HTML5:

They have more descriptive kind of html tags.

Main, header, footer, nav, video, audio, section, article.

Descriptive structure to your html

Easy to read and for SEO

Main tags tells about the main content in your web page

<main>

<h1>Hello World</h1>

<p>Hello Paragraph</p>

</main>

Image:

<img src="https://www.your-image-source.com/your-image.jpg" alt="Author standing on a beach with two thumbs up.">

Self closing tag with src to set image url

Alt to set alternate text for display if no image present

Anchor:

Used to specify the link

<a href="https://freecodecamp.org">this links to freecodecamp.org</a>

Anchor with internal links

<a href="#contacts-header">Contacts</a>

...

<h2 id="contacts-header">Contacts</h2>

To make internal link define anchor tag with href as id of the element you want to navigate remove target attribute

Make dead link href=”#”

This is useful where you want to block normal behaviour of your link and handle with javascript

Bulleted Unordered list:



Ordered list:

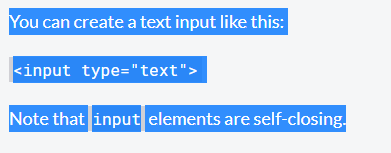
Numere list

<ol>

<li></li>

</ol>

Text input

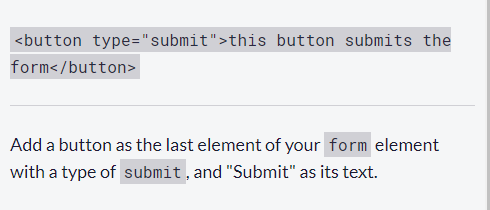


 <input type="text" placeholder="cat photo URL">

We can setup water mark text for the control by setting place holder value.

Htm form tag:

Used to submit/post the data to server



Make your input field required.

 <input type="text" placeholder="cat photo URL" required>

Radio button and radio button group:

Each of your radio buttons can be nested within its own label element. By wrapping an input element inside of a label element it will automatically associate the radio button input with the label element surrounding it.

<label>

<input type="radio" name="indoor-outdoor">Indoor

</label>

Suppose if you want to have radio button group then you have to make same name for all radio button belongs to one group.

Always good to have for attribute which id the ID of the input control we are associating this will have better linking between label and control.

<label for="indoor">

<input id="indoor" type="radio" name="indoor-outdoor">Indoor

</label>

So to make radio button group both should have same name attribute value.

Checkbox:

Using for attribute between control and containing label element we can establish relation ship between these 2 control.

<label for="loving">

<input id="loving" type="checkbox" name="personality"> Loving</label>

In above both type of control when the for submitted value from the value attribute of the control will be submitted to the server

So we need to set the value attribute properly.

HTML property to check the any option by default we use the checked attribute.

<input type="radio" name="test-name" checked>

Container Div:

DOCTYPE

Basically used to indicate the version of the HTMl used in the document



CSS: cascading style sheet:

Deals with the presentation of the page rendered on the web browser

It tell what font to take colour to take etc..

3ways to apply:

1. In line
2. Document level
3. External style sheet

We can used font family out side the system define font family

For that first refer the font family link inside the project and specify the value inside the style element

font-family: Font-family\_name, Generic\_name

Generic\_name optional value used a fall back family name if main specified family name Is not present in the font family list

Names are case sensitive if any name present with the more than one word may be we can enclose it within the double quotes.

If the font family has multiple name then basically it’s a order in which we have to fall back the font family if the one style is not present

For standard css font made of two words we don’t have to enclose inside double quotes.

3 css property deals with spacing

An element's padding controls the amount of space between the element's content and its border.

An element's margin controls the amount of space between an element's border and surrounding elements

Unit of length in Css

Absolute we use px this is mapped to the actual physical measurements (mm, cm etc)

Relative unit: we use em, rem this is relative to the elements font size

Creating css variable:

Put two hyphen followed by the name

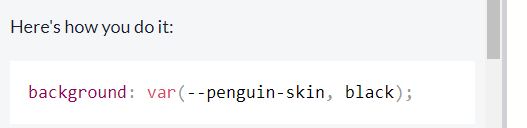
--penguin-skin:grey



Using this variable can be done with the help of var function

Background-color:Var(--penguin-skin)

Fall back value to css variable





Inheriting css variable:

When you define variable inside the css class that can be used by the selected inside which it is defined.

