HTML

Alakh Pandya

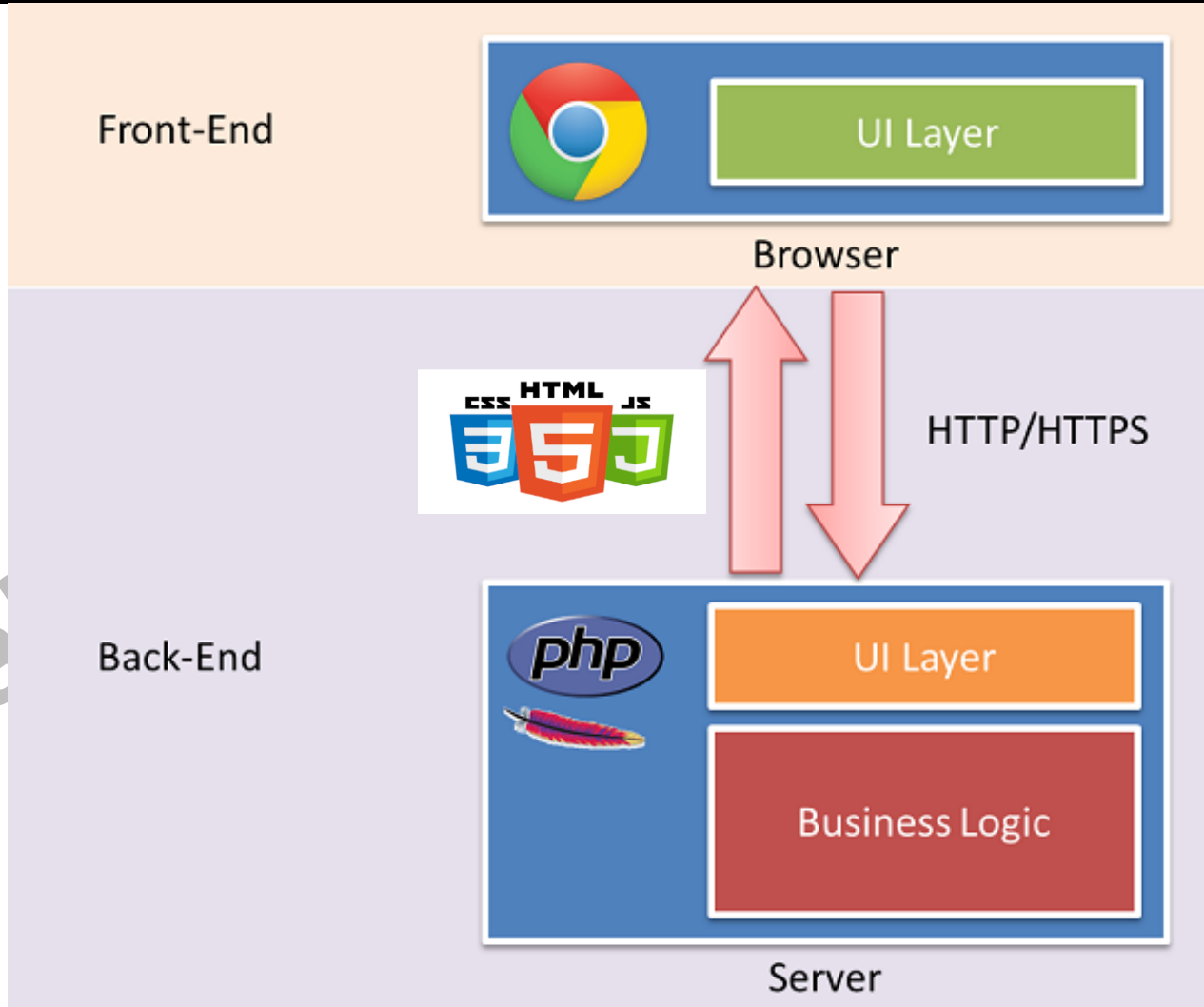# Basic Structure of a Website

Front End Mechanism

Understanding <head>

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta name="description" content="In this website you are going to learn advanced concepts of html">
    <meta name="keywords" content="html, web development, front end">
    <meta name="robots" content="INDEX, FOLLOW">
    <title>Document</title>
</head>
<body>

</body>
</html>
Comments in html
```

<h> & <p> tags

- <h1> to <h6> : where to use <h> tags & where not to use

- <p> tag: lorem

- Emmet: p*4

Text modification tags

- \<b\> tag = bold   - not recommended
- Strong tag = Important/bold
- \<i\> tag = italic   - not recommended
- Em tag = Emphasize/italic
- \<u\> tag = underline
- Del tag = strike through - not recommended
- Strike tag = strike through
- Sup tag = superscript
- Sub tag = subscript
- Mark tag = high light

Miscellaneous Tags

- \
: break

- \<hr>: horizontal rule

- Body tag attribute: bgcolor = "Yellow"

  - \<body bgcolor = "Yellow">

- Font Tag: \<font size = "15" color = "white">

- Marquee tag – moving text, image, behavior = "scroll", "slide", "alternate"; direction = "left", "right", "up", "down"; scrollamount = "20"; bgcolor; width; height; loop="1", hspace & vspace;

Links & Images

- \<img> tag:
- src
- External source
- Internal source
- alt = "alternate message"
- Width = " "   - css
- Height = " "   - css

- \<a> tag:
- href
- External Links
- Internal Links
- Name of link
- target = "_blank" attribute
- Image in place of 'name of link'

Lists

- OL:
  -
  - <ol type = "A">
  - <ol reversed>
  - <ol start = 10>
  - <li value = 400>

- UL:
  -
  - Type = "disc", "square", "circle"
  - Nesting of lists

Form

- \<form action = "form.php"\>

– \<div\>

- Name: \<input type = "text" name="myName"\>

– \</div\>

– \<div\>

