东方瑞通 终身学习 Founded in 1998

CSS

1. What is CSS?

- **CSS** stands for **C**ascading **S**tyle **S**heets
- CSS describes **how HTML elements are to be displayed on screen, paper, or in other media**
- CSS **saves a lot of work**. It can control the layout of multiple web pages all at once
- External stylesheets are stored in **CSS files**

2. Why Use CSS?

CSS is used to define styles for your web pages, including the design, layout and variations in display for different devices and screen sizes.

Why Use CSS?

2.1. CSS Example

```
body {
 background-color: lightblue;
}

h1 {
 color: white;
 text-align: center;
}

p {
 font-family: verdana;
 font-size: 20px;
}
```

3. CSS Solved a Big Problem

HTML was NEVER intended to contain tags for formatting a web page!

HTML was created to **describe the content** of a web page, like:

`<h1>This is a heading</h1>`

`<p>This is a paragraph.</p>`

When tags like `<font>`, and color attributes were added to the HTML 3.2 specification, it started a nightmare for web developers. Development of large websites, where fonts and color information were added to every single page, became a long and expensive process.

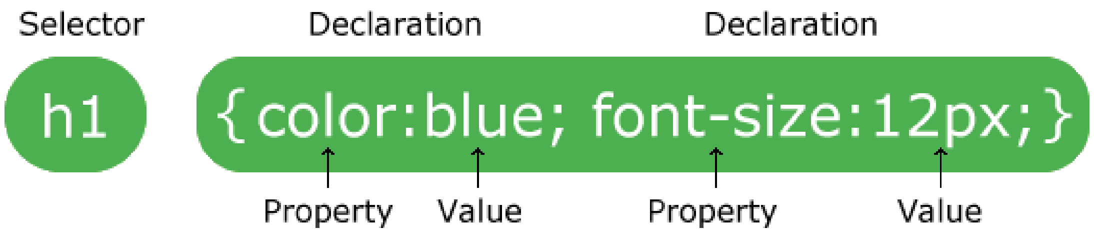
To solve this problem, the World Wide Web Consortium (W3C) created CSS.

CSS removed the style formatting from the HTML page!

> 🔥 CSS Saves a Lot of Work!
>
> The style definitions are normally saved in external .css files.
>
> With an external stylesheet file, you can change the look of an entire website by changing just one file!

4. CSS Syntax

A CSS rule-set consists of a selector and a declaration block:



- 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.

In this example all `<p>` elements will be center-aligned, with a red text color:

```
p {
 color: red;
 text-align: center;
}
```

Example Explained

- `p` is a **selector** in CSS (it points to the HTML element you want to style:

).