Also inside the descendants of the parent selectors

:root pseudo class selector used to point to the root element in the document. Usually Htm element

If any variable define in the parent selector then it cab be accessible anywhere in child hierarchy

So defining it in root level makes it kind of global variable.

It also possible to override this value in specific level

By setting the value specific to that section.

Tag to make text italic

<em> content tot be used </em>

Say you want to strike the text in display

<s></s>

Style:

Text-decoration:line-through

Colors:

rgba()

rgba stands for:  
  r = red  
  g = green  
  b = blue  
  a = alpha/level of opacity

each color value range from 0-255

and opacity value range from 0-1

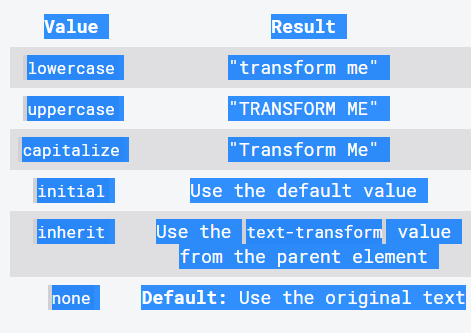
Box radios

The box-shadow property takes values for

* offset-x (how far to push the shadow horizontally from the element),
* offset-y (how far to push the shadow vertically from the element),
* blur-radius,
* spread-radius and
* color, in that order.

We can set the opacity

Text transformation for changing the case in text.



Line height:

Vertical space taken by the text in each line.

Pseudo classes in css

a:hover {

color: red;

}

Transformation on mouse over

Positioning in CSS:

Css treats every element as its own box. That why its called as css box model

Block level items always falls in the next line.

Inline items sits with the surrounding element

By setting the item position value as relative we can move the element from its actual flow location

We can also specify the offset position for left, right, top, bottom

Moving the element using relative will not remove it from the normal flow so the space allocated for the element remain as it is.

Absolute positioning:

Positioning of the element is with respect to its absolute or relative positioned parent. If no such parent exists then it will be positioned from the document root.

Here element will be moved from it normal flow position and next element will fill the normal flow position of the element.

Fixed:

One key difference between the fixed and absolute positions is that an element with a fixed position won't move when the user scrolls.

Similar to fixed element will be moved out of normal flow position and location fixed even if you scroll the scroll bar. Eg: menu which is always visible on scroll

Float: right or left

Used with width property normally used to move the element from its normal float position wrt parent container.

CSS 3 uses color wheel as colo picker

Where value for Rgb specified in %

Colour transition on the element in web page using the gradient

Background colour can be set with gradient

background: linear-gradient(gradient\_direction, color 1, color 2, color 3, ...);

we can also repeat the gradient patter with repeating-linear-gradient

URL method to load background

  body {

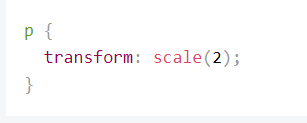
background :url('https://cdn-media-1.freecodecamp.org/imgr/MJAkxbh.png')

  };

</style>

content transformation with scale

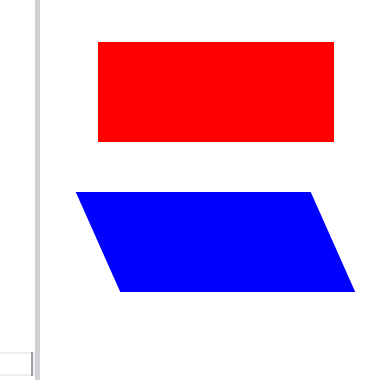
Transform: scale(value);



We can scale up and down wrt size

Also can be combined with the other pseudo selectors

Element skew



transform :skewx(24deg);

transform :skewy(24deg);

angle to the skew direction

Pseudo elements

::before : apply something before the element

::after : apply something after the element.

