Introduction to Front-End Development

(Week-1)

1.Web server:

-It is stored in data center with hundred or thousand of other servers, all running different services connected to the internet.

- Different systems in data center ensures that servers have a) continuous power b) kept cool 24 hour a day c) continuous internet power

(Function)

-Website storage and administrator

-Managing Email

-Security

-Data storage

2. Website and Webpage

Webpage:

A webpage is a document that displays images, texts, videos and other content in the web browser.

Website:

It is a collection of related webpages that link together.

3. Web Browser:

A web browser is a software application that is used to browse the world wide web.

URL ->> Uniform Resource Locator

It consists protocol, Domain, Path

Web Server

Web Browser

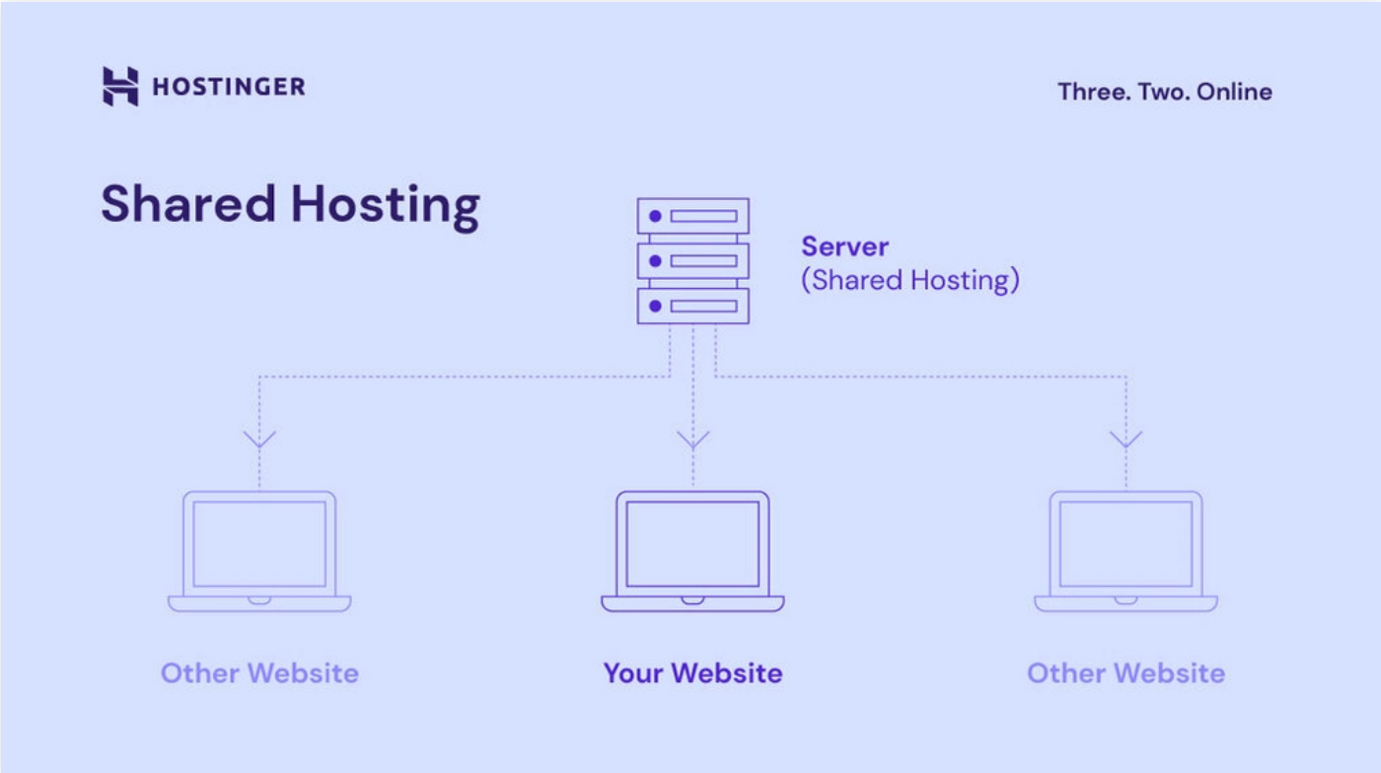
Request

Response

4. Web Hosting:

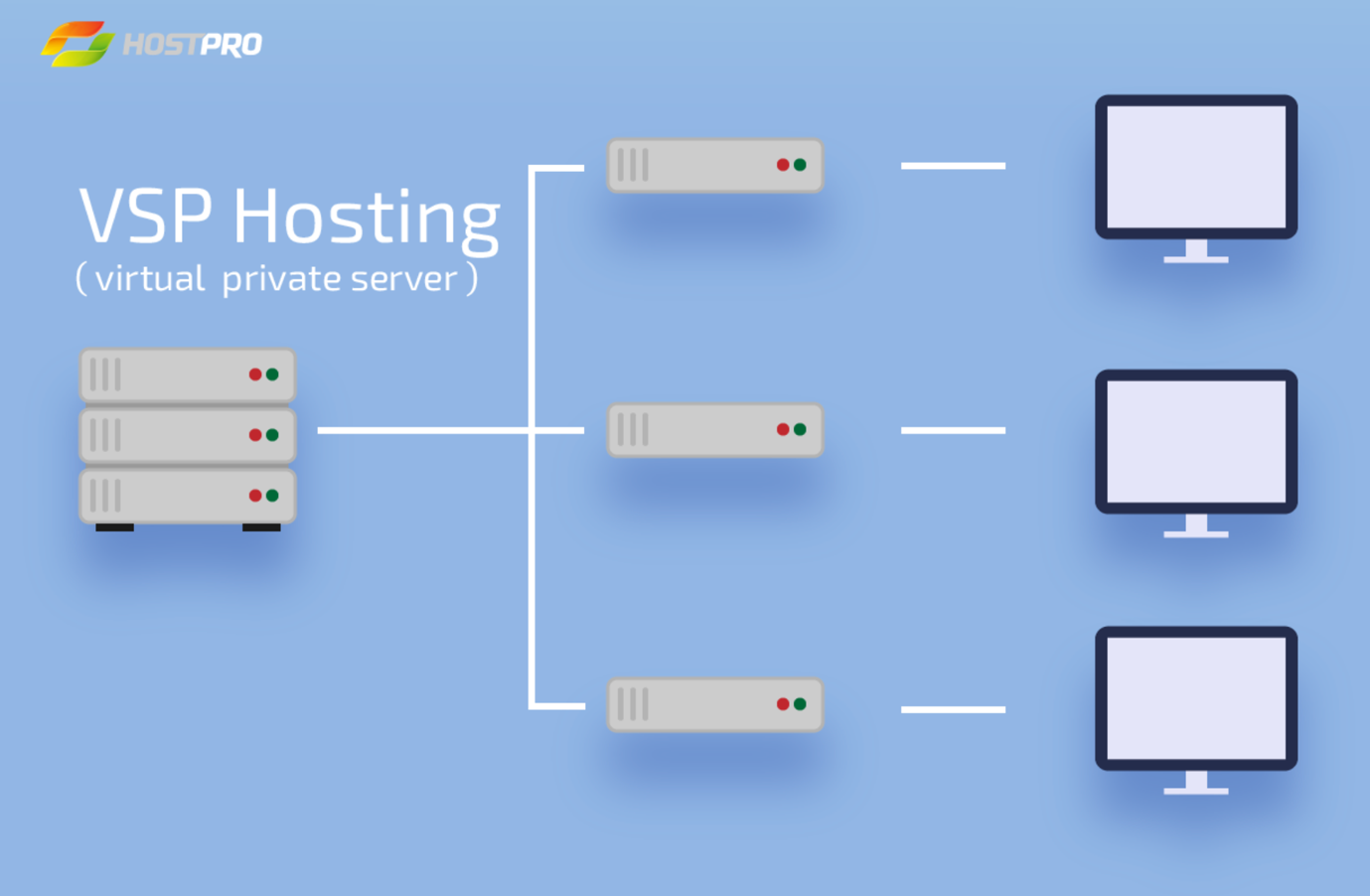
-It is a service where we place our website and files on the hosting companies web server.

1. Shared Hosting



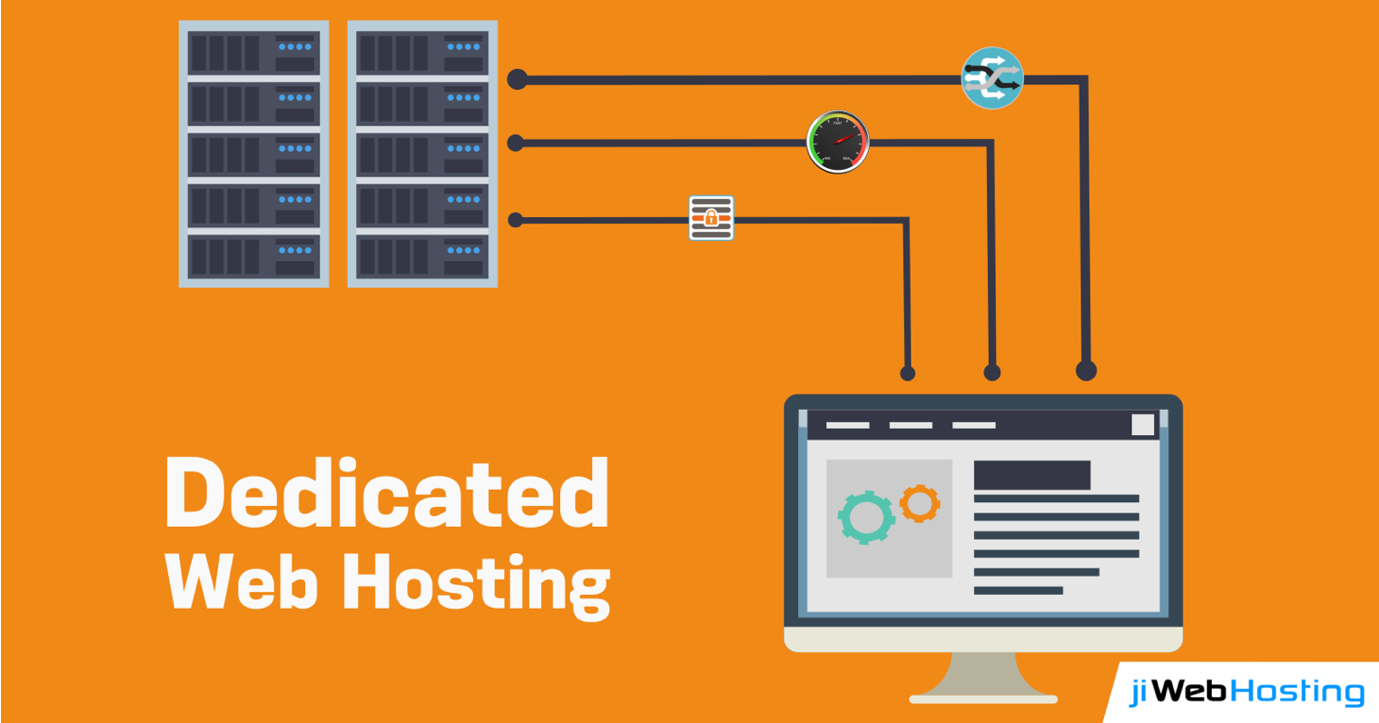
* Share service processing power, memory and bandwidth

b)Virtual Private Server (VPS) Hosting:



