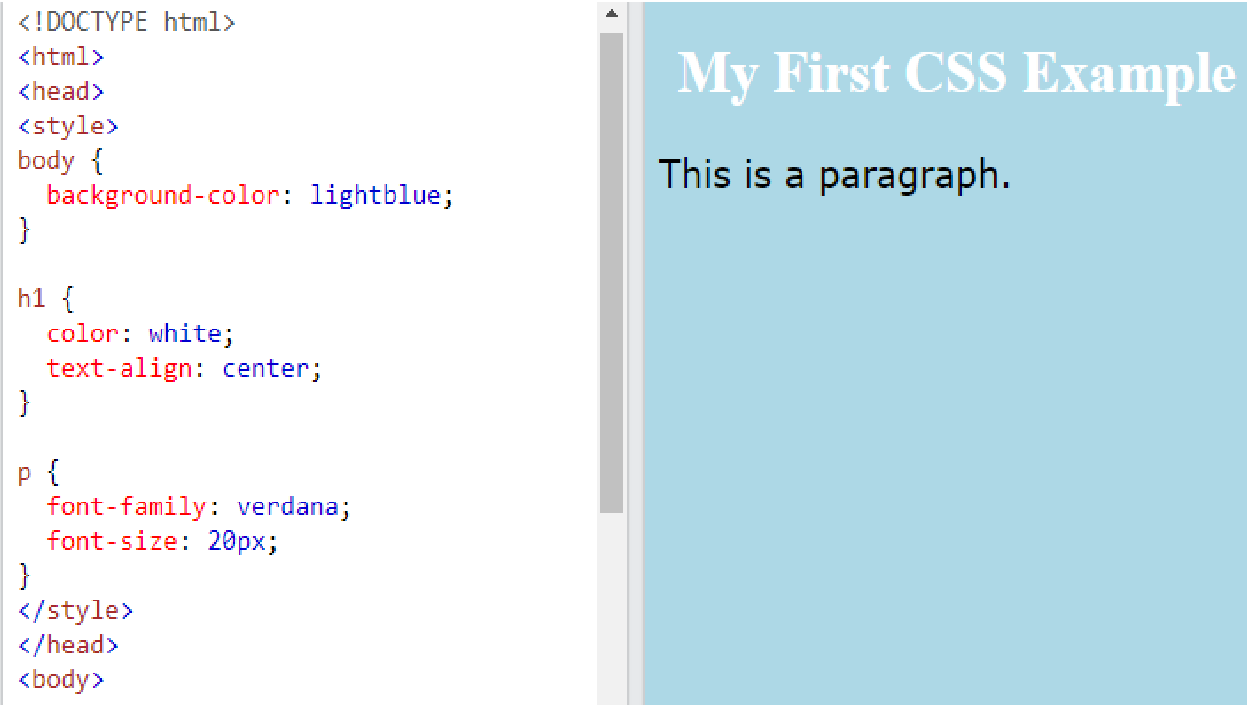
What is CSS?

CSS is the language we use to style a Web page.

- CSS stands for Cascading Style Sheets
- 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

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.



```
<!DOCTYPE html>
<html>
<head>
<style>
body {
  background-color: lightblue;
}

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

p {
  font-family: verdana;
  font-size: 20px;
}
</style>
</head>
<body>
```

My First CSS Example

This is a paragraph.

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!

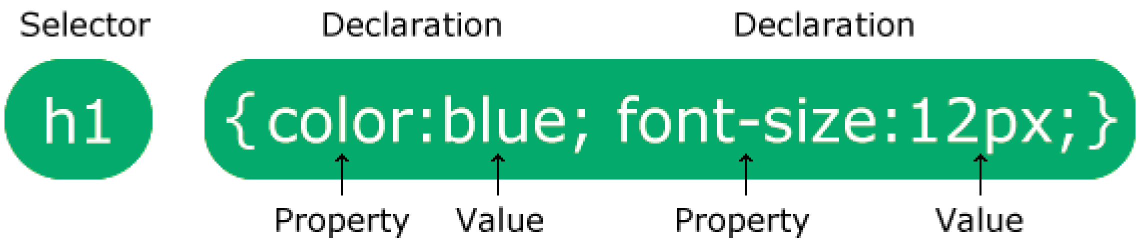
CSS Saves a Lot of Work!

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!

<u>CSS Syntax</u>

A CSS rule consists of a selector and a declaration block.

