CS 4032 – Web Programming

National University
Of Computer and Emerging Sciences

Course Objectives

- Understanding of modern web application development technologies
- Develop and design Web interfaces using latest UI frameworks
- Understand the best web development practices being followed in the industry
- Training on industry-oriented web frameworks
- Utilize the incredible power of web technologies
  - Develop web based of MEAN, MERN, and advance technologies.

Course is really different mini-courses

Frontend Technologies

HTML5, CSS3, JavaScript
Angular, ReactJS etc

Backend Technologies

NodeJS, ExpressJS,
MongoDB and others

.net Technologies
Asp.net MVC

Web Service Technologies

Amazon AWS
Google Cloud
Azure Services

Target Students

- Want to pursue good career as a **Full Stack Web Developer/Web Designer**
- Want to learn cutting-edge Web development technologies
- Want to get hands-on practice on industry-oriented web frameworks
- Feel Comfortable in coding (e.g., C++, Java, Python...)
  - Has built, or could build, a single-user application
- Ready to take coding challenges

Contents and Organization

Topics to be covered:

List of Topics	No. of Weeks	Contact Hours	CLO(s)
Introduction to Web Development Front-end vs Back-end Development	1	3	1,2,7,12
HTML, HTML5, CSS, CSS3, Bootstrap, tailwind	2	6	1,2,3,5,8
JavaScript fundamentals , JQuery	1	3	1,2,3,5,8
AJAX, JQUERY, FETCH, AXIOS	1	3	1,2,3,5,8
State management and Data Bindings	1	3	1,3,4,12
MongoDB	1	3	5,12,6
Introduction to Node	1	3	1,2,3,5
Introduction to Express	1	3	1,2,3,5
Programming with Angular and Typescript	2	6	1,3,4,5,11,12
Introduction to React Asynchronous JavaScript	2	6	1,2,3,5
Introduction to Vue	1	3	1,2,3,5
Advance topics- Serverless , GraphQL	1	6	1,2,3,4, 5,12
Deployment and Web Programming practices and Demos	1	3	1,6,7,9,10,11
Total	16	48	

Tentative Marks Distribution

Assessment Item	Number	Weight (%)
Quizzes/Tasks	>=8	15
Assignments	>=5	12-15
Sessional Exam	2	20-30
Project	1	10-15
Final Exam	1	40-50

Grading policy: *Absolute grading scheme*

Class Policies and Guidelines

Attendance policy:

- **Will be taken at the start of the class**. Students ***appearing late in the class after the attendance*** will be marked **"Absent"**
- **80%** attendance is compulsory

Plagiarism policy:

- Plagiarism in **midterm/final** exam may result in **F grade** in the course.
- Plagiarism in an assignment items (**assignments, quizzes & project**) will result in zero marks in the whole assignments items category. If fore mentioned act is repeated more than once the instructor can refer a case to the Department Disciplinary Committee (the maximum punishment can be award of **'F' grade** in that course.)

Class Policies and Guidelines

Course retake policy:

- **Sessionals/final exam retake**
  - The examination assessment and retake committee decide the exam retake/pretake cases.
- **Assignments/quizzes retake**
  - There will be **no retake** of any assignment or quiz.

Class Policies and Guidelines

Quiz Policy:

- Class Tasks would be considered as QUIZ.
- Submission of Class Task would be required.
- Demo of Class Tasks are also required to get marked.
- **Re-take of Demo would NOT be taken.**
- **On Missing Demo**
  - **Evaluation would be based on Code submission**
  - **40% deduction would be applied.**

Class Policies and Guidelines

Tasks, Assignments or project submission policy:

- Use the following rules for your submissions
  - Combine all files in one **zip** file
  - Name the zip file as **ROLLNO_NAME_SECTION.zip**
  - Submit the zip file on **github-Classroom** before the deadline
    - Your Github Account should be with your fullName
- **On Missing Assignment/Project Demo:**
  - **Evaluation would be based on Code submission**
  - **40% deduction would be applied.**

Class Policies and Guidelines

Late Submission policy:

- Late for First hour – 10 % deduction
- Late for 12 hours – 20% deduction
- Late for 24 hours – 30% deduction
- No submission after 24 hours – zero marks

Assignments and Projects

- Where **~80%** of your learning will take place
- Posted to **Google Classroom**
- All **tasks**, **assignments** will be individual or announced otherwise
- **Project** will be in group (max. 2-3 students)
- Program must work, compile errors / runtime errors lose all correctness points
- Copying solution code or giving code to someone else is **CHEATING -> F** in the course.

Class Policies and Guidelines

- **Don'ts**
  - Use of cell phones
  - Discussion with fellows during class
  - Early leave
  - Frequent movements In-out of class

- **Do's**
  - Bring your own laptops along with chargers- **Mandatory**
  - Be interactive, ask questions
  - Participate in the lecture especially during hands-on practice/Tasks

Textbooks and Reference Material

- **Web Application Architecture Principles, protocols and practices** by Leon Shklar and Richard Rosen
- **Learning JavaScript**, 3rd Edition by Todd Brown
- Internet is best to learn web programming

Google Classroom

- Class code
  - **Gcnmedj (sec C/D)**
  - **Kgy7h46 (Sec A)**
  - **2occ3dx   (Sec B)**
- Lectures, reference material, announcements

Contact Information

About me:

Sana Aurangzeb
– **Ph.D.** from NUCES-FAST, 2024
– **Specialization**: Malware Analysis

Contact Information:

– Office: C-502G
– Phone Ext: 698
– Email: sana.aurangzeb@nu.edu.pk

Introduction to Web

CORE TECHNOLOGIES

HTML CSS JavaScript

CSS PRE-PROCESSORS

Sass Less

JS FRAMEWORKS & LIBRARIES

React Redux Angular Vue Backbone Ember jQuery

TASK RUNNERS

Gulp Grunt

DEPENDENCY & PACKAGE MANAGERS

NPM Yarn Webpack

CSS FRAMEWORKS

Bootstrap Foundation Bulma

MISC TECH

node.js GraphQL

TESTING

Mocha Chai Jest

ADDITIONAL SKILLS

Accessibility

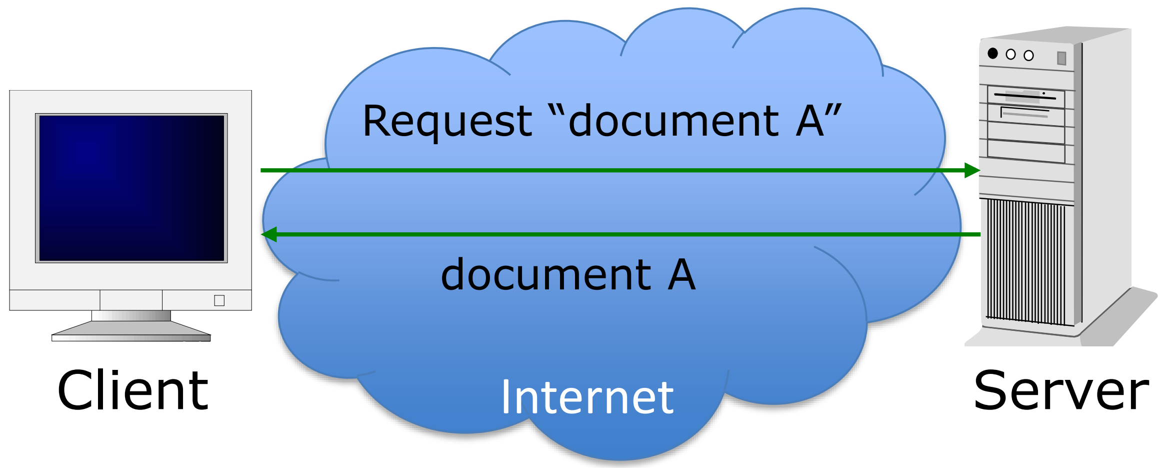
Agile

GitHub & Git

Search Engine Optimization (SEO)

Web Essentials

- **Client**: web browsers, used to surf the Web
- **Server systems**: used to supply information to these browsers
- **Computer networks**: used to support the browser-server communication



Internet v.s. Web

The Internet: a inter-connected computer networks, linked by wires, cables, wireless connections, etc.

Web: a collection of interconnected documents and other resources.

The world wide web (**WWW**) is accessible via the Internet, as are many other services including email, file sharing, etc.

National University
Of Computer and Emerging Sciences

How does the Internet Work?

Through communication protocols

A **communication protocol** is a specification of how communication between two computers will be carried out

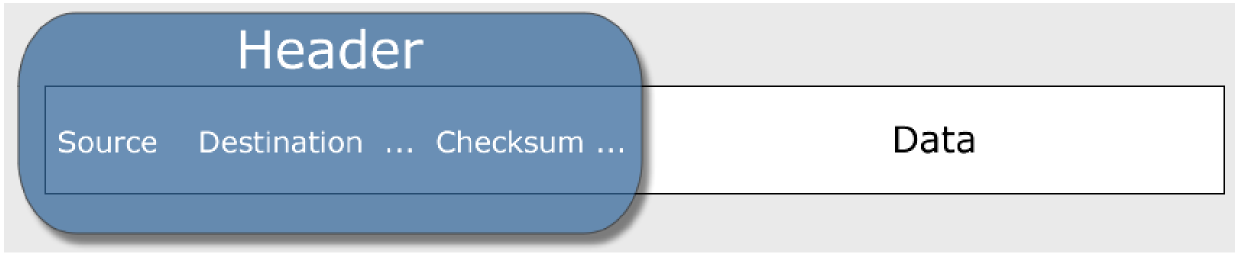
IP (Internet Protocol): defines the packets that carry blocks of data from one node to another

TCP (Transmission Control Protocol) and **UDP** (User Datagram Protocol): the protocols by which one host sends data to another.

Other application protocols: **DNS** (Domain Name Service), **SMTP** (Simple Mail Transfer Protocol), and **FTP** (File Transfer Protocol)