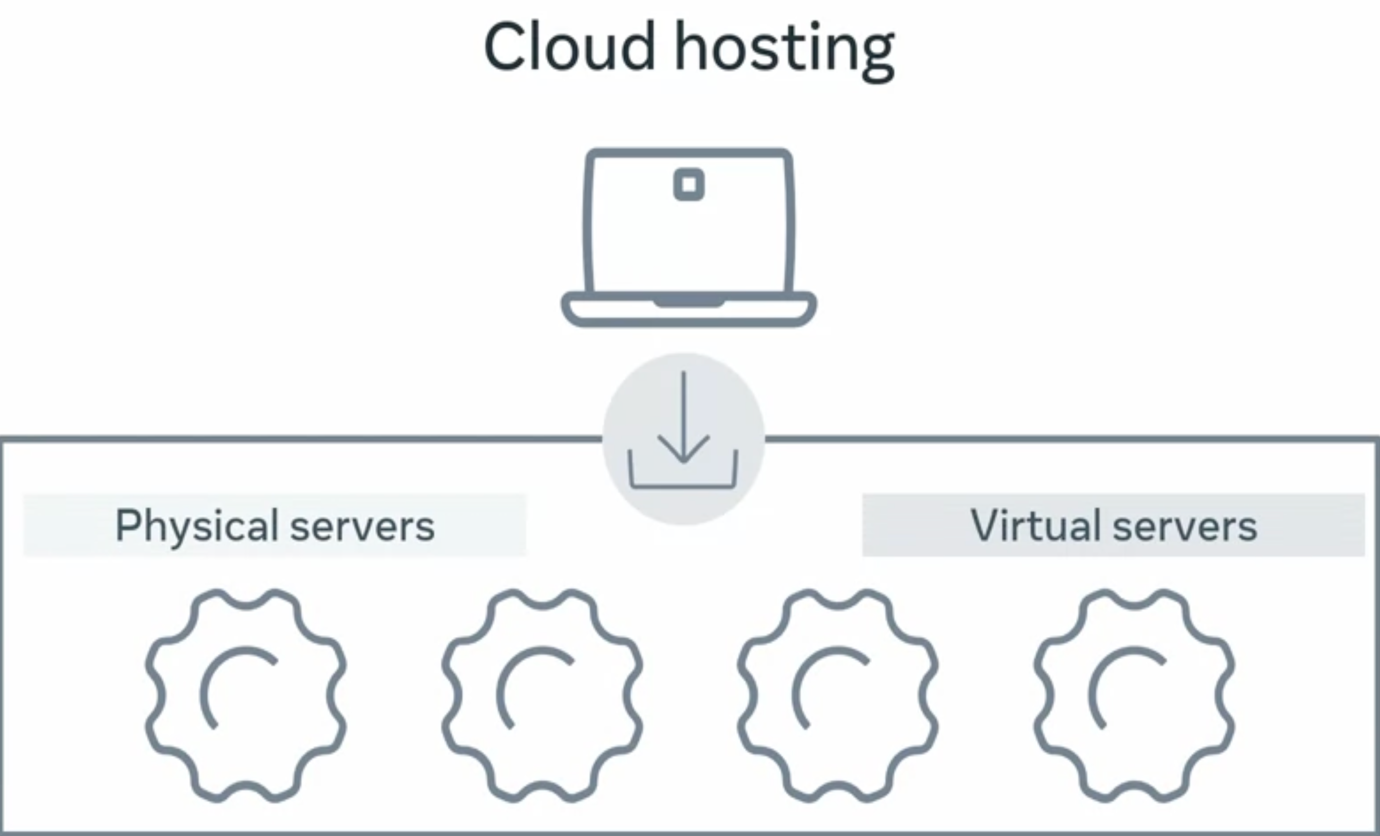
* It is running on a hardware server with other VPS instances but as the resources are fixed per VPS instances.

c)Dedicated Hosting:



-All hardware, CPU, memory and bandwidth resources are yours to use.

d)Cloud Hosting



-With Cloud hosting, your website is run in something called a Cloud environment, which spans across multiple physical and virtual servers. If a physical or virtual server fails,

your website will run on a different server and stay online. The main advantage of Cloud hosting is that you can use as many resources as you need without hardware limitations.

However, you pay based on resource use.

5. Introduction to Internet Protocol:

Source

Destination

Network

Transmission of data from source to destination in a network is only possible with the help of Internet Protocol.

IP version4: 192.168.0.1

IP version6:

A000:B000:C000:D000:E000:F000:G000:H000

when we send data across a network, we send the data as a series of message called IP packets. Also known as data gram.

At a high level of IP packets contains:



IP packets include source IP address and destination IP address. These addresses are in header along with some additional information to help deliver the packet.

