HTML NOTES E-BOOK

1. What is UI?
   a. User Interface
2. What is UX?
   a. User Experience

3. What is UI?
   a. User Interface
4. What is UX?
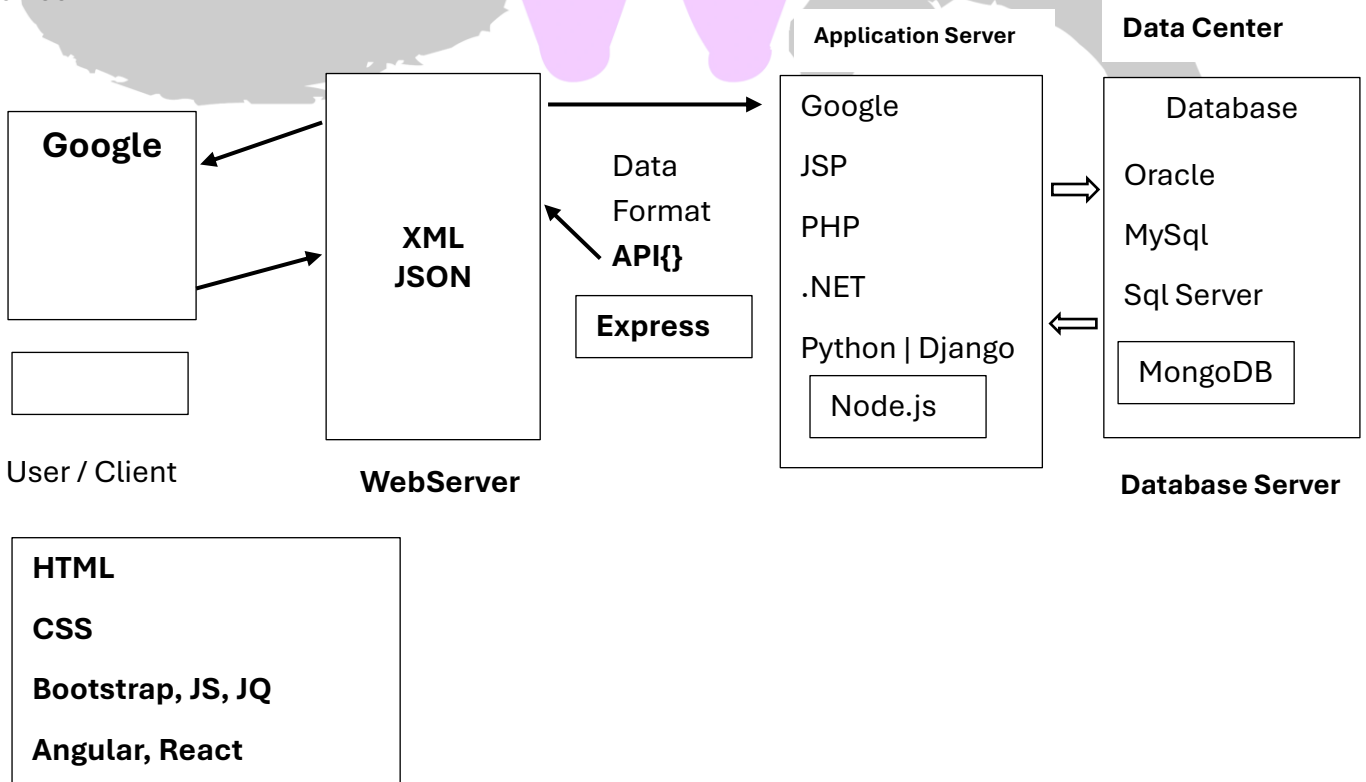   a. User Experience is a part of UI.

5. Note: Ul is presentation and UX is behaviour
   a. UI => HTML, CSS, JavaScript
   b. UX => jquery, responsive design react, angular

What is Full-Stack?

