**USEFUL HTML|CSS RESOURCES**

* [**character refrences: https://html.spec.whatwg.org/multipage/named-characters.html**](character%20refrences:%20https://html.spec.whatwg.org/multipage/named-characters.html)
* **Check if a tag/element is valid:** 
  + [**https://caniuse.com/**](https://caniuse.com/)
* **Check if your html code is valid:**
  + [**https://validator.w3.org/#validate\_by\_input**](https://validator.w3.org/#validate_by_input)
* **Check if css code is valid:** 
  + [**https://jigsaw.w3.org/css-validator/#validate\_by\_input**](https://jigsaw.w3.org/css-validator/#validate_by_input)
* **Everything html:**
  + [**https://www.w3schools.com/html/default.asp**](https://www.w3schools.com/html/default.asp)
  + [**https://developer.mozilla.org/en-US/docs/Web/HTML**](https://developer.mozilla.org/en-US/docs/Web/HTML)
* **Everything css:**
  + [**https://css-tricks.com/almanac/**](https://css-tricks.com/almanac/)

**Basic HTML outline**

<!DOCTYPE html>  
<html>  
<head>

</head>

<body>  
  
  
</body>  
</html>

* <!-- --> COMMENT
* lang attribute inside the <html> tag, to declare the language of the Web page
  + example:

<html lang="en">

* HTML headings are defined with the <h1> to <h6> tags
  + Example:

<h1>Heading 1</h1>

* <div> tag is known as a block level element
* <span> tag is known as an inline element
* HTML paragraphs are defined with the <p> tag
  + <br> is used to break apart paragraphs, no need for end tag
  + style attribute is used to add styles to an element, such as color, font, size, and more
    - example:

<p style="color:red;">This is a red paragraph.</p>

<body style="background-color:powderblue;"> (changes background color of the entire webpage)

* + - * + color formatting: <https://www.w3schools.com/html/html_colors.asp>

<p style="font-family:courier;">This is a paragraph.</p>

<p style="font-size:160%;">This is a paragraph.</p>

<p style="text-align:center;">Centered paragraph.</p>

* + - Formatting tags: <https://www.w3schools.com/html/html_formatting.asp>
  + title attribute defines some extra information about an element when you mouse over a sentence
    - example:

<p title="I'm a tooltip">This is a paragraph.</p>

* HTML links are defined with the <a> tag
  + The href attribute specifies the URL of the page the link goes to
    - example:

<a href="https://www.w3schools.com">Visit W3Schools</a>

* HTML images are defined with the <img> tag,the source file (src), alternative text (alt), width, and height are provided as attributes:
  + example:

<img src="w3schools.jpg" alt="W3Schools.com" width="104" height="142">

Linking external CSS style sheet

<!DOCTYPE html>  
<html>  
<head>  
  <link rel="stylesheet" href="styles.css">  
</head>

* + rel = linking to a css stylesheet
  + href = name of css doc
* CSS
* Comments : /\* …… \*/
  + Selector{

Property:value;

}

* \*{} – universal selector
* p{} – type selector
* #id{} – ID selector
* .class{} – class selector
* Descendant selector- when theres 2 or more selectors within a selector
  + Id id{}
* Pseudo-classes: ( <https://developer.mozilla.org/en-US/docs/Web/CSS/:focus> )keyword added to a selector that specifies a special state of the selected element(s). For example, the pseudo-class [:hover](https://developer.mozilla.org/en-US/docs/Web/CSS/:hover) can be used to select a button when a user's pointer hovers over the button and this selected button can then be styled.
  + <https://css-tricks.com/almanac/selectors/a/after-and-before/>
    - <https://www.toptal.com/designers/htmlarrows/arrows/>
  + button:hover {

color: blue;

