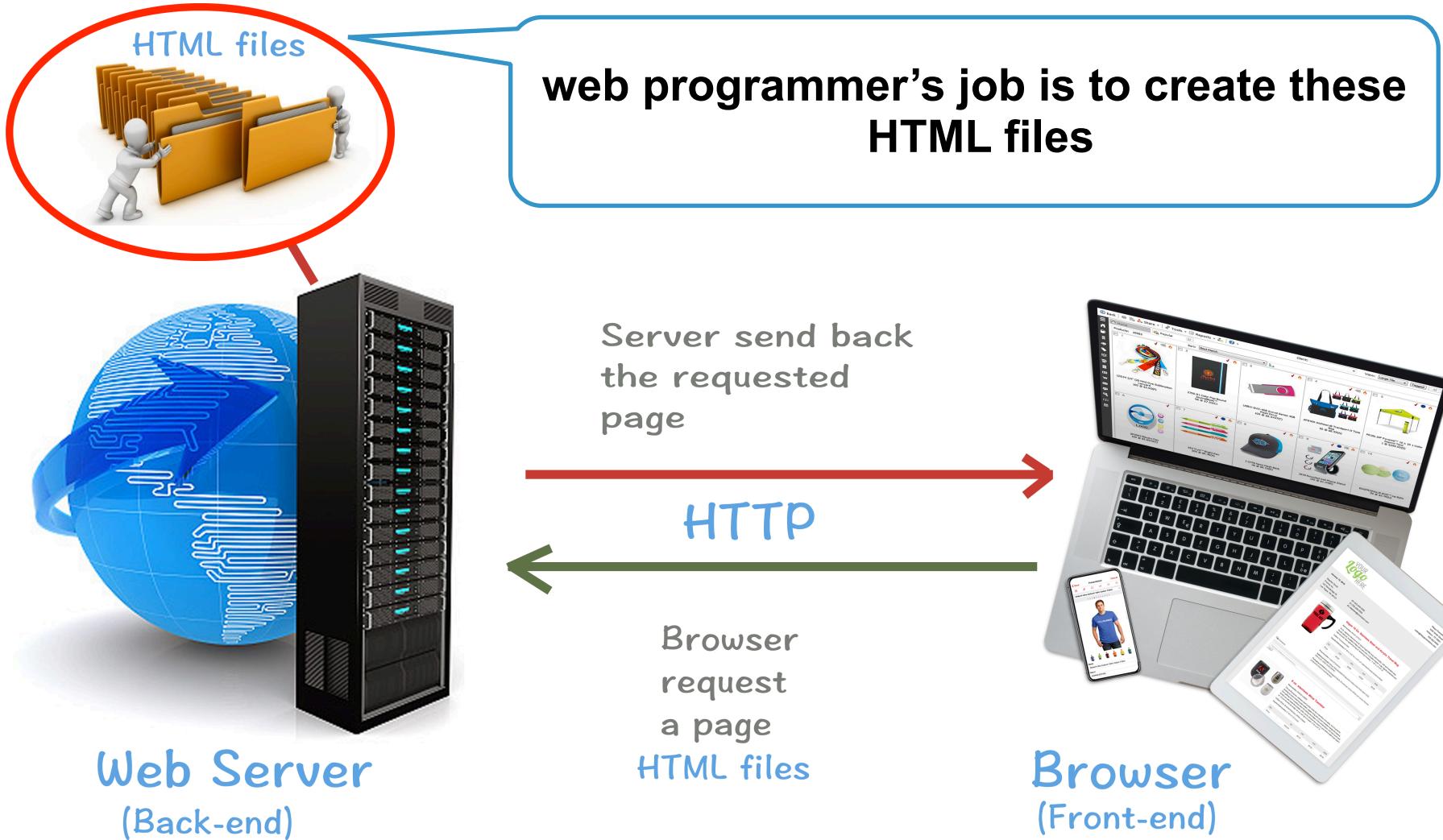
The language web application and web server
to talk to each other on the internet



Web



Web



Web



Web Server
(Back-end)

Browser's job is to render HTML files

Server send back
the requested
page

HTTP

Browser
request
a page
HTML files



Browser
(Front-end)

Web



Web Server
(Back-end)

what are web programmers
(you) supposed to do?

2 things

Server send back
the requested
page

HTTP

Browser
request
a page
HTML files

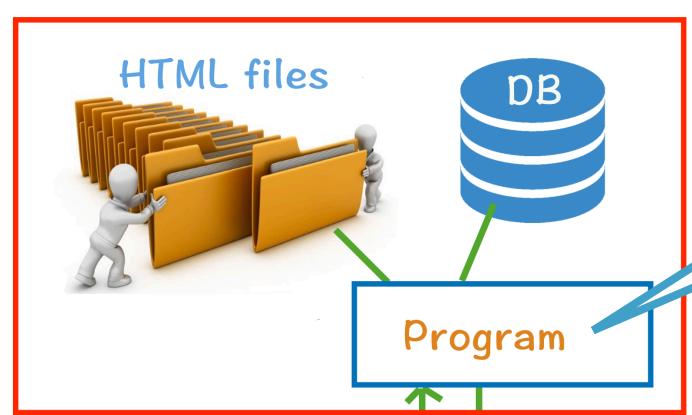


Browser
(Front-end)

Web

1

Write program on server
to generate HTML
dynamically
(back-end programming)



Web Server
(Back-end)

Server send back
the requested
page

HTTP

Browser
request
a page
HTML files

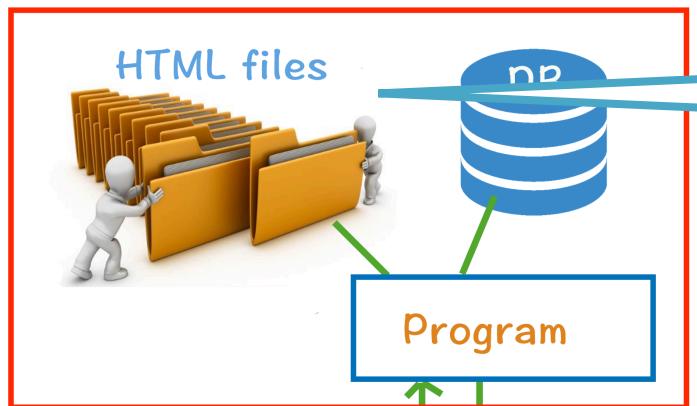


Browser
(Front-end)

Web

2

Add javascript to HTML file, so that browser can execute and create dynamic page (front-end programming)



Web Server
(Back-end)

