

HTML

Alakh Pandya

Basic Structure of a Website

Ala

Front-End

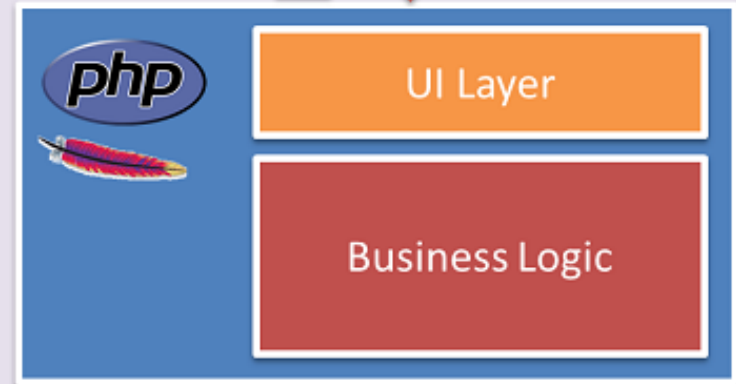


Browser



HTTP/HTTPS

Back-End



Server

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">
```

```
  <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

- 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:
- 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

• 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", "none"

-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 = “radio”

- 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
- Heading tags

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
- < – less than
- > – greater than
- £
- © – copyright logo
- Google: [html entities reference](#)

<button> & tag

.Button tag – <a> tag in button to redirect

Alakh Pandya

<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

<iframe> tag

```
<iframe src="iframeExample.html"  
frameborder="1"></iframe>
```

```
<a href="iframeExample.html" target="mobilePic">Click here</a>  
to see my next mobile
```

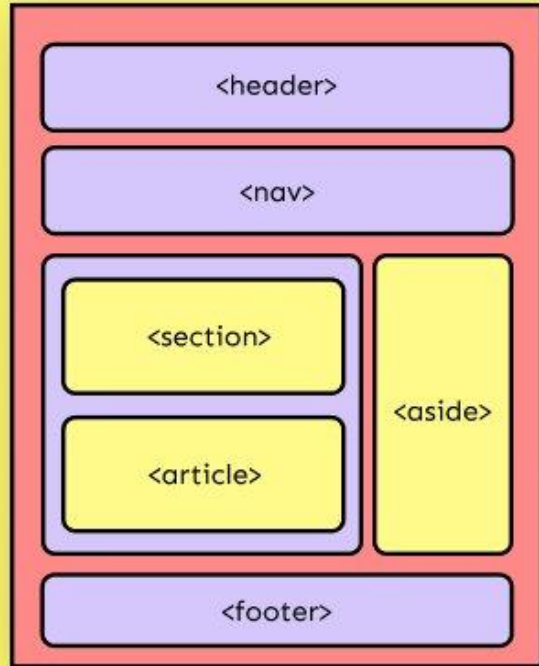
```
<iframe src="" frameborder="1" width="50%" height="315px"  
name="mobilePic"></iframe>
```

Embedding youtube videos

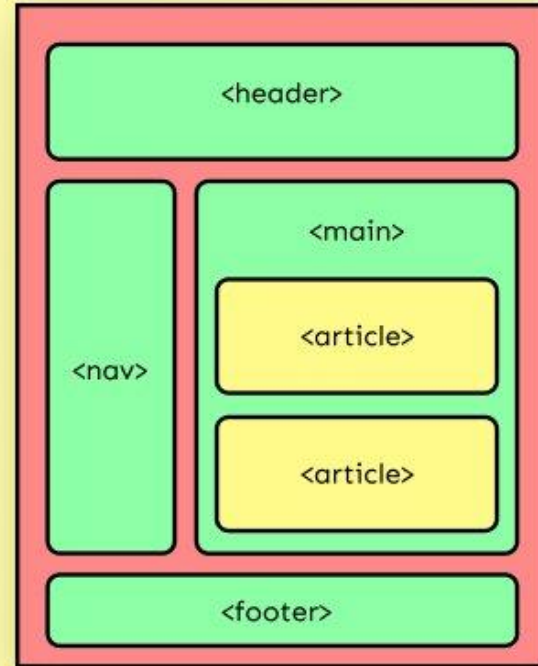
Youtube embed generator

Semantic Tags

SEMANTIC ELEMENTS IN HTML



SEMANTIC BLOG STRUCTURE



Semantic Tags

•<details>

–Lorem200

•</details>

•Vs.

