HTML
5

About your instructor

- 14+ years of experience
- Worked in various domains using different technologies
- Developed
  - 180+ mobile applications on iOS and Android platforms
  - Various websites using PHP, MEAN and MERN stacks
  - Various machine learning solutions (using Python)
- Certification completed
  - Certified Kubernetes Application Developer (CKAD)
  - Certified Kubernetes Administrator (CKA)
  - Certified Jenkins Engineer (CJE)
  - Docker Certified Associate (DCA)

Instructions

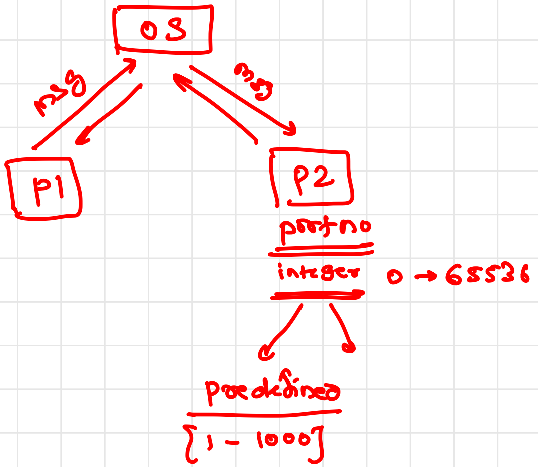
- Be at least 15 mins before in office for training sessions
- Avoid leaves during training days
- Ask and clarify your query even if it is not clear after first explanation
- Complete your assignments on time and send the code for review to mentor on time
- Send daily status to your mentor every day without fail
- In case of issue, before reaching out to mentor, first you do your research, discuss in your group, try to resolve issue by your own
- Focus on training, learning and practicing the topics taught everyday
- Avoid spending time mingling in cafeteria or office areas without reason
- Avoid sitting idle if you are done with assignment. Ask trainer for next/new assignment.
- Teams' application for communication is for official use. Avoid spending time in casual chit chat
- Respect others time. Check mentor's availability before asking them your queries
- Follow basic discipline and etiquettes for not to disturb office environment