Server send back
the requested
page

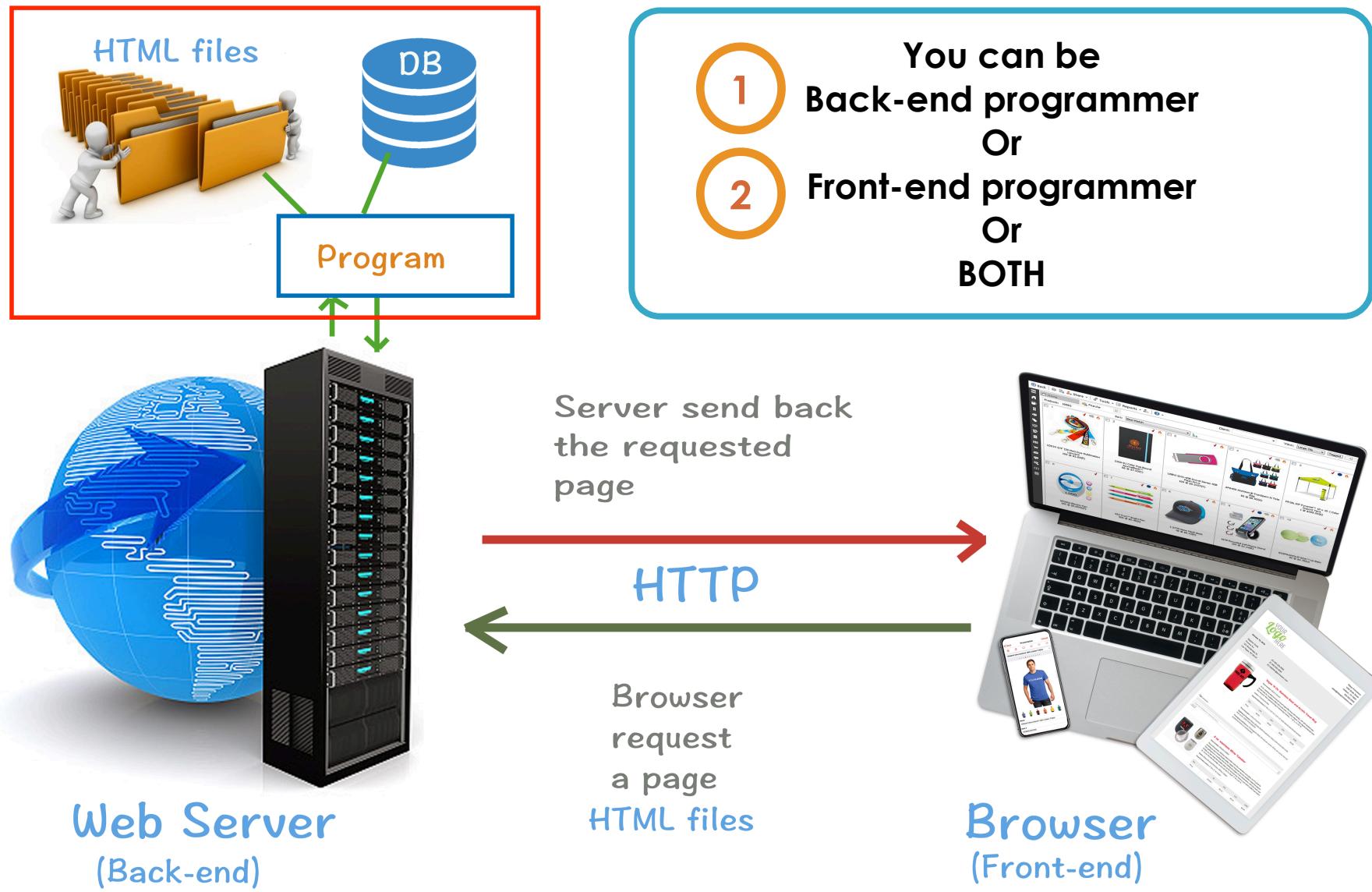
HTTP

Browser
request
a page
HTML files

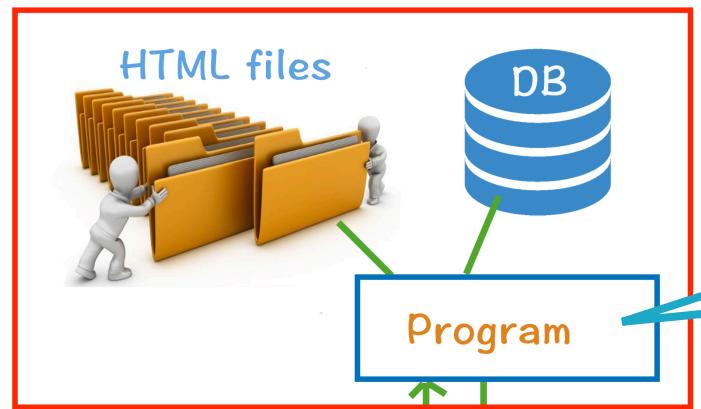


Browser
(Front-end)

Web



Web



Languages / Frameworks for server side programming:

Python / Django

PHP / Laravel

Node.js / Express

C# / ASP.NET

Ruby / Sinatra or Ruby on Rails

we are using this



Web Server
(Back-end)

Server send back
the requested
page

HTTP



Browser
request
a page
HTML files

Browser
(Front-end)

Internet - HTTP

what is inside
Request Package

“enforced” by
HTTP protocol



GET /doc/test.html HTTP/1.1

Host: www.test101.com

Accept: image/gif, image/jpeg, */*

Accept-Language: en-us

Accept-Encoding: gzip, deflate

User-Agent: Mozilla/4.0

Content-Length: 35

bookId=12345&author=Tan+Ah+Teck

Request Line

Request Headers

Request
Message
Header

A blank line separates header & body

Request Message Body

Internet - HTTP

another example of Request Package

```
01 POST /foo.php HTTP/1.1
02 Host: localhost
03 User-Agent: Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US; rv:1.9.1.5) Gecko/20091102
04 Firefox/3.5.5 (.NET CLR 3.5.30729)
05 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
06 Accept-Language: en-us,en;q=0.5
07 Accept-Encoding: gzip,deflate
08 Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
09 Keep-Alive: 300
10 Connection: keep-alive
11 Referer: http://localhost/test.php
12 Content-Type: application/x-www-form-urlencoded
13 Content-Length: 43
14 first_name=John&last_name=Doe&action=Submit
```



Internet - HTTP

Request

How to address the request: **IP address**

URL: Uniform Resource Locator
(human readable IP address)

URL format:
protocol://server/request-URI

commonly used protocols		
Protocol	Protocol indicator	Description
ftp	ftp://	allows downloading of files from a remote FTP server
file	file://	Allows accessing of files from local hard drive
gopher	gopher://	Allows access to documents located on a gopher server
http	http://	allows access to documents located on the web
mailto	mailto:	although it is not a standard, most browsers supports it to open user's email client
news	news:	this protocol is used to connect to USENET newsgroup
telnet	telnet://	This protocol allows a telnet session to a telnet host

Internet - HTTP

HTTP URL:

`http://host [: port] / [file path]`

port: for **host** to know what specific service on host you are requesting

Example

`http://www.domain.com:1234/path/to/resource`

protocol

host (Domain name)

resource path



Internet - HTTP

The city hall building
is one server machine

IP address and port number analogy

IP address (URL)	
City Hall (300 City Street)	Room
Office of the City Social Welfare & Development Officer	192
Office of the City Treasurer	1929
Office of the City Vice-Mayor	3805
Office of the Disaster Risk Reduction & Management Officer	9954
Office of the City Engineer	2761
Office of the Human Resource Management Officer	2758
Office of the Local Civil Registrar Officer	2973
Office of the Secretary to the Sanggunian panlungsod	2843
Office of the Senior Citizen Affairs	8282
Office of the City Veterinarian	1291
Office of the Public Order & Safety Officer	2308

Internet - HTTP

Domain names

- .com** Represents business, commercial, and personal web sites. Examples include www.honda.com or www.potterybarn.com.
- .edu** Used by educational institutions, like technical schools, colleges, and universities. Examples include Pitt's address at www.pitt.edu and Penn State's address at www.psu.edu.
- .net** Used by networked organizations or internet service providers like www.webtv.net or www.netzero.net.
- .org** Organizations that are usually non-profit. Our address, www.carnegielibrary.org, is one example of this type of site.
- .mil** Used by military agencies.
- .gov** Denotes government agencies: state, national or international. Examples are www.whitehouse.gov and www.irs.gov

Internet - HTTP

dmv.com/?tg1=DVA&utm_content=dmv.com&utm_medium=dmv_&tg7=dmv_&utm_source=dmv.com&tg9=dmv.com&utm_term=organic_dmv&utm_campaign=organic_dmv

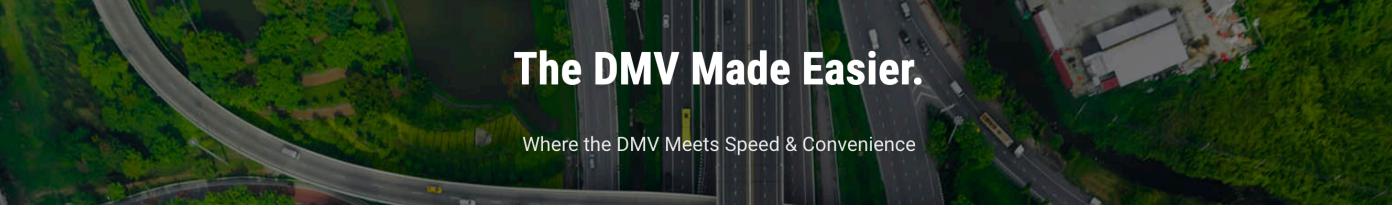
DMV.com is a privately owned website that is not affiliated with any government agencies.

DMV
a non-government site

Licenses ▾ Registration & Titles ▾ Violations & Regulations ▾ Records ▾ Insurance ▾ Education ▾ Buy/Sell a Car ▾ DMV Locations Select State

The DMV Made Easier.

Where the DMV Meets Speed & Convenience



dmv.ca.gov/portal/

REAL ID Online Services Translate MyDMV Search here... →

Vehicle Registration Driver's License & ID Cards Vehicle Industry Services Driver Education & Safety Appointments Locations

JOIN THE PILOT

Learn More

mDL

CA DMV Wallet Apple Wallet Google Wallet®



Use our Service Advisor to find the info you need, fast.

• renewals • replacements • permits • tests • change address • more!