}

* Percentages are great for flexible layouts
* Font size measurements:
  + Em: size is relative to the font-size of the parent size. When used on other properties, its relative to the font-size of the element itself
    - <http://pxtoem.com/>
    - Parent size: 20px
    - Em size: 1.5em = 20\*1.5\*2
  + Rem:  short for root em is a relative unit that’ll always be based upon the font-size value of the root element
    - <https://nekocalc.com/px-to-rem-converter>
    - Parent size: 20px
    - Rem size: 2rem = 20\*2
* Font colors: <https://developer.mozilla.org/en-US/docs/Web/CSS/named-color>
* How the css cascade settles conflict
  + Origin & importance
    - User agent style sheet: css a browser applies by default
    - User style sheet: styles set by a user
    - Author style sheet: css the author writes (most importance)
  + Selector specificity
    - How the style was written, over riding less specific styles by the more specific styles
      * Id selectors are more specific and will override a type and class selctors
  + Source Order
    - If there is one or more selector, the one that was written last will be applied
* Inherited properties in CSS means that an HTML element’s style values are copied from its parent (or containing) element.
  + <https://developer.mozilla.org/en-US/docs/Web/CSS/inheritance>
* CSS Box Model
  + Content
    - Padding: inside the box, space between content and border
    - Border
      * Short hand- width, style, color
    - Margin: outside the box, space around the border
  + Short hand
    - Tag 2: top bottom;
    - Tag 3: top left right bottom;
    - Tag 4: top right bottom left;
* CSS Layout
  + Display:none;
    - Hides something from the website so user can no longer see it
  + Display:block;
    - Occupy 100% of the parent container by default and are placed underneath one other
  + Display:inline;
    - Will be on the same line as other content, can’t change width or height using css
  + Display:inline-block;
    - Elements set to this layout will be on the same line but you can set the width and height using css
* Position property
  + Static: positioned according to the normal flow of the document
  + Relative: positioned according to the normal flow of the document, THEN offset relative to itself based on the values of top, right, bottom and left. Therefore, the space given for the element is the same position as if it were given static positioning until changed by the top, right, bottom and left values.
  + Absolute: removes an element from the normal document flow and allows you to place an element at the edge of the view port. The element will be positioned relative to its closet positioned ancestor.
  + Fixed: element is removed from the normal doc flow and no space is created for the element and the element scrolls with the screen and never moves.
  + Sticky: element is positioned according to the normal flow of the doc and then offset to its nearest scrolling ancestor and containing block, based on the top, right, bottom and left values. The offset does not affect the position of any other elements.
  + Float: places an element on the left or right side of its container, allowing text and inline elements to wrap around it.
* Media Queries
  + @media type{css styles}
  + @media screen and (min-width:px){}
    - <https://css-tricks.com/a-complete-guide-to-css-media-queries/>
  + allow you to apply CSS styles depending on a device's general type.
  + In the head of an html file, add a <meta> tag that lets the website know that it has been modified for mobile usage so it doesn’t automatically zoom out.
    - <meta charset=”UTF-8”>
    - <meta name =”viewport” content=”width=device-width, initial-scale=1”>
  + Oreientation: landscape or portrait
  + Breakpoints is where the current layout must adjust to properly display content
    - Threshold where the layout should change
* Typography
  + <https://developer.mozilla.org/en-US/docs/Web/CSS/font-family>
  + <https://www.cssfontstack.com/>
  + <https://fonts.google.com/>
  + Importing fonts

@font-face {

  font-family: "Inter";

  font-weight: normal;

  src: url("Inter-Regular.woff2") format("woff2");

}

* + Line height and letter-spacing
    - <https://css-tricks.com/almanac/properties/f/font/>
    - h2 {

font: 400 4rem/1.1 'Abolition Regular', Impact, Charcoal, sans-serif;

}

* + - * font: font weight size font/line height ‘type face’
    - letter-spacing: 1px;
  + text shadow
    - text-shadow: 1px 1px 1px #000;
      * horizontal, vertical, blur, and color
        + positive numbers move the shadow to the right, negative to the left
* box-shadow
  + <https://css-tricks.com/almanac/properties/b/box-shadow/>
  + box-shadow: 0 3px 10px rgb(0 0 0 / 0.2);
    - box-shadow: [horizontal] [vertical] [blur] [optional spread radius] [color];
* border-radius
  + <https://css-tricks.com/almanac/properties/b/border-radius/>
  + border-radius: 12px 12px 0 0;
    - top left, top right, bottom right, bottom left
* background images
  + background-image: url(“”);
    - <https://developer.mozilla.org/en-US/docs/Web/CSS/background-image>
  + Background-repeat
    - <https://developer.mozilla.org/en-US/docs/Web/CSS/background-repeat>
  + Background-position
    - <https://developer.mozilla.org/en-US/docs/Web/CSS/background-position>
  + Background-size
    - <https://developer.mozilla.org/en-US/docs/Web/CSS/background-size>
  + Background-blend-mode
    - <https://css-tricks.com/almanac/properties/b/background-blend-mode/>
  + Background shorthand:
    - <https://css-tricks.com/almanac/properties/b/background/>

body {

background:

url(sweettexture.jpg) */\* image \*/*

top center / 200px 200px */\* position / size \*/*

no-repeat */\* repeat \*/*

fixed */\* attachment \*/*

padding-box */\* origin \*/*

content-box */\* clip \*/*

red; */\* color \*/*

}

* Gradient transitions
  + <https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Images/Using_CSS_gradients>
  + background-image:linear-gradient(color1, color2);
* Adding a wrapper centers the body of a website, it can be common to have more than one wrapper
  + Width: 70%;
  + Margin: 0 auto;
* Vertical margins only collapse when there is no border, padding or content area to interrupt two touching margins
* A CSS reset removes all vrowswer inconsistencies to ensure that your layout displays as consistent as possible across all browsers.
* Responsive web design:
  + collection of techniques for building websites that work on multiple screen sizes. The scope of responsive design sometimes also includes websites that work at different pixel densities, such as traditional screens or high DPI screens.
  + Responsive design consists of fluid grids, fluid images and media queries
  + Maintaining multiple versions of the same website is not a sustainable strategy
  + Relative units: percentages or ems

**FLEX BOX**

[**https://css-tricks.com/snippets/css/a-guide-to-flexbox/**](https://css-tricks.com/snippets/css/a-guide-to-flexbox/)

* Flex containers
  + Sets the content for flexbox layout
    - display:flex;
    - display:inline-flex;
  + Contains the flex items, the actual elements you layout using flexbox
  + Can be any block level or inline element
* Flex items
  + Every direct child of a flex container
  + Unlimited amount in a container
* Flexbox axes
  + Main axis
    - Runs horizontally
  + Cross axis
    - Runs perpendicular to the main axis

Applies to only the flex container

* Flex-direction
  + flex-direction:row; (default behavior)
  + flex-direction:row-reverse;
  + flex-direction:column;
  + flex-direction:column-reverse;
* flex-wrap
  + flex-wrap:nowrap;
  + flex-wrap:wrap;
  + flex-wrap:wrap-reverse;
* justify-content (aligns items on the main axis)
  + - <https://developer.mozilla.org/en-US/docs/Web/CSS/justify-content>
  + A margin set to auto will absorb any extra space around a flex item and push other flex items into different positions.
* A margin value of auto affects flex item alignment because it absorbs any extra space around a flex item and pushes other flex items into different positions.
  + justify-content:flex-end;
  + justify-content:flex-start;(default behavior)
  + justify-content:space-between;
  + justify-content:center;
  + justify-content:space-around;
  + justify-content:space-evenly;
* align-items (aligns items on the cross axis)
  + align-items:center;
  + align-items: flex-start;
  + align-items:stretch;
  + align-items:flex-end;

Applies to only the Flex items

* order
  + The order [CSS](https://developer.mozilla.org/en-US/docs/Web/CSS) property sets the order to lay out an item in a flex or grid container. Items in a container are sorted by ascending order value and then by their source code order.
  + The default order of all flex items is 0
  + To place an item in front of another item, it should have a lower order value
  + To place a flex item after another item, it should have a higher order value
* Flex-grow
  + Specified how much of the flex containers remaining space should be assigined to the flex items main size
  + Default flex grow value is 0
  + The higher the value given to an item the more an item grows relative to the other items
* Flex
  + Shorthand property
    - Flex: flex-grow value, flex-basis value, flex-shrink value
  + Sets how an item will grow or shrink to fit the space available
  + Flex-basis
    - Sets the initial size of an item and then sets the size of the content box unless otherwise set with box-sizing
  + Flex-shrink
    - sets the flex shrink factor of a flex item
* Align-self
  + align-self: flex-start;
  + align-self: stretch;
  + align-self:center;
  + align-self: flex-end;

Graphical user interface, text, website

Description automatically generated

**HTML Forms**

* <form action=”url” method=”post”></form>
  + - Cannot nest a form inside the form attribute
  + Action
  + Method
    - Post : Data from the body of the form is sent to the server (most commonly used)
    - Get : data is sent inside the url and parameters are separated by a question mark
* <input type=”text” id=”name” name=”user\_name”>
  + <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/input>
  + Type: what kind of input
  + ID : not required but its helpful for targeting elements with html/css. Its also for specific form controls
  + Name is needed so when the data is submitted to server side code the server can understand the data and process the data correctly
* <textarea id=”bio” name=”user\_bio”></textarea>
  + <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/textarea>
  + Accepts multiple line input
* <button type=”submit”></button>
  + <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/button>
* Radio buttons