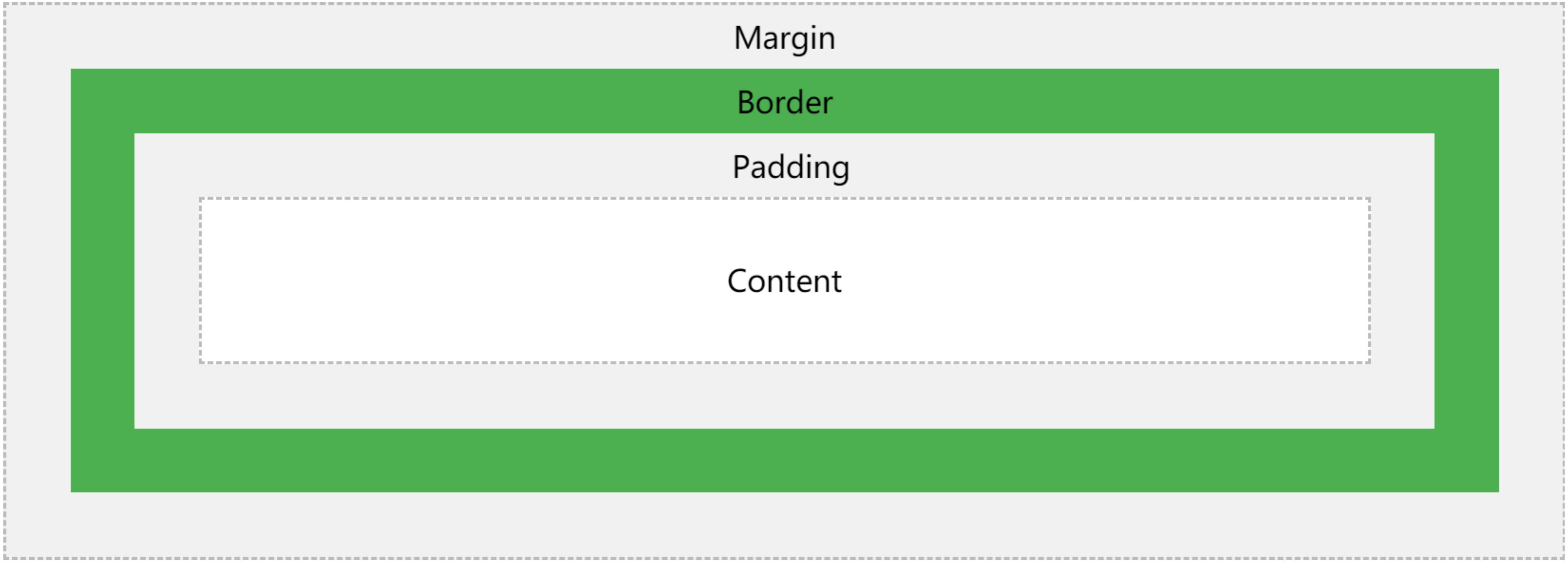
CSS Syntax

- `color` is a property, and `red` is the property value
- `text-align` is a property, and `center` is the property value

5. The 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. The image below illustrates the box model:



Explanation of the different parts:

- **Content** - The content of the box, where text and images appear
- **Padding** - Clears an area around the content. The padding is transparent
- **Border** - A border that goes around the padding and content

The CSS Box Model

- **Margin** - Clears an area outside the border. The margin is transparent

The box model allows us to add a border around elements, and to define space between elements.

```
div {
  width: 300px;
  border: 15px solid green;
  padding: 50px;
  margin: 20px;
}
```

5.1. Width and Height of an Element

In order to set the width and height of an element correctly in all browsers, you need to know how the box model works.

> **Important:** When you set the width and height properties of an element with CSS, you just set the width and height of the **content area**. To calculate the full size of an element, you must also add padding, borders and margins.

This `<div>` element will have a total width of 350px:

```
div {
  width: 320px;
  padding: 10px;
  border: 5px solid gray;
  margin: 0;
}
```

Here is the calculation:

> ⚡ 320px (width) + 20px (left + right padding) + 10px (left + right border) + 0px (left + right margin) **= 350px**

The total width of an element should be calculated like this:

`Total element width = width + left padding + right padding + left border + right border + left margin + right margin`

The total height of an element should be calculated like this:

`Total element height = height + top padding + bottom padding + top border + bottom border + top margin + bottom margin`

The CSS Box Model

6. CSS Selectors

> CSS selectors are used to “find” (or select) the HTML elements you want to style.

We can divide CSS selectors into five categories:

- Simple selectors (select elements based on name, id, class)
- Combinator selectors (select elements based on a specific relationship between them)
- Pseudo-class selectors (select elements based on a certain state)
- Pseudo-elements selectors (select and style a part of an element)
- Attribute selectors (select elements based on an attribute or attribute value)

6.1. Simple selectors

6.1.1. The CSS element Selector

The element selector selects HTML elements based on the element name.

```
p {
  text-align: center;
  color: red;
}
```

6.1.2. The CSS id Selector

The id selector uses the id attribute of an HTML element to select a specific element.

The id of an element is unique within a page, so the id selector is used to select one unique element!

To select an element with a specific id, write a hash (#) character, followed by the id of the element.

```
#para1 {
  text-align: center;
  color: red;
}
```

6.1.3. The CSS class Selector

The class selector selects HTML elements with a specific class attribute.

To select elements with a specific class, write a period (.) character, followed by the class name.

```
.center {
  text-align: center;
  color: red;
}
/* You can also specify that only specific HTML elements should be affected by a class.*/
p.center {
  text-align: center;
  color: red;
}
```

HTML elements can also refer to more than one class.

```
<p class="center large">This paragraph refers to two classes.</p>
```

6.1.4. The CSS Universal Selector

The universal selector (*) selects all HTML elements on the page.

```
* {
  text-align: center;
  color: blue;
}
```

6.1.5. The CSS Grouping Selector

The grouping selector selects all the HTML elements with the same style definitions.

CSS Selectors

```css
h1, h2, p {
  text-align: center;
  color: red;
}
```

6.2. Combinator selectors

> A combinator is something that explains the relationship between the selectors.

A CSS selector can contain more than one simple selector. Between the simple selectors, we can include a combinator.

There are four different combinators in CSS:

- descendant selector (space)
- child selector (>)
- adjacent sibling selector (+)
- general sibling selector (~)

6.2.1. Descendant Selector

The descendant selector matches all elements that are descendants of a specified element.