Try Service Advisor

Internet - HTTP

IP Address:

http://en.wikipedia.org/wiki/Uniform_resource_locator

<http://www.scu.edu/oncampus/studentlife.cfm>

<http://www.facebook.com>

Don't forget: <http://localhost>

We will be using this
to run our web
applications locally



Internet - HTTP

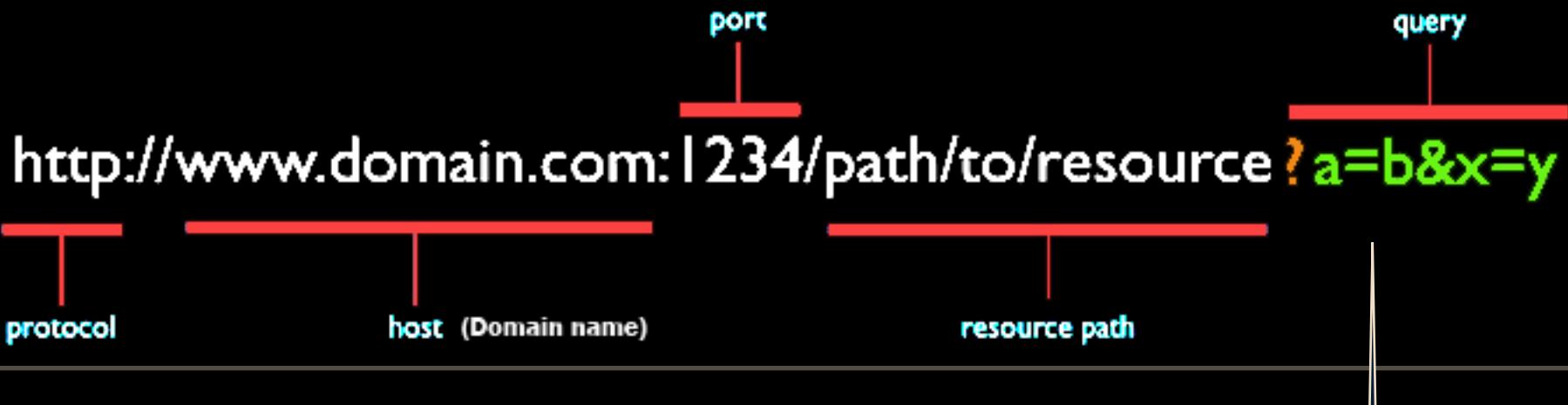
query string

How about this?

http://search.scu.edu/search?q=coen276

IP Address with query

http://host [: port] / [file path [? query]]



Pass parameters: **name/value pairs**
field1=value1&field2=value2&field3=value3

Internet - HTTP

HTTP response:
detail content



Internet - HTTP

HTTP response

Head information

HTTP/1.1 200 OK
Date: Thu, 20 May 2004 21:12:58 GMT
Connection: close
Server: Apache/1.3.27
Accept-Ranges: bytes
Content-Type: text/html
Content-Length: 170
Last-Modified: Tue, 18 May 2004 10:14:49 GMT

Status Line

General Headers

Response Headers

Entity Headers

HTTP
Response

<html>
<head>
<title>Welcome to the Amazing Site!</title>
</head>
<body>
<p>This site is under construction. Please come
back later. Sorry!</p>
</body>
</html>

Message Body

web page

Internet - HTTP

HTTP response

Another
important
message ?



Internet - HTTP

HTTP response: Status Code

Client Request Successful

200 — OK [Success! This is what you want.]

201 — Created [Successfully created the URI specified by the client.]

202 — Accepted [Accepted for processing but the server has not finished processing it.]

203 — Non-Authoritative Information [Information in the response header did not originate from this server. Copied from another server.]

204 — No Content [Request is complete without any information being sent back in the response.]

205 — Reset Content [Client should reset the current document. Ie. A form with existing values.]

206 — Partial Content [Server has fulfilled the partial GET request for the resource. In response Range request from the client. Or if someone hits

Client Request Incomplete

400 — Bad Request [The server detected a syntax error in the client's request.]

401 — Unauthorized [The request requires user authentication. The server sends the WWW-Authenticate header to indicate the authentication type and realm for the requested resource.]

402 — Payment Required [reserved for future.]

403 — Forbidden [Access to the requested resource is forbidden. The request should not be repeated by the client.]

404 — Not Found [The requested document does not exist on the server.]

405 — Method Not Allowed [The request method used by the client is unacceptable. The server sends the Allow header stating what methods are acceptable to access the requested resource.]

Server Errors

500 — Internal Server Error [A general configuration setting or an application error caused an error.]

501 — Not Implemented [The server does not support the function required to fulfill the request.]

502 — Bad Gateway [The server encountered an invalid response from a downstream server or proxy.]

503 — Service Unavailable [The service is temporarily unavailable. The server can send a Retry-After header to indicate when the service may become available again.]

504 — Gateway Time-Out [The gateway or proxy has timed out.]

505 — HTTP Version Not Supported [The version of HTTP used by the client is not supported.]

Request Redirected

300 — Multiple Choices [Requested resource corresponds to a set of documents. Server sends information about each one and a URL to request them from so that the client can choose.]

301 — Moved Permanently [Requested resource does not exist on the server. A Location header is sent to the client to redirect it to the new URL. Client continues to use the new URL in future requests.]

302 — Moved Temporarily [Requested resource has temporarily moved. A Location header is sent to the client to redirect it to the new URL. Client continues to use the old URL in future requests.]

303 — See Other [The requested resource can be found in a different location indicated by the Location header, and the client should use the GET method to retrieve it.]

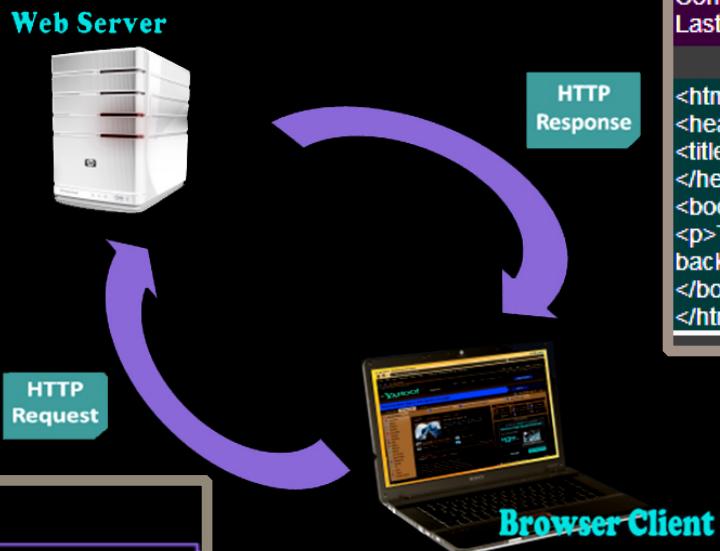
304 — Not Modified [Used to respond to the If-Modified-Since request header. Indicates that the requested document has not been modified since the specified date, and the client should use a cached copy.]

305 — Use Proxy [The client should use a proxy, specified by the Location header, to retrieve the URL.]

307 — Temporary Redirect [The requested resource has been temporarily redirected to a different URL.]

Internet - HTTP

Request/Response Summary



HTTP Request

```
GET /doc/test.html HTTP/1.1
Host: www.test101.com
Accept: image/gif, image/jpeg, /*
Accept-Language: en-us
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0
Content-Length: 35
```

bookId=12345&author=Tan+Ah+Teck

HTTP Response

```
HTTP/1.1 200 OK
Date: Thu, 20 May 2004 21:12:58 GMT
Connection: close
Server: Apache/1.3.27
Accept-Ranges: bytes
Content-Type: text/html
Content-Length: 170
Last-Modified: Tue, 18 May 2004 10:14:49 GMT
```

```
<html>
<head>
<title>Welcome to the Amazing Site!</title>
</head>
<body>
<p>This site is under construction. Please come
back later. Sorry!</p>
</body>
</html>
```