•<details>

–<summary> lorem50 </summary>

–Lorem150

•</details>

CSS

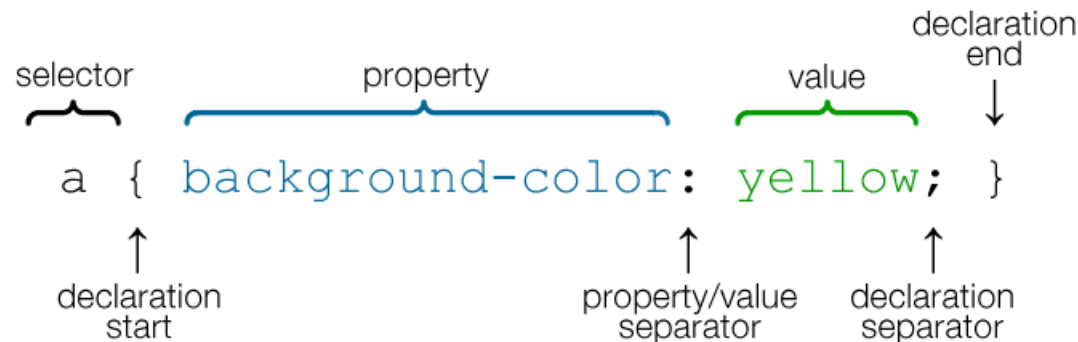
Alakh Pandya

CSS – Cascading Style Sheet

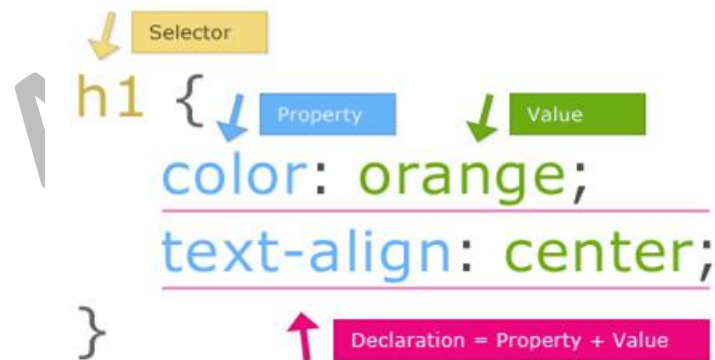
- Beautification & Modern looking website
- Responsiveness – 1% comes through html, 95% comes through CSS
- We will use HTML only for structure, all styling responsibility we will put on shoulder of CSS

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: space
- Multiple Selectors using 'or': comma

Nesting of Selectors

```
div li a{  
    background-color: brown;  
}
```

```
div a{  
}
```

```
div a li{  
    background-color: yellow;  
}
```

<div>

<a>Home

</div>

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

:link

:hover

:visited

:active

```
.a[target = '_blank']{
```

```
}
```

```
.input[type = 'search']{
```

```
}
```

```
.a[target]{
```

```
}
```

Google Chrome Developer's Tools

Spcificity

Alakh Pandya

Fonts: font-family

- Web-safe fonts (Built-in fonts)

- Web fonts – Google fonts

- Using CDN

- By Downloading fonts:

- Download, 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 custom css 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;
- letter-spacing
- Text-transform: uppercase

Responsive Design Requirements

.Viewport

.max-width/max-height

.Media query

.Units like vh, vw, rem, em & % instead of px

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

- em : in multiple of font size

- Used when you want to resize the child objects according to font size

- rem : multiplies with html's font size; 1 rem = 16px

- vh : in percentage of viewport height

- vw : in percentage of viewport width

Colors

- By names
- RGB colors
- Hex colors
- Online Color pickers, Color Picker Extension
- Color & background-color properties
- Opacity using `rgba()`

Height, Width & Borders

- Height : 100px;
- Width : 500px; and in % of parent, %vw, rem, em
- 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, round
- Background-position: 20px 100px
- Background-attachment: fixed;
- Background-position: x y
 - X: top, center, bottom
 - Y: right, center, left

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 **actual 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)

