**COFFEE MENU**

* You can add style to an element by specifying it in the style element and setting a property for it like this:

element {

property: value;

}

Center your h1 element by setting its text-align property to the value center.

* You can add the same group of styles to many elements by creating a list of selectors. Each selector is separated with commas like this:

selector1, selector2 {

property: value;

}

* For the styling of the page to look similar on mobile as it does on a desktop or laptop, you need to add a meta element with a special content attribute.Add the following within the head element:

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

* body tag ----- body{

background-colour:brown;

} ----------- keeps background as brown.

* The div element is used mainly for design layout purposes unlike the other content elements you have used so far.
* the div not take up the entire width of the page. The CSS width property is perfect for this. pixel. div {

width:300px; OR width:80%;

}

* we can change the bg colur from the body selector to a customize form by adding the elements in the div element after adding the elements we need to change .
* If you want to center the div horizontally. You can do this by setting its margin-left and margin-right properties to auto. Think of the margin as invisible space around an element. Using these two margin properties, center the div element within the body element.
* A class selector is defined by a name with a dot directly in front of it, like this:

.class-name {

styles

}

* To apply the class's styling to the div element, add a class attribute to the div element's opening tag and set its value to menu.
* background image: background-image:url(https://cdn.freecodecamp.org/curriculum/css-cafe/beans.jpg);
* elements commonly contain multiple elements that have related information.

<article>

h2>Google Chrome</h2>

<p>Google Chrome is a web browser developed by Google, released in . Chrome is the world's most popular web browser today!</p>

</article>

* <p class="clss name">French Vanilla</p>
* p elements are block-level elements, so they take up the entire width of their parent element.
* To get them on the same line, you need to apply some styling to the p elements, so they behave more like inline elements.
* The p elements are nested in an article element with the class attribute of item. You can style all the p elements nested anywhere in elements with a class named item like this:

.item p { }

Using the above selector, add a display property with value inline-block so the p elements behave more like inline elements

* You can give your menu some space between the content and the sides with various padding properties.

Give the menu class a padding-left and a padding-right with the same value 20px.

padding-left: 20px;

padding-right: 20px;

padding-top: 20px;

padding-bottom: 20px;

* Since all 4 sides of the menu have the same internal spacing, go ahead and delete the four properties and use a single padding property with the value 20px.

padding: 20px;

* Add a max-width property to the menu class with a value of 500px to prevent it from growing too wide.
* You can add a fallback value for the font-family by adding another font name separated by a comma. Fallbacks are used in instances where the initial is not found/available.
* Below are some commonly used font fallbacks, organized by the 5 generic font families:

Serif

Sans-serif

Monospace

Cursive

Fantasy

* h1, h2 {

font-family: Impact, serif;

}

* You can use an hr element to display a divider between sections of different content.hr elements are self closing. The <hr> tag defines a thematic break in an HTML page (e.g. a shift of topic).The <hr> element is most often displayed as a horizontal rule that is used to separate content (or define a change) in an HTML page.
* The default properties of an hr element will make it appear as a thin light grey line. You can change the height of the line by specifying a value for the height property.

<hr style="width:50%;text-align:left;margin-left:0">

* Make all the edges of the hr element the same color as the background of it using the border-color property in the hr selector. after this our height increases .so change the size.
* The default color of a link that has not yet been clicked on is typically blue. The default color of a link that has already been visited from a page is typically purple.

To make the footer links the same color regardless if a link has been visited, use a type selector for the anchor element (a) and use the value black for the color property. change properties of a link when the link has actually been visited by using a pseudo- selector that looks like a:visited { propertyName: propertyValue; }.

* You change properties of a link when the mouse hovers over them by using a pseudo-selector that looks like a:hover { propertyName: propertyValue; }
* You change properties of a link when the link is actually being clicked by using a pseudo-selector that looks like a:active { propertyName: propertyValue; }.
* The image you added is not centered horizontally like the Coffee heading above it. img elements are "like" inline elements.

To make the image behave like heading elements (which are block- level), create an img type selector and use the value block for the display property and use the applicable margin-left and margin- right values to center it horizontally.

img{

display:block;

margin-left:auto;

margin-right:auto; }

* The h2 elements have default top and bottom margin space, so you could change the bottom margin of the h2 elements to say 0 or another number.
* There is an easier way, simply add a negative top margin to the img elements to pull them up from their current positions. Negative values are created using a - in front of the value.
* img {

display: block;

margin-left: auto;

margin-right: auto;

margin-top:-25px;

}