Response document - HTML

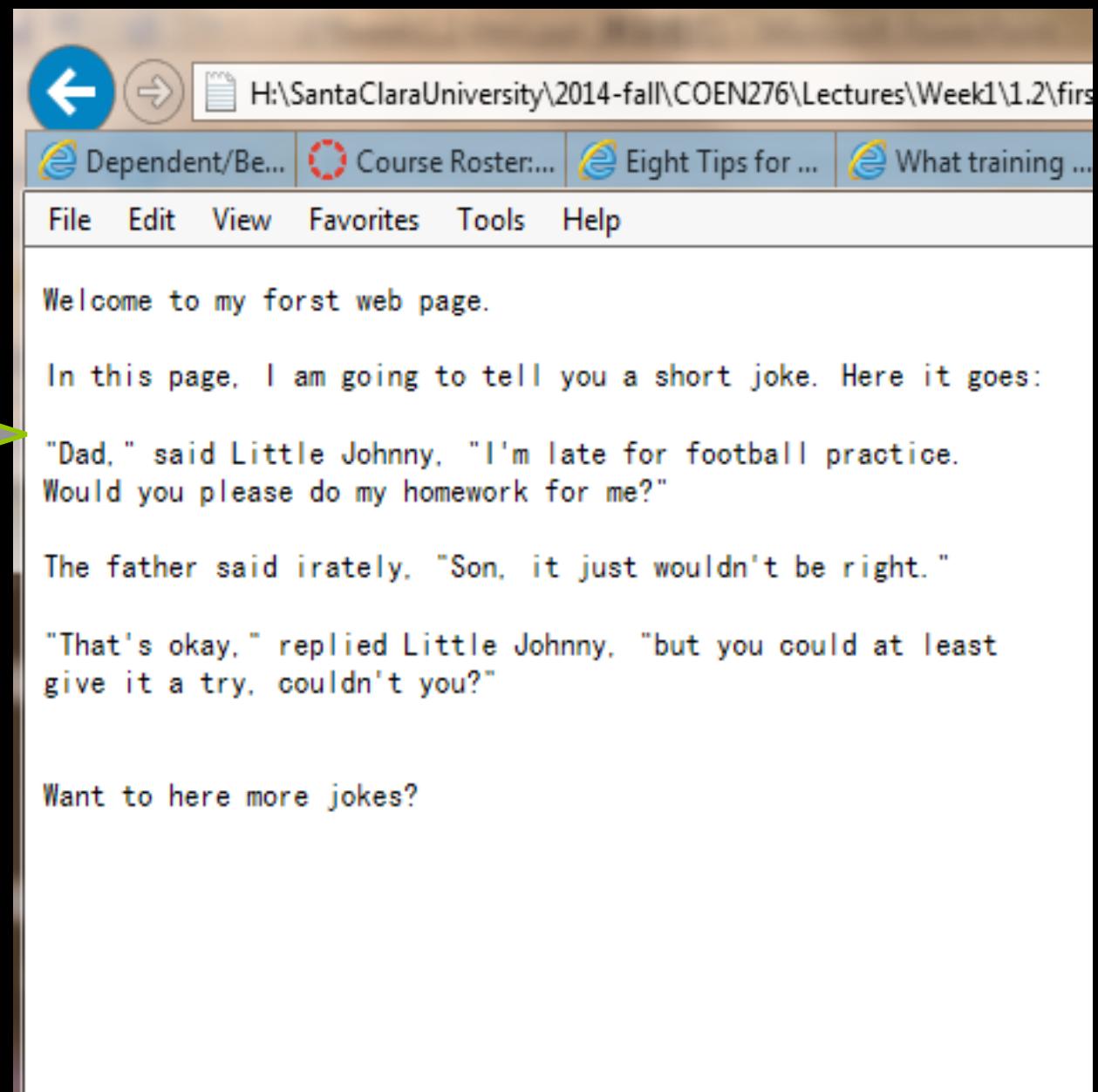
Response document
format
for browser to render



HTML

Plain text
document
is
not
good enough

no font
no link
no image
no controls



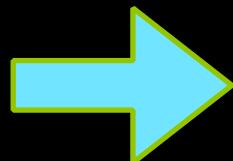
HTML

Use **tag** to tell browser how to display your content:

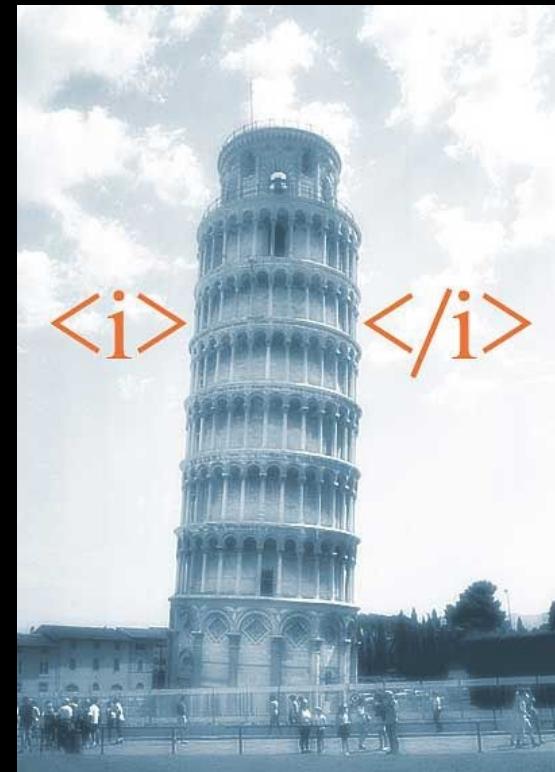
```
<tag-name>your content </tag-name>
```

Example: Italic

```
<i> My Italic Text </i>
```



My Italic Text



HTML

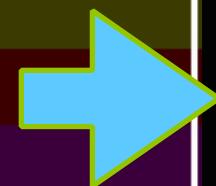
Response received by browser.

Browser will take the response
HTML document and render
in browser window

HTTP Response

```
HTTP/1.1 200 OK
Date: Thu, 20 May 2004 21:12:58 GMT
Connection: close
Server: Apache/1.3.27
Accept-Ranges: bytes
Content-Type: text/html
Content-Length: 170
Last-Modified: Tue, 18 May 2004 10:14:49 GMT
```

```
<html>
<head>
<title>Welcome to the Amazing Site!</title>
</head>
<body>
<p>This site is under construction. Please come
back later. Sorry!</p>
</body>
</html>
```

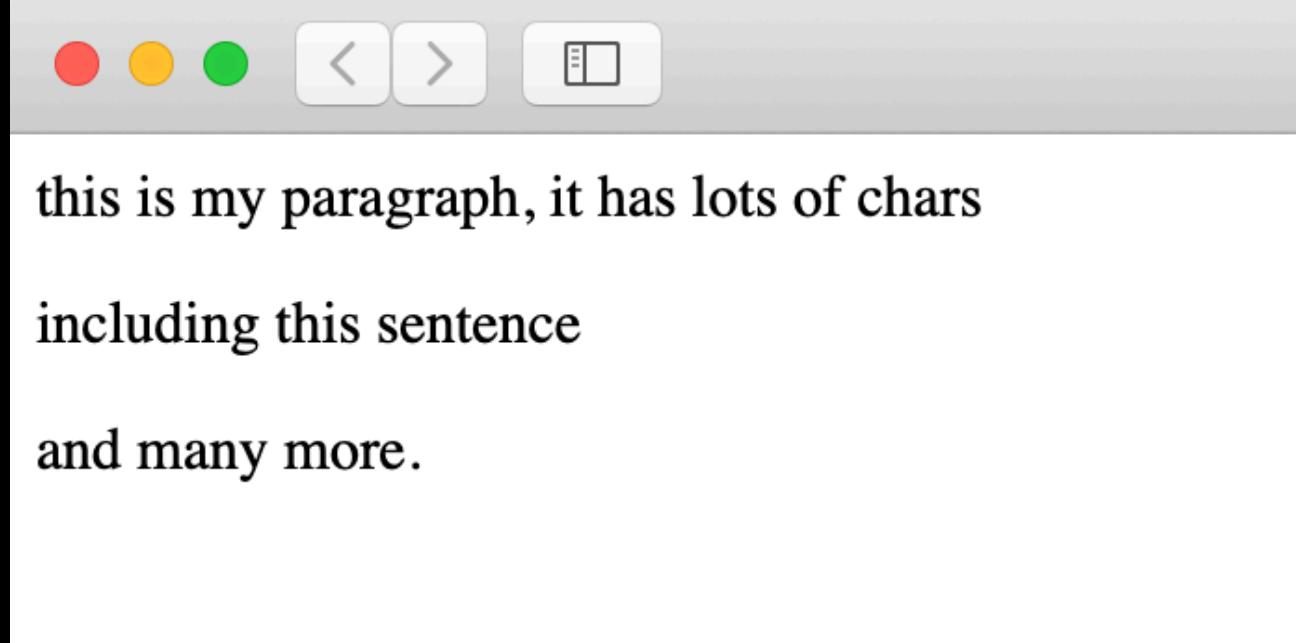


HTML

More Examples: Paragraph

this is my paragraph, it has lots of chars
<p>including this sentence</p> and many more.

in browser:



HTML -tag

More Examples

 - line break

<hr> - horizontal line

<h1>...<h6>

different size headers

<h1>size h1 header</h1>

size h1 header

size h2 header

size h3 header

size h4 header

size h5 header

size h6 header

HTML file



HTML file:
myfile.html

```
<h1><i>Welcome to my forst web page.</i></h1>
<hr/>
<h2> my name is preeti </h2>
<h6> i like jokes </h6>

<p>
In this page, I am going to tell you a short joke. Here it goes:
</p>

<p>
"Dad," said Little Johnny, "I'm late for football practice.
Would you please do my <br>homework for me?"
</p>
<p>
The father said irately, "Son, it just wouldn't be right."
</p>
<p>
"That's okay," replied Little Johnny, "but you could at least
give it a try, couldn't you?"
</p>

Want to here more jokes?
```

