Chapter 36: Functions

Section 36.1: calc() function

Accepts a mathematical expression and returns a numerical value.

It is especially useful when working with different types of units (e.g. subtracting a px value from a percentage) to calculate the value of an attribute.

`+`, `-`, `/`, and `*` operators can all be used, and parentheses can be added to specify the order of operations if necessary.

Use `calc()` to calculate the width of a div element:

```
#div1 {
    position: absolute;
    left: 50px;
    width: calc(100% - 100px);
    border: 1px solid black;
    background-color: yellow;
    padding: 5px;
    text-align: center;
}
```

Use `calc()` to determine the position of a background-image:

```
background-position: calc(50% + 17px) calc(50% + 10px), 50% 50%;
```

Use `calc()` to determine the height of an element:

```
height: calc(100% - 20px);
```

Section 36.2: attr() function

Returns the value of an attribute of the selected element.

Below is a blockquote element which contains a character inside a `data-*` attribute which CSS can use (e.g. inside the `::before` and `::after` pseudo-element) using this function.

```
<blockquote data-mark='"'></blockquote>
```

In the following CSS block, the character is appended before and after the text inside the element:

```
blockquote[data-mark]::before,
blockquote[data-mark]::after {
    content: attr(data-mark);
}
```

Section 36.3: var() function

The var() function allows CSS variables to be accessed.

```
/* set a variable */
:root {
```

```
    --primary-color: blue;
}

/* access variable */
selector {
    color: var(--primary-color);
}
```

This feature is currently under development. Check caniuse.com for the latest browser support.

Section 36.4: radial-gradient() function

Creates an image representing a gradient of colors radiating from the center of the gradient

```
radial-gradient(red, orange, yellow) /*A gradient coming out from the middle of the
gradient, red at the center, then orange, until it is finally yellow at the edges*/
```

Section 36.5: linear-gradient() function

Creates a image representing a linear gradient of colors.

```
linear-gradient( 0deg, red, yellow 50%, blue);
```

This creates a gradient going from bottom to top, with colors starting at red, then yellow at 50%, and finishing in blue.