Text

Description automatically generated with medium confidence

Text

Description automatically generated

* Checkboxes



Text

Description automatically generated

* <label for=”name”></label>
  + For : matches with the ID that has the same value
* <fieldset></fieldset>
  + Used to group several controls including labels within a form
  + <legend></legend>
    - Labels a fieldset
* Select Menus
  + <select id=”” name=””></select>
    - Drop down menu
    - <option value=””></option>
      * Choices inside a menu
  + <optgroup label=””></optgroup>
    - Wraps elements to create logical groups

**ACCESSIBILITY**

Graphical user interface, text, application, email

Description automatically generated

Text

Description automatically generated with medium confidence

* Web Content Accessibility Guidelines
  + - <https://www.w3.org/WAI/standards-guidelines/wcag/>
    - <https://webaim.org/resources/contrastchecker/>
    - Checklist: <https://treehouse-project-downloads.s3.amazonaws.com/Accessibility-for-Web-Developers/Accessibility-Checklist-for-Web-Developers.pdf>
  + Perceivable
    - guideline 1.1
      * Text alternatives
    - guideline 1.2
      * Captions and transcripts
    - guideline 1.3 adaptable
      * Present content using semantic HTML
      * Portrait and landscape orientation
    - guideline 1.4 distinguishable
      * Sufficient color contrast
      * Combine multiple visual cues, not just color
      * Avoid auto starting audio or video
      * Responsive body copy
      * Legible at 200% zoom
  + Operable
    - Guideline 2.1 keyboard accessible
      * Keyboard navigation
    - Guideline 1.4.13 content on hover or focus (AA)
      * Obvious focus indicators
    - Guideline 2.2 enough time
      * Allow enough time
    - Guideline 2.3.3 animation from interactions (AAA)
      * Disable distracting content
    - Guideline 2.3 seizures and Physical Reactions
      * Careful with flashing animations
    - Guideline 2.4 Navigable
      * Make link purpose clear
      * Semantic heading structure
      * descriptive page <title>
    - Guideline 2.5 Input Modalities
      * Provide alternatives to complex interactions
      * Generous target sizes
  + Understandable
    - Guideline 3 Readable
      * Set HTML language attribute
      * Use plain language
        + Strong simple words
        + Short paragraphs
        + Clearly organized info
        + A conversational tone
    - Guideline 3.2 Predictable
      * Create predictable navigation
      * No unexpected changes in content
    - Guideline 3.3 Input Assistance
      * Make errors easy to spot and correct
        + Make messages easy to notice and understand
        + Highlight errors visually
        + Keep instructions simple
  + Robust
    - Write valid HTML
    - Use semantic markup
    - Changes in content status
* Principles of an accessible form
  + - Avoid unnecessary questions
    - Minimize questions per screen
    - Provide content clues
    - Group related questions
    - Use clear input labels
      * Don’t use placeholders to label input boxes
    - Make focus indicators obvious
    - Help users spot and correct errors
  + When chunking a form
    - Remind user of form purpose
    - Clearly label the current step
    - How many tasks have been completed?
    - How many steps remain?

**CSS SELECTORS**

* Attribute selectors
  + <https://developer.mozilla.org/en-US/docs/Web/CSS/Attribute_selectors>
  + can target an element by any of its HTML attributes
  + we can define specific child and sibling selectors with combinators
  + One colon (:) is a pseudo classes and (::) is a pseudo element
  + attr() function inserts an element attribute value as page content
    - <https://developer.mozilla.org/en-US/docs/Web/CSS/attr>

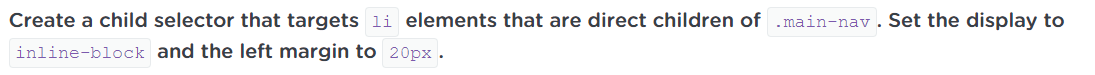
 A screenshot of a computer program

Description automatically generated with low confidence

A picture containing text, font, screenshot, algebra

Description automatically generated

* + - (>) <https://developer.mozilla.org/en-US/docs/Web/CSS/Child_combinator>
    - (+) <https://developer.mozilla.org/en-US/docs/Web/CSS/Adjacent_sibling_combinator>
    - (~) <https://developer.mozilla.org/en-US/docs/Web/CSS/General_sibling_combinator>

 A screen shot of a computer code

Description automatically generated with low confidence

A screenshot of a computer

Description automatically generated with medium confidence

* + - <https://developer.mozilla.org/en-US/docs/Web/CSS/:first-child>
    - <https://developer.mozilla.org/en-US/docs/Web/CSS/:last-child>