Chrome/Edge
Browser

Google

User / Client

WebServer

XML
JSON

Data
Format
API{}

Express

Application Server

Google

JSP

PHP

.NET

Python | Django

Node.js

Data Center

Database

Oracle

MySql

Sql Server

MongoDB

Database Server

HTML
CSS
Bootstrap, JS, JQ
Angular, React

Web Applications

1. What is a network?
    a. A group of computers connecting with each other for sharing of information and resources.
2. What are the ious types of networks?
    a. LAN, MAN, WAN
    b. What is internet?
    c. It is a WAN - all over the world.

How internet works When we type www.google.com at your address bar:
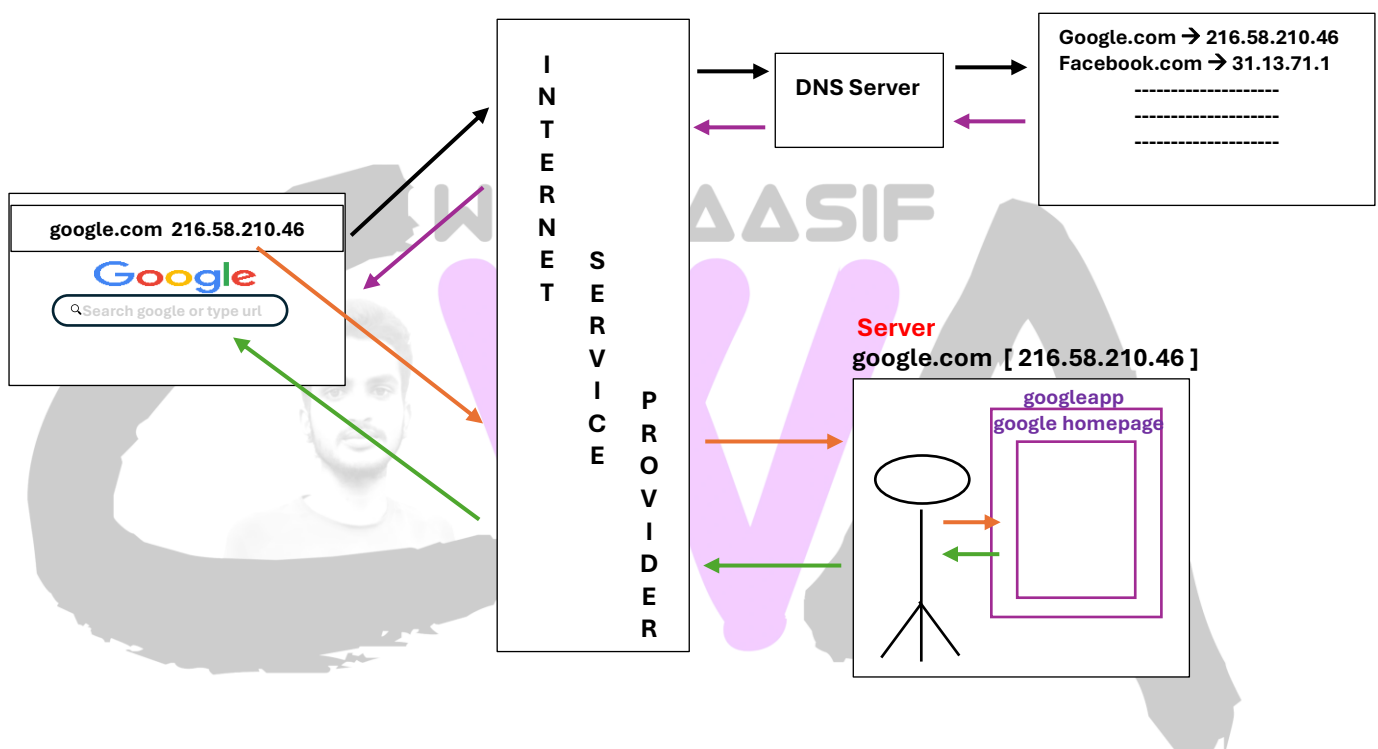
→ First request will be sent to our Internet Service provide [ISP]

→ Internet Service provider will connect with DNS[Domain Naming Server] Server which has the database of doman names and their IP addresses.

→ Internet Service provider will get IP address of www.google.com and send to the computer browser from which the request is coming.

