Good font== Arial or Helvetica

To answer the question "What is clean design?" most succinctly: a clean design is one that supports visual thinking so people can meet their informational needs with a minimum of conscious effort.

CSS Selectors [CSS reference - CSS: Cascading Style Sheets | MDN (mozilla.org)](https://developer.mozilla.org/en-US/docs/Web/CSS/Reference)

<https://developer.mozilla.org/en-US/docs/Web/CSS>

Note: margin-color is not a color property, box-width is not box property

Typography: The style and appearance of text on the page is called **Typography.**

You can use CSS to control these properties. The properties are:

* font-size
* line-height
* max-width (for controlling the length of each line)
* color and background-color

<http://practicaltypography.com/>

Key rules for typography:

* Point size should be 15-25 pixels
* Line spacing should be 120–145% of the point size
* Line length should be 45–90 characters (including spaces)
* Less color is more effective

Text anf font styling link on MDN—

<https://developer.mozilla.org/en-US/docs/Learn/CSS/Styling_text/Fundamentals>

using diff font: in html: <link href="https://fonts.googleapis.com/css2?family=Merriweather:ital,wght@0,400;0,700;1,400&display=swap" rel="stylesheet">

in css: body {

font-family: 'Merriweather', serif;

}

Google Fonts: <https://fonts.google.com/>

MDN web fonts --- <https://developer.mozilla.org/en-US/docs/Learn/CSS/Styling_text/Web_fonts>

* Using font-face CSS property to declare your own fonts
* Details about how font loading works
* Different kinds of font files

LINKS

* Links are a good time to use an accent color to style text.
* You can change the text-decoration - you can remove the underline, or show it only on hover.
* You can style other properties, like background-color and border-bottom.
* Link states: hover, visited, active

Links can show differently when they are hovered over, or after they’ve been clicked. To apply styles to particular states, you need a new kind of selector: the :hover and :visited pseudo-selectors.

It’s also nice to be able to make the cursor show differently when hovering something clickable. By default, clickable links have cursor: pointer to make the pointer into a little hand icon. You can apply this yourself to other elements that are meant to be clicked.

Styling links MDN --- <https://developer.mozilla.org/en-US/docs/Learn/CSS/Styling_text/Styling_links>

Styling lists MDN --- <https://developer.mozilla.org/en-US/docs/Learn/CSS/Styling_text/Styling_lists>

COLORS

NB: to change colors of icons, use **fill: red;**

Color: rgba(255, 255, 255, 0.5) ---- where a is the opacity(highest being 1 ie 0 – 1)

Hex(#FF0000) or usx name(red)

What colors do good sites have?

When designers create a color palette, they usually pick:

* **Theme** color (or several related **theme colors**) for background colors for the header and footer.
* **Accent** color, for links and highlights.
* **Background** color (usually an off-white, like #eeeeee).
* **Body font color** (usually a dark gray, like #222222).
* Your primary goal should be to create *contrast* between the background color and the text color. Dark text on a dark background is hard to read. So is light text on a light background.
* Usually, the best choice is black or dark gray text, on a white or near-white background.

MDN color: <https://developer.mozilla.org/en-US/docs/Web/CSS/color_value>

Color palette: <https://coolors.co/generate>

**CSS variables**. Here’s how they work.

html {

--theme: #D231A0;

}

a:hover {

color: var(--theme);

text-decoration-color: var(--theme);

}

Trending color palettes: <https://coolors.co/palettes/trending>

BOX

Box-sizing: border box; ensures that the height and width given to that object encompases both the border and padding. Note that the margin is not part of this arrangement.

Border: width style color. Eg 4px dashed red

Margin or padding: top right bottom left;

MDN Box ---- <https://developer.mozilla.org/en-US/docs/Learn/CSS/Building_blocks/The_box_model>

JAVASCRIPT

Alert – alert(“hello”) or window.alert(“hello”)

Variables: let x = 10

let name = "Wasiu"

let numbers = [1,2,3]

conditions: let x = 10

let name = "Wasiu"

let numbers = [1,2,3]

loops: for (let i = 0; i < 100; i++) {

console.log(i)

}

Functions: function greetPerson(name) {

console.log("Hello, " + name)

}

greetPerson("Tolu") // logs "Hello, Tolu"

greetPerson("Rosemary") // logs "Hello, Rosemary"

greetPerson("Oluwaseyi") // logs "Hello, Oluwaseyi"

Objects: let party = {

"name": "Birthday Party",

"date": "November 20",

"guests": ["Mom", "Dad", "Brother"]

}

console.log(party["date"]) // November 20

 The function querySelector asks the page for the first element that matches the CSS selector you give it.

You write document.querySelector to find elements inside the document. The document represents the whole page, so all of the elements should be inside it.

document.querySelector('h1.menu-title')

changx properties: document.querySelector('h1.menu-title').style = "background-color: lavender"

Gettimg text in h1: document.querySelector('h1.menu-title').innerText

Changing innertext: document.querySelector('h1.menu-title').innerText = "Intro to Gardening 🌱"

Styling using javascript, do this: document.querySelector('h1.menu-title').style = "background-color: lavender; padding: 12px; font-family: monospace"

But its good to use CSS to style then give different classes for different style types to be used in js

Eg adding the .retro class: document.querySelector('h1.menu-title').classList.add('retro')

document.querySelector('h1.menu-title').classList.remove('retro')

*Events* happen when the user does something, like click on the button.

addEventListener lets you set up code to run when an event happens. JavaScript will run the code between the curly braces {} when the button is clicked.

button.addEventListener('click', function() {

console.log("clicked");

})

Link --- <https://developer.mozilla.org/en-US/docs/Learn/JavaScript/Building_blocks/Events>

More types of events other than click --- <https://developer.mozilla.org/en-US/docs/Web/Events>