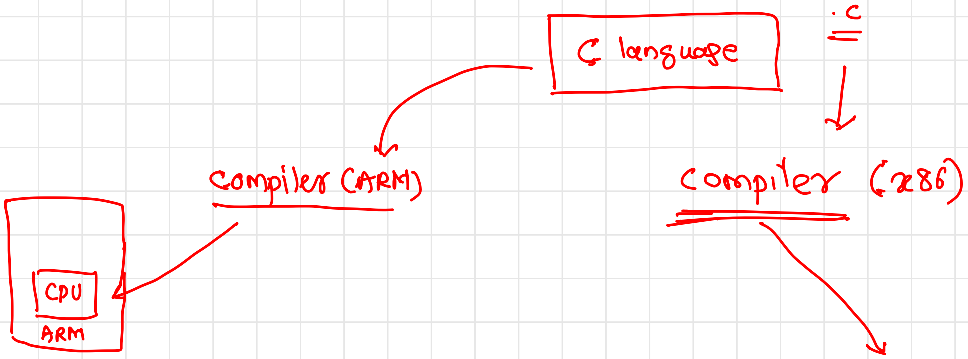
Fundamentals

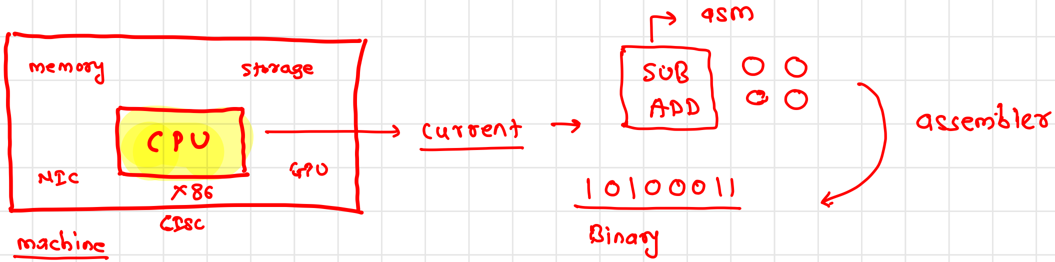
program → process

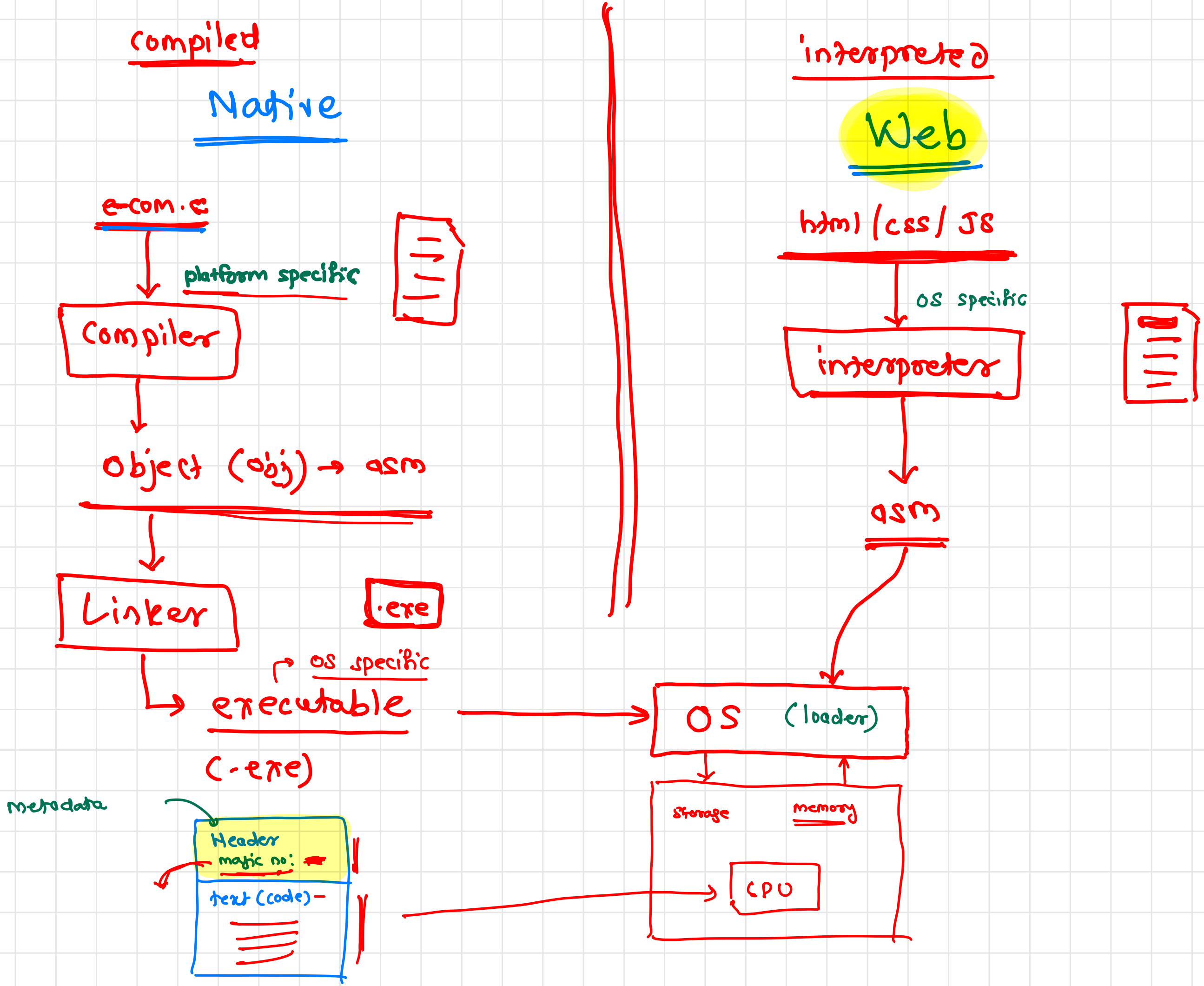
(file)