Things that can wrong with IP Packet:

a)IP Packets arrives out of order

b)Become corrupted or damaged

c)Dropped or lost during transit

To solve this problem payload(IP data) part of IP packet contains other protocols too. The two most common protocols are:

i)TCP:

TCP stands for transmission control protocol. TCP solve all three of the problem mentioned above but at the cost of a small delay when sending the data. Reason for delays are:

-Connection Setup

-Congestion Control

-Retransmission timeouts

-Buffering

This protocol is used for sending the data that must arrive correctly and in order such as image and text files.

ii) UDP:

UDP stands for User Datagram Protocol. UDP solves the corrupt packet issue but packet can still arrive out of order or not arrive at all. This protocol is used for sending data that tolerate some data loss such as voice calls or live video streaming.

Both of these protocols contain payloads that contain further protocol inside of them.

6. Introduction to HTTP:

HTTP stands for Hypertext Transfer Protocol. It is used for transferring web resources such as HTML documents, images, styles and other files. It is a request-response protocol.

HTTP Request

GET /HTTP/1.1

Host: developer.mozilla.org

Accept-Language: en

HTTP Methods

|  |  |
| --- | --- |
| GET | To retrieve |
| POST | To send |
| PUT | To update |
| DELETE | To remove |

HTTP Response

HTTP/1.1 200 OK

Content-Type: text/HTML

Content-length: 50

<html>

<body>

<p>Hello World! </p>

</body>

</html>

Groups of status code

|  |  |
| --- | --- |
| Informational | 100-199 |
| Successful | 200-299 |
| Redirection | 300-399 |
| Client Error | 400-499 |
| Server Error | 500-599 |

7. Other Internet Protocol

|  |  |
| --- | --- |
| DHCP (Dynamic Host Configuration Protocol) | It is a network protocol used to dynamically assign IP addresses and other network configuration parameters to device on a network |
| DNS (Domain Name System) | It is used to translate domain names to IP addresses |
| IMAP (Internet Message Access Protocol) | It is used to retrieve email messages from a mail server |
| SMTP (Simple Message Transfer Protocol) | It is used for sending email messages between mail server. |
| POP (Post Office Protocol) | It is used for retrieving email message from a remote server to a local client. |
| FTP (File Transfer Protocol) | It is used for transferring files between a client and a server on a computer network. |
| SSH (Secure Shell Protocol) | It is a cryptographic network used for secure data communication, remote command-line login and remote command execution between two computer systems. |
| SFTP (Secure File Transfer Protocol) | -It is used to securely transfer files between computer. |

8. Framework and Libraries

|  |  |
| --- | --- |
| Framework | Libraries |
| Provides a complete solution for building application | Provides a set of reusable functions or classes. |
| Enforces a specific architecture or design pattern | Doesn’t enforce any specific architecture or design pattern |
| Typically provide tools for managing application flow, routing and data persistence | Typically focuses on specific functionality, such as UI components or data manipulation |
| Example include Angular, React, and Ruby on Rails | Example include JQUERY, NumPy and Pandas |

9. API and Services

API is the acronym for Application Program Interface. A service, application or interface offering advanced functionality with simple syntax. An API is a set of functions and procedures for creating applications that access the features or data of an operating system, application or others service.

API

API

API are the bridge between different components or system. It is also called gateway or middleware.

Common types of API are:

a) Browser API:

They extend the functionality of the browser by adding new services and are designed to simplify complex functions and provide easy syntax for building advanced features. A good example of the DOM API. The DOM API turns the html document into a tree of nodes that are represented as javascript objects. Another example is the geolocation API that returns coordination of where the browser is located. There are also other API’s available for fetching data known as fetch API. Drawing graphics or canvas API, keeping history or history API, web storage API.

b) Restful API:

This kind of API provides data for popular web and mobile apps. They are also called web servers. REST or representational state transfer, is a set of principles that help build highly efficient API’s. One of the main responsibilities of these kinds of API’s is sending and receiving data to and from a centralized database.

Database

PC

REST API

c)Sensor-Base API:

This is what the internet of things also known as IOT is based on. These are actual physical senses that are interconnected with each other. The sensors can communicate through API and track and respond to physical data. Some examples are Philips hue, smart light and node bots.

How API Works?

<https://example.com/?t=info>

These API’s uses endpoints to specify how different resources can be accessed. The endpoint is built into the URL when accessing the API. Once the endpoint is hit, the API performs whatever service side processing is needed to build the response. Two common forms of response are full web pages, data form based on Javascript called JSON.

(Week-2)

1.Hyper Text Markup Language:

HyperText Markup Language is a standard markup language used to create web pages and web applications. HTML is the backbone of the internet, and every web page that we see on the world wide web is written in HTML.

HTML is used to structure the content of a webpage by using a set of tags or elements that define the different parts of the page such as headings, paragraphs, images, links and more. These tags are written using a specific syntax that is understood by web browsers, which interpret the HTML code and display the web page to the users.

HTML also allows web developers to create forms, tables, lists and other interactive elements that enables users to interact with the web pages. It is often used in conjunction with other web technologies such as CSS and JavaScript to create modern, dynamic, and responsive web pages.

2.HTML Document:

An HTML Document is a text file that contains HTML code that is used to create a web page. The document is saved with the file extension .html and can be opened by a web browser, which interprets the HTML code and displays the web page to the user.

An HTML document usually contains two main sections: the head section and the body section.