A screenshot of a computer

Description automatically generated with low confidence

* + - <https://developer.mozilla.org/en-US/docs/Web/CSS/:only-child>
    - <https://developer.mozilla.org/en-US/docs/Web/CSS/:empty>

A picture containing text, font, algebra

Description automatically generated

A screen shot of a computer code

Description automatically generated with low confidence

A picture containing text, font, screenshot, line

Description automatically generated

A screenshot of a computer program

Description automatically generated with low confidence

* + - <https://developer.mozilla.org/en-US/docs/Web/CSS/:disabled>
    - <https://developer.mozilla.org/en-US/docs/Web/CSS/:checked>
    - :nth-child()
      * <https://developer.mozilla.org/en-US/docs/Web/CSS/:nth-child>
      * Targets a combination of child elements
      * Even or odd as parameters or can use item number

A picture containing text, font, white, design

Description automatically generatedA screenshot of a computer

Description automatically generated with low confidence

* B integer will be the first one selected then it will select every A value after that

A picture containing text, font

Description automatically generated

* :nth-of-type()
  + <https://developer.mozilla.org/en-US/docs/Web/CSS/:nth-of-type>
  + targets an element based on its position within a parent, but only if it’s a specific type of element

A picture containing text, font, white, handwriting

Description automatically generated

* :root
  + <https://developer.mozilla.org/en-US/docs/Web/CSS/:root>
  + Selects the highest level or top-most parent element in a document
  + Has more specificity than the HTML type selector
* :target
  + <https://developer.mozilla.org/en-US/docs/Web/CSS/:target>
  + Selects an element when it’s the target of a link

A screen shot of a computer code

Description automatically generated with low confidence

* :not() -- called a negation pseudo-class
  + <https://developer.mozilla.org/en-US/docs/Web/CSS/:not>
  + It selects everything except the element we specify in the selector
* ::first-line
  + <https://developer.mozilla.org/en-US/docs/Web/CSS/::first-line>
  + Targets the first line of text, no matter how big or small the web page is

A picture containing text, font, white

Description automatically generated

* :first-letter
  + <https://developer.mozilla.org/en-US/docs/Web/CSS/::first-letter>
  + Targets the first character in a line of text

A screen shot of a computer code

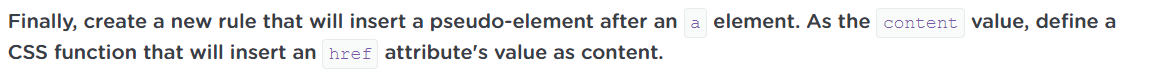
Description automatically generated with low confidence

* ::before & ::after are visible to the user and are style able with CSS but they do not appear in the HTML source code
  + These are pseudo-elements

A picture containing text, font, screenshot

Description automatically generated

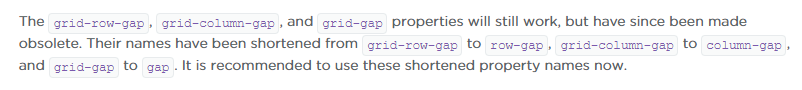
* ::before
  + <https://developer.mozilla.org/en-US/docs/Web/CSS/::before>
  + Inserts virtual element before an elements content
* ::after
  + <https://developer.mozilla.org/en-US/docs/Web/CSS/::after>
  + Inserts virtual element after an elements content
* Content
  + <https://developer.mozilla.org/en-US/docs/Web/CSS/content>

 A picture containing font, text, screenshot, graphics

Description automatically generated

**CSS Grid Layout -** <https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Grid_Layout>

* Two dimensional layout
* Grid Containers - <https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Grid_Layout/Basic_Concepts_of_Grid_Layout#grid_container>
  + Establishes the grid formatting context
  + Manages how its direct child elements are spaced, sized and aligned
* Grid Items
  + The direct children of a grid container
  + There can be any number of grid items inside a grid container
* Grid Lines - <https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Grid_Layout/Basic_Concepts_of_Grid_Layout#grid_lines>
  + Horizontal grid lines position the rows
  + Vertical grid lines position the columns
  + Grid line exists on either side of a column or row
    - Grid Tracks: The space between two adjacent grid lines forms either a column or a row
      * grid-template-columns : <https://developer.mozilla.org/en-US/docs/Web/CSS/grid-template-columns>
      * grid-template-rows : <https://developer.mozilla.org/en-US/docs/Web/CSS/grid-template-rows>
    - Track List: 
    - Gap: <https://developer.mozilla.org/en-US/docs/Web/CSSgap>