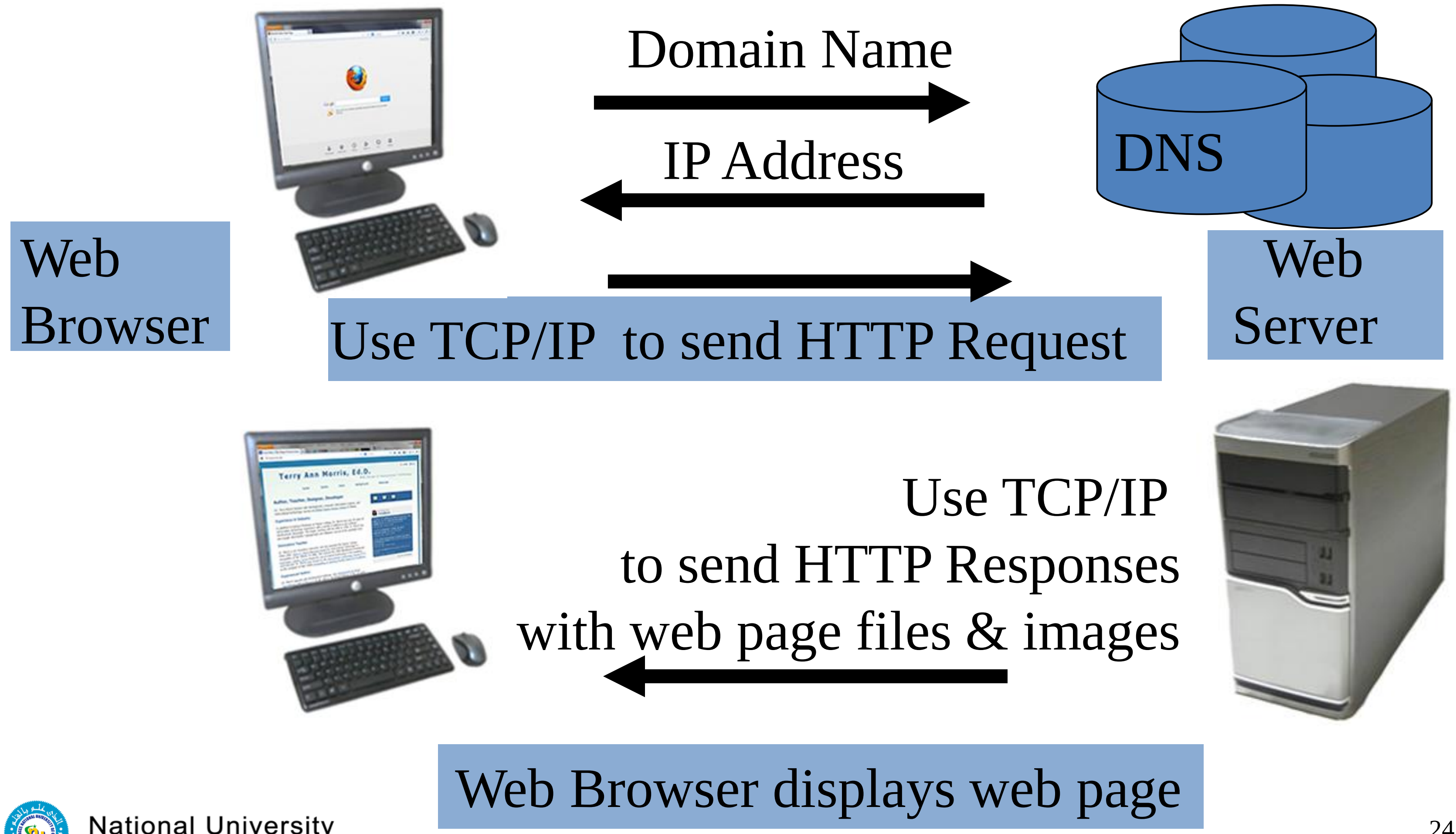
The Internet Protocol (IP)

- A key element of IP is IP address, a 32-bit number
- The Internet authorities assign ranges of numbers to different organizations
- IP is responsible for moving packet of data from node to node
- A packet contains information such as the data to be transferred, the source and destination IP addresses, etc.
- Packets are sent through different local network through gateways
- A checksum is created to ensure the correctness of the data; corrupted packets are discarded
- IP-based communication is unreliable



Domain Name System

- The Domain Name System (DNS) associates Domain Names with IP addresses.


Domain Name
IP Address
DNS
Web Browser
Web Server
Use TCP/IP to send HTTP Request
Use TCP/IP to send HTTP Responses with web page files & images
Web Browser displays web page

Domain Name

- Locates an organization or other entity on the Internet
- Domain Name System (DNS)
  - Divides the Internet into logical groups and understandable names
  - Associates unique computer IP Addresses with the text-based domain names you type into a web browser
  - Browser: http://google.com
  - IP Address: 173.194.116.72

The Transmission Control Protocol (TCP)

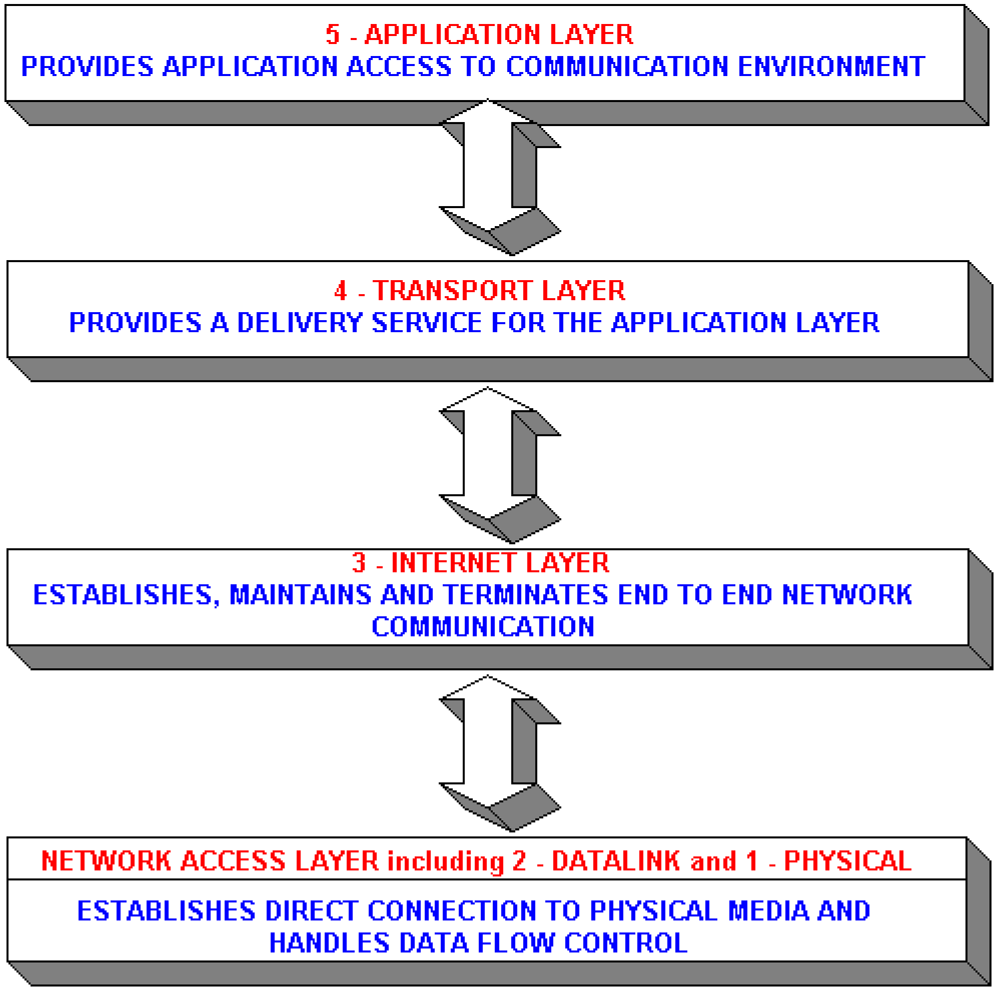
- TCP is a higher-level protocol that extends IP to provide additional functionality
  - reliable communication
  - two-way (full duplex) communication