- email: \<input type = "email" name="myEmail"\>

– \</div\>

– \<div\>

- \<input type = "submit" value="Submit Now"\>

– \</div\>

Form – <input>

- Values of "type" attribute:
- Type = "password" : invisible
- Type = "date" : Gives calendar to pick date
- Type = "number" :
- Gives incrementer/decrementer
- Allows 'e' as number as exponent
- <input type = "checkbox" name= "myCheckbox" checked> : checked by default
- Type = "raido"
- Give same name to enable selecting only one
- Type = "reset"
- Type = "file" : to upload file
- Type = "url"
- Readonly attribute:
- User can not modify it but can highlight & copy text from it

Form

- <textarea name= "myText" cols= "10" rows= "10" maxlength = 200 wrap= "hard">
  - Wrap =
- "soft" (default) – does not wrap while submitting the form
- "hard" – submitted form contains new line characters wherever wrapped
- <label for = "u_name"> Name </label>
  - <div>
- <input type = "text" name = "myName" id = "u_name">
  - </div>
- <select name= "myCountry" id = "country">
  - <option value = "ind" selected>India</option>
  - <option value = "sw">Switzerland</option>
- Placeholder & Required attributes:
  - <input type = "password" placeholder = "Enter Your Password Here" required>
- Autofocus Attribute:
  - Can be used in: button, input, select & textarea tags

Form – GET & POST Method

• <form action = "form.php" method = "get">

– Non-secure

– Has size limit of approx 3000 characters

– Everything will be visible in url

– Used in: search queries (eg. Google), where user wants to bookmark the results

• <form action = "form.php" method = "post">

– Secure, data is not visible in url. It is hidden inside the body of HTTPS request.

– No limit of characters

– Used to send sensitive data such as password or anything that you don't want to disclose openly.

– Form submission through POST can not be bookmarked.