The head section contains information about the web page that is not displayed on the page itself, such as the title of the page, metadata, and links to external files such as stylesheets and scripts.

The body section contains the actual content of the web page, such as text, images, videos, and other elements. HTML tags are used to define the structure and formatting of the content, such as headings, paragraphs, lists, tables, and forms.

<!DOCTYPE html>

<html>

<head>

<title>My Web Page</title>

</head>

<body>

<h1>Welcome to my web page</h1>

<p>This is a paragraph of text.</p>

<img src="myimage.jpg" alt="My Image">

<a href="http://www.example.com">Visit Example.com</a>

</body>

</html>

In this example, the <!DOCTYPE html> declaration specifies that the document is and HTML5 document. The <html> tag defines the beginning and end of the HTML document, while the <head> and <body> tags define the head and body sections respectively. The <title> tag sets the title of the web page, while the <h1>, <p>, <img> and <a> tags are used to define heading, paragraphs, images and links respectively.

3. Forms:

HTML provides several elements that can be used to create forms on a web page. Forms are used to collect input from users, such as their name, email-address, and other information, and then submit this information to a server for processing. Here are some of the most commonly used form elements in HTML:

a) Text

b) Password

c) Checkbox

d) Radio

e) Textarea

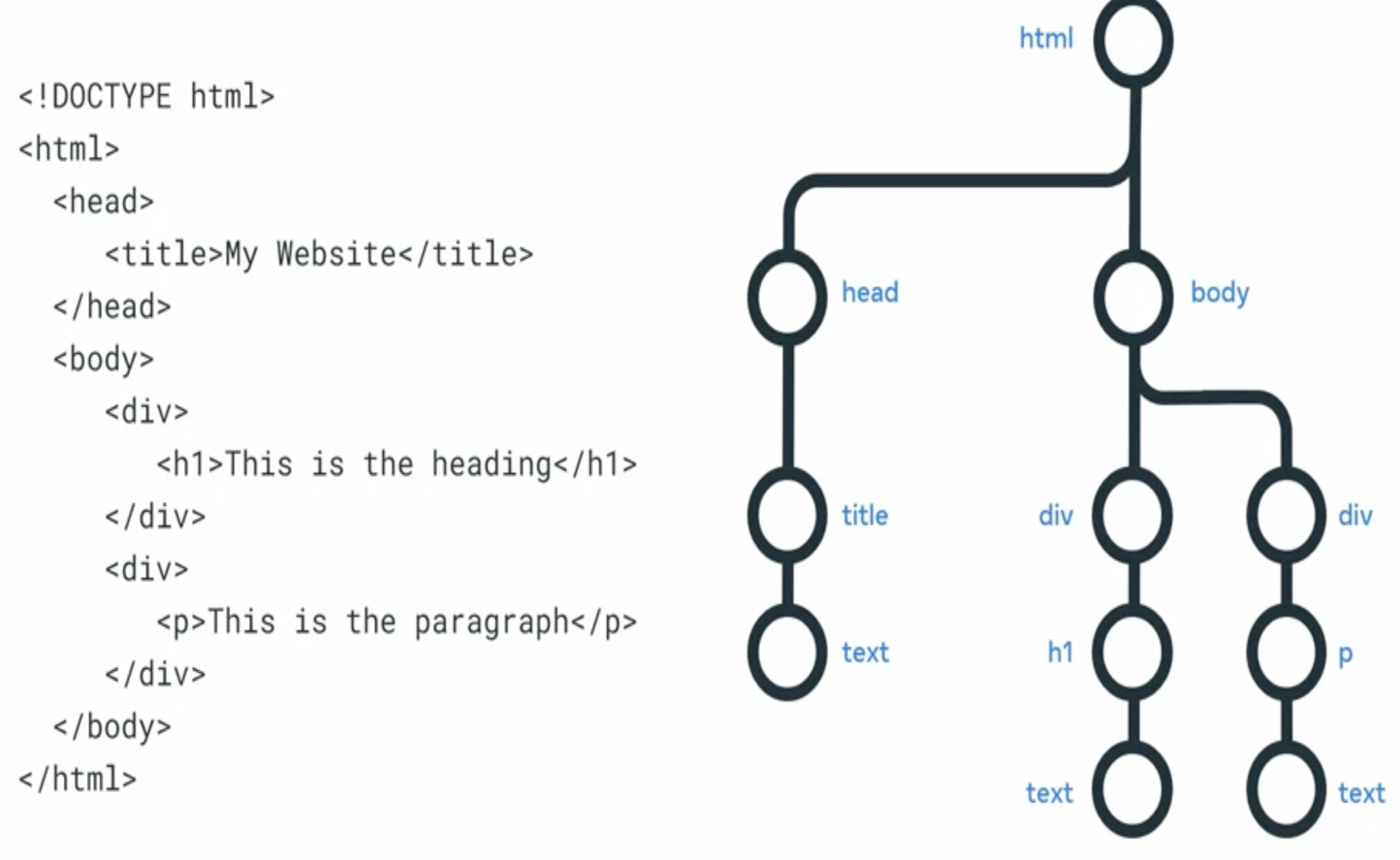
f) Select

4.Introduction to DOM:

DOM stands for Document Object Model. It is a programming interface for HTML and XML documents that represents the structure of a document as a tree of objects. The DOM is an essential part of web programming, as it allows developers to access, manipulate, and update the content and structure of a web page dynamically using scripting language such as JavaScript.