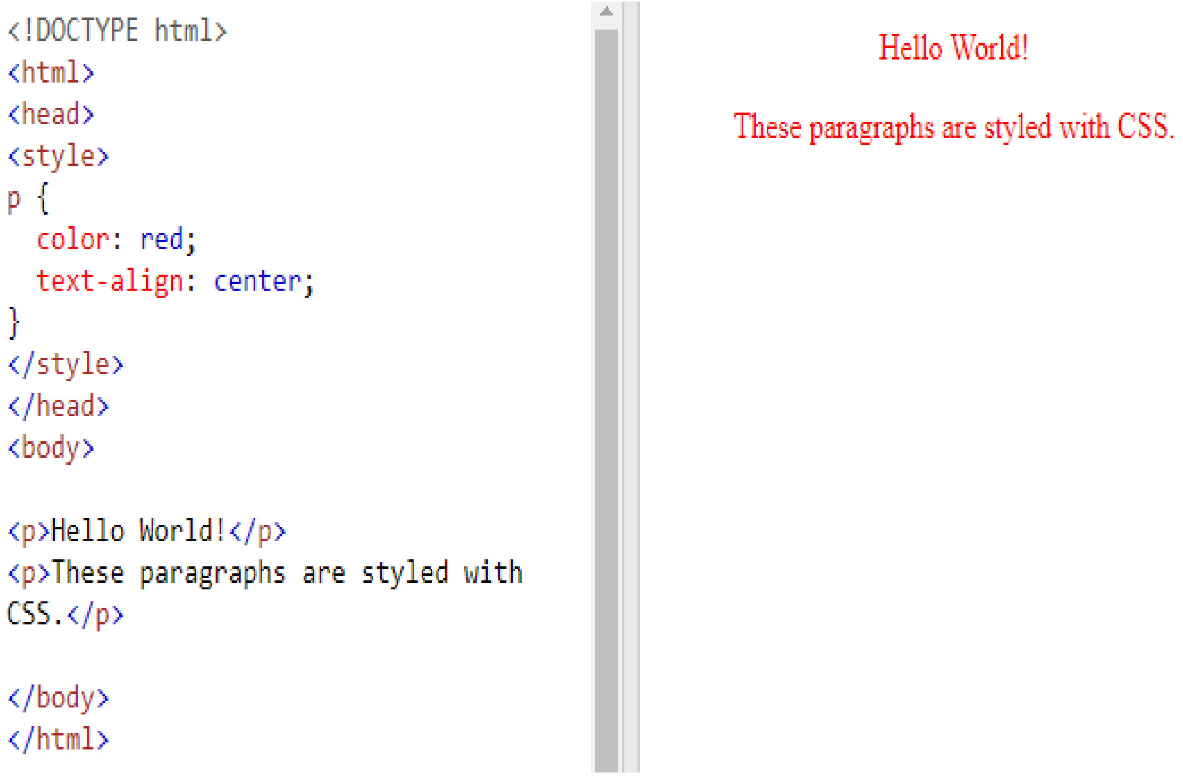


The selector points to the HTML element you want to style.

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.



```
<!DOCTYPE html>
<html>
<head>
<style>
p {
  color: red;
  text-align: center;
}
</style>
</head>
<body>

<p>Hello World!</p>
<p>These paragraphs are styled with
CSS.</p>

</body>
</html>
```

```
Hello World!

These paragraphs are styled with CSS.
```

CSS Selectors

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

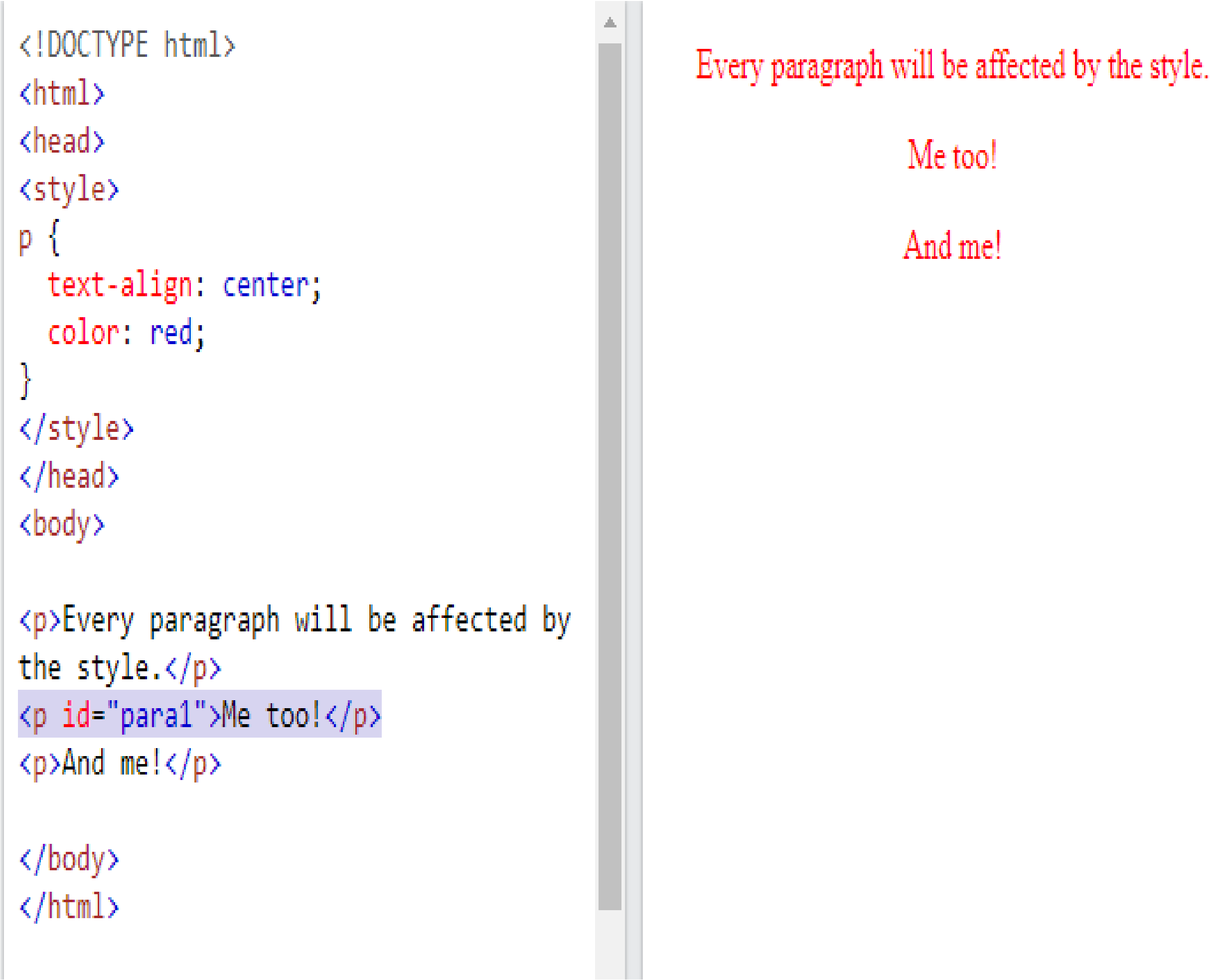
We can divide CSS selectors into five categories:

- 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

The CSS element Selector

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

Here, all <p> elements on the page will be center-aligned, with a red text color:



```
<!DOCTYPE html>
<html>
<head>
<style>
p {
  text-align: center;
  color: red;
}
</style>
</head>
<body>

<p>Every paragraph will be affected by
the style.</p>
<p id="para1">Me too!</p>
<p>And me!</p>

</body>
</html>
```

Every paragraph will be affected by the style.

Me too!

And me!

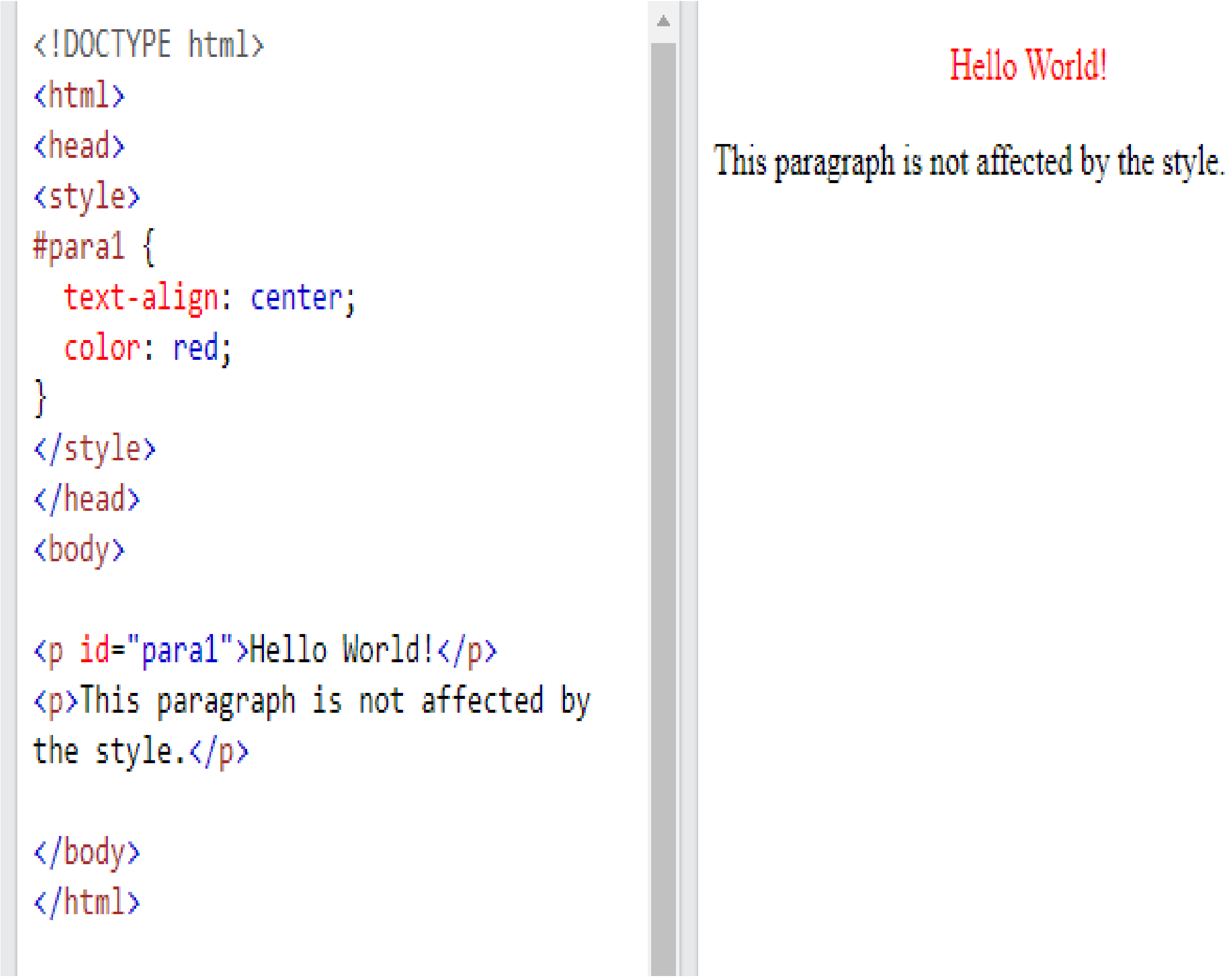
