CSS FUNCTION

Functions are reusable pieces of code which perform specific tasks. Functions are passed "arguments" between parentheses, each of which is used by the function in a specific way. CSS functions become important when thinking about responsive design.

1. calc()
    To perform calculations. The most powerful use cases for calc include:

    - Mixing units
    - The ability to nest `calc( calc () - calc () )`

    Example:
    ```css
    1  :root {
    2  --header: 3rem;
    3  --footer: 40px;
    4  --main: calc(100vh - calc(var(--header) + var(--footer)));
    5  }
    ```

    In the example above, calc() is handling the math for us even though we are mixing vh, rem and px units.

2. min()
    To select the smallest value within the parentheses.
    Example:
    ```css
    1  #iconHolder {
    2     width: min(150px, 100%);
    3     height: min(150px, 100%);
    4     box-sizing: border-box;
    5     border: 6px solid blue;
    6  }
    ```

3. max()
    To select the largest value within the parentheses.
    Example:
    ```css
    1  width: max(100px, 4em, 50%);
    ```

4. clamp()
    To specify a value within a range. Clamp() takes 3 values:
    ```css
    font-size: clamp(320px, 80vw, 60rem);
    ```
    1. the smallest value (320px)
    2. the ideal value (80vw)
    3. the largest value (60rem)

    clamp() is a great way to make elements fluid and responsive.

Note: We don't need to call calc() when using calculation inside a min(), max(), or clamp().

A. Uses cases & example (https://web.dev/articles/min-max-clamp)

1. Perfect width
   - Use the clamp() function to set a minimum and maximum width.
   - Use the min() function to set a maximum width.
   - Use the max() function to set a minimum width.

   Note: from the visual example, in this case clamp() is recommended than min() and max()

2. Manage padding
   This idea let an element have additional padding at larger screen sizes, but keep a minimum padding at smaller screen sizes.

   To do this, we can yous either:

   ```
   : calc((100vw - var(--contentWidth)) / 2), or max(2rem, 50vw - var(--contentWidth) / 2).
   ```

3. Fluid/responsive typography
   Uses clamp() to set a minimum font size, maximum font size, and allow scaling between those sizes.

   Note: Limiting maximum font sizes using max() or clamp() might prevent users from scaling the text to 200% of its original size. Make sure to test the results with zoom.

B. More uses cases & example
   - https://ishadeed.com/article/css-min-max-clamp/
   - https://moderncss.dev/practical-uses-of-css-math-functions-calc-clamp-min-max/