HTML document is
composed of
elements

HTML



Don't forget that
the whole html
document itself,
is an
element

<html>
Everything
</html>

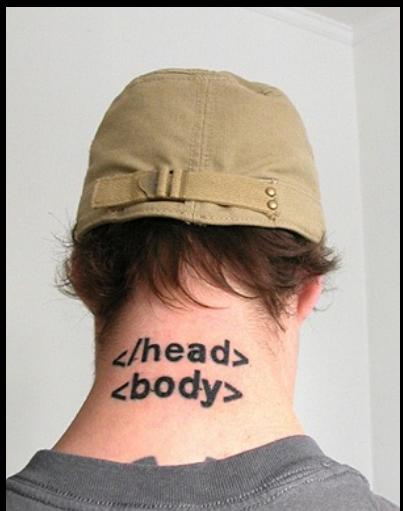
```
<html>
<h1><i>Welcome to my forast web page.</i></h1>
<hr/>
<h2> my name is preeti </h2>
<h6> i like jokes </h6>

<p>
In this page, I am going to tell you a short joke. Here it goes:
</p>

<p>
"Dad," said Little Johnny, "I'm late for football practice.
Would you please do my <br>homework for me?"
</p>
<p>
The father said irately, "Son, it just wouldn't be right."
</p>
<p>
"That's okay," replied Little Johnny, "but you could at least
give it a try, couldn't you?"
</p>

Want to here more jokes?
</hmtl>
```

HTML file structure



head

<HTML>

<Head>

Info about this document :
title, script, style, meta info

</Head>

<Body>

Content

</Body>

</HTML>

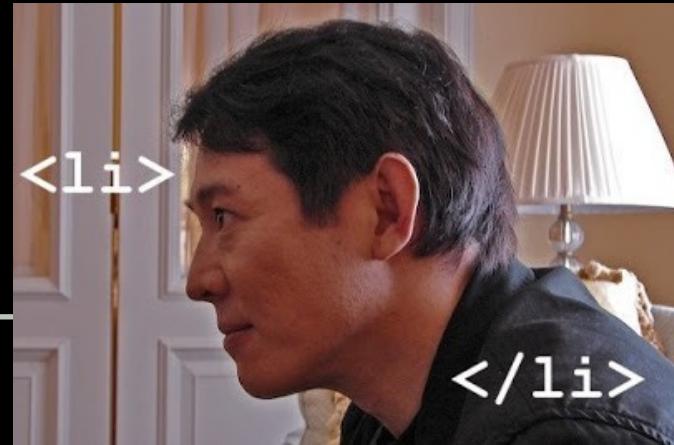
HTML - element

Example: List

Unordered List

- list item
- list item
- list item
- list item

```
<ul>
<li>list item</li>
<li>list item</li>
<li>list item</li>
<li>list item</li>
</ul>
```

Ordered List

- The first list item
- The second list item
- The third list item
- The fourth list item

```
<ol>
```

```
<li> The first list item </li>
<li> The second list item </li>
<li> The third list item </li>
<li> The fourth list item </li>
</ol>
```

HTML - element

Table

```
<table>
  <tr>
    <td>Madura</td>
    <td>1026981</td>
    <td>ComEng</td>
  </tr>
  -----
  </table>
```

<table>

Name	ID	Section
Madura Acharya	00001026981	Computer Engineering 276:Web Programming I-TR 07:10 PM
Neeta Bhat	00001033696	Computer Engineering 276:Web Programming I-TR 07:10 PM
Priya Bhat	00001065636	Computer Engineering 276:Web Programming I-TR 07:10 PM
Ahmad Boostani	00001107202	Computer Engineering 276:Web Programming I-TR 07:10 PM
Kavyaramesh Chandra	00001027284	Computer Engineering 276:Web Programming I-TR 07:10 PM

</table>

HTML - element

- image attributes



```

```

```
<img src= "url" alt="image here">
```

attributes

HTML - element

Attribute - Further specify element's property.

Example:

font color

font size

width

background color

border (e.g. table)

colspan/rowspan

type (e.g. list)

id

class



HTML - element

hyperlink - anchor

<**aa**>

with property:

<**a href="**http://somewhere**"> this is a link **

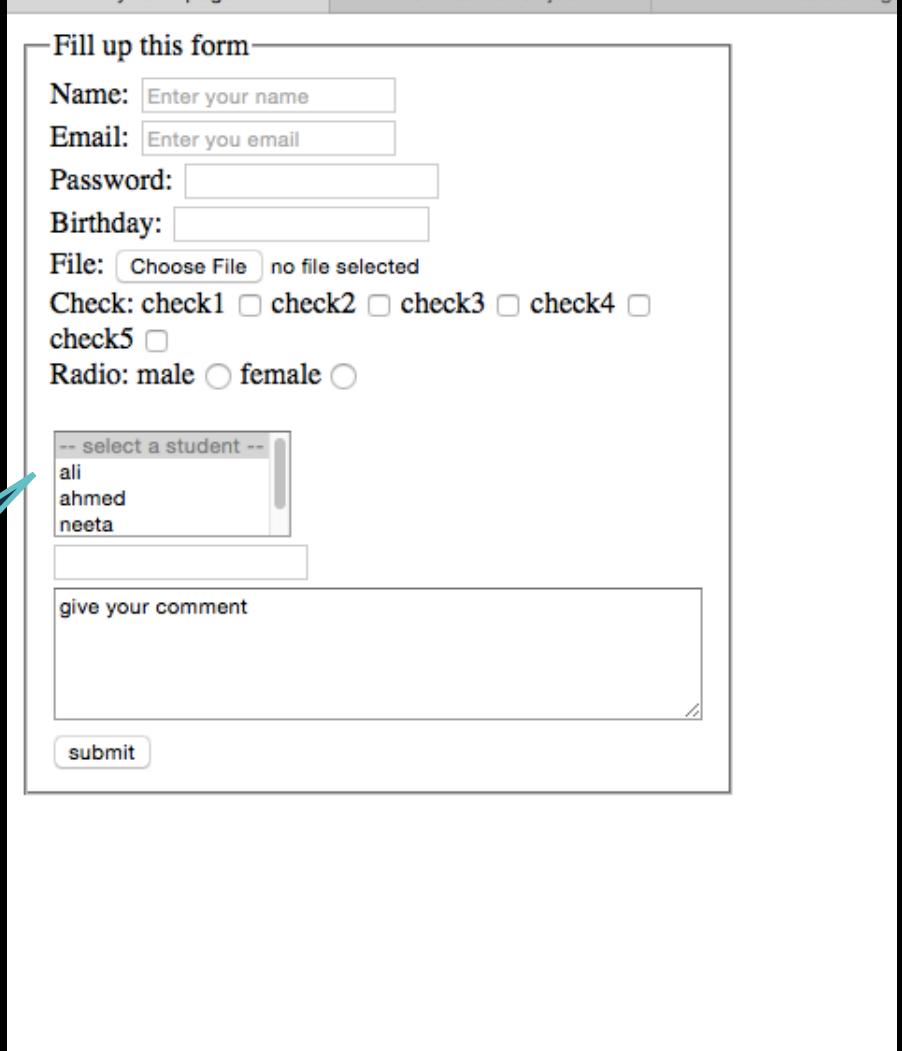
<**a href="**http://somewhere**"> **

HTML - element

- **form:**
**for user to submit
information**

<form attributes=...>
different types of <input>
</form>

**different types of
input**



Fill up this form

Name:

Email:

Password:

Birthday:

File: no file selected

Check: check1 check2 check3 check4 check5

Radio: male female

-- select a student --
ali
ahmed
neeta

give your comment

submit

HTML - element

- **form:**

attributes

Sr.No	Attribute & Description
1	action Backend script ready to process your passed data.
2	method Method to be used to upload data. The most frequently used are GET and POST methods.
3	target Specify the target window or frame where the result of the script will be displayed. It takes values like _blank, _self, _parent etc.
4	enctype You can use the enctype attribute to specify how the browser encodes the data before it sends it to the server. Possible values are – application/x-www-form-urlencoded – This is the standard method most forms use in simple scenarios. multipart/form-data – This is used when you want to upload binary data in the form of files like image, word file etc.