(running program)
program loaded in the memory









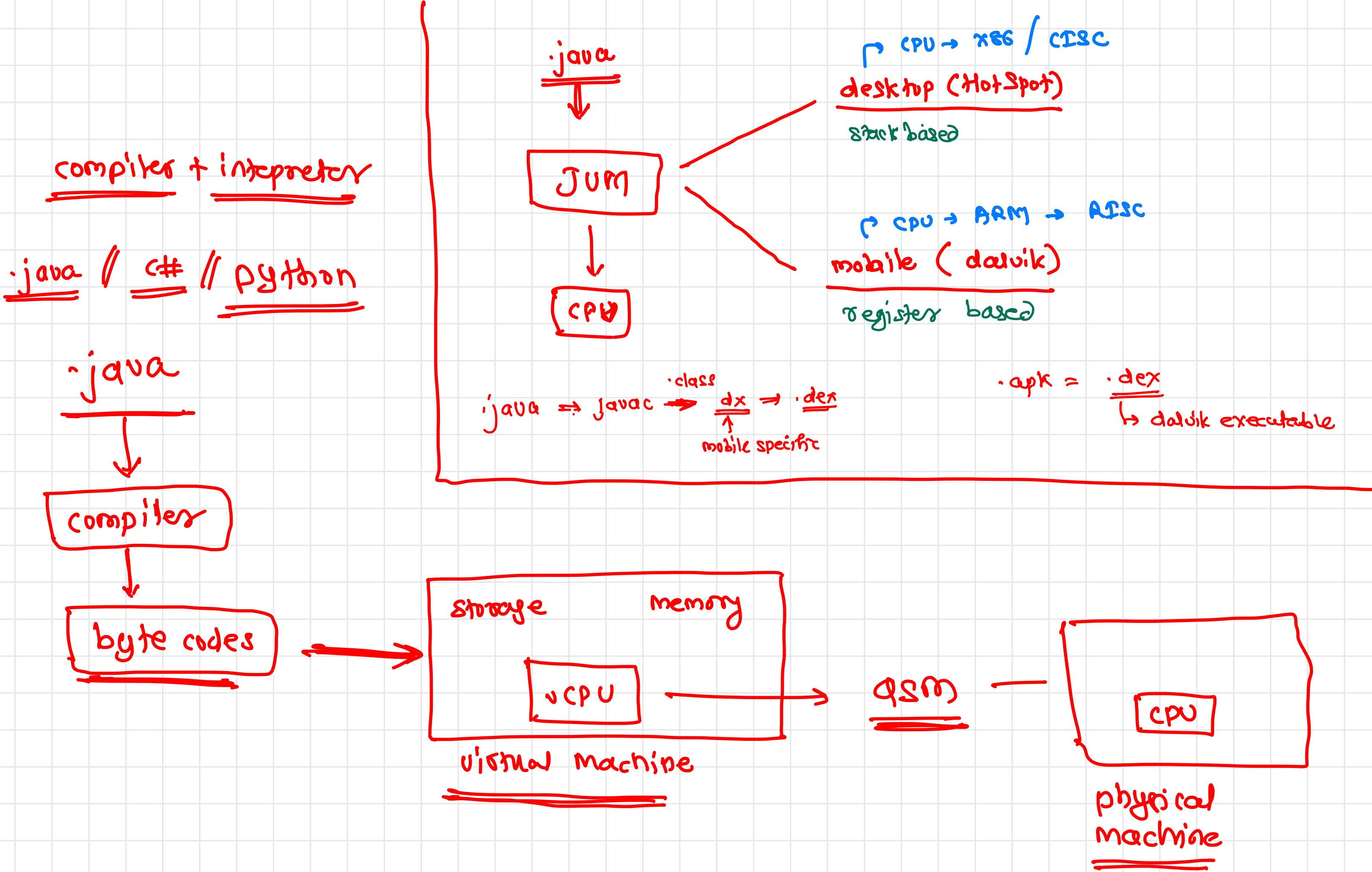
OS

- windows
- Linux
- mac OS

Executable	macOS	linux	windows
library → static → dynamic	.ar .dlib	.ar .so	.lib .dll
Self executable → CUI → GUI	.a .out	.a .out	.exe .bat
format	Mach-O XNU	ELF Embedded & Linkable format	PE/Coff portable Executable Common object file format