The CSS id Selector

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.

The CSS rule below will be applied to the HTML element with id="para1":



```
<!DOCTYPE html>
<html>
<head>
<style>
#para1 {
  text-align: center;
  color: red;
}
</style>
</head>
<body>

<p id="para1">Hello World!</p>
<p>This paragraph is not affected by
the style.</p>

</body>
</html>
```

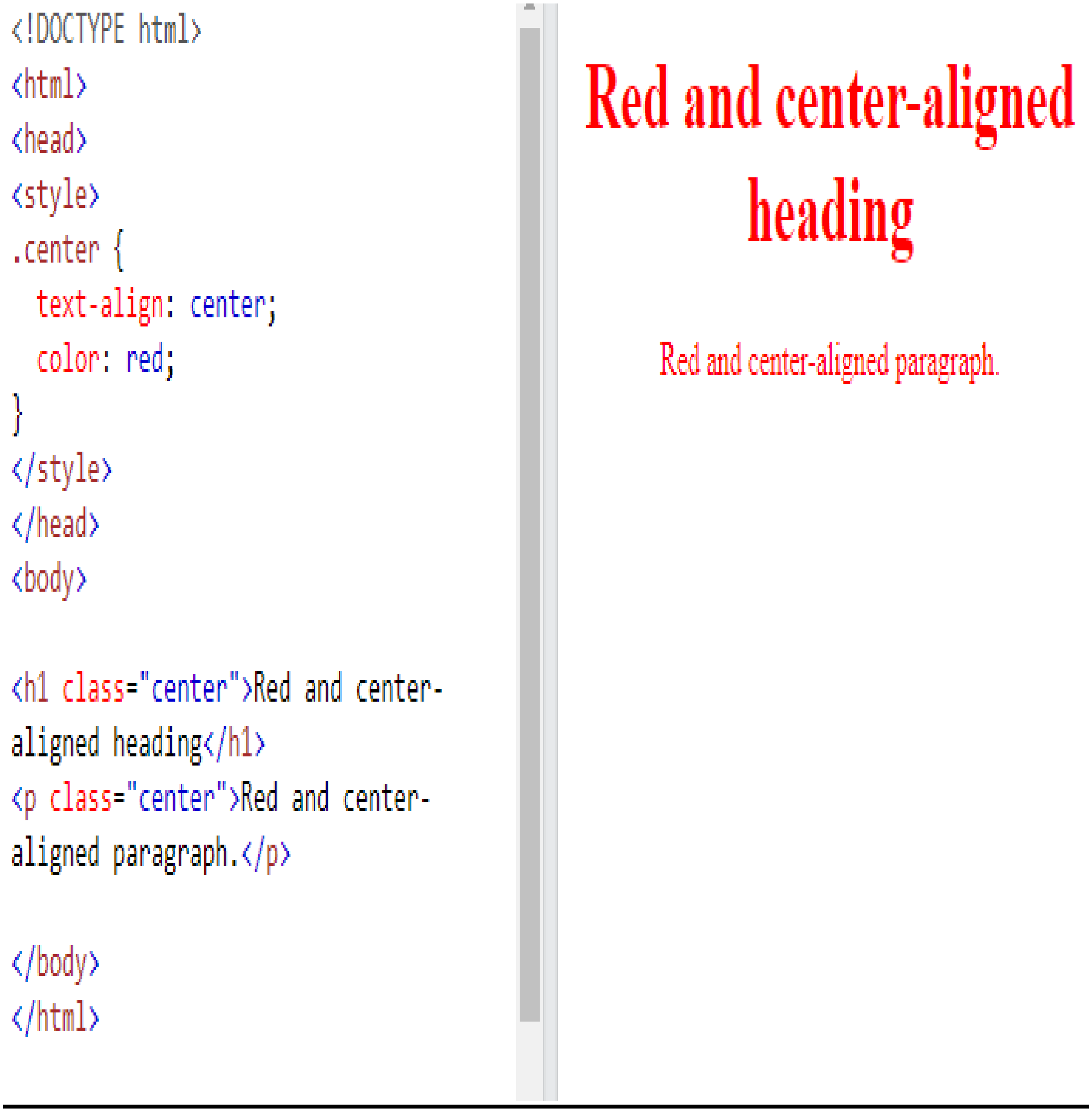
```
Hello World!

This paragraph is not affected by the style.
```

<u>The CSS class Selector</u>

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.