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

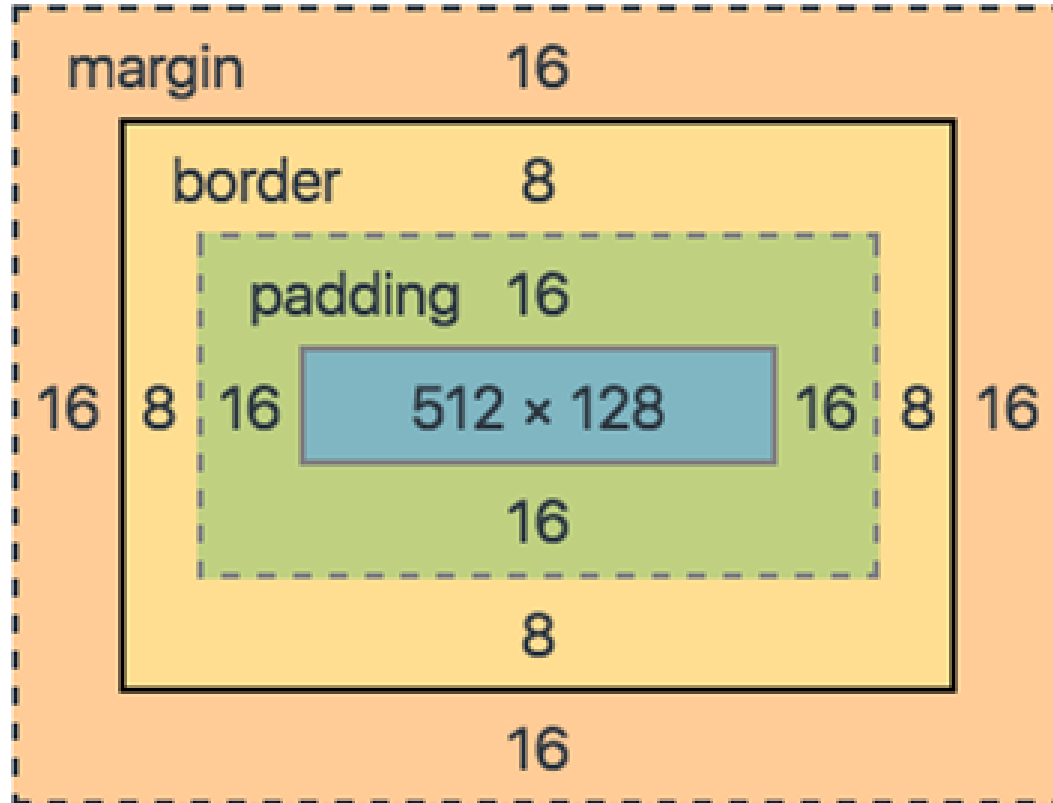
Before and After pseudo element

- Applications

- To put * after required class content
- To show tool 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)
- Overlapping margins in block elements & Non overlapping margins in inline elements
- 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

```
*, *::before, *::after
{
margin: 0;
padding: 0;
box-sizing: border-box;
}
body{
background-color: lightpurple;
}
```

