CSS

CONTENTS

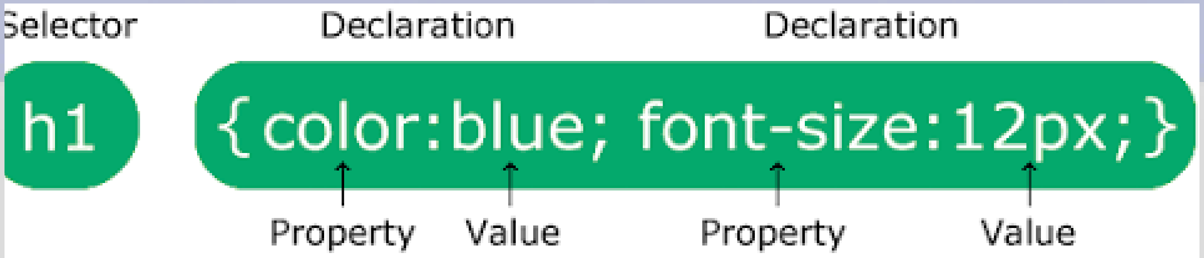
- INTRODUCTION
- CSS COMMENTS
- CSS COLORS
- CSS TEXT
- CSS ID AND CLASS
- POSITIONING
- OVERLAPPING ELEMENTS
- CSS ALIGN
- CROSS BROWSERS COMPATIBLE ISSUES

INTRODUSTION

* CSS stands for Cascading Style Sheets

* Styles define how to display HTML elements
* Styles were added to HTML 4.0 to solve a problem
* External Style Sheets can save a lot of work
* External Style Sheets are stored in CSS files

CSS Syntax



```
h1 {color:blue; font-size:12px;}
```

- The selector points to the HTML element you want to style.
- The declaration block contains one or more declarations separated by semicolons.
- Each declaration includes a CSS property name and a value, separated by a colon.
- Multiple CSS declarations are separated with semicolons, and declaration blocks are surrounded by curly braces.

There are three types of CSS which are:

- Inline CSS
- Internal or Embedded CSS
- External CSS

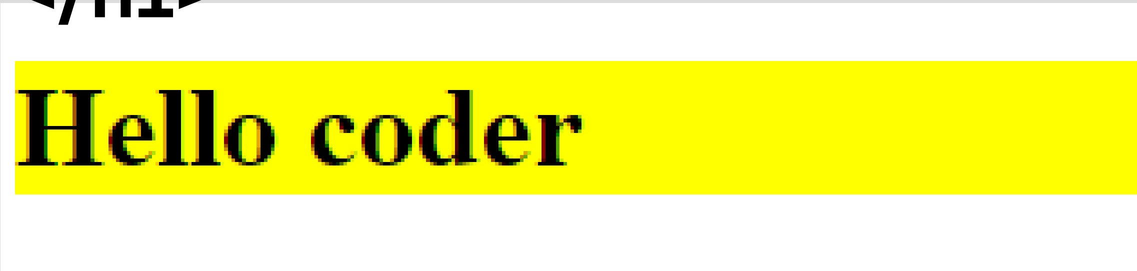
Inline CSS

- Inline CSS contains the **CSS** property in the body section attached to the element is known as inline CSS.

- This kind of style is specified within an HTML tag using the style attribute.

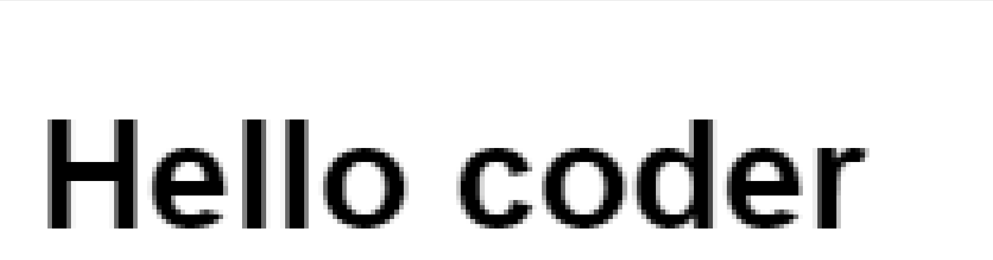
Syntax of Inline CSS

```
<h1 style="background-color:yellow">
        Hello coder
 </h1>
```



Hello coder

```
<h1 style="font-family:arial">
        Hello coder
 </h1>
```



Hello coder

Internal or Embedded CSS:

- This can be used when a single HTML document must be styled uniquely.