In this example all HTML elements with class="center" will be red and center-aligned:

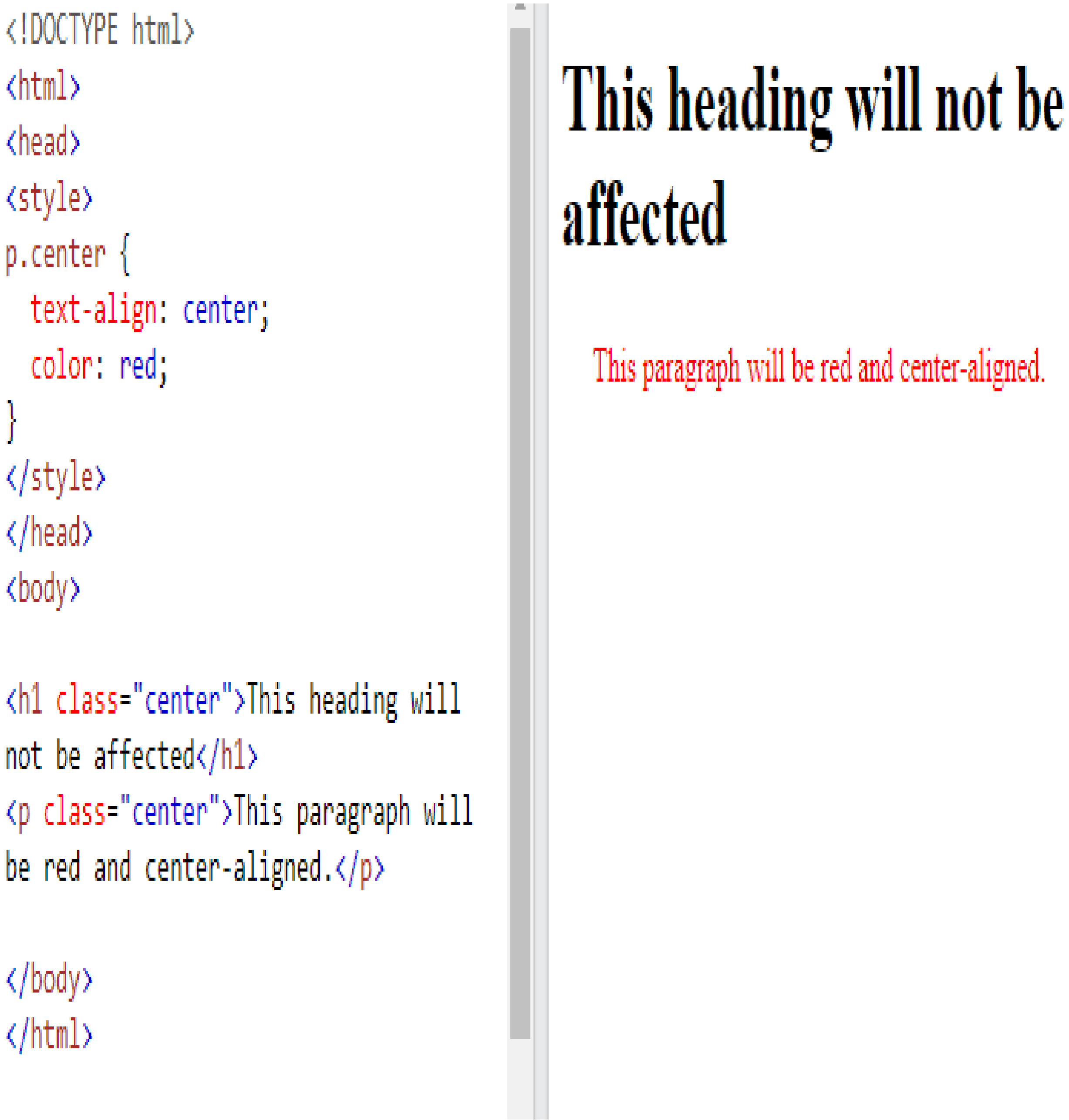


```
<!DOCTYPE html>
<html>
<head>
<style>
.center {
  text-align: center;
  color: red;
}
</style>
</head>
<body>

<h1 class="center">Red and center-
aligned heading</h1>
<p class="center">Red and center-
aligned paragraph.</p>

</body>
</html>
```

Red and center-aligned heading

Red and center-aligned paragraph.