compiler + interpreter

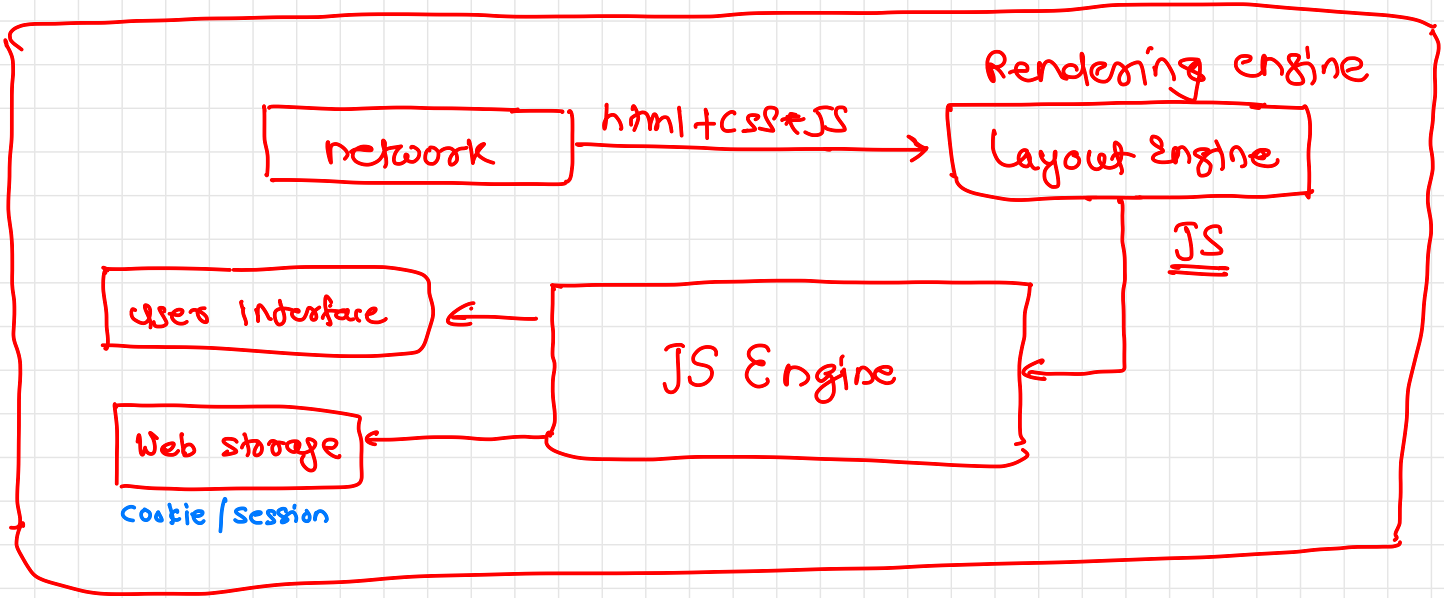
.java // c# // python



① Browser

- interpreter for html / css / JS
- native application
- developed in c, c++
- eg. chrome / firefox / safari / opera / IE / Edge

		JS engine	Rendering
safari	:	Nitro Engine	webkit
firefox	:	Spider Monkey	Gecko
IE/Edge	:	chakra	EdgeHTML
chrome	:	V8 ✓	Blink



(#) URL

→ Uniform Resource Locator

→ http://google.com, https://apple.co.in, . http:// 18.5.6.7 :8080/ index.html ? name = abc

scheme | IP address | port | file | query string

→ components

1] scheme : protocol ; http(80) port, https(443), file

2) domain name or IP address of server

3] port no of server process [web server]

4] file path or file name

5] Query string: used to send input [key-value pair]