Inline & Block Elements

- Inline elements
  - Takes up only required space
  - Span
  - Input
  - strong/em/u etc
  - <a>
  - Img

- Block Elements
  - Takes up entire width
  - Paragraph
  - Division
  - li

Table

- \<table>
  - \<thead>
    - \<tr>
      - \<th>
      - \<th>
      - \<th>
      - \</tr>
  - \</thead>
- \<tbody>
  - \<tr>
    - \<td>
    - \<td>
    - \<td>
    - \</tr>
  - \<tr>
    - \<td>
    - \<td>
    - \<td>
    - \</tr>
- \</tbody>
- \</table>

- \<table border = "1">
- \<table bgcolor = "yellow">
  - \<thead bgcolor = "blue">
  - \<tr bgcolor = "red">
  - \<td bgcolor = "green">
  - \<tr align = center>
  - \<td colspan = 2>
  - \<td rowspan = 2>
- \<img> tag in \<td>

IDs & Classes

- ID : Your Aadhar Card Number/ Passport number

    - Id must be unique for each element

    - Each element can have only one id

- Class : Name of its class (boy/girl/dog etc)

    - More than one elements can share the same class

    - One element can have more than one classes

- Emmet for creating any tag with some class

- Emmet to create div with some class

- Emmet for creating any tag with some id

- Emmet to create div with some id

    - a.class1.class2.class3#myLink*4

HTML Entities

-   – non breaking space
- &lt – less than
- &gt – greater than
- &pound
- &copy – copywrite logo
- Google: html entities reference

\<button\> tag

- Button tag – \<a\> tag in button to redirect
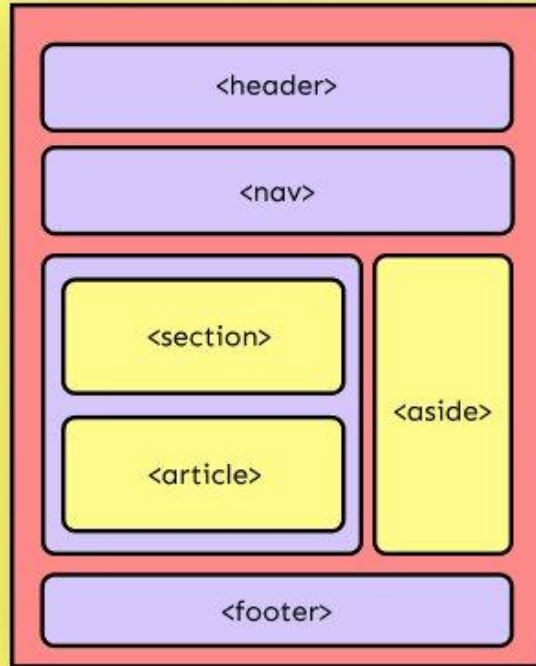
<video> tag

- Will not work on ie8 or lower
- Supported formats: mp4, webM, ogg
- Poster
- Autoplay
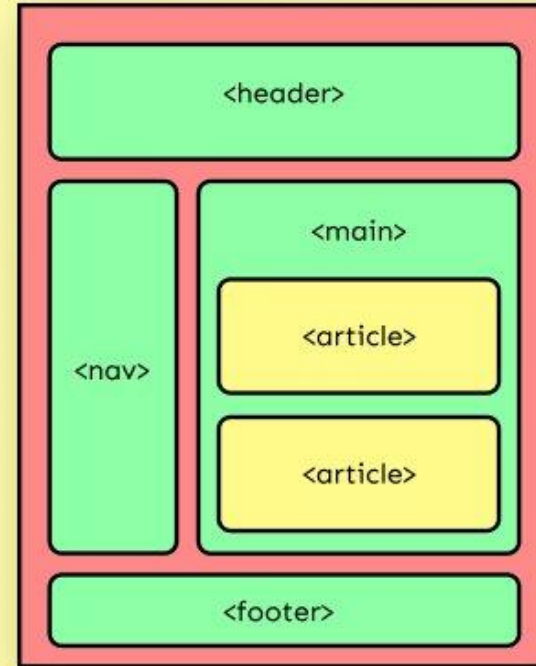- Muted
- Loop
- controls

\<audio> tag

- Will not work on ie8 or lower

- Supported formats: mp3, wav, ogg

- Autoplay

- Muted

- Loop

- controls

Semantic Tags

SEMANTIC ELEMENTS IN HTML

SEMANTIC BLOG STRUCTURE

Semantic Tags

- <details>
- Lorem200
- </details>
- Vs.