- The CSS rule set should be within the HTML file in the head section i.e. the CSS is embedded within the <style> tag inside the head section of the HTML file.

Syntax of Internal CSS

```
<head>
    <style>
        .main {
            text-align: center;
            }
        #first {
            color: #009900;
            font-size: 50px;
            font-weight: bold;
            }
</style></head>
```

External CSS:

- External CSS contains separate CSS files that contain only style properties with the help of tag attributes (For example class, id, heading, ... etc).

- CSS property is written in a separate file with a .css extension and should be linked to the HTML document using a link tag.

- It means that, for each element, style can be set only once and will be applied across web pages.

How to link .css file with HTML file?

```
<head>
<link rel="stylesheet" href="mystyle.css">
</head>
```

CSS Colors

Colors are displayed combining RED, GREEN, and BLUE light.

COLOR VALUES

CSS colors are defined using a hexadecimal (hex) notation for the combination of Red, Green, and Blue color values (RGB). The lowest value that can be given to one of the light sources is 0 (hex 00). The highest value is 255 (hex FF).

Hex values are written as 3 double digit numbers, starting with a # sign.

CSS TEXT

Text Color

The color property is used to set the color of the text. The color can be specified by:

* name - a color name, like "red"
* RGB - an RGB value, like "rgb(255,0,0)"
* Hex - a hex value, like "#ff0000"

The default color for a page is defined in the body selector.

Example

```
body {color:blue;}
h1 {color:#00ff00;}
h2 {color:rgb(255,0,0);}
```

Text Alignment

The text-align property is used to set the horizontal alignment of a text. Text can be centered, or aligned to the left or right, or justified.

When text-align is set to "justify", each line is stretched so that every line has equal width, and the left and right margins are straight

Example

```
h1 {text-align:center;}
p.date {text-align:right;}
p.main {text-align:justify;}
```

Text Transformation

The text-transform property is used to specify uppercase and lowercase letters in a text.

It can be used to turn everything into uppercase or lowercase letters, or capitalize the first letter of each word.

Example

```
p.uppercase {text-transform:uppercase;}
p.lowercase {text-transform:lowercase;}
p.capitalize {text-transform:capitalize;}
```

Text Indentation

The text-indentation property is used to specify the indentation of the first line of a text.