Alakh Pandya

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

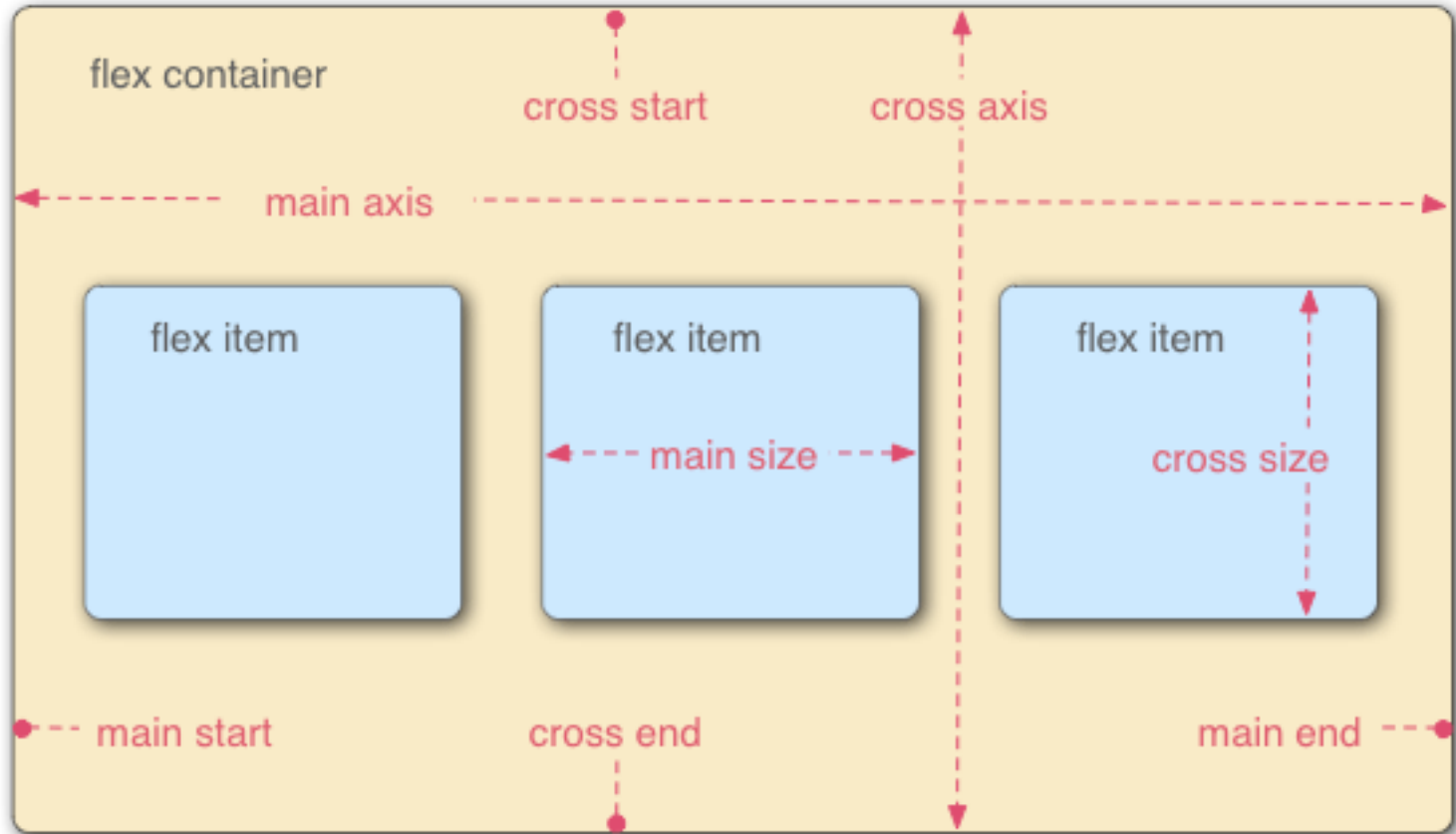
Display

- Block – can set width, margin auto doesn't work without display block
- Inline
- Inline-block: can set width
- box-sizing: border-box;

Miscellaneous Properties

- Border: none;
- Text-decoration: none;
- Cursor: pointer;
- List-style: none;/ list-style-type: none;
- Outline: 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)

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

Alakh Pandya

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.

Variables 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.

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

Alakh Pandya

Overflow – when you give fix width to container

- Hidden: Hides the overflowing data
- Scroll: Gives a scrollbar (horizontal/vertical) from the beginning
- Auto: Gives scrollbars once the data starts overflowing
- overflow-x, overflow-y and overflow

Animation

- Can be used for progress bar

Alakh Pandya

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

Alakh Pandya

Owl Carousel

Alakh Pandya



Daniel Clifford
Verified Graduate

I received a job offer mid-course, and the subjects I learned were current, if not more so, in the company I joined. I honestly feel I got every penny's worth.

"I was an EMT for many years before I joined the bootcamp. I've been looking to make a transition and have heard some people who had an amazing experience here. I signed up for the free intro course and found it incredibly fun! I enrolled shortly thereafter. The next 12 weeks was the best - and most grueling - time of my life. Since completing the course, I've successfully switched careers, working as a Software Engineer at a VR startup."



Jeanette Harmon
Verified Graduate

An overall wonderful and rewarding experience

"Thank you for the wonderful experience! I now have a job I really enjoy, and make a good living while doing something I love."



Patrick Abrams
Verified Graduate

Awesome teaching support from TAs who did the bootcamp themselves. Getting guidance from them and learning from their experiences was easy.

"The staff seem genuinely concerned about my progress which I find really refreshing. The program gave me the confidence necessary to be able to go out in the world and present myself as a capable junior developer. The standard is above the rest. You will get the personal attention you need from an incredible community of smart and amazing people."



Jonathan Walters
Verified Graduate

The team was very supportive and kept me motivated

"I started as a total newbie with virtually no coding skills. I now work as a mobile engineer for a big company. This was one of the best investments I've made in myself."



Kira Whittle
Verified Graduate

Such a life-changing experience. Highly recommended!

"Before joining the bootcamp, I've never written a line of code. I needed some structure from professionals who can help me learn programming step by step. I was encouraged to enroll by a former student of theirs who can only say wonderful things about the program. The entire curriculum and staff did not disappoint. They were very hands-on and I never had to wait long for assistance. The agile team project, in particular, was outstanding. It took my learning to the next level in a way that no tutorial could ever have. In fact, I've often referred to it during interviews as an example of my development experience. It certainly helped me land a job as a full-stack developer after receiving multiple offers. 100% recommend!"

Alakh Pandya

[illegible]