- <details>
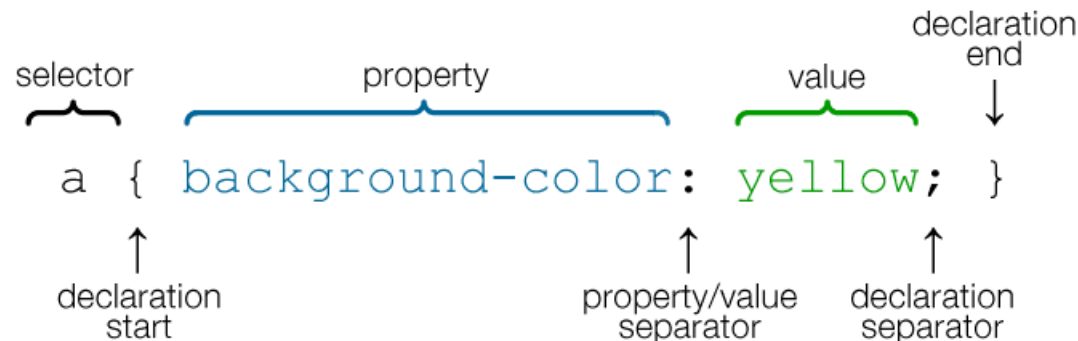- <summary> lorem50 </summary>
- Lorem150
- </details>

CSS

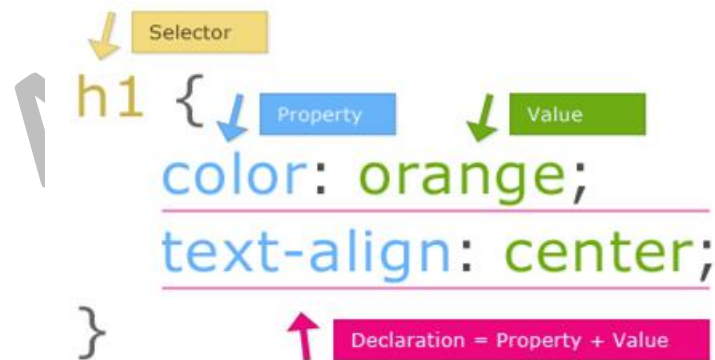*Alakh Pandya*

CSS – Cascading Style Sheet

- Beautification & Modern looking website

- Responsiveness

- We will use HTML only for structure, all styling responsibility we will put on shoulder of CSS

A typical CSS Block

- Example 1

- Example 2

Types of CSS

- Inline CSS

- Internal CSS

- External CSS

- Order of Priority

CSS Selectors

- Tag selector

- Class selector

- ID selector

- Multiple Selectors using 'and': space

- Multiple Selectors using 'or': comma

Nesting of Selectors

- div li a{
- }
- div a{
- }
- div a li{
- }

Descendant selectors

- div > a
  - Applies to only those \<a> tags which are **direct child** of any \<div>
- li:nth-child(3)
  - Applies to all the third \ child
- Li:nth-child(even/odd)
  - Applies to all the \ child who are on even/odd positions
- li:nth-child(3n+1)
  - Starts from putting n=0 in the formula
- div + a
  - Applies to those \<a> whose previous sibling is \<div>

pseudo classes & selecting by attributes

- :hover

- :visited

- :active

- a[target = '_blank']{

- }

- input[type= 'search']{

- }

- a[target]{

- }

Google Chrome Developer's Tools

Alakh Pandya

Fonts: font-family

- Web-safe fonts (Built-in fonts)

- Web fonts – Google fonts

– Using CDN

– By Downloading fonts:

- Downloade, unzip, paste in root/fonts folder

- Create fonts.css with:

- @font-face{

Font-family: custom-name;

Src: url('/fonts/font-folder/font-name.ttf');

- }

- Insert in html page: @import url("/css/fonts.css")

Fonts

- Font-size – in px (196th part of an inch), in rem

- Line-height: 1.5rem;

- Font-weight: bold;

- Font-style: italic;

- Font-variant: small-caps;

Responsive Design Requirements

- **Viewport**

- max-width/max-height

- Media query

- **Units like vh, vw, rem instead of px**

Size units – used in margin, padding, font size etc

- em : in multiple of parent class's px size

  – Used when you want to resize the child objects according to parent's size change

- rem : multiplies with html's font size

- vh : in percentage of viewport height