In this example only <p> elements with class="center" will be red and center-aligned:



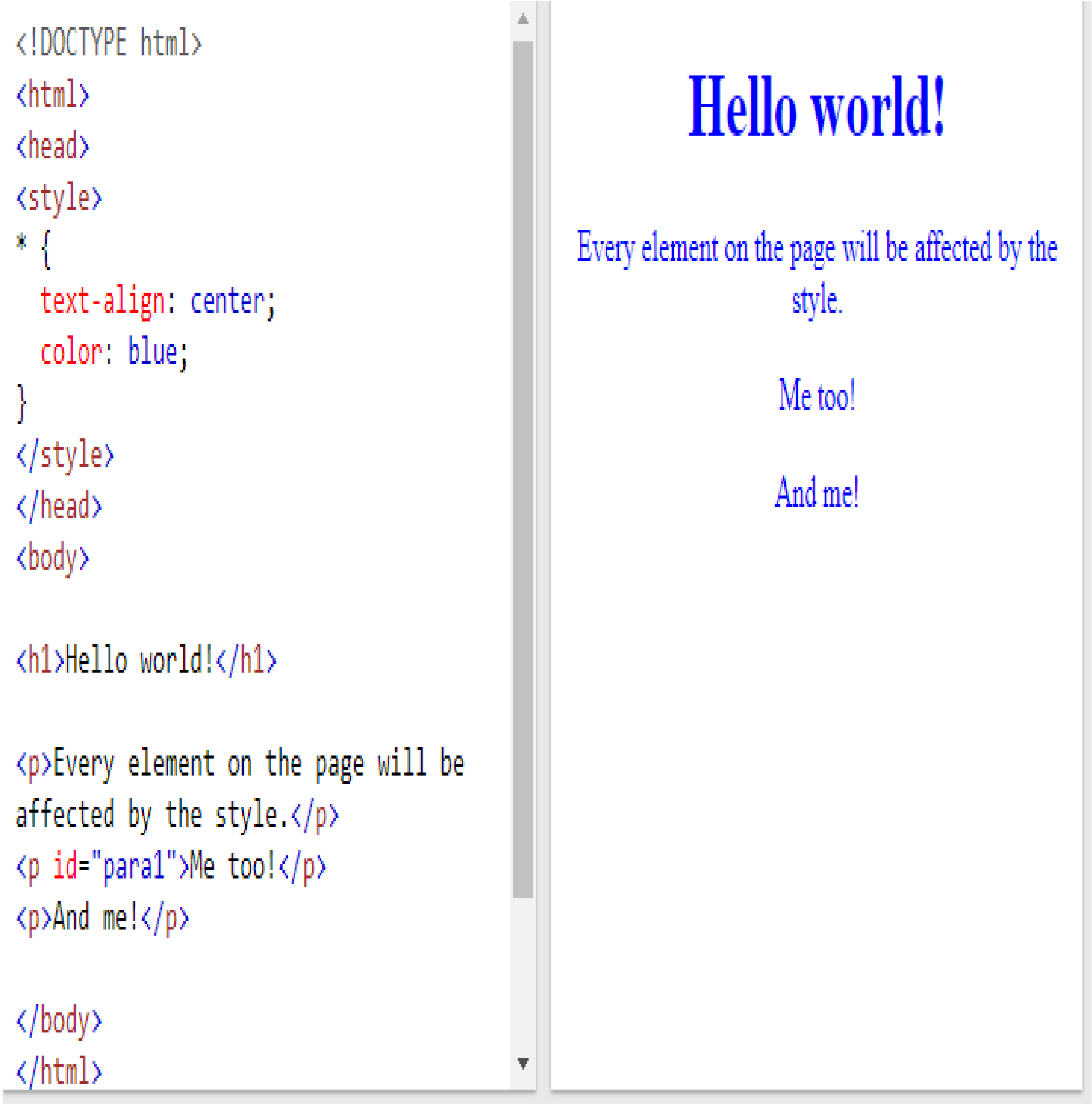
```
<!DOCTYPE html>
<html>
<head>
<style>
p.center {
  text-align: center;
  color: red;
}
</style>
</head>
<body>

<h1 class="center">This heading will
not be affected</h1>
<p class="center">This paragraph will
be red and center-aligned.</p>

</body>
</html>
```

```
This heading will not be affected

This paragraph will be red and center-aligned.
```

Note: A class name cannot start with a number!

The CSS Universal Selector

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



```
<!DOCTYPE html>
<html>
<head>
<style>
* {
  text-align: center;
  color: blue;
}
</style>
</head>
<body>

<h1>Hello world!</h1>

<p>Every element on the page will be
affected by the style.</p>
<p id="para1">Me too!</p>
<p>And me!</p>

</body>
</html>
```

Hello world!

Every element on the page will be affected by the style.