6] hash component: used to link sections within page

http	https
- plain text	- encrypted
	→ SSL ↳ certificate

public

encryption

private

decryption

salt

amit → bnju → amit

plain text | cipher text | plaintext

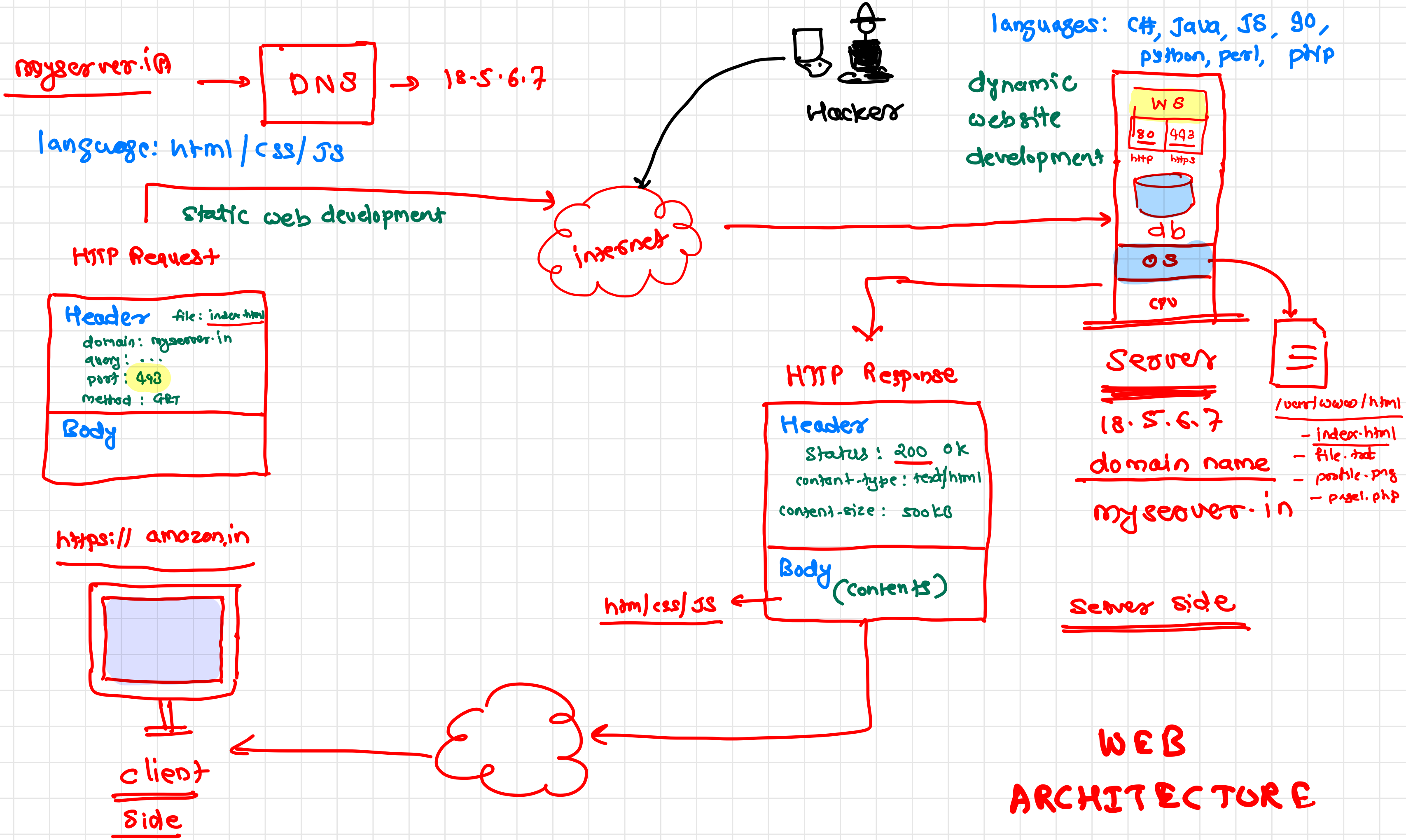
certificate

↓

random data

⊛ Response status

- 1xx : debugging the protocol
- 2xx : success (200, 201)
- 3xx : rediction (301, 302)
- 4xx : client error (400, 401, 403, 404)
- 5xx : server error (500, 501, 502..)