- TCP adds support to detect errors or lost data and to trigger retransmission until the data is correctly and completely received

- Connection

- Acknowledgment

TCP/IP Protocol Suites


5 - APPLICATION LAYER
PROVIDES APPLICATION ACCESS TO COMMUNICATION ENVIRONMENT
4 - TRANSPORT LAYER
PROVIDES A DELIVERY SERVICE FOR THE APPLICATION LAYER
3 - INTERNET LAYER
ESTABLISHES, MAINTAINS AND TERMINATES END TO END NETWORK COMMUNICATION
NETWORK ACCESS LAYER including 2 - DATALINK and 1 - PHYSICAL
ESTABLISHES DIRECT CONNECTION TO PHYSICAL MEDIA AND HANDLES DATA FLOW CONTROL

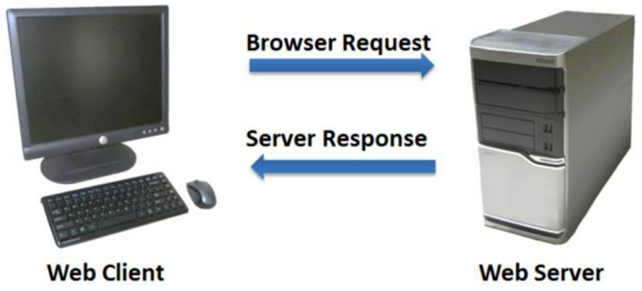
HTTP, FTP, Telnet, DNS, SMTP, etc.

TCP, UDP

IP (IPv4, IPv6)

The World Wide Web (WWW)

- WWW is a system of interlinked, hypertext documents that runs over the Internet
- Two types of software:
  - **Client**: a system that wishes to access the information provided by servers must run client software (e.g., web browser)
  - **Server**: an internet-connected computer that wishes to provide information to others must run server software
  - Client and server applications communicate over the Internet by following a protocol built on top of TCP/IP – HyperText Transport Protocol (HTTP)

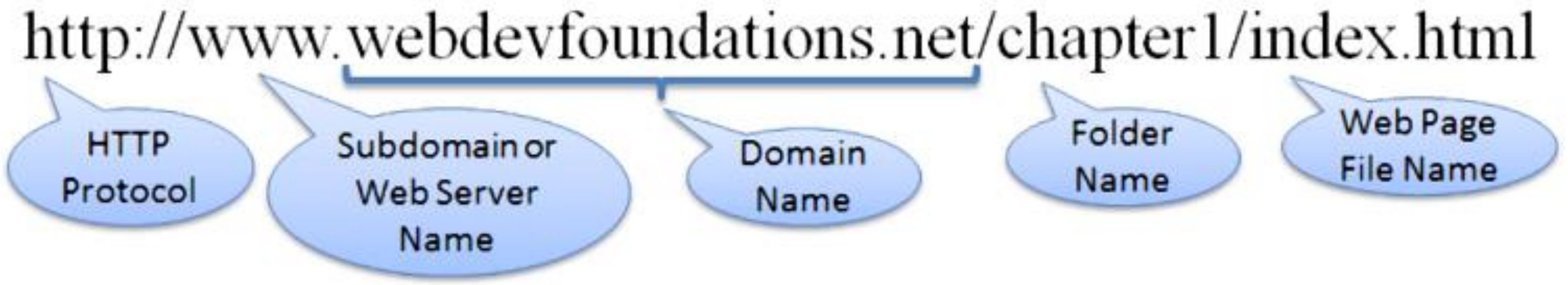


Basics of the WWW

- **Hypertext**: a format of information which allows one to move from one part of a document to another or from one document to another through hyperlinks
- **Uniform Resource Locator (URL)**: unique identifiers used to locate a particular resource on the network
- **Markup language**: defines the structure and content of hypertext documents

Uniform Resource Identifier

- URI – Uniform Resource Identifier
  - identifies a resource on the Internet either by location, name
- URL – Uniform Resource Locator
  - a type of URI which represents the network location of a resource such as a web page, a graphic file, or an MP3 file.


http://www.webdevfoundations.net/chapter1/index.html
HTTP Protocol
Subdomain or Web Server Name
Domain Name
Folder Name
Web Page File Name

Top-Level Domain (TLD) Name

- A top-level domain (TLD) identifies the right-most part of the domain name.

- Examples of generic TLDs:
  .com, .org, .net, .mil, .gov, .edu, .int, .aero, .asia, .cat, .jobs, .name, .biz, .mobi, .museum, .info, .coop, .post, .pro, .tel, .travel

County Code TLDs

- Two character codes originally intended to indicate the geographical location (country) of the web site.
- In practice, it is fairly easy to obtain a domain name with a country code TLD that is not local to the registrant.
- Examples:
  - .tv, .ws, .au, .jp, .uk
  - See http://www.iana.org/cctld/cctld-whois.htm