→ Client browser will send request with Server IP address to Internet Service Provider then Internet Service provider will forward request to the actual Server, that is, google.com.

→ Upon recieving request, Server will identify the requested resource[webpage] and server will send back that requested resource[web page] to client.
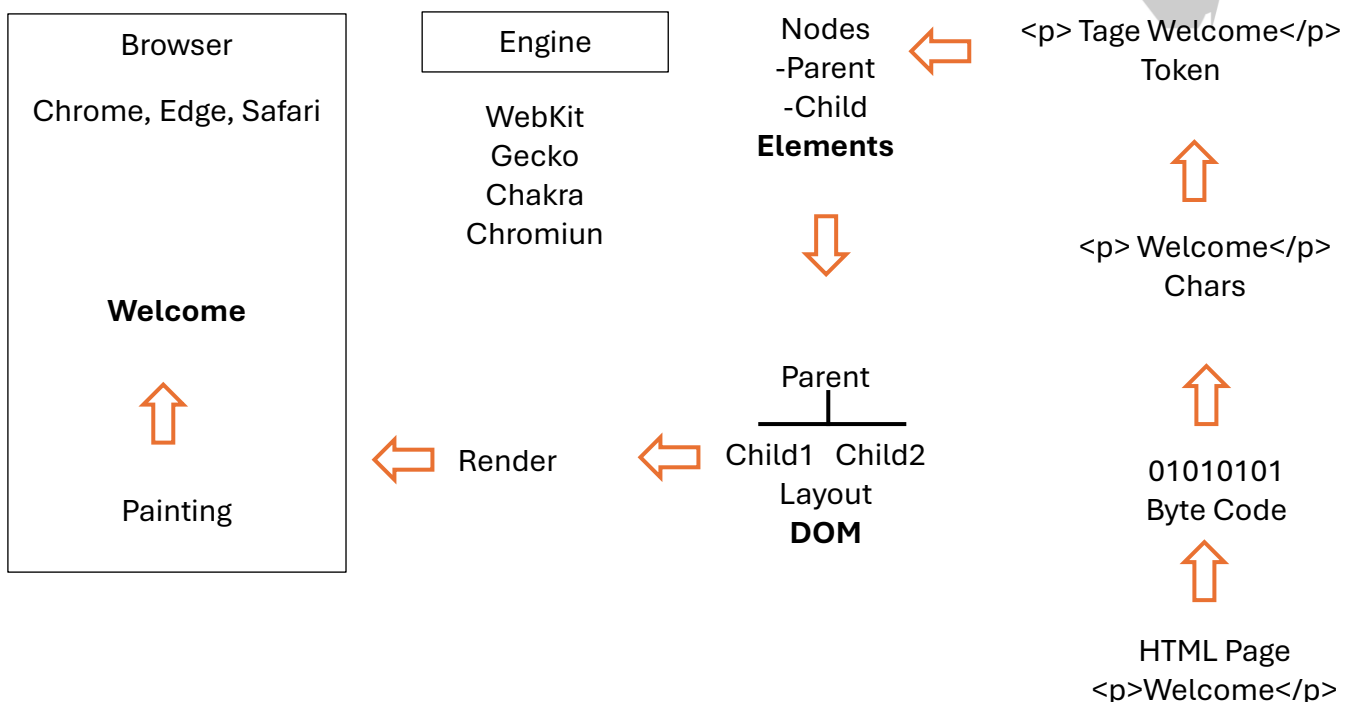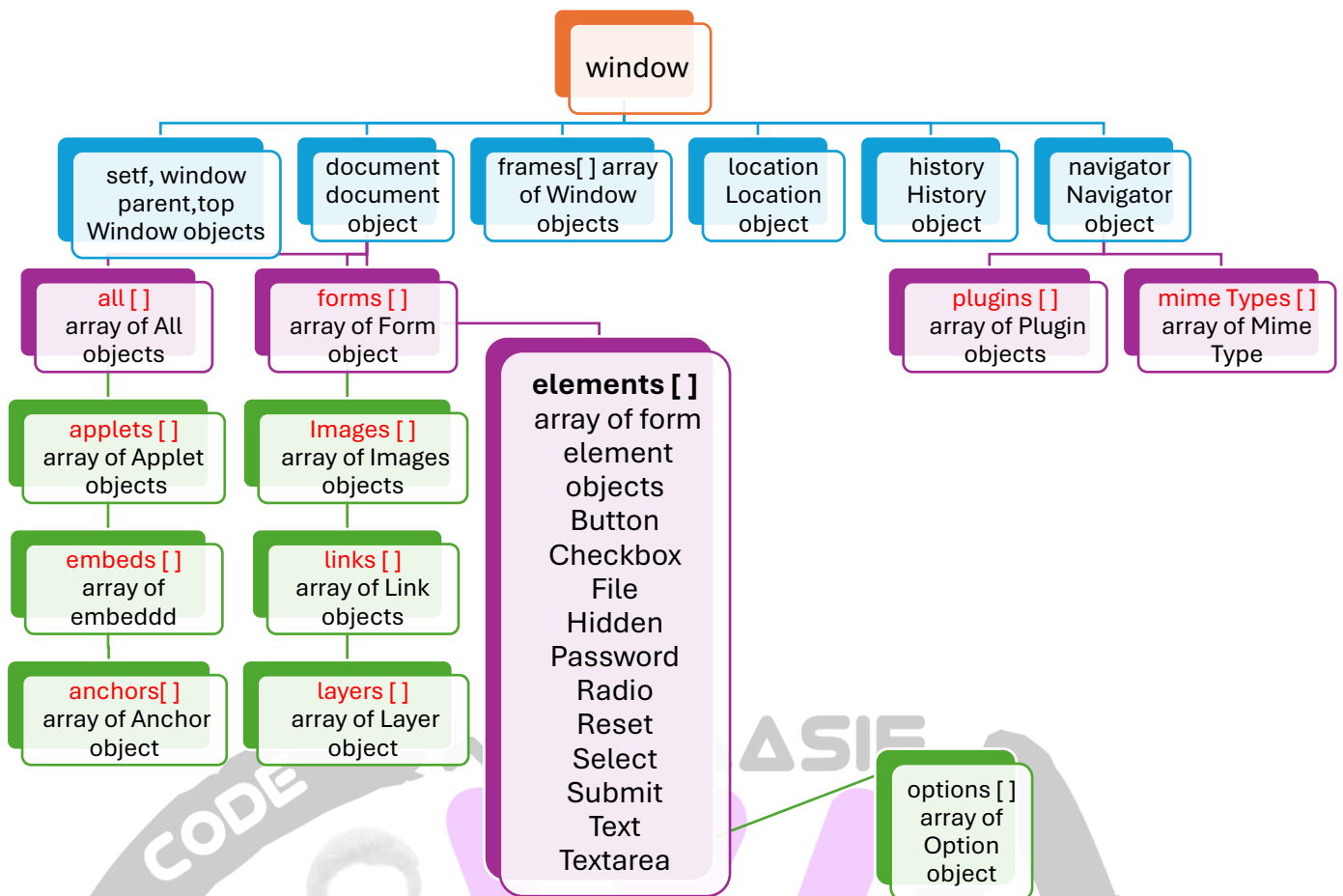
HTML
[Hyper Text Markup Language]