HTML - element

form:

```
<form action = “/login” method = “post”>  
    User ID : <input type = "text" name = "user_id" value = “yuan”>  
    <br>  
    Password: <input type = "password" name = "password" value = “123”>  
</form>
```

name

is used to identify this input

which is sent to the server to recognize and get the value.

name/value pair is used for this information:

user_id: yuan

password: 123

HTML – Tag (Element) Summary

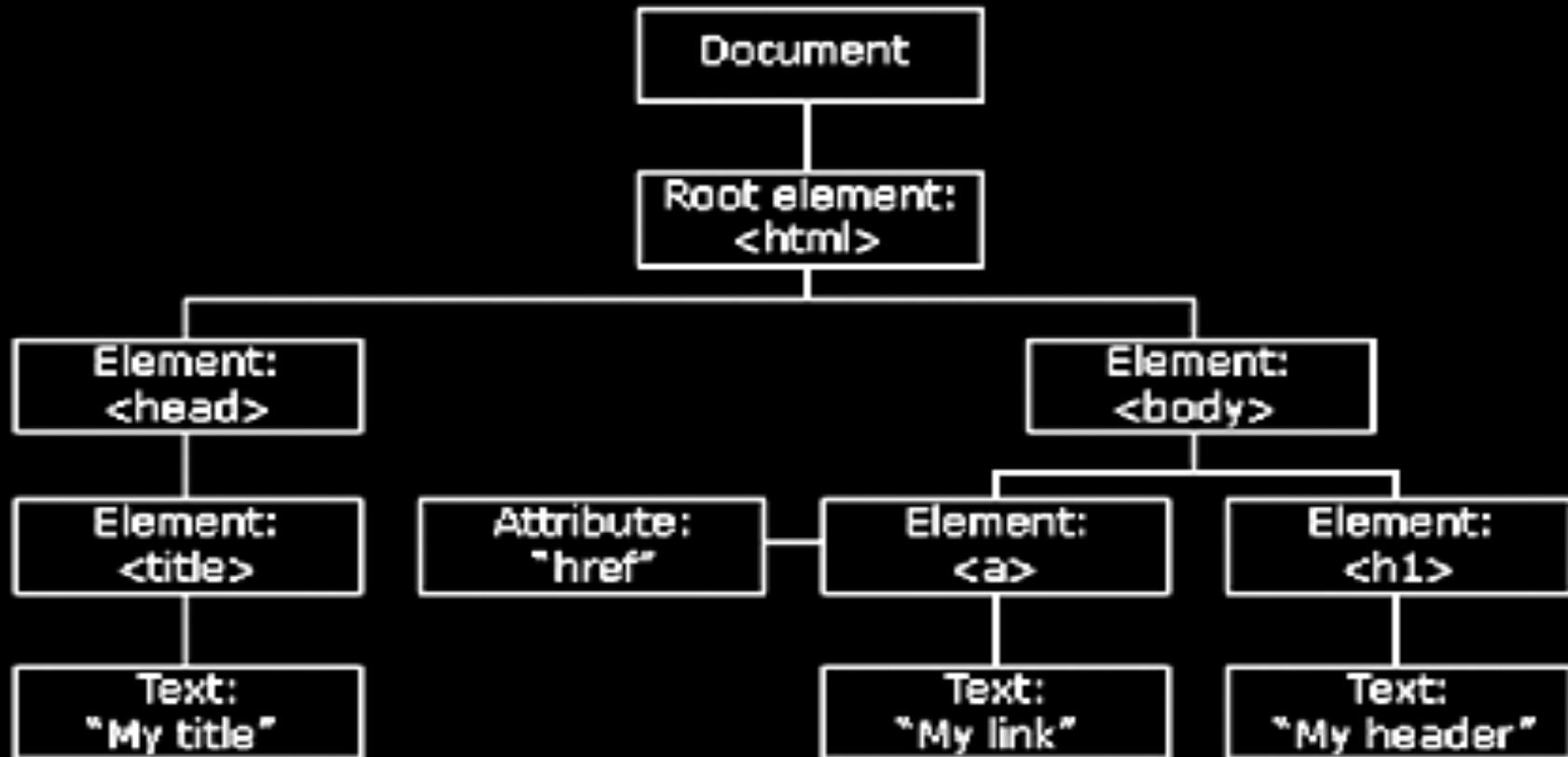
Block – rectangular area, with newline
example: paragraph, list, div

Inline -- within a line

example: img, anchor, span

Empty – no content
no closing tag or <empty />
example: ,

HTML – structure in different view



HTML5

- **Audio and Video**

<audio controls>

<source src="audio.mp3" type="audio/mpeg"

</audio>

<video controls>

<source src="video.mp4" type="video/mp4"

</video>

HTML - validation

- **<https://validator.w3.org/>**

**check if your file meet
official specification**

- provide a link
- upload your file
- copy and paste your code

HTML - <tag> for today

- <h>
- <p>
- <hr>
-
- <a>anchor (link)
- <list>
- <table>
- <pre>
- entity
- <Audio> / <Video>
- don't forget about <html> <head> <body>

More HTML <tag>

Optional

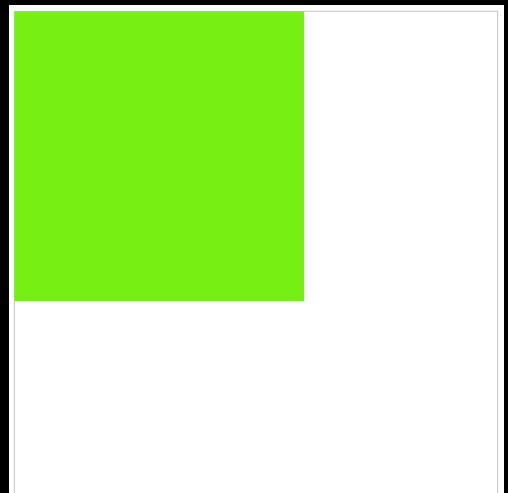
More HTML <tag>

- HTML5 Canvas Element

```
<canvas id="myCanvas" style="width:500;height:220">  
    Your browser does not support the HTML5 canvas tag.  
</canvas>
```

<script>

```
var c = document.getElementById("myCanvas");  
var ctx = c.getContext("2d");  
ctx.fillStyle = "#00FF00";  
ctx.fillRect(0,0,200,200);  
</script>
```



<audio>

```
<audio controls loop autoplay muted>
  <source src="yourbody.mp3" type="audio/mpeg">
    Your browser does not support audio.
</audio>
```

type: audio/wav, audio/ogg, audio/mpeg

text: if audio is not supported, will be displayed

controls: play/pause/volume button

loop:

autoplay:

src: audio file



<video>

```
<video width="320" height="240" controls poster="flower.jpg">
  <source src="tennis.mp4" type="video/mp4">
    Your browser does not support video.
</video>
```

text: if video is not supported, will be displayed

type: video/ogg, video/mp4

controls: play/pause/volume/full screen button

loop:

autoplay:

src: video file

width, height: dimension

poster: specify an image while downloading or until user click play

SVG **(Scalable Vector Graphics)**

Can be used to draw different shapes like
rectangle, circle, curves

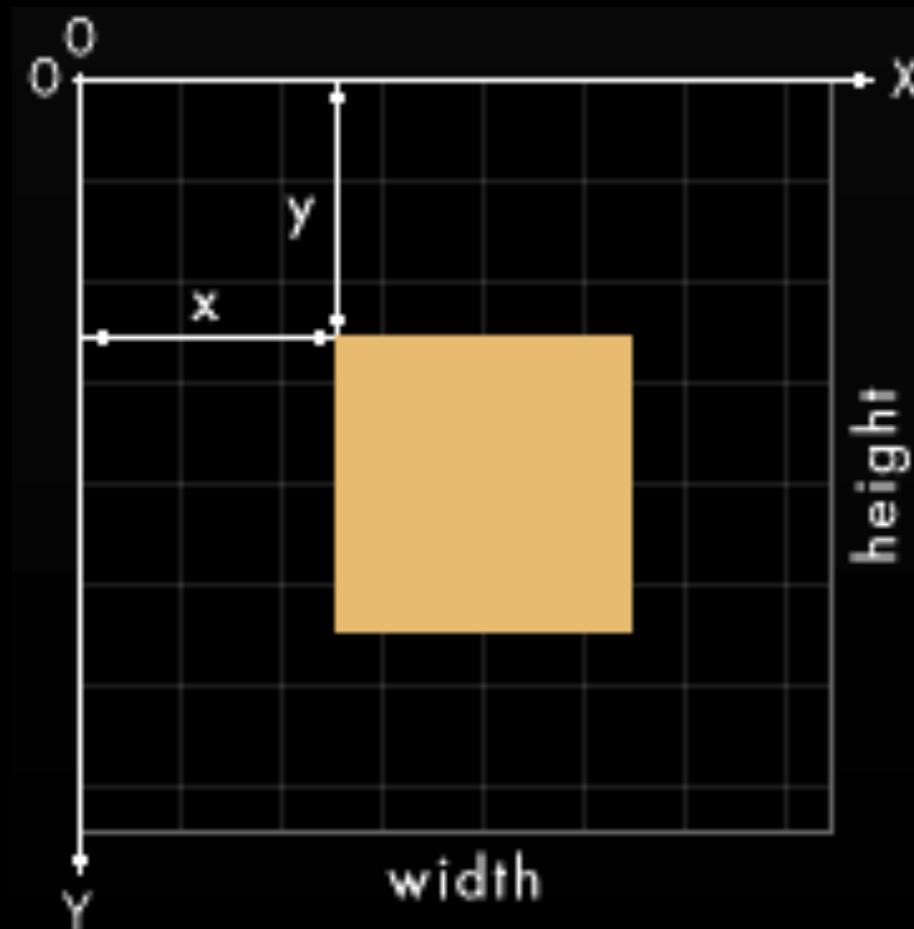
Use <**svg**> in document

You can use that to create **.svg** file on its own.