Request inside envelope

- GET / HTTP / 1.1
- Host: www.example.com

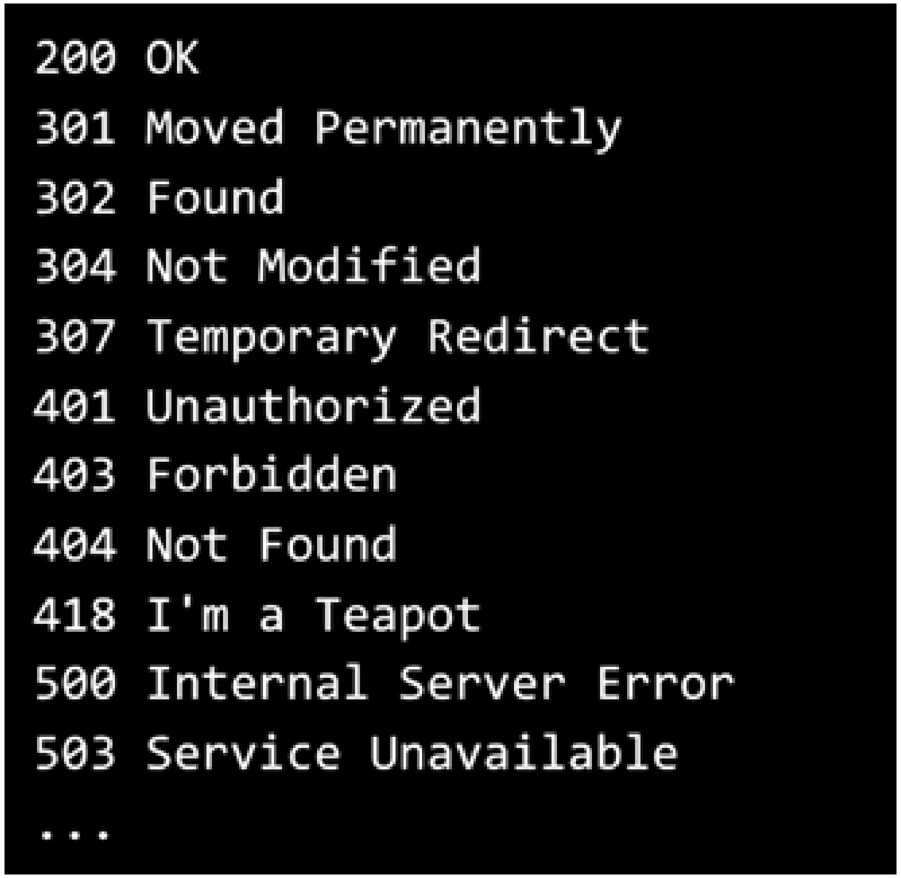
- GET /index.html HTTP/ 1.1
- Host: www.example.com

Response

- HTTP / 1.1 200 OK
- Content-type: text/html

Request inside envelope

- GET / HTTP / 1.1
- Host: www.harvaed.edu

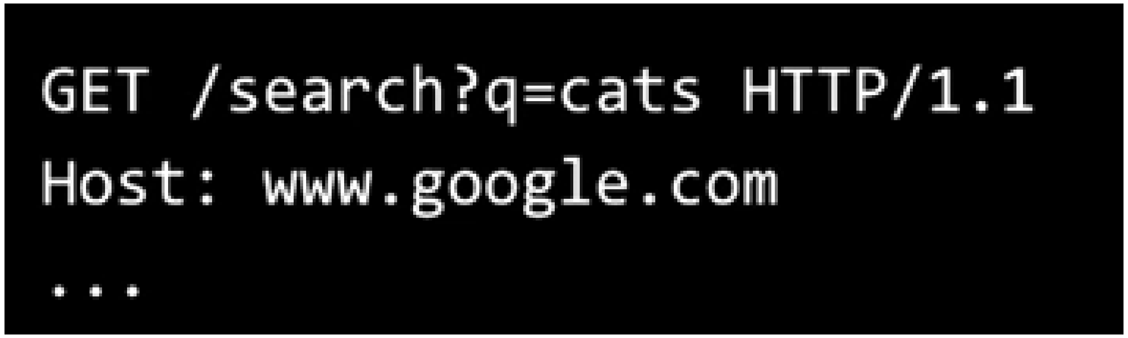


```
200 OK
301 Moved Permanently
302 Found
304 Not Modified
307 Temporary Redirect
401 Unauthorized
403 Forbidden
404 Not Found
418 I'm a Teapot
500 Internal Server Error
503 Service Unavailable
...
```

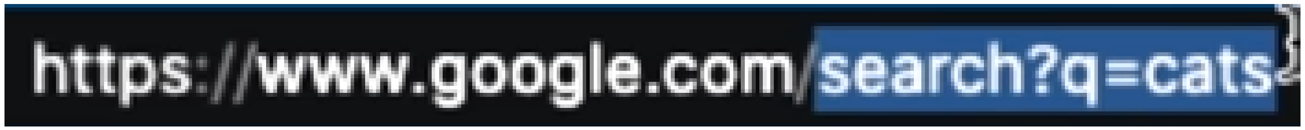
Check status code

- Try Open Safetyschool.org and check the status code

Other commands



```
GET /search?q=cats HTTP/1.1
Host: www.google.com
...
```



```
https://www.google.com/search?q=cats
```

Web Client: Browser

- Makes HTTP requests on behalf of the user
  - Reformat the URL entered as a valid HTTP request
  - Use DNS to convert server's host name to appropriate IP address
  - Establish a TCP connection using the IP address
  - Send HTTP request over the connection and wait for server's response
  - Display the document contained in the response
    - If the document is not a plain-text document but instead is written in HTML, this involves rendering the document (positioning text, graphics, creating table borders, using appropriate fonts, etc.)

Web Servers

- Main functionalities:
  - Server waits for connect requests
  - When a connection request is received, the server creates a new process to handle this connection
  - The new process establishes the TCP connection and waits for HTTP requests
  - The new process invokes software that maps the requested URL to a resource on the server
  - If the resource is a file, creates an HTTP response that contains the file in the body of the response message
  - If the resource is a program, runs the program, and returns the output

Static Web: HTML/XHTML, CSS

- **HTML** stands for **H**yper**T**ext **Ma**rkup **L**anguage
  - It is a text file containing small markup tags that tell the Web browser how to display the page

- **XHTML** stands for e**X**tensible **H**yperText **M**arkup **L**anguage
  - It is identical to HTML 4.01
  - It is a stricter and cleaner version of HTML
  - E.g., <!DOCTYPE>, <html>, <head>, and <body> are mandatory

- **CSS** stands for **C**ascading **S**tyle **S**heets
  - It defines how to display HTML elements

Static web limitations

- What is the drawback to simple document model?
  - Static
  - Assume that documents are created before they are requested

- What are examples of information that might be part of web documents that may not be known before they are requested?