1. The term "Hyper" means "Beyond"
2. Hyper Text comprises of content beyond what you see.
3. Markup is general computer term derived from "Marking Up"
4. Markup language is a language used for presentation.
5. It is used for presenting on Browser.

Evolution

-The first Markup language used for Internet was GML built by CERN

-After GML [Generic Markup Language) - We use SGML [Standard] -

-1990's "Tim Berner Lee" introduced the concept of "Web" & HTML.

-HTML is superset to GML and SGML.

- IETF Group started developing HTML upto 3.2

-2004 W3C and WHATWG started developing HTML

World Wide Web Consortium

Web Hyper Text Application Technology Work Group

-HTML Version 4,

-HTML Latest Version 5.

Browser Architecture
================

Browser	Engine	Nodes	Tage Welcome
Chrome, Edge, Safari	WebKit Gecko Chakra Chromiun	-Parent ⬅ -Child **Elements**	Token

⬆

<p> Welcome</p>
Chars

⬆

Parent
|
Child1 Child2
Layout
DOM

01010101
Byte Code

⬆

Welcome

⬆

Painting   ⬅ Render ⬅

HTML Page
<p>Welcome</p>

```
                              ┌──────────────┐
                              │    window    │
                              └──────────────┘
    ┌──────────┬──────────┬──────────┬──────────┬──────────┐
┌─────────┐┌─────────┐┌─────────┐┌─────────┐┌─────────┐┌─────────┐
│setf,    ││document ││frames[ ] ││location ││history  ││navigator│
│window   ││document ││array of  ││Location ││History  ││Navigator│
│parent,  ││object   ││Window    ││object   ││object   ││object   │
│top      ││         ││objects   ││         ││         ││         │
│Window   │└─────────┘└─────────┘└─────────┘└─────────┘└─────────┘
│objects  │
└─────────┘
```

window

| setf, window parent, top Window objects | document document object | frames[ ] array of Window objects | location Location object | history History object | navigator Navigator object |

all [ ] array of All objects

forms [ ] array of Form object

plugins [ ] array of Plugin objects

mime Types [ ] array of Mime Type

applets [ ] array of Applet objects

Images [ ] array of Images objects

elements [ ]
array of form element objects
Button
Checkbox
File
Hidden
Password
Radio
Reset
Select
Submit
Text
Textarea

embeds [ ] array of embeddd

links [ ] array of Link objects

anchors[ ] array of Anchor object

layers [ ] array of Layer object

options [ ] array of Option object

Browser Architecture
=============

Markup => Bytes => Chars => Token => Elements => DOM => Layout => Render => Paint

HTML is a collection of Elements arranged in a Hierarchy called DOM. [Document Object Model]

HTML is a collection of Elements arranged in a Hierarchy called DOM. [Document Object Model]

HTML Elements are classified into 5 Groups
   1. Normal Elements

HTML Elements are classified into 5 Groups

1. Normal Elements
2. Void Elements
3. RC Data Elements
4. Raw Text Elements
5. Foreign Elements

Normal Elements:
- It returns a presentation directly on call back.
- It doesn't require additional attributes.
- It starts but can't stop implicitly.
- You have to stop explicitly by using end token.

start tag   \<b\>
end tag   \</b\>

Void Elements:
-The term void means no return value.
- It will not paint anything. [Empty]
- It will paint only content asked to.
-Doesn't require end tag

\<img\>

RC Data Elements
-Rich Content Data Elements
These elements will not allow any another element with in the context.

\<p\>

Welcome to \<b\> HTML \</b\> It is a mark-up language.

\</p\>

\<textarea\>

Welcome to \<b\> HTML\</b\> It is a mark-up language.

Raw Text Elements

Copy, copyright 2022

&copy copyright 2022

&#8377, 5600.00/-

Above is these are not tokens. these are normally group of chars

Foriegn elements

1.such type of elements can't work directly, such element depends on External library

That a called Foreign elements.