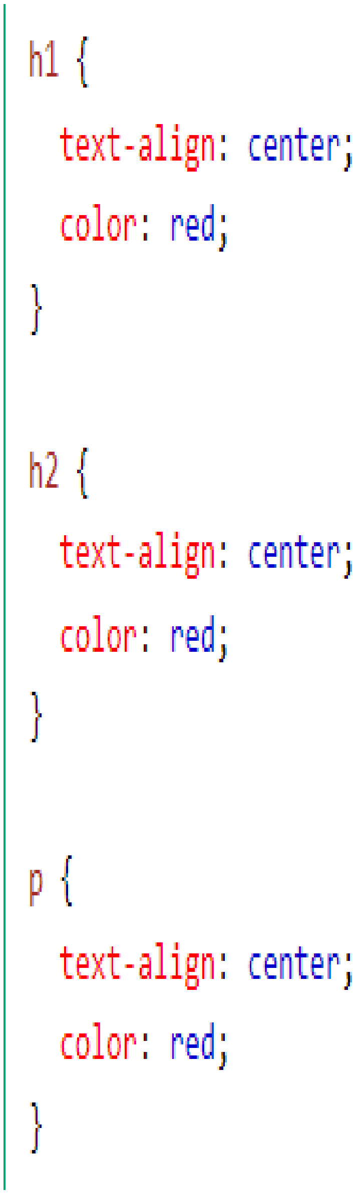
Me too!

And me!

The CSS Grouping Selector

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

Look at the following CSS code (the h1, h2, and p elements have the same style definitions):