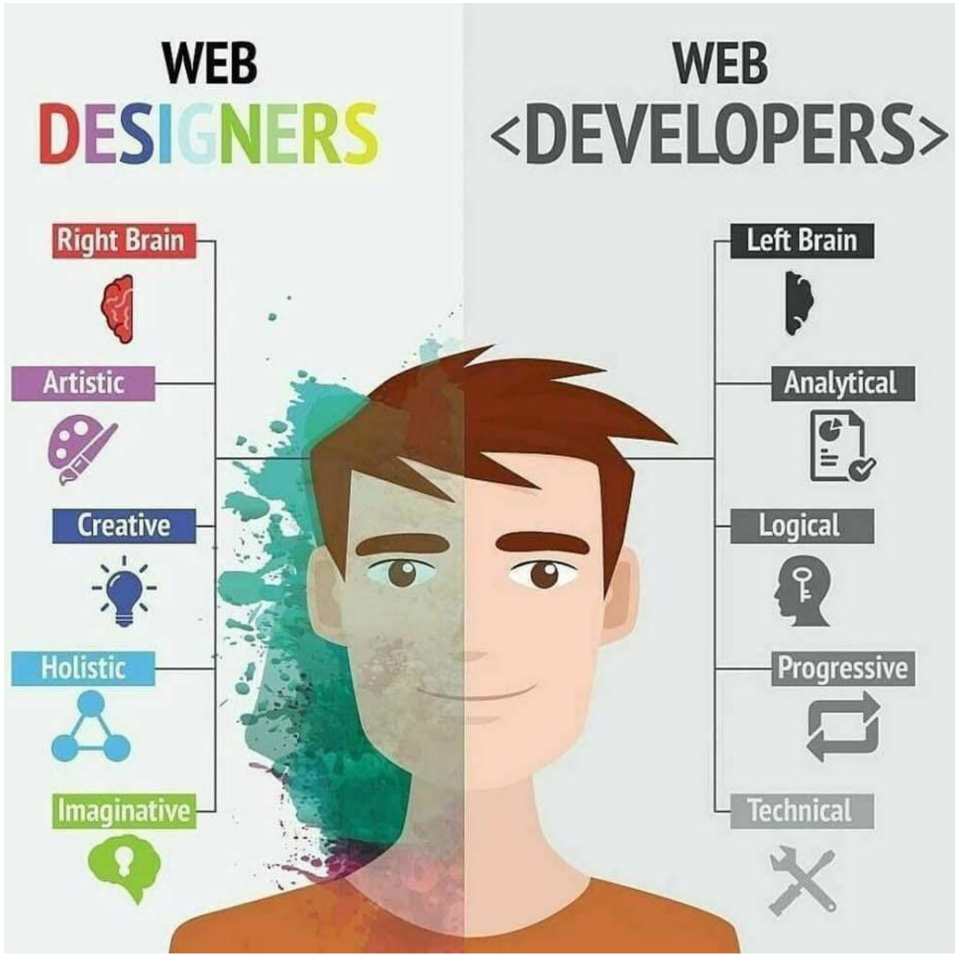
Client-Side Programming

- Scripting language: a lightweight programming language
- Browser scripting: **JavaScript**
  - Designed to add interactivity to HTML pages
  - Usually embedded into HTML pages
  - What can a JavaScript Do?
    - Put dynamic text into an HTML page
    - React to events
    - Read and write HTML elements
    - Validate data before it is submitted to a server
    - Create cookies
    - …

Server-Side Programming

- The requests cause the response to be generated
- Server scripting:
  - ASP.Net MVC: Microsoft product, uses .Net framework (*.asp)
  - CGI/Perl: Common Gate Way Interface (*.pl, *.cgi)
  - PHP: Open source, strong database support (*.php)
  - Java via JavaServer Pages (*.jsp)
  - …

Web Development vs Designing

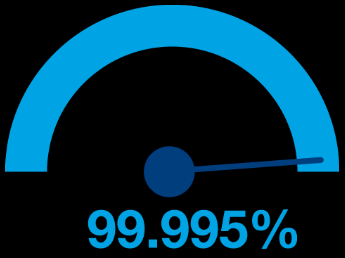


Scalability

We currently receive c10,000 trace requests each month.

We currently have capacity to take on an additional 10,000 trace requests each month.

With un-capped earning potential and heavy bonus penalties for inaccurate results, our teams only real restriction is the number of hours in the day.