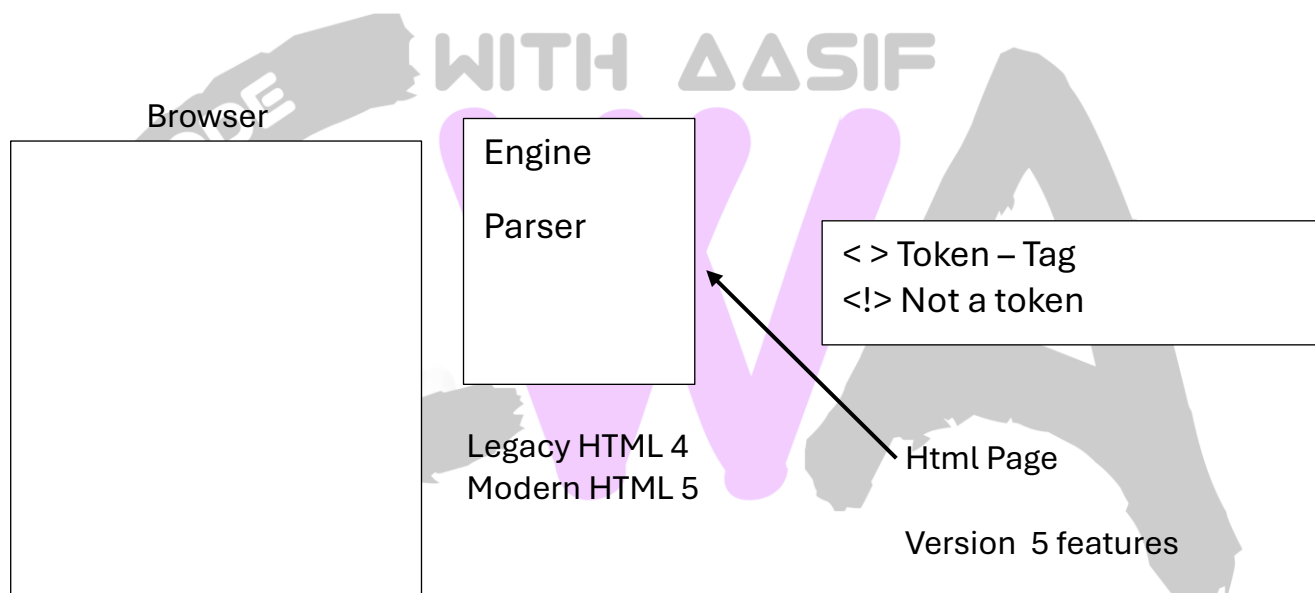**For example: SVG, MathML, Canvas etc**

HTML Page Structure

1. HTML Static page will have extension ".html or .htm"

FAQ: What is difference between .html and .htm?
Ans: No-difference

2. Static Page starts with "Document Declaration

Browser

Engine

Parser

< > Token – Tag
<!> Not a token

Legacy HTML 4
Modern HTML 5

Html Page

Version 5 features

2. Static Page starts with "Document Declaration"

<!DOCTYPE html>

It defines that we are using HTML 5

3. Page must contain Document Scope
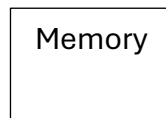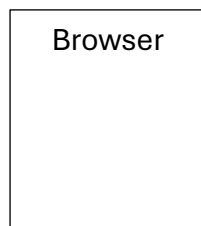
        <html>

        </html>

4. A document can display contents belonging to various regional language.

5. Every Document scope Comprtses of 2 Sectione

   a) Head <head> </head>
   b) Body <body> </body>

Browser	Memory	HTML Page
		Head
		content-1
		Body
		content-2
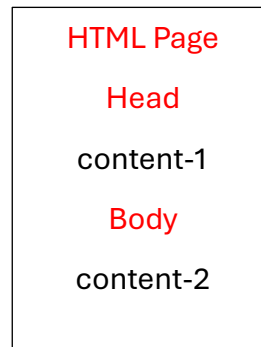
Head Section :-

It comprises of content which is intended to load into browser memory

HTML lead section usually comprises.

1. **title**
2. **link**
3. **meta**
4. **script**
5. **style**

1. **Title :-** Title contains text to display in browser title bar.
    Title is also used in bookmark.

2. **Link :-** It is used to link external link to webpage.

        <head>

            <link rel="stylesheet" href="style.css">

        </head>

3. **Meta in Head :-**
    meta defined meta data.
    It contains, information about your application given to browser and SEO.
    SEO (Search Engine Optimization]
    SEO makes your page search engine friendly.
    Search is done by "web spiders and web crawlers" [BOTS]
    meta is one of the SEO feature that maker page SEO friendly
    meta is also used to design responsive page.