When a web page is loaded into a browser, the browser parses the HTML code and creates a representation of the page in memory, which is known as the Document Object Model. This model is made up of hierarchical tree-like structure of objects, where each object corresponds to an element, attribute, or piece of text in the HTML code.

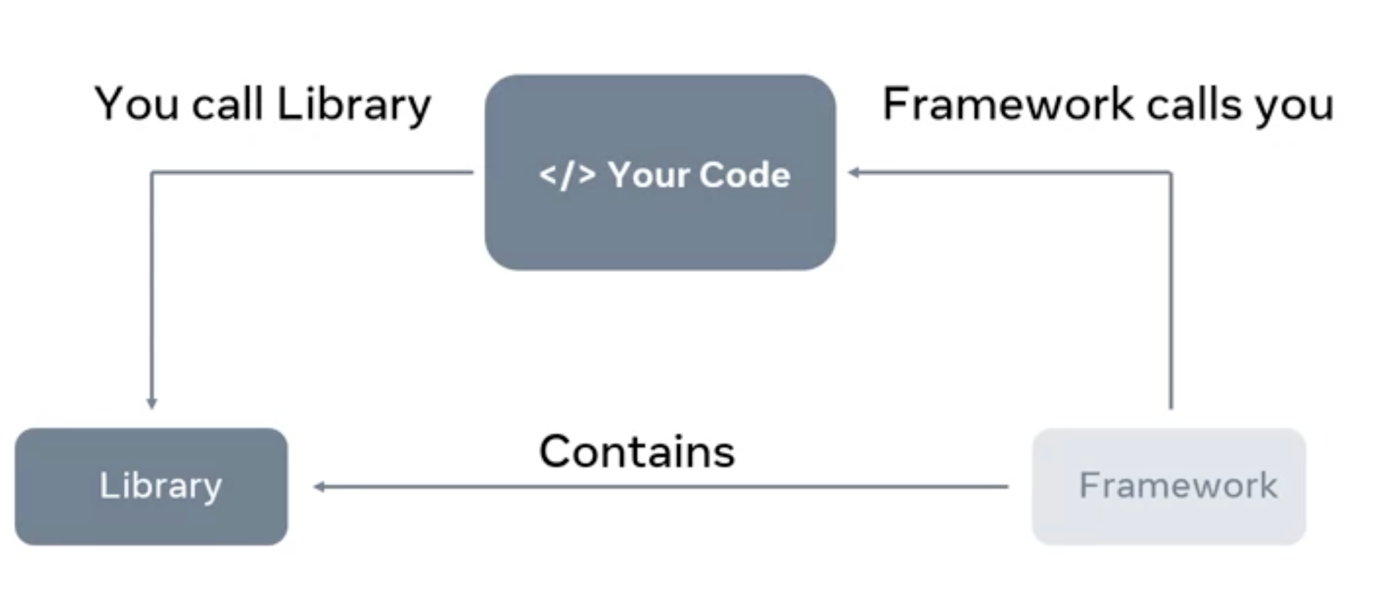
Using the DOM, developers can access and manipulate these objects using scripting language such as JavaScript, allowing them to modify the content and structure of a web page dynamically in response to user interactions, data input or other events.



5.Web Accessibility

Web Accessibility refers to the practice of making websites and web content accessible to people with disabilities. This include making websites usable by people with visual, auditory, physical and cognitive disabilities.

