Html

Anchor tag ---

<a href = “link” target = “…”> text</a>

* The href is an anchor tag attribute that contains the URL address of the link.
* target is an anchor tag attribute that specifies where to open the link. The value \_blank specifies to open the link in a new tab.
* Replace the href attribute value with a #, also known as a hash symbol, to create a dead link.

For example: href="#"

<a href="https://www.freecodecamp.org" target="\_blank"> ... </a>

Input tag

<input type="text" placeholder="this is placeholder text">

* Placeholder text is what is displayed in your input element before your user has inputted anything.

Radio button/checkbox

* All related radio buttons should have the same name attribute to create a radio button group.
* for attribute on the label element should match the value of the id attribute of the input element.
* Why? --- > create a linked relationship between the label and the related input element.
* Inputs of type radio and checkbox report their values from the value attribute. When the user submits the form with the indoor option selected, the form data will include the line: indoor-outdoor=indoor. This is from the name and value attributes of the "indoor" input.

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

* If you omit the value attribute, the submitted form data uses the default value, which is on. In this scenario, if the user clicked the "indoor" option and submitted the form, the resulting form data would be indoor-outdoor=on, which is not useful.
* “checked” keyword -🡪 makes the radio and checkboxes checked by default.

Declare doctype of html

* The <!DOCTYPE ...> tag on the first line, where the ... part is the version of HTML. For HTML5, you use <!DOCTYPE html>.

css

* An id also has a higher specificity (importance) than a class so if both are applied to the same element and have conflicting styles, the styles of the id will be applied.
* padding: 10px 20px 10px 20px;

These four values work like a clock: top, right, bottom, left.

* [attr=value] attribute selector

[type='radio'] {

margin: 20px 0px 20px 0px;}

browsers read CSS from top to bottom in order of their declaration.

the keyword !Important --- overrides all other styles.

Css - variables

To create a CSS variable, you just need to give it a name with two hyphens in front of it

--css\_variable: value;

After you create your variable, you can assign its value to other CSS properties by referencing the name you gave it.

background: var(--penguin-skin);

When using your variable as a CSS property value, you can attach a fallback value that your browser will revert to if the given variable is invalid.

background: var(--penguin-skin, black);

When you create a variable, it is available for you to use inside the selector in which you create it. It also is available in any of that selector's descendants.

This happens because CSS variables are inherited, just like ordinary properties.

:root is a *pseudo-class* selector that matches the root element of the document

By creating your variables in :root, they will be available globally and can be accessed from any other selector in the style sheet.

When you create your variables in :root they will set the value of that variable for the whole page.

You can then over-write these variables by setting them again within a specific element.

Applied Visual Design

**Width: width of an element.**

**Height: height of an element.**

**Text-align: center, right, left, justify.**

* **Strong tag:**

 With the strong tag, the browser applies the CSS of font-weight: bold.

* To underline text, you can use the u tag.With the u tag, the browser applies the CSS of text-decoration: underline; to the element.
* em tag -- displays text as italicized, as the browser applies the CSS of font-style: italic.
* the s tag --- the browser applies the CSS of text-decoration: line-through; to the element.
* The box-shadow property applies one or more shadows to an element.

box-shadow: 0 10px 20px rgba(0,0,0,0.19)

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.

Multiple box-shadows can be created by using commas to separate properties of each box-shadow element.

The blur-radius and spread-radius values are optional.

* opacity property in CSS is used to adjust the opacity, or conversely, the transparency for an item.

A value of 1 is opaque, which isn't transparent at all.  
A value of 0.5 is half see-through.  
A value of 0 is completely transparent.

* The font-size property in CSS is not limited to headings, it can be applied to any element containing text.