Example

```
p {text-indent:50px;}
```

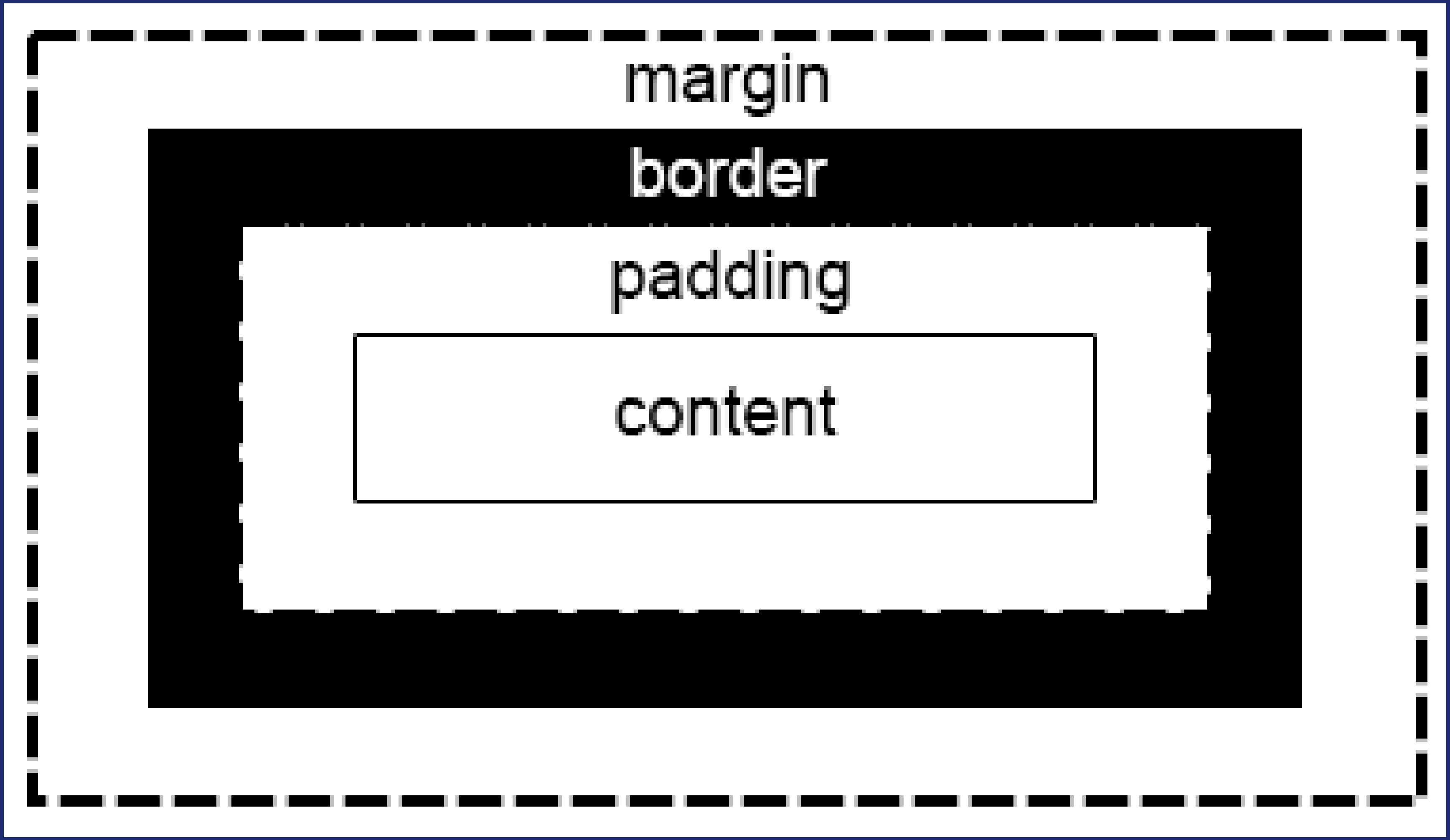
CSS Example

```
body
{
background-color:#d0e4fe;
}
h1
{
color:orange;
text-align:center;
}
p
{
font-family:"Times New Roman";
font-size:20px;
}
```