- vw : in percentage of viewport width

Colors

- By names

- RGB colors

- Hex colors

- Online Color pickers

- Color & background-color properties

Height, Width & Borders

- Height : 100px;
- Width : 500px; and in % of container, %vw, rem
- Border-width: 4px;
- Border-style: solid;
- Border-color: violet;
- Border: 4px solid violet
- Border-radius: 15px;

- Border-top, border-right, border-bottom & border-left with different width & color
- Border-top-left-radius
- Border-top-right-radius
- Border-bottom-left-radius
- Border-bottom-right-radius

Background image

- Background-image: url('/img/logo.png');

- Background-size: 100%, 500px, auto(default), 100vh

– Cover: cover the entire element; Auto: Actual Size of the image

– Contain: Tries to cover without changing aspect ratio & full image will be contained inside the element

- Background-repeat: repeat-x, repeat-y, no-repeat

- Background-position: 20px 100px

- Background-position: x y

– X: top, center, bottom

– Y: right, center, left

Before and After pseudo element

- All the pseudo elements requires to have 'content'

- 'content' may be null

- Pseudo elements remains inside the target element, they just go 'before' or 'after' the content of the target element. So we can not apply 'before' or 'after' to any empty tags like <input>, <br>, <img> etc

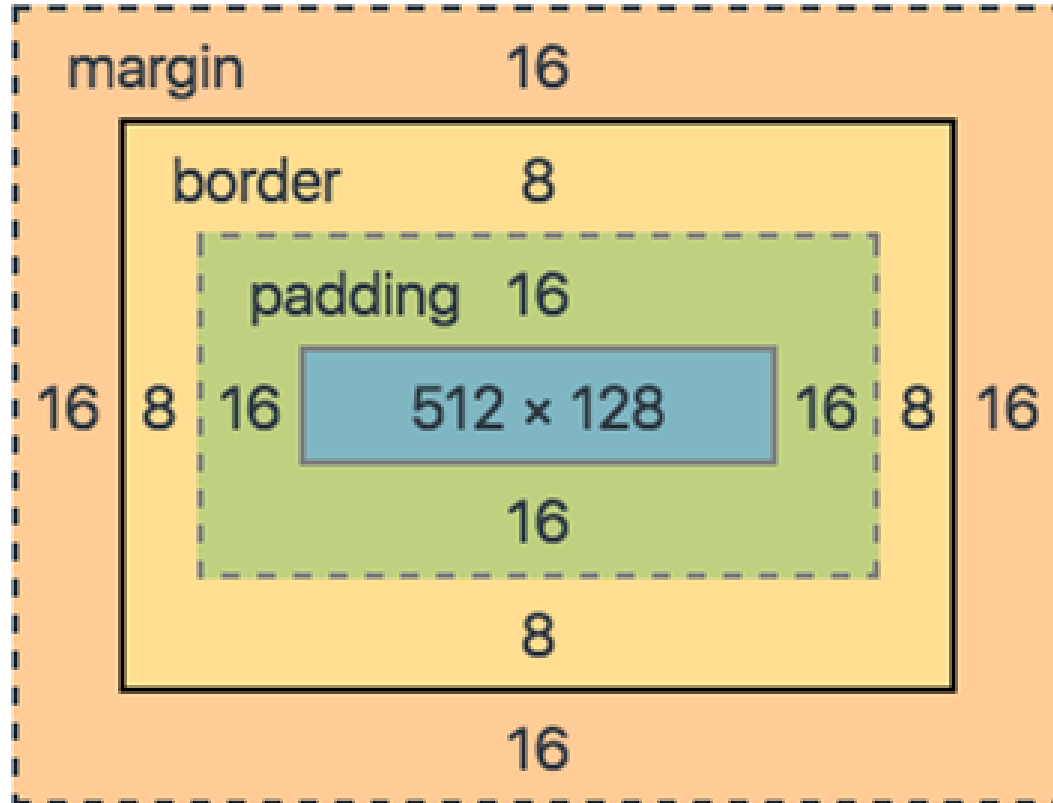- You can have only 1 'after' & only 1 'before' to each element on your page

Before and After pseudo element

- Applications

– To put * after required class content

– To show tips when hover

– Background image's opacity

Margin, Border & Padding – Box Model

- See in
- Inspect
- element
- in google
- chrome

Margin & Padding

- margin: 30px
- margin-top, margin-bottom, margin-left, margin-right
- margin: top-bottom left-right
- margin : top right bottom left (clock-wise)
- padding: 40px
- padding-top, padding-bottom, padding-left, padding-right
- padding: top-bottom   left-right
- padding : top   right   bottom   left (clock-wise)

Box-sizing: border-box

- Width of an element = content + padding + border

- Hence, if you change padding, width will also change.