.heart::before {

content: "";

background-color: yellow;

border-radius: 25%;

position: absolute;

height: 50px;

width: 70px;

top: -50px;

left: 5px;

}

While using this pseudo element the content property is mandatory because while dealing with share we should have those property with empty values.

Animation with css:

@keyFrame

Here we say during the specified animation duration what action should happen.

For this we need name of the animation specified in the control selector

animation-name: ‘name value’

animation-duration: 3s

then define a key frame

@keyframes name

{

0%:{}

100%:{}

}

Inside this we use percentage values to indicate the animation over the period of time.

After running the animation once it reached the 100% it reset back to old value suppose if you want to maintain value for some action

Use: animation-fill-mode:forwad;

nimation-iteration-count:

say you want to iterate animation from multiple times then we have this property

if you want to iterate forever use infinite as value

animation-timing function:

indicates how the element changes over the given animation time

speed of the animation cn be controlled with this property.

ease

ease-int

ease-out

linear

HTML5 semantic elements:

Header: Any content to be displayed on the header section goes in here

Footer: Page footer content will be displayed here

Main: Content to be displayed on each page inside the body should go in here

Nav: navigation panel comes here

Section: is thematical related content can be placed within each other

Article: Self-contained section inside the web page. Wrap independent, self-contained section

If one book is articling each chapter inside it is a section

<div> - groups content

<section> - groups related content

<article> - groups independent, self-contained content

With semantic elements html structure will looks like this.

<!Doctype html>

<head></head>

<body>

<header>

<nav> </nav>

</header>

<main>

<section>

<article></article>

<article></article>

</section>

</main>

<footer></footer>

</body>

</html>

**Audio:**

To play audio on the web browser

<audio id=”” controls>

<source src=’’ type=’’/>

<source src=’’ type=’’/>

</audio>

<audio id="meowClip" controls>

<source src="audio/meow.mp3" type="audio/mpeg" />

<source src="audio/meow.ogg" type="audio/ogg" />

</audio>

Controls will tell you that whether you want to show player controls.

Improve accessibility of the figure element with fig and fig caption.

<figure>

<img src=”” alt=””/>

<br/>

<figcaption>

Sd kf kfkkf al

</figcation>

</figure>

User label element for the text labels;

Say you want to group certain elements together and put a description text over it

Then we can add a tag called Field set

And description can be given with legend

Access key attribute helps you to access the element from the keyboard key.

Tab index property used to

Media query:

Present content based on different viewport size

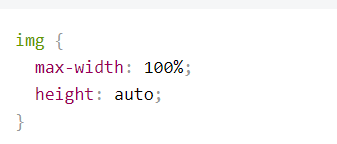
Syntax for writing such CSS

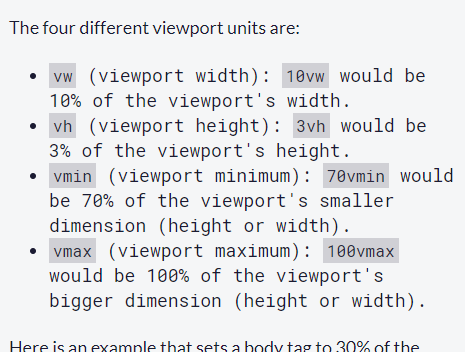
@media{mention the screen size}

Max-width: abc px (render this style for all the device where the width is up to this )

Min-width: abc px (render this style for the device where the width of the device is more than or equal to this)

Make image responsive





Flex box or flexible boxes used for ui alignment. And predicts and put the element which are dynamic in nature

Display flex used to set flex property on the element.

So if I add property display: flex this will make the container as flex container.

With that we can make any element inside the container move inside the container in any direction

Flex-direction: used to set the direction for the parent element

Row

Column

row-reverse

column-reverse

puts then child in respective direction.