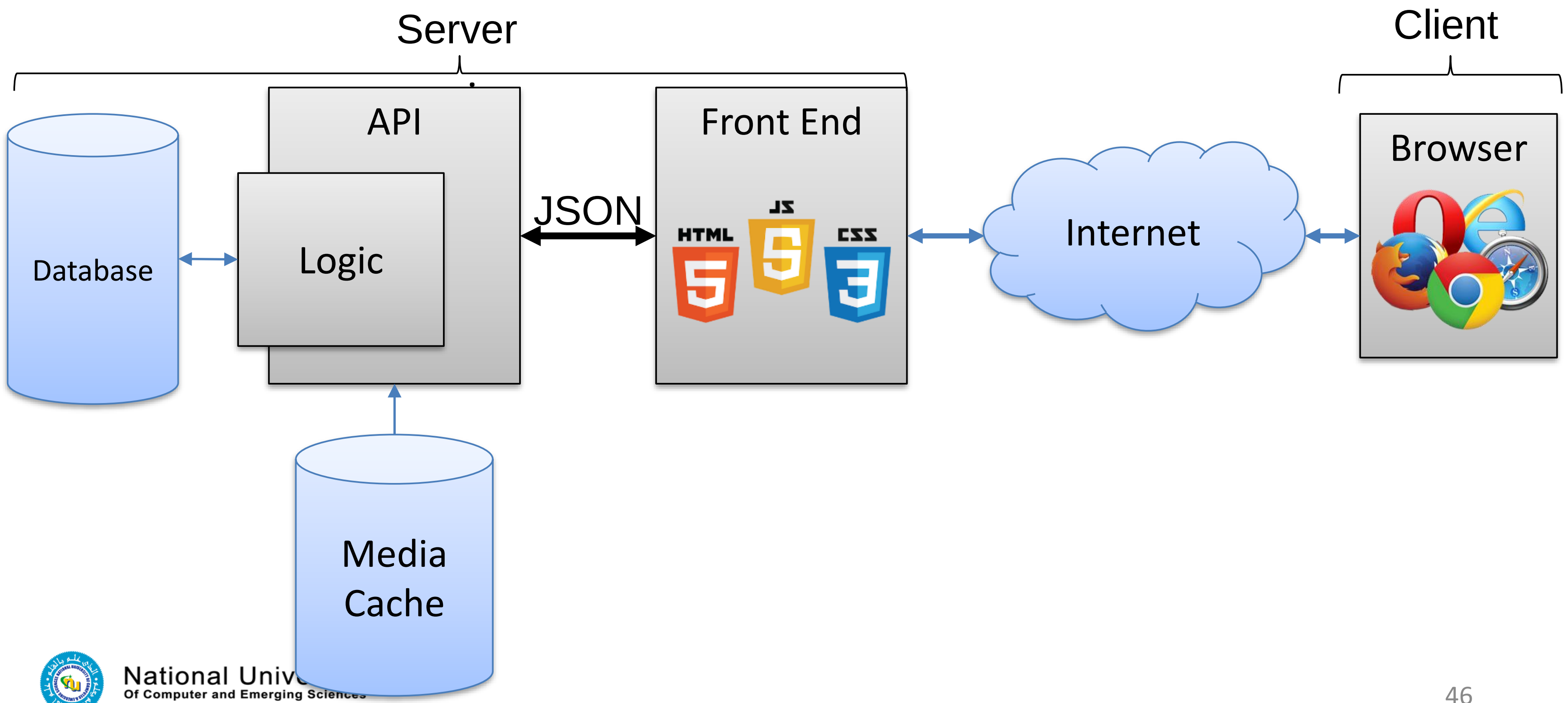


Principles of Web Design

- Availability
- Performance
- Reliability
- Scalability
- Manageability
- Cost

Web Applications

- UI (Front End (DOM, Framework))
- Request Layer (Web API)
- Back End (Database, Logic)



FRONTEND DEVELOPMENT

HTML CSS JS

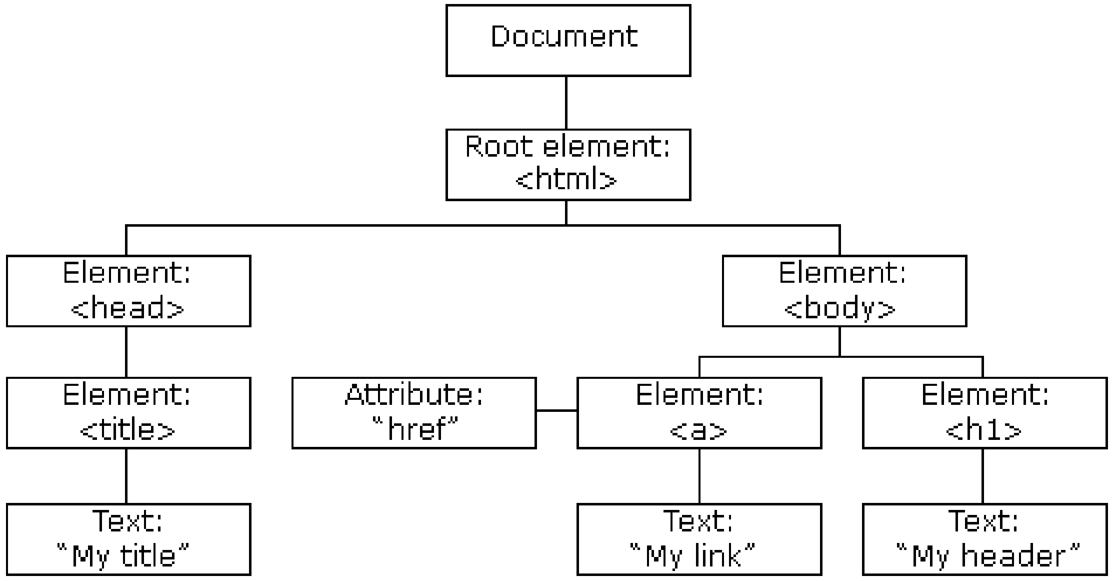
Front End Languages

- HTML/CSS
- JavaScript
- Java (applets)

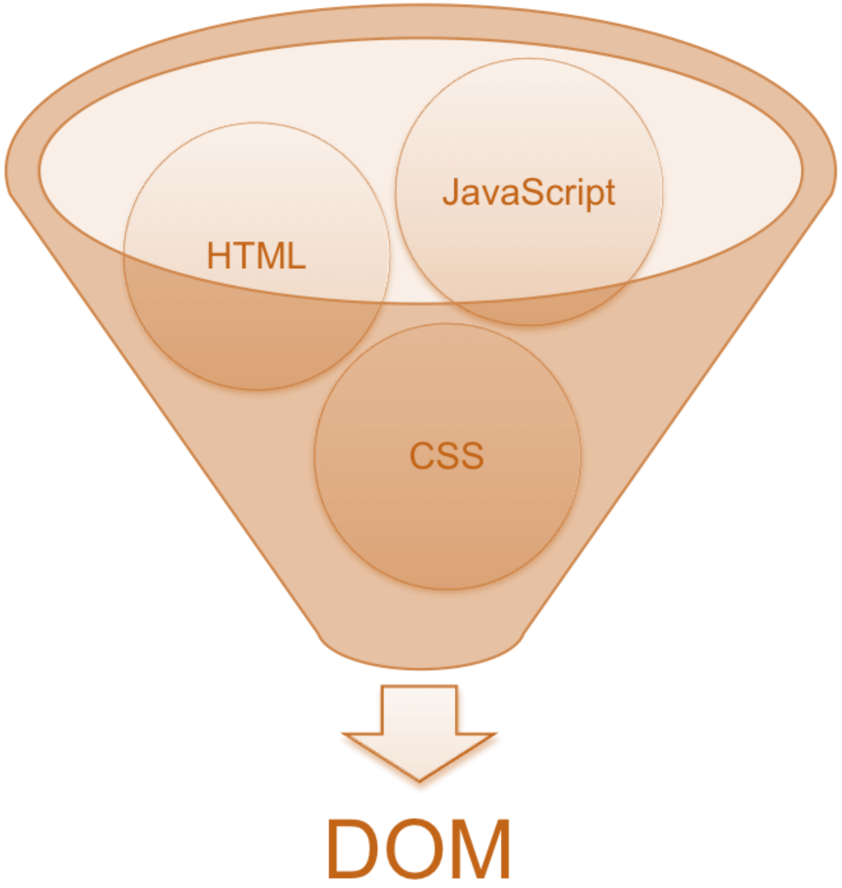
- What is the most popular?
- Answer: JavaScript/HTML/CSS is the only real option for front-end native languages and is basically the standard. But there are many variations on JavaScript that are used.