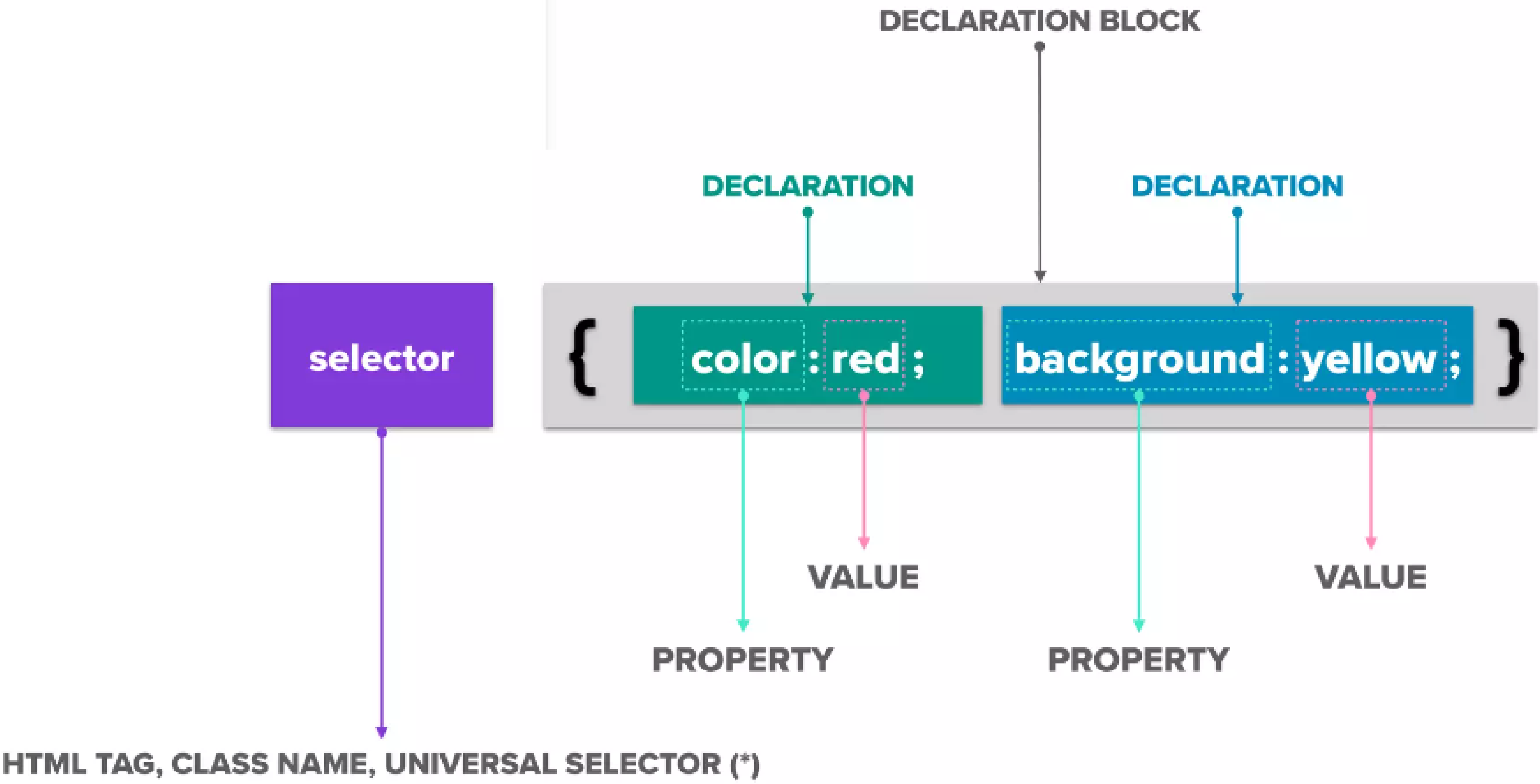
CSS Box Model

- All HTML elements can be considered as boxes. In CSS, the term "box model" is used when talking about design and layout.
- The CSS box model is essentially a box that wraps around every HTML element. It consists of: margins, borders, padding, and the actual content.

CSS Box Model



CSS Selectors



CSS Selectors

Types of Selectors

- **Universal Selector(*)**
- **Element Selector(p)**
- **Id Selector(#id)**
- **Class Selector(.class)**
- **Group Selector(h1,h2,p)**

POSITIONING

The CSS positioning properties allow you to position an element. It can also place an element behind another, and specify what should happen when an element's content is too big.

Elements can be positioned using the top, bottom, left, and right properties. However, these properties will not work unless the position property is set first. They also work differently depending on the positioning method.

There are four different positioning methods.

Static Positioning

HTML elements are positioned static by default. A static positioned element is always positioned according to the normal flow of the page.

Static positioned elements are not affected by the top, bottom, left, and right properties.

Relative Positioning

A relative positioned element is positioned relative to its normal position.

Example

```
h2.pos_left
{
position:relative;
left:-20px;
}
h2.pos_right
{
position:relative;
left:20px;
}
```

Absolute Positioning

An absolute position element is positioned relative to the first parent element that has a position other than static. If no such element is found, the containing block is <html>:

Example

```
h2
{
position:absolute;
left:100px;
top:150px;
}
```

Fixed Positioning

An element with fixed position is positioned relative to the browser window.

It will not move even if the window is scrolled:

Example

```
p.pos_fixed
{
position:fixed;
top:30px;
right:5px;
}
```

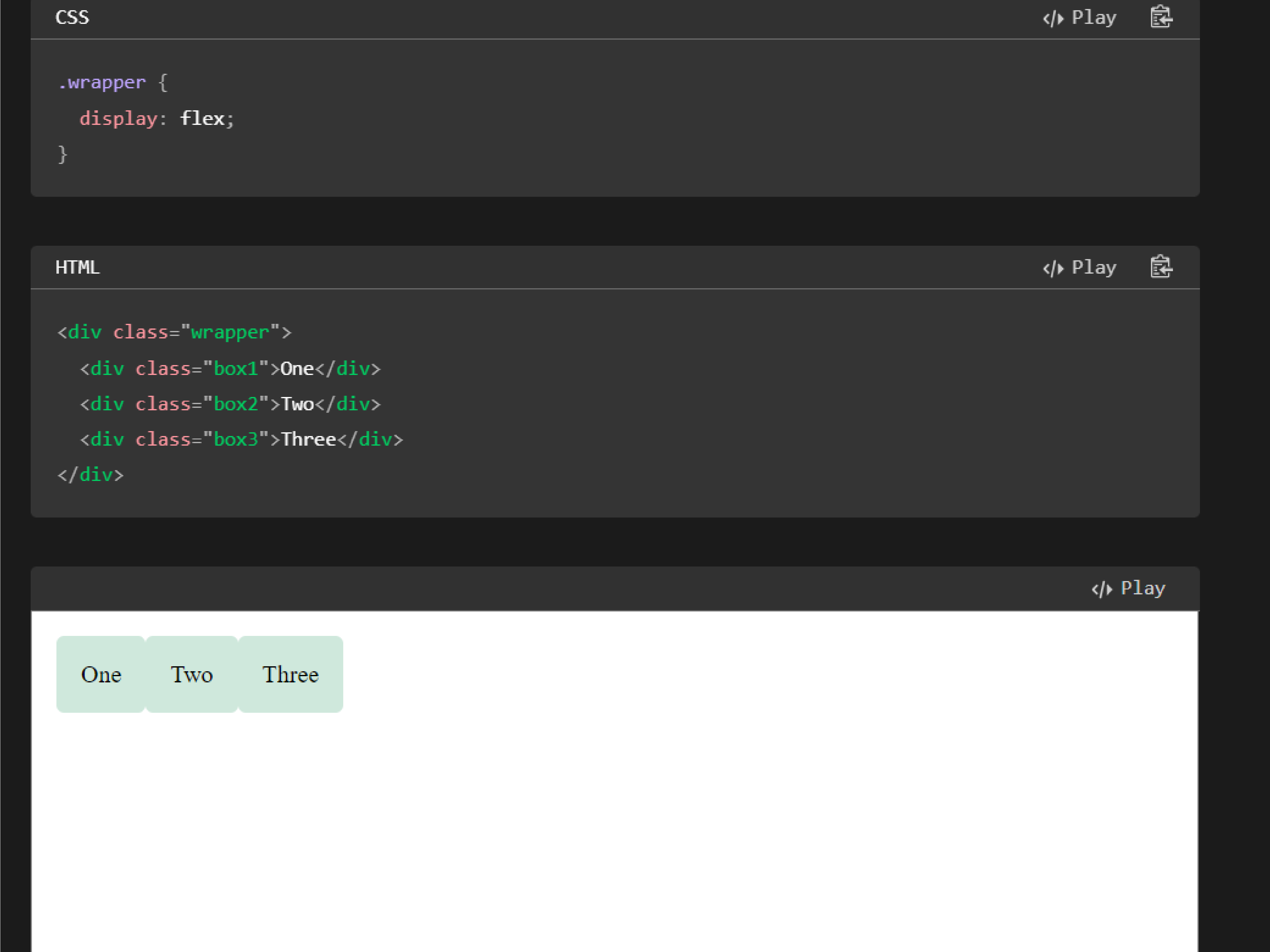
CSS LAYOUT

CSS page layout techniques allow us to take elements contained in a web page and control where they're positioned relative to the following factors: their default position in normal layout flow, the other elements around them, their parent container, and the main viewport/window. The page layout techniques we'll be covering in more detail in this module are:

- Normal flow
- The display property
- Flexbox
- Grid
- Floats
- Positioning
- Table layout
- Multiple-column layout

Flexbox

Flexbox is the short name for the Flexible Box Layout CSS module, designed to make it easy for us to lay things out in one dimension — either as a row or as a column. To use flexbox, you apply display: flex to the parent element of the elements you want to lay out; all its direct children then become flex items.



```css
.wrapper {
  display: flex;
}
```

```html
<div class="wrapper">
  <div class="box1">One</div>
  <div class="box2">Two</div>
  <div class="box3">Three</div>
</div>
```