CSS for Flex :

Justify-content:center

Flex-start: for row align move content to extreme left, column to the top

Flex-end: move content to right for row, bottom to the column

Space-around: similar to space between but here first and last elements are not fixed to the extreme end of the

Space-between: all the items inside the flex container are aligned main axis. First and the last element moved to the extreme corner with the empty space distributed evenly between the elements.

Space-evenly: Distribute space evenly between the elements max space shown at the beginning and end of the flex container.

Align-items: used to align the item against the cross axis.

Similar to above property

Flex-start:

Flex-end:

Center:

Stretch:

Baseline:

Wrapping of the flex:

Using css we can wrap the items into multiple lines

Initially everything will be put in single line.

Flex-wrap:

Nowrap: default option

Wrap:

Wrap-reverse: wraps in opposite direction of above

Property for the flex items:

When the screen size is reduced items will be shrinked respectively

Flex-grow: opposite if the shrink we have used

Flex-basis: indicates the size to be applied before specifying any css

Its can be auto which will set the size based on content

Else put based on the unit

Short hand representation for flex grow, shrink, basis

Using flex:

Order of the flex boxes:

Align-self:

Align the items to individual

Float:

Clear:

Vertical: align:

CSS grid system:

Turn any element into a grid by using the display: grid css property helps is putting all the css associated with the grid.

Css grid system parent is call container and children’s are called items

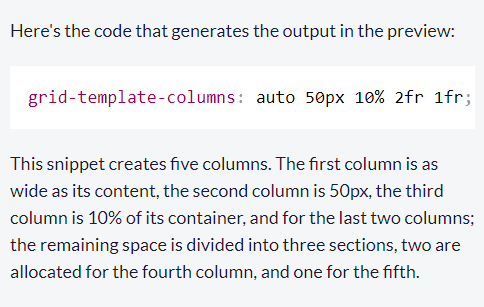
Simply putting grid system will not have any effect

We need to define template column basically indicates how many boxes should be displayed in each rows

Grid-template-columns: 10px 10px 10px;

Indicates total 3 columns to be displayed in a row each width the width 10px;

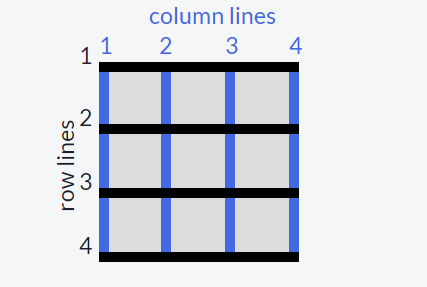
Grid-template-rows: used to set the rows



Grid-column-gap: 10px used to give the gap between the columns

Grid-row-gap: usd to give gap between the rows;

Short hand representation for this is grid-gap: row px column px



Gaps inside the grid system are number by line count. Using this count we can as well specifies the column can be used by the content.

Grid-column: 1/3 it says use the grid column line from 1 to 3 to fill the element.

Justify-self: used to align content inside the allotted space

Values:

Stretch

Center

Start

End

Align -slef: for the vertical alignment of the element

Justify-items this will set the css to all items in the grid system

We can also group the cells into different areas.

Grid-template-areas



Top 3 cells merged as one area

Bottom 3 cells merge as another row

Centre one cell is merged into 2 different areas.

Once the template area is setup we can refer the area in css using

.temp{

Grid-area: header;

}

Define the area on go:

grid-area: horizontal line to start at / vertical line to start at / horizontal line to end at / vertical line to end at;

we can generate repetitive rows columns using repeate function.

Grid-template-rows: repeate(100,50px)

Alos column

grid-template-columns: repeat(2, 1fr 50px) 20px;

this is equivalent of grid-template-columns: 1fr 50px 1fr 50px 20px;

minmax(min value, max value)

used to specify the minimum and the maximum value that a cell can take when the size is resized.

We can also automatically generate the number of rows and column in my grid system using

Auto-fill option in repeat that takes width height and makes as many cells as possible based on the available space.