Or embed <**svg**> into **.html** file

SVG (Scalable Vector Graphics)

Coordinate



SVG

How to use: set a graphics area
and specify shapes you want within the area

```
<svg width="320" height="240">  
specify shapes here  
</svg>
```

<svg> defined an area for graphics
width, height: dimension of your whole
graphics area.

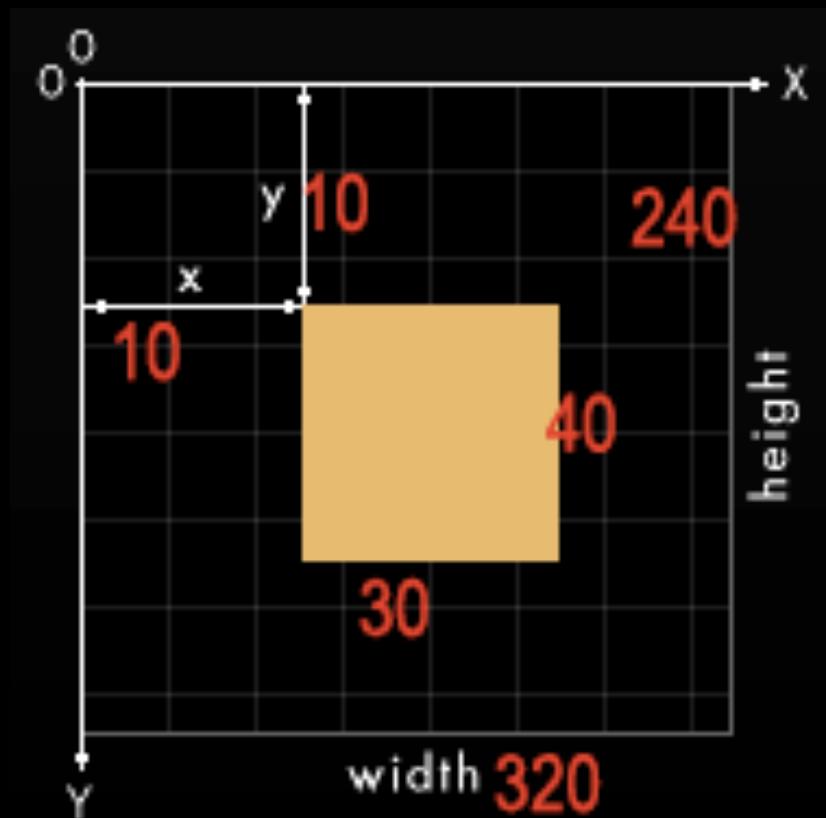
SVG - different shapes

```
<svg width="320" height="240">  
<rect />  
<circle />  
<line />  
<ellipse />  
<polygon />  
</svg>
```

Of course, those shapes need **attributes** to specify

SVG - ranctangle

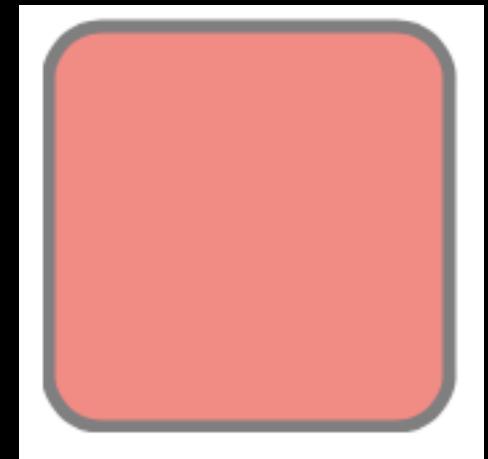
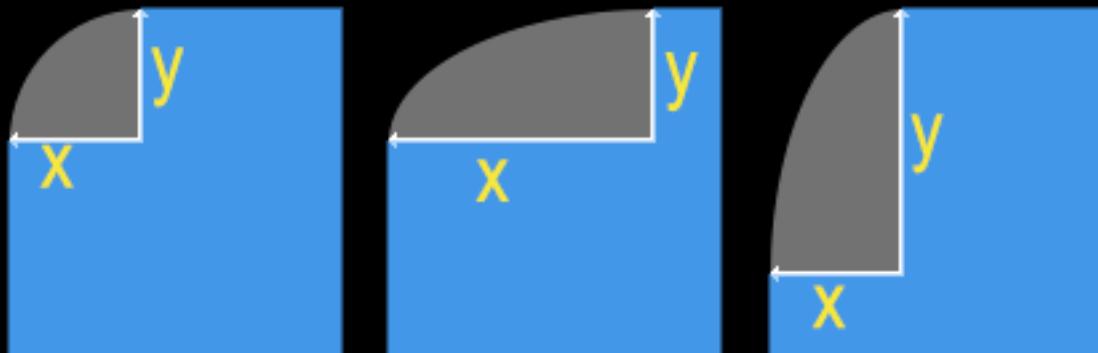
```
<svg width="320" height="240">  
<rect x="10" y="10" width="30" height="40"  
style="fill:rgb(0,0,255);  
stroke-width:3;  
stroke:rgb(0,0,0)">  
</svg>
```



SVG - rectangle (round corner)

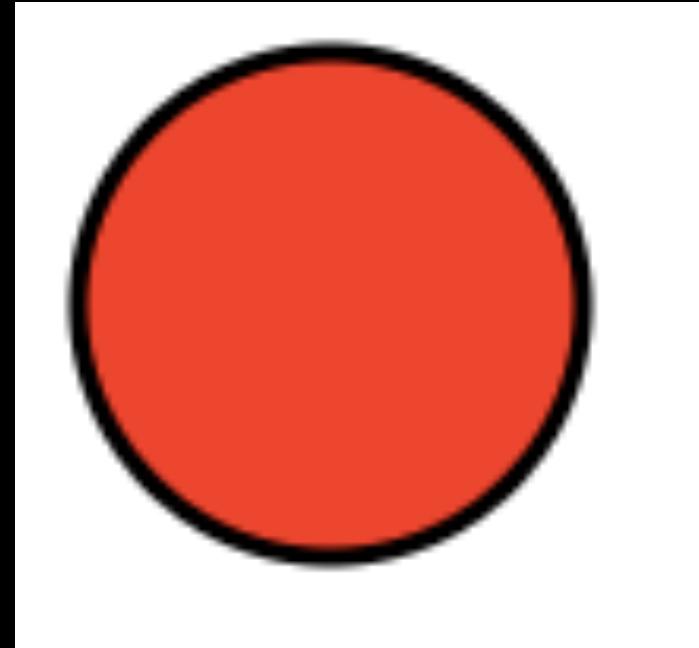
```
<svg width="320" height="240">  
<rect x="10" y="10" rx="20" ry="30"  
      width="30" height="40"  
      style="fill:rgb(0,0,255);  
              stroke-width:3;  
              stroke:rgb(0,0,0); opacity:0.5"/>  
</svg>
```

rx, ry: dimension for round corner



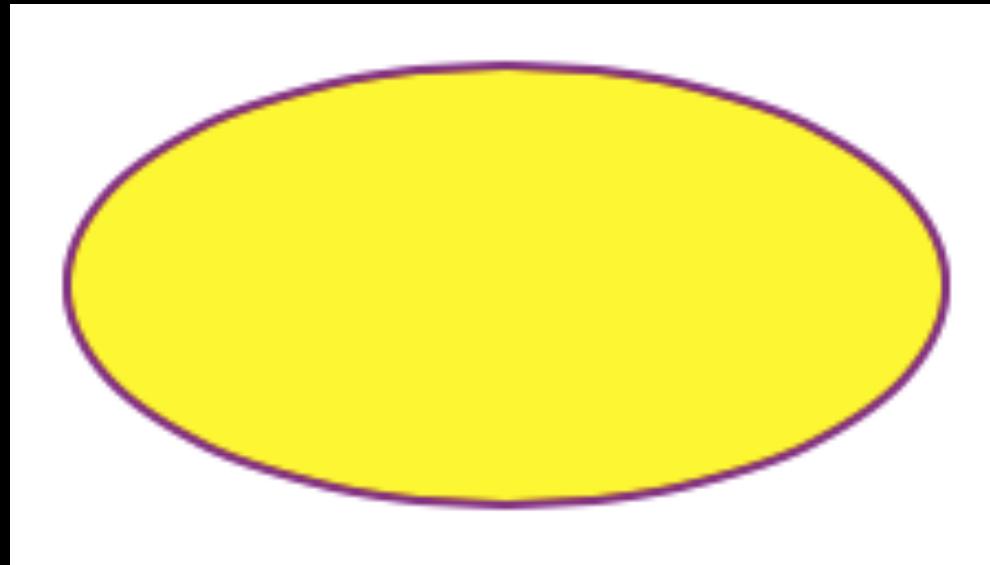
SVG - circle

```
<svg width="320" height="240">  
<circle cx="60" cy="60" r="40"  
      style="fill:red;  
              stroke-width:3;  
              stroke:black"/>  
</svg>
```