A screen shot of a computer code

Description automatically generated with low confidenceA picture containing text, font

Description automatically generated

* The fraction unit (fr) - <https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Grid_Layout/Basic_Concepts_of_Grid_Layout#the_fr_unit>
  + Flexible unit for responsive design that expands and contracts based of the free space in the grid
* Repeat() - <https://developer.mozilla.org/en-US/docs/Web/CSS/repeat>
  + Shortcut for repeating patterns of tracks

A computer code on a white background

Description automatically generated with low confidence

* Minmax() - <https://developer.mozilla.org/en-US/docs/Web/CSS/minmax>
  + Lets you set a grid tracks min and max size

A picture containing text, font, screenshot

Description automatically generated

* Auto-fill & Auto-fit
  + Have similar behavior, they both generate as many columns as will fit the available space without causing the grid to overflow
  + The biggest difference between the two is how they handle the empty repeated tracks
    - Auto-fill generates as many columns as necessary to fill the empty space even if it means generating empty tracks
    - Auto-fit expands the existing tracks to cover all of the available space without generating empty tracks

A computer code on a white background

Description automatically generated with low confidence

* Explicit and Implicit Grid - <https://css-tricks.com/difference-explicit-implicit-grids/>
  + When you create a grid layout, you are defining whats called an explicit grid
  + An implicit grid is automatically generated by the grid container to position any extra items outside the explicit grid
* Grid-auto rows & grid-auto-columns
  + <https://developer.mozilla.org/en-US/docs/Web/CSS/grid-auto-rows>
  + <https://developer.mozilla.org/en-US/docs/Web/CSS/grid-auto-columns>

A screen shot of a computer code

Description automatically generated with low confidence

A picture containing text, font, white, algebra

Description automatically generated

<https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Grid_Layout/Basic_Concepts_of_Grid_Layout#positioning_items_against_lines>

A screen shot of a computer code

Description automatically generated with low confidence

* Negative Grid Lines & shorthand syntax

A picture containing line, parallel, rectangle, diagram

Description automatically generated

A picture containing text, font, line, diagram

Description automatically generated

* Grid Template Areas - <https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Grid_Layout/Grid_Template_Areas>

A screen shot of a computer code

Description automatically generated with low confidence

* + If you want to have an empty space, just use a period to demonstrate the empty area
* Flexbox and Grid are complementary layout tools
  + <https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_grid_layout/Relationship_of_grid_layout_with_other_layout_methods#grid_and_flexbox>

**HTML TABLES -** <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/table>

A picture containing text, screenshot, font, algebra

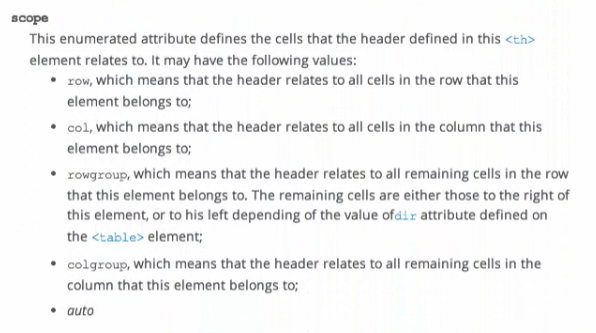
Description automatically generated

A picture containing text, white, algebra, design

Description automatically generated A picture containing text, font, white, screenshot

Description automatically generated





A picture containing text, font, algebra

Description automatically generated

A picture containing text, font, screenshot, algebra

Description automatically generated

A picture containing text, font, algebra

Description automatically generated



* Table footer goes in between the table head and the table body tags

A white background with black text

Description automatically generated with low confidence

A white background with black text

Description automatically generated with low confidence



* Caption element should come immediately after the opening table tag



**HTML VIDEO & AUDIO**





* <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/video>
* <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/source>
  + Need to provide different sources in case a browser is not compatible with only the one source



A picture containing text, font

Description automatically generated



* <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/audio>

A picture containing text, font

Description automatically generated



* Programs that can help format media for the web
  + VLC Media Player: <https://www.videolan.org/vlc/index.html>
  + Adobe Media Encoder: <https://www.adobe.com/products/media-encoder.html>
* Captioning/ Subtitles on Videos:
  + Web VTT Validator - <https://w3c.github.io/webvtt.js/parser.html>

A screenshot of a computer code

Description automatically generated with low confidence 