One Two Three

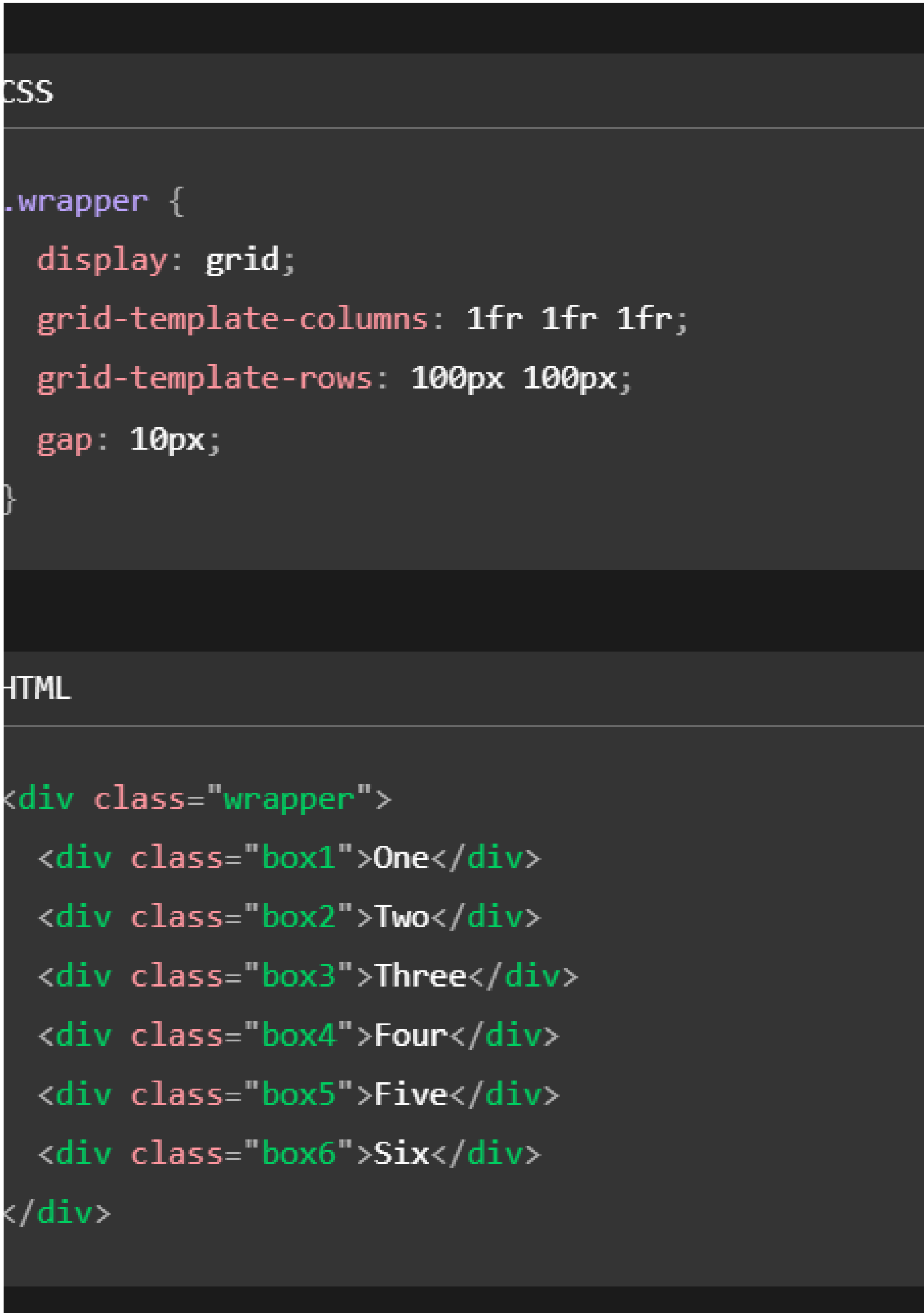
Nested Flexbox

It's possible to create some pretty complex layouts with flexbox. It's perfectly OK to set a flex item to also be a flex container, so that its children are also laid out like flexible boxes.

CSS Grid

It's possible to create some pretty complex layouts with flexbox. It's perfectly OK to set a flex item to also be a flex container, so that its children are also laid out like flexible boxes.

Similar to flexbox, we enable Grid Layout with its specific display value — display: grid. The below example uses similar markup to the flex example, with a container and some child elements. In addition to using display: grid, we also define some row and column tracks for the parent using the grid-template-rows and grid-template-columns properties respectively.



CSS

```css
.wrapper {
  display: grid;
  grid-template-columns: 1fr 1fr 1fr;
  grid-template-rows: 100px 100px;
  gap: 10px;
}
```

HTML

```html
<div class="wrapper">
  <div class="box1">One</div>
  <div class="box2">Two</div>
  <div class="box3">Three</div>
  <div class="box4">Four</div>
  <div class="box5">Five</div>
  <div class="box6">Six</div>
</div>
```

Responsive Web Design

Responsive web design is an approach to web development aimed at creating websites that adapt and respond to different screen sizes and devices.

The primary goal of responsive design is to provide an optimal and consistent user experience across a wide range of platforms, including desktop computers, laptops, tablets, and smartphones.

THANK YOU