HTTP Request

Header file: index.html

domain: myserver.in

query: ...

port: 443

method: GET

Body

HTTP Response

Header

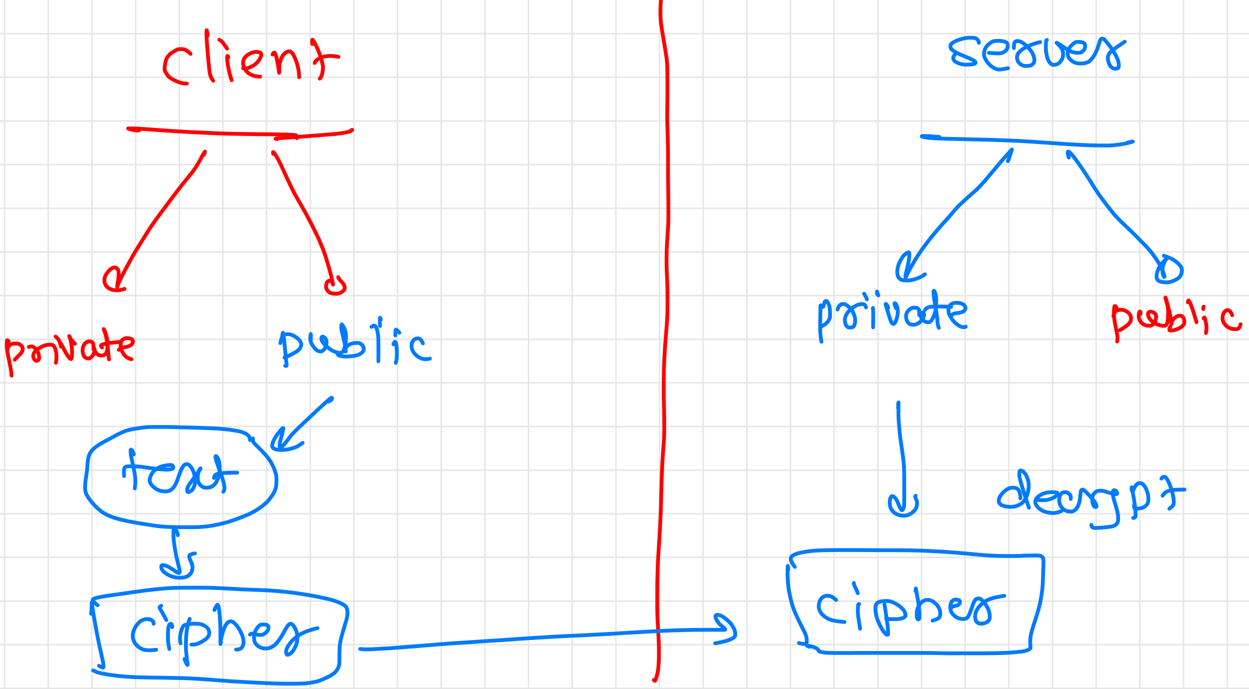
status: 200 ok

content-type: text/html

content-size: 500KB

Body (contents)

WEB ARCHITECTURE



public → trusted → publicaly signed

public → untrusted → self signed

2 Server

→ application which serves a request

- types

1. Web server
   - → serves web requests
   - → apache, nginx, IIS

2. Database server
   - → RDBMS → MS-SQL server, mySQL, oracle...
   - → NoSQL → MongoDB, CouchDB, Firebase .....

Programming Language

- Language is a mode of communication that is used to share ideas, opinions with each other
- A programming language is a computer language that is used by programmers (developers) to communicate with computers
- It is a set of instructions written in any specific language (C, C++, Java, Python) to perform a specific task
- A programming language is mainly used to develop desktop applications, websites, and mobile applications