- If you don't want that to happen, use

- box-sizing: border-box

- Used in nearly all website to make them responsive in css reset section.

CSS reset

```
* {
magin: 0;
padding: 0;
box-sizing: border-box;
}
body{
background-color: lightpurple;
}
```

Alignment

- Container class --> Margin: auto
- Centers horizontally
- Margin: 100px auto
- 100px margin in Y-direction, center in X-direction
- Float: left / right – along with border-box
- Clear: left / right / both
- Text-align: left / right / center / justify

Display

- Block – can set width, margin auto doesn't work without setting width

- Inline

- Inline-block: can set width, used along with

- box-sizing: border-box;

Position

●Position: absolute;

–Can use top, bottom, left & right properties relative to its **parent's position**. Will **not leave gap** in its original position.

●Position: relative;

–Can use top, bottom, left & right properties relative to its **normal position**. Will **leave a gap** in its original position.

●Position: fixed;

–Position is fixed **relative to browser window**. Used in chatbot. Element is converted into inline.
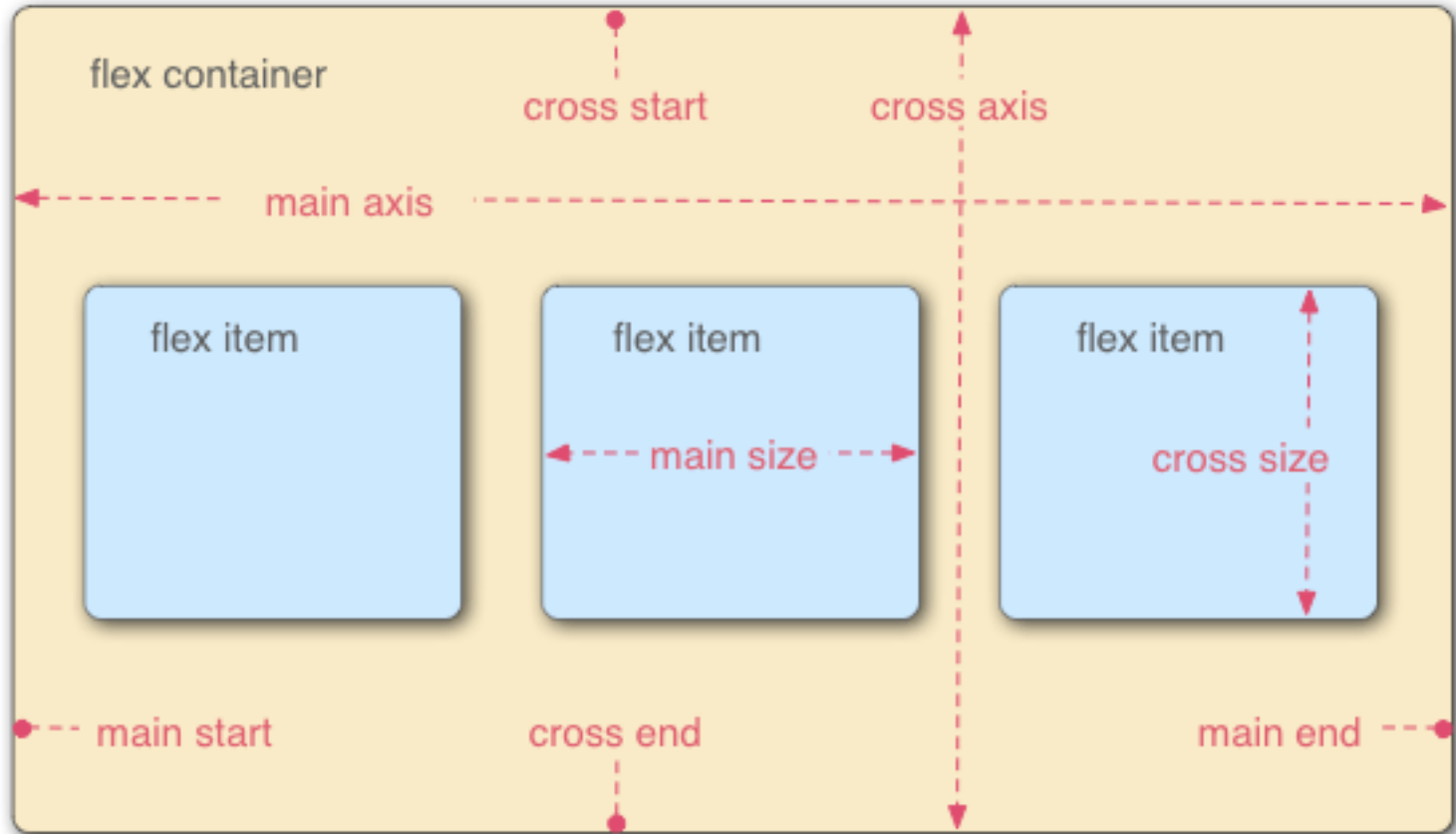
●Position: sticky;

–Keep element stuck **relative to Parent** while scrolling. Does not work without top/bottom/right/left properties (condition).

●Position: static; (default)

Miscellaneous Properties

- Border: none;

- Text-decoration: none;

- Cursor: pointer;

- List-style: none;

Flexbox – flexible box

Flexbox – flexible box

- Flex Container properties: display: flex;

- Flex-direction (default is row)

- Flex-wrap (default is no-wrap)

- Flex-flow: column wrap (combination of flex-direction and flex-wrap)

- Justify-content: center, space-between, space-evenly, space-around

- Align-items: center, flex-end, (default=flex-start)

- stretch – give even space vertically

Flexbox – flexible box

- Flex items' properties:

– Order: number; Higher the order, item will appear at the end

– Flex-grow: number; higher the number, item will grow with higher proportion when screen width increases