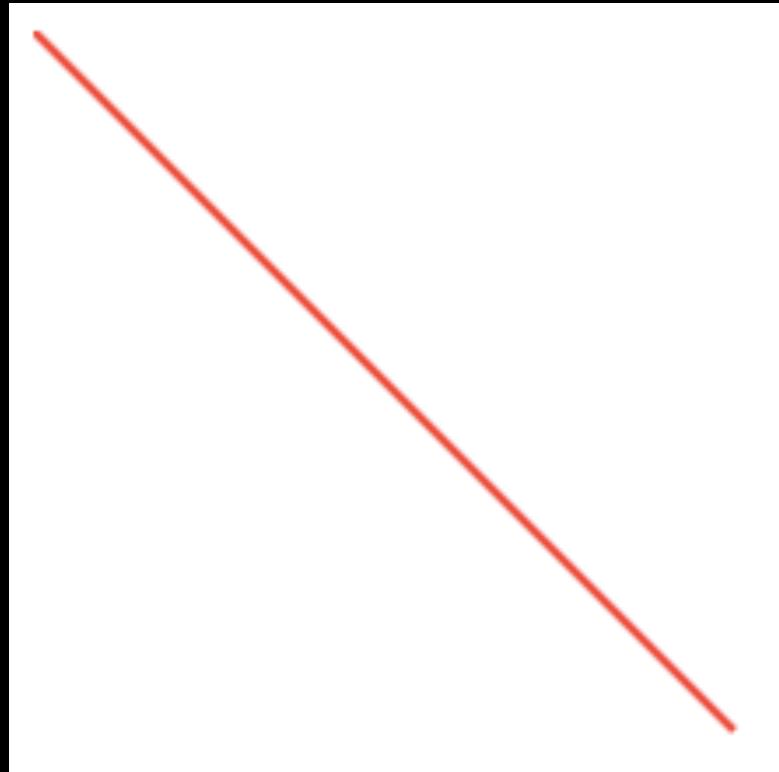
SVG - ellipse

```
<svg width="320" height="240">  
<circle cx="60" cy="60" rx="40" ry="60"  
    style="fill:yellow;  
    stroke-width:2;  
    stroke:purple"/>  
</svg>
```



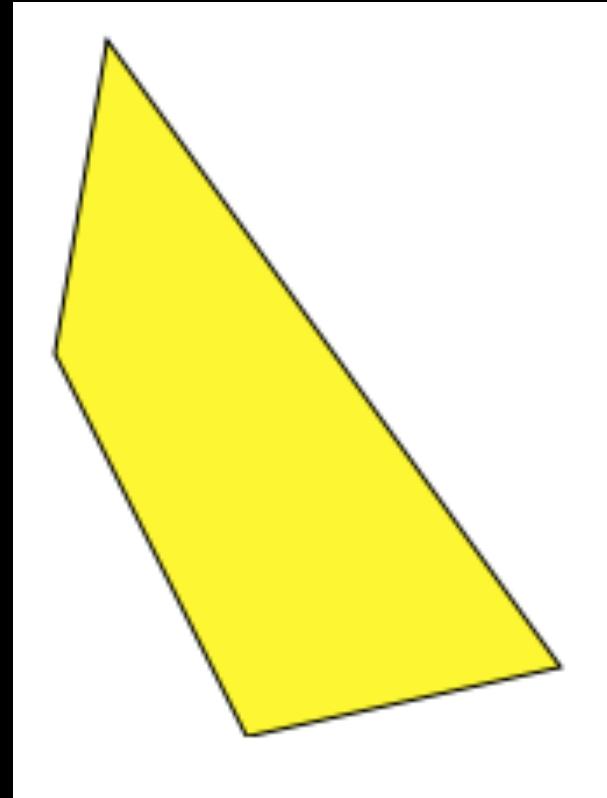
SVG - line

```
<svg width="320" height="240">  
<line x1="20" y1="20" x2="70" y2="70"  
      style="stroke-width:2;  
              stroke:red"/>  
</svg>
```



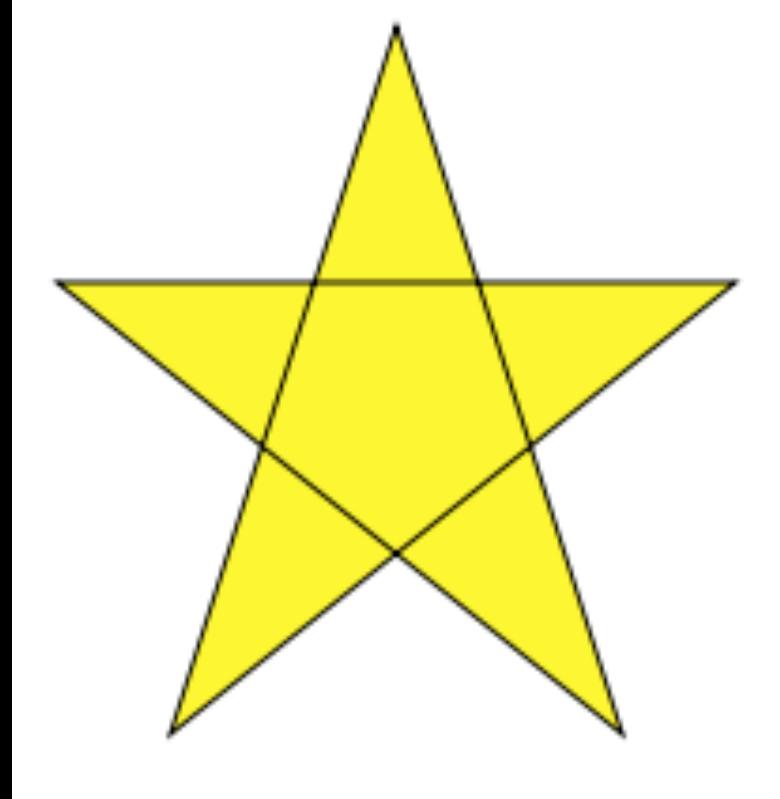
SVG - polygon

```
<svg width="320" height="240">  
<polygon points="20,10 150,190 60,210 5 100"  
         style="stroke-width:2;  
                 stroke:black;  
                 fill:yellow"/>  
</svg>
```



SVG - polygon

```
<svg width="320" height="240">  
<polygon points="100,10 40,198 190,78 10,78 160,198"  
style="stroke-width:2;  
stroke:black;  
fill:yellow"/>  
</svg>
```



SVG - polyline

```
<svg width="320" height="240">
```

```
<polyline points=
```

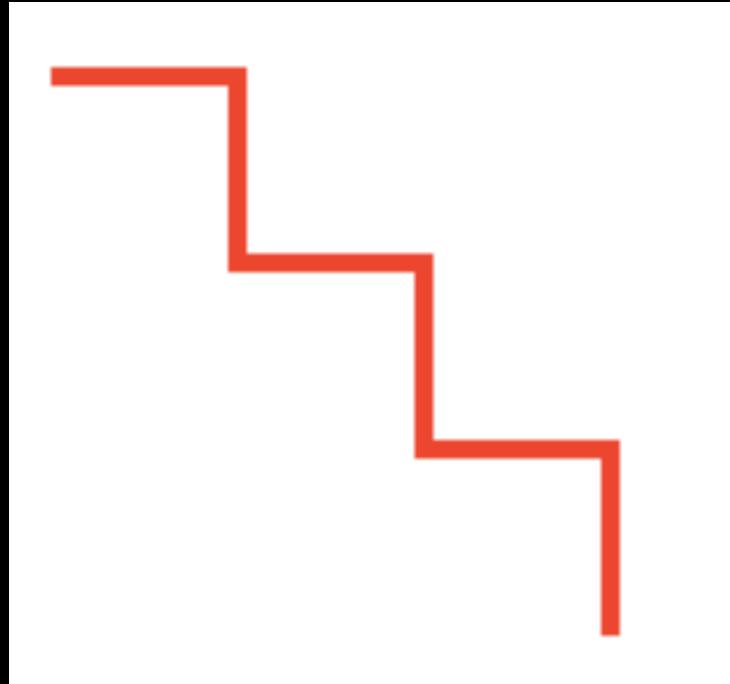
```
“0,40 40,40 40,80 80,80 80,120 120,120 120,160”
```

```
style=“stroke-width:3;
```

```
stroke:red;
```

```
fill:none/>
```

```
</svg>
```



SVG - text

```
<svg width="320" height="240">  
<text x="10" y="10" transform="rotate(30)"  
      style="fill:red> I love you  
</text>  
</svg>
```

rotate

Rotate all characters by this degree.



END