6. Working with libraries:

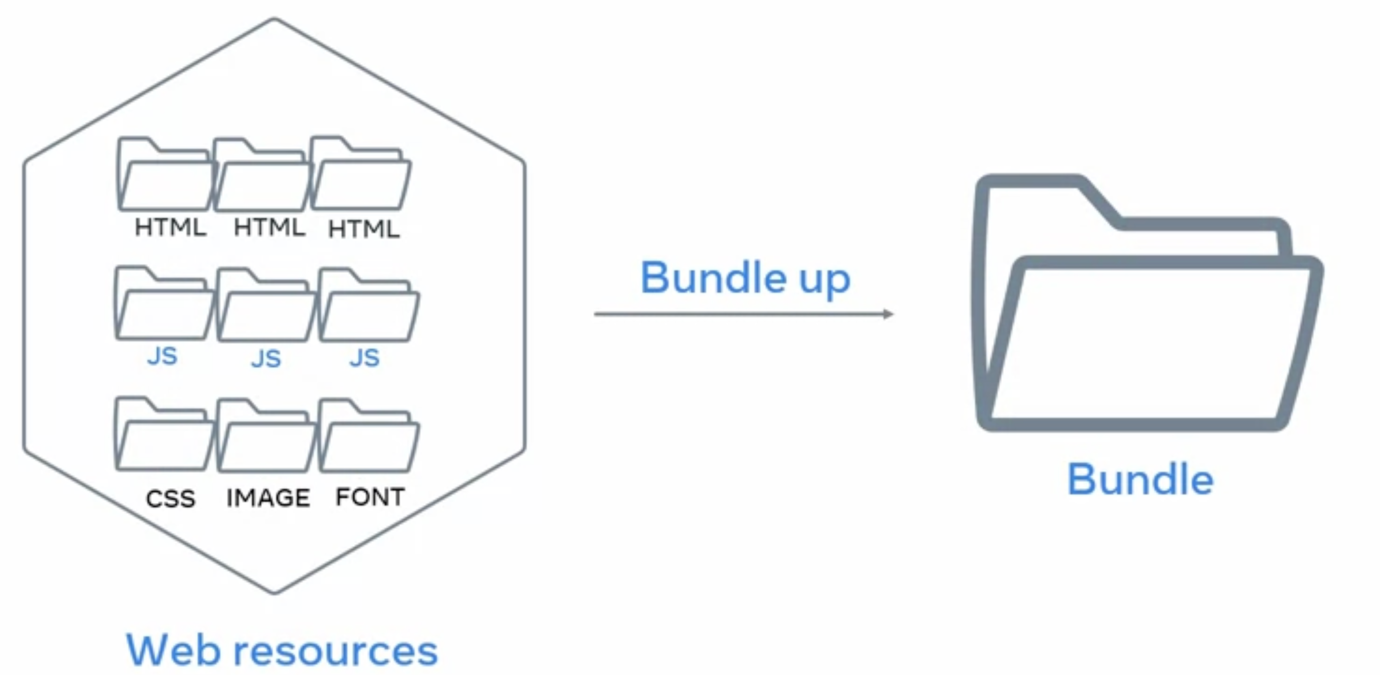


Package Manager:

A package manager is a tool that automatically downloads and installs dependencies. A package manager also provides the capability to publish your own packages.

NPM is the most common package manager.

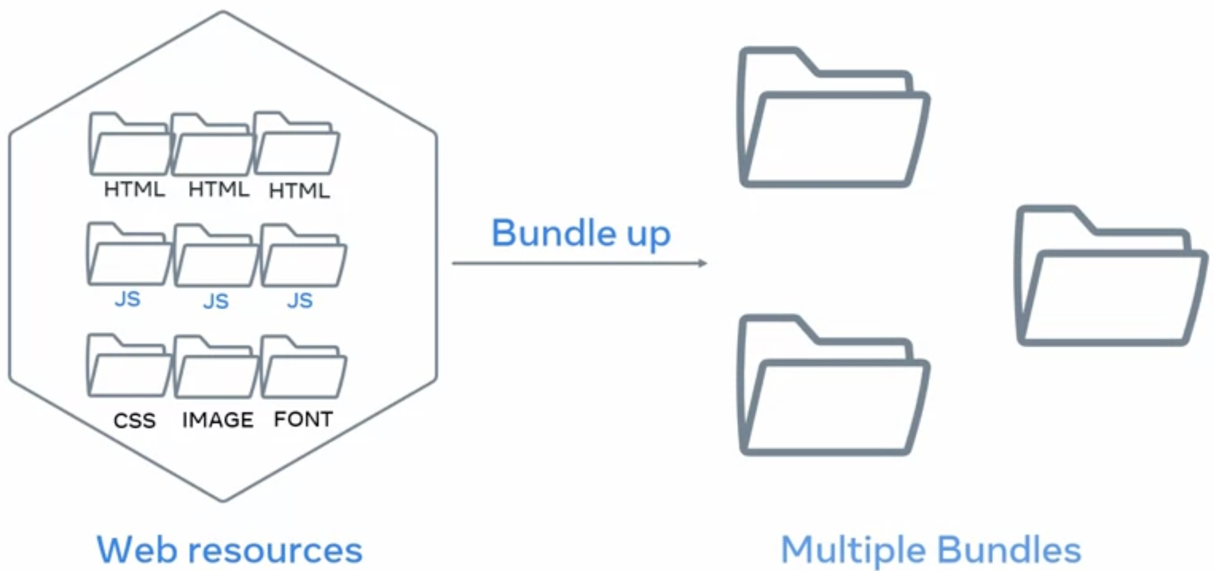
Bundler:



The purpose of Bundler is to automatically combine web resources into a single file.

If you’re bundle is significantly large many bundles can split your dependencies into multiple bundles.

Gulp, Webpack are the example of bundler.



7. Introduction to responsive design:

Responsive design means that a web page can automatically stretch or shrink depending on the screen is displayed on.

Responsive design is a set of 3 practices that allow website to automatically change its visuals. It is a combination of three techniques:

1. Flexible Grids:

Flexible grids are made up of columns, gutters and margins.

-Not pixel based

-Flexibility

-Percentages

b) Fluid Images:



By setting the CSS Max-width property of image to 100%. The images will scale down smaller if they’re containing column becomes narrows than the images size but never grow larger.

c)Media query:

They allow developers to the display size, orientation and aspect ratio to conditionally apply CSS rule.

Breakpoint:

In responsive design, the pixel value specifies is often referred to as the breakpoint.

A breakpoint is the point at which a website’s content and layout will adapt to provide the best possible user experience.

A breakpoint can function in different ways across three different grids:

1. Fixed grid:

Fixed grid has fixed content with that doesn’t change in a specific breakpoint range while the flexible margins occupy the remaining space on screen.

b)Fluid grid:

The fluid grid has a flexible content with that goes edge to edge as per the screen size. In a fluid grid, columns either grow or shrink or adapt to the available space.

c)Hybrid grid:

Hybrid grids that have both fluid width and fixed with components.

8. Bootstrap:

A collection of pre-written code chunks in css and javascript.

How to use modifiers:

Modifiers are CSS classes that can be applied to existing Bootstrap components to modify their appearance or behavior. These modifiers allow you to customize the default styles and adapt components to better suit your specific requirements.

<div class= ”alert alert-primary” role=”alert”>

Contextual class

9. Bootstrap Grid system:

For the grid, bootstrap uses a 12 column grid system that can be fluid or fixed.