DOM (Document Object Model)

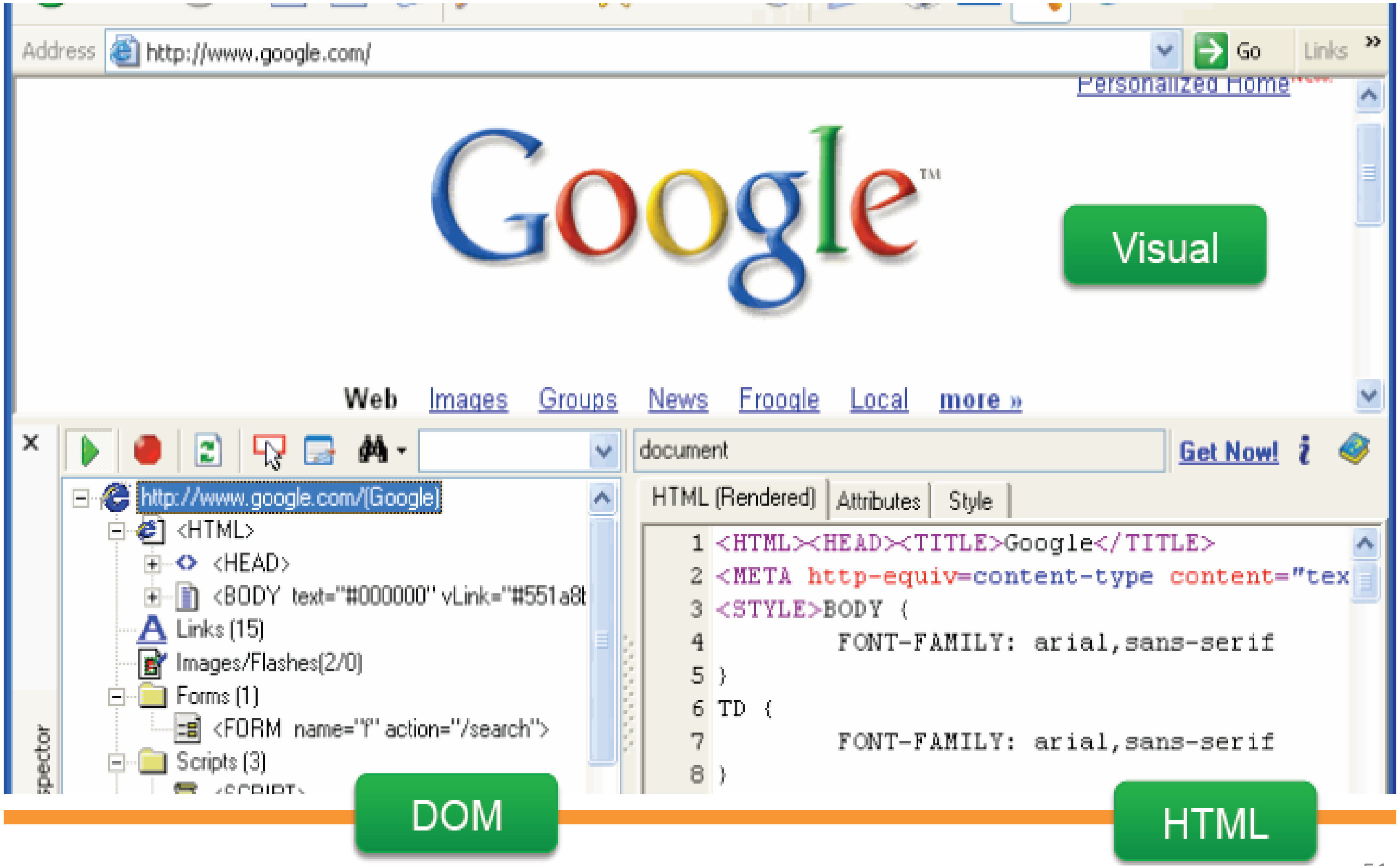
- Document Object Model makes every addressable item in a web application an Object that can be manipulated for color, transparency, position, sound and behaviors.
- Every HTML Tag is a DOM object



DOM (Document Object Model)



Three representations of same page



```
<HTML><HEAD><TITLE>Google</TITLE>
<META http-equiv=content-type content="tex
<STYLE>BODY {
        FONT-FAMILY: arial,sans-serif
}
TD {
        FONT-FAMILY: arial,sans-serif
}
```

Thank you!

National University
Of Computer and Emerging Sciences

Class Exercise- Developer Console

- **The 1st rule of web development**: Always keep the Developer Console open on your web
- Follow the link

  https://fullstackopen.com/en/part0/fundamentals_of_web_apps#traditional-web-applications
- Open this example app
- https://studies.cs.helsinki.fi/exampleapp/notes
- https://studies.cs.helsinki.fi/exampleapp/