    <HTML lang="eng">

    <meta charset="utf-8 / utf-16 / utf-32">
     utf = Unicode transformation format

8-Bit	16-Bit	32-Bit
Eng	Korean	Arabic
	China	
	Japan	

Charset :- It is required to defined the type of language used. **8-bit English**, **16-bit-kaca, China, Japan 32-bit Arabic**

4. **Script :-** It is used to embed script into page.

    <script type="script.js"></script>

5. **Style :-** It is used to embed style into page.

    <style type="style.css"></style>

    Ex:-

    <head>

    <style>

    h1{
    calor: white;
    backgrowd-Color: green;
    border: 2px solid black;
    padding: 10px;
    text-align: center;
    border-radius: 30px;
    }

    </style>

Bady section:-

→The body Section is defined by using <body> element.

→ Body is defined by using fallowing attributes.

1. bgcolor
2. background
3. text
4. align
5. alink
6. vlink
7. leftmargin
8. rightmargin
9. bottommargin
10. topmargin

<html lang >   Property

<meta content>   →   Attribute

Background :-   `<body background="./anji.jpg">`

                                       `</body>`

→ It is used to set background storage Image for body section.

Syntax :-

        `<body background="image-Path">`

Note :- Abways use only virtual et Reth for resources.

        [image]

        Don't we physical path

        Q. what is virtual Path?

        Ans.   Virtual Path is the local reference Path of resource in Project.

                1. First copy the image into your project local folder.
                2. Access from the folder without mentioning drive name. L.D, E).

                E: \Image shose-jp   Physical.

                Image/shase. Jpg   Virtual.

        back, slash only physical Path.

        forward slash only is used virtual Path.

To control background image you can apply the Following attributes.

        background-repeat: repeat / no-repeat/repeat-x/repeat-y;

        background-position: top, center, bottom, left, center, right;

        background-size: width px, hight px/cover/contain;

        background-attachment: fixed/scroll;

Q. Can we make background image lighter, darker or transparent?

Ans. You can't define by using any css attributes. You can implement by using alpha colors with containers.

 # Syntax:

```
<head>
    <style>
        body{
            background-repeat: no-repeat;
            background-position: center center;
            height: 500px;
            background-attachment: fixed;
        }
        div{
            background-color: rgba(0,0,0.05);
            heigh: 600px;
            margin: -50px;
        }
    </style>
</head>
<body background="/image.jpg">
<div>
</div>
</body>
```

Body Semantic Elements

→ Upto HTML 4 body section is designed by using Tables.

→Tables lead to a situation called "Kiss-of-Death".

→Tables are not SEO Friendly

→HTML 5 Introduced new elements into Body section to design a lag layout that can be SEO Friendly.

→ The new elements of HTML 5 body section are:

1. aside
2. article
3. dialog
4. figure
5. figcaption
6. header
7. footer
8. Main
9. nav
10. menu
11. section
12. div
13. span

→All are containers

```
<aside>

        your content

</aside>
```

→Div defines blocked content

→span defines inline content.

1. Add <style> container in <head> section

    <head>

        <style>

        </style>

    </head>

2. You can add style into styles container

    selector {

        attribute: value;

        attribute: value;

        }

Note: Select definer where you want to apply effects, to which element.

style Attributes

    background-color: set background color;

    color: set text color;

    text-align: set alignment left, center, right, justify;

    border :set border;

    [size, style, color]

    border: 2px Salid/dotted/dashed red;

margin-left   padding-left

margin- right   padding-right

margin-top   padding-top

margin-bottom   padding-bottom

margin: all directions   padding: all directions

border-radius: curved border [Pixel];

height: height of clement,

Width: width of element;

→Distribute Contents into multiple columns:

display: grid

grid-template-columns how many columns and size max 12 columns

Note: Columns width in pixels of Fractions.

pixels: max 1200px Fixed size

fr: max 12

adjust according to browser

</html>

Columns can be defined by using.

a) display: grid
b) columns

Display: Grid: Every column is individual. Contents of one Colum will not span to next column

Columns: It is used for continous columns. The content will span to next Column when if it's the Previous.

Syntax :-

```
#container {
    columns: 3;
}


#container {
    display: grid;
    grid-template-columns: 2fr 10fr;
}
```