The following example selects all `<p>` elements inside `<div>` elements:

```
div p {
  background-color: yellow;
}
```

6.2.2. Child Selector

The child selector selects all elements that are the children of a specified element.

The following example selects all `<p>` elements that are children of a `<div>` element:

```
div > p {
  background-color: yellow;
}
```

6.2.3. Adjacent Sibling Selector

The adjacent sibling selector selects all elements that are the adjacent siblings of a specified element.

Sibling elements must have the same parent element, and “adjacent” means “immediately following”.

The following example selects all `<p>` elements that are placed immediately after `<div>` elements:

```
div + p {
  background-color: yellow;
}
```

6.2.4. General Sibling Selector

The general sibling selector selects all elements that are siblings of a specified element.

The following example selects all `<p>` elements that are siblings of `<div>` elements:

```
div ~ p {
  background-color: yellow;
}
```

6.3. CSS Pseudo-classes

A pseudo-class is used to define a special state of an element.

For example, it can be used to:

- Style an element when a user mouses over it
- Style visited and unvisited links differently
- Style an element when it gets focus

6.3.1. Anchor Pseudo-classes

Links can be displayed in different ways:

```
/* unvisited link */
a:link {
  color: #FF0000;
}

/* visited link */
a:visited {
  color: #00FF00;
}

/* mouse over link */
a:hover {
  color: #FF00FF;
}

/* selected link */
a:active {
  color: #0000FF;
}
```

6.3.2. Pseudo-classes and CSS Classes

Pseudo-classes can be combined with CSS classes:

When you hover over the link in the example, it will change color:

```
a.highlight:hover {
  color: #ff0000;
}
/* Hover on <div> */
div:hover {
  background-color: blue;
}
```

6.3.3. Simple Tooltip Hover

```
<!DOCTYPE html>
<html>
<head>
<style>
p {
  display: none;
  background-color: yellow;
  padding: 20px;
}

div:hover p {
  display: block;
}
</style>
</head>
<body>

<div>Hover over me to show the p element
  <p>Tada! Here I am!</p>
</div>

</body>
</html>
```

6.3.4. The :first-child Pseudo-class

The `:first-child` pseudo-class matches a specified element that is the first child of another element.

CSS Selectors

```
p:first-child {
  color: blue;
}
p i:first-child {
  color: blue;
}
p:first-child i {
  color: blue;
}
```

6.4. CSS Pseudo-elements

A CSS pseudo-element is used to style specified parts of an element.

For example, it can be used to:

- Style the first letter, or line, of an element
- Insert content before, or after, the content of an element

6.4.1. Style the first letter, or line, of an element

```
p::first-line {
  color: #ff0000;
  font-variant: small-caps;
}
p::first-letter {
  color: #ff0000;
  font-size: xx-large;
}
```

6.4.2. Insert content before, or after

```
h1::before {
  content: url(smiley.gif);
}
h1::after {
  content: url(smiley.gif);
}
```

6.4.3. The ::selection Pseudo-element

The ::selection pseudo-element matches the portion of an element that is selected by a user.

The following CSS properties can be applied to ::selection : color , background , cursor , and outline .

The following example makes the selected text red on a yellow background:

```
::selection {
  color: red;
  background: yellow;
}
```

6.5. CSS Attribute Selectors

t is possible to style HTML elements that have specific attributes or attribute values.

- The `[attribute="value"]` selector is used to select elements with a specified attribute and value.
- The `[attribute~="value"]` selector is used to select elements with an attribute value containing a specified word.
- The `[attribute|="value"]` selector is used to select elements with the specified attribute starting with the specified value.
- The `[attribute$="value"]` selector is used to select elements whose attribute value ends with a specified value.
- The `[attribute*="value"]` selector is used to select elements whose attribute value contains a specified value.

```
a[target="_blank"] {
  background-color: yellow;
}
[title~="flower"] {
  border: 5px solid yellow;
}
[class|="top"] {
  background: yellow;
}
[class$="test"] {
  background: yellow;
}
[class*="te"] {
  background: yellow;
}
```

CSS Selectors

```
input[type="text"] {
  width: 150px;
  display: block;
  margin-bottom: 10px;
  background-color: yellow;
}

input[type="button"] {
  width: 120px;
  margin-left: 35px;
  display: block;
}
```

Thank you

Thank you

全国统一咨询热线：400-690-6115

北京|上海|广州|深圳|天津|成都|重庆|武汉|济南|青岛|杭州|西安