– Flex-shrink: number; higher the number, item will shrink with higher proportion when screen width decreases

– Flex-basis: 50px; when flex-direction: row, flex-basis will control width of flex item but when flex-direction: column, it will control height of the flex item

– Flex: grow_number   shrink_number   basis_px;

– Align-self: flex-start / center / flex-end

Project 1 – Design a landing page

Visibility, Display:none & z-index

- Visibility: hidden;

– Hides the element but leaves gap there

- Display: none;

– Hides the element & doesn't leave the gap

- Z-index

– Does not work with default value of position that is: position: static;

– Z-index is like layer number. The layer with highest z-index will be on top most position.

Responsive Design Requirements

- *Viewport*

- max-width/max-height

- **Media query**

- *Units like vh, vw, rem instead of px*

Media Query

- @media (max-width:500px){
.nav-items{
display: none;
}
- }
- Max-width: width 0 to max_width
- Min-width: any width that is more than min_width
- Max-width < 750

Media Query media types

- Only screen and (*condition)
- No parameter – for all uses: screen, printer etc

Vairables in CSS

- Why to use variables?

- Defining a variable:

--brand-color: rgb(223, 223, 223);

- Using a variable:

color: var(--brand-color)

- Where to declare & define variables- scope of variable – global & local

- : root{

--brand-color: rgb(223, 223, 223)

- }

- What can take in variables? Color, height, width, anything.

Animation

- Can be used for progress bar

Misc Topics

- Box-shadow: x-offset, y-offset, blur-radius, spread-radius, color

- Box-shadow: inset, x-offset, y-offset, blur-radius, spread-radius, color

- Multicolor box shadow:

- Box-shadow: x-offset, y-offset, blur-radius, spread-radius, color1, x-offset, y-offset, blur-radius, spread-radius, color2

- text-shadow: x-offset, y-offset, blur-radius, spread-radius, color

Important functions in CSS

- Invert function to invert the image colors