Low Level Programming Language

- Low-level language is machine-dependent (0s and 1s) programming language
- The processor runs low- level programs directly without the need of a compiler or interpreter, so the programs written in low-level language can be run very fast
- **Machine Language**
  - Machine language is a type of low-level programming language
  - It is also called as machine code or object code
  - Machine language is easier to read because it is normally displayed in binary or hexadecimal form (base 16) form
  - It does not require a translator to convert the programs because computers directly understand the machine language programs
  - The advantage of machine language is that it helps the programmer to execute the programs faster than the high-level programming language
- **Assembly Language**
  - Assembly language (ASM) is also a type of low-level programming language that is designed for specific processors
  - It represents the set of instructions in a symbolic and human-understandable form
  - It uses an assembler to convert the assembly language to machine language
  - The advantage of assembly language is that it requires less memory and less execution time to execute a program

High-level programming language

- High-level programming language (HLL) is designed for developing user-friendly software programs and websites
- This programming language requires a compiler or interpreter to translate the program into machine language (execute the program)
- The main advantage of a high-level language is that it is easy to read, write, and maintain
- High-level programming language includes
    - Python, Java, JavaScript, PHP, C#, C++, Objective C, Cobol, Perl, Pascal, LISP, FORTRAN, Swift etc.

Middle-level programming language

- Middle-level programming language lies between the low-level programming language and high-level programming language
- It is also known as the intermediate programming language and pseudo-language
- A middle-level programming language's advantages are that it supports the features of high-level programming, it is a user-friendly language, and closely related to machine language and human language
- Example: C, C++

Application

- Written using one or more programming languages for a specific purpose
- Types
  - Native Application
    - Written using compiled languages like c, c++ etc.
    - Source code gets compiled into an executable
    - Faster than Web applications
  - Web Application
    - Written using interpreted languages like html, JS, python etc.
    - Source code gets executed directly
    - Slower than native application

Web Architecture

Web Architecture

- Web uses client server architecture
- Client
  - The machine which uses the browser to browse a website
  - Client machine sends a request to the server demanding the required resource
- Server
  - The machine which runs a server program
  - The server program serves a request
  - Types
    - Web server
    - File server
    - Database server

Web Architecture

- It uses request-response pattern
- Request
  - An object created by the client (browser)
  - Contains all the information required by the server to send the response
  - Header contains
    - HTTP Method, url, port, required resource, user-agent etc
  - Body contains the body parameters
- Response
  - An object created by the server which contains the requested resource
  - Header contains
    - Status code, server type, content-type, content size
  - Body contains the requested resource

Browser

- A special application used to browse the websites
- It contains parsers for HTML, XML, CSS, Text etc. which are used to parse HTML and CSS and can produce the UI
- It executes the JS code using JS engine
- Any browser contains
  - User interface component
  - Browser Engine (JS Engine)
  - Rendering engine
  - Networking component
  - JS interpreter
  - Data persistence component

SDK [software Development kit]

✓1] header files / packages / namespaces

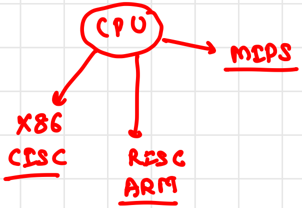
✓2] essential libraries

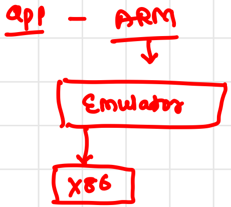
✓3] tool chain → collection of tools [compiler, interpreter, assembler, debugger, dis-assembler ...]

✓4] documentation → offline / online

✓5] Editor / IDE —
- → Integrated Development Environment - [Editor → .....]
- → Integrated Drive Electronics [type of HD]

✓6] Runtime / emulator / simulator





HTML

History

- HTML was created by Sir Tim Berners-Lee in late 1991
- HTML 1.0 was released in 1993 with the intention of sharing information that can be readable and accessible via web browsers
- Then comes the HTML 2.0, published in 1995, which contains all the features of HTML 1.0 along with that few additional features, which remained as the standard markup language for designing and creating websites until January 1997 and refined various core features of HTML
- Then comes the HTML 3.0. It included improved new features of HTML, giving more powerful characteristics for webmasters in designing web pages
- Then comes HTML 4.01, which is widely used and was a successful version of HTML before HTML 5.0, which is currently released and used worldwide
- HTML 5 can be said for an extended version of HTML 4.01, which was published in the year 2012