To specify the column width add the suffix-n to the col css class

<div class=”col-4”>

<div class=”col-12 col-lg-6”>

10. Static and dynamic content:

Static content is files that the server transfers just as they are stored on the web server, such as video or images.

Dynamic content, on the other hand is generated when the HTTP request is made. For example, the content may be generated based on input from a user, or when you visit a news website.

What actually happens is that the web server communicates with another kind of server called an application server or backend. The application server generated the dynamic content that the web server send back to the user’s browser.

11. How website update static and dynamic content.

Say for instance, you want to watch a video on a website, you click on the play button and a request is send to the webserver. The webserver responds by sending the file to your browser.

Because dynamic content is generated while you use a website. It typically takes longer to generate than it takes to send back static content. For example, when you log into the course, the web server communicates with an application server to check that you are in face enrolled. The application server confirms your enrollment and specifies what content should shown for your profile specifically.

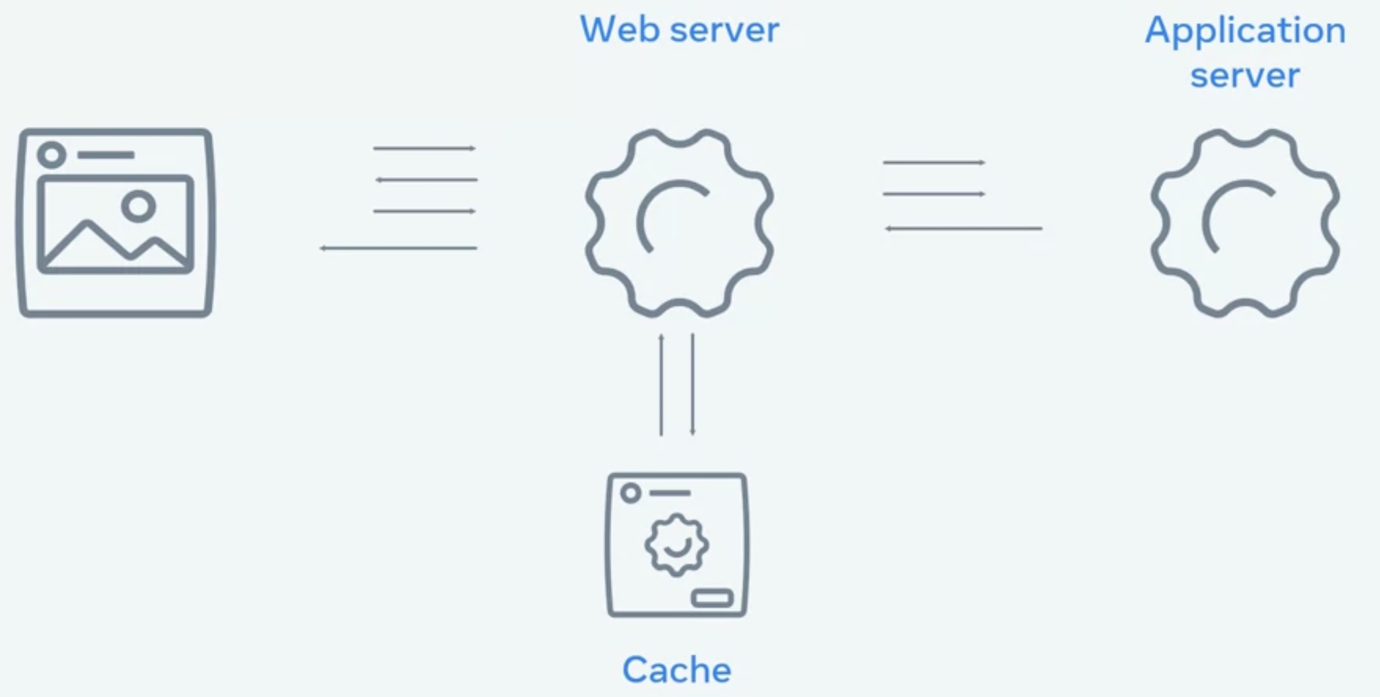
Application server perform more complex processing than web servers. For instance:

-They have to run the application logic

-Communicated with database.

-And check permission.

Application server typically have a limited capacity on how many requests they can process per second. But fortunately, this is where the web server can help out. Web server uses a process called caching instead of generating content dynamically for every request. Caching means the web server keeps copy of dynamic content.



12. Single page Application

A spa allows the user to interact with the website without the application needing to download entire web page. Instead, it rewrites the current web page as the user interact with it.

SPA Approach:

-Bundling:

When the browser requests the application, the server returns and loads all necessary HTML, CSS and JavaScript immediately.

-Lazy Loading:

With lazy loading, the browser requests the application and server returns only the minimum HTML, CSS and JavaScript needed to load application. Additional resources are download as required.

13. React :

React component is basically a small piece of user interface such as a music player or photo gallery.

How reacts works:

React builds a representation of the browser Document Object Model or DOM in memory called the Virtual DOM. As components are updated, React checks to see if the component’s HTML code in the virtual DOM matches the browser DOM. If a change is required, the browser DOM is updated. If nothing has changed, then no update is performed.

This process is called reconcillation.

React fiber Architecture:

React can optimize when and where updates occur to the browser DOM to significantly improve application performance and responsiveness to user input. Think of it as a priority system. The highest priority changes, the elements visible to the user, are updated first. While lower priority changes, the elements not currently displayed, are updated later.