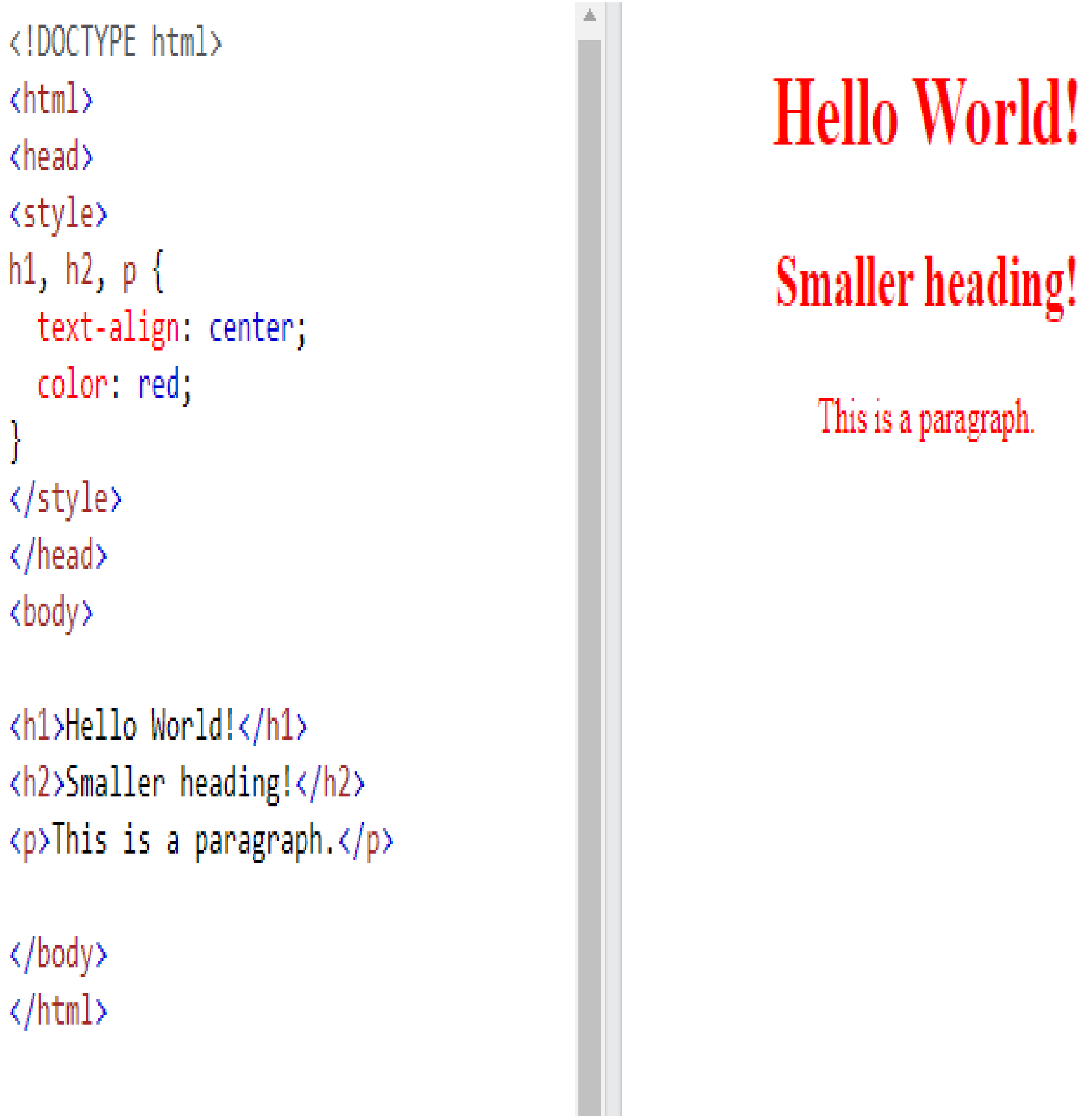


```
h1 {
  text-align: center;
  color: red;
}

h2 {
  text-align: center;
  color: red;
}

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

It will be better to group the selectors, to minimize the code.

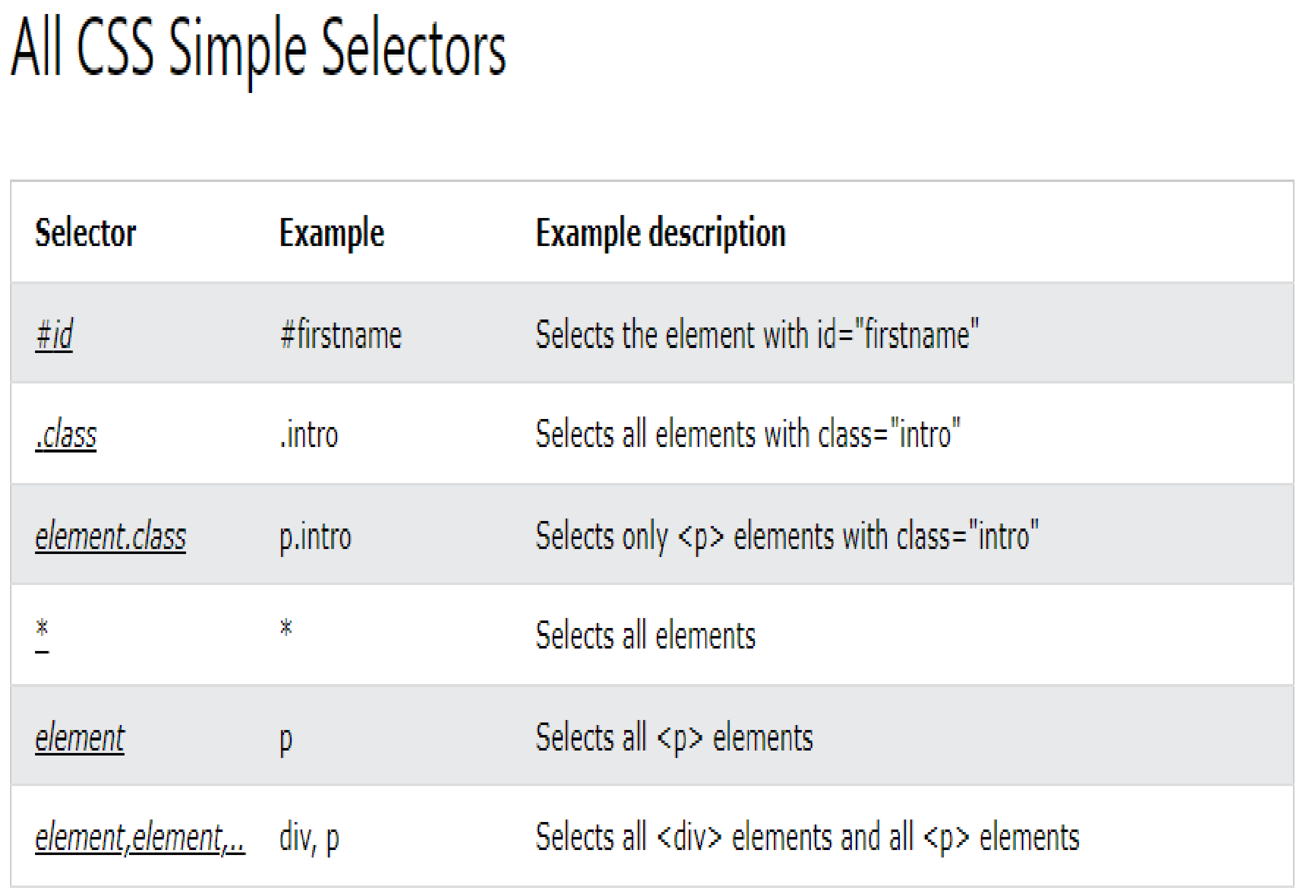
To group selectors, separate each selector with a comma.

In this example we have grouped the selectors from the code above:



```
<!DOCTYPE html>
<html>
<head>
<style>
h1, h2, p {
  text-align: center;
  color: red;
}
</style>
</head>
<body>

<h1>Hello World!</h1>
<h2>Smaller heading!</h2>
<p>This is a paragraph.</p>

</body>
</html>
```

```
Hello World!

Smaller heading!

This is a paragraph.
```



All CSS Simple Selectors

Selector	Example	Example description
#id	#firstname	Selects the element with id="firstname"
.class	.intro	Selects all elements with class="intro"
element.class	p.intro	Selects only <p> elements with class="intro"
*	*	Selects all elements
element	p	Selects all <p> elements
element,element,..	div, p	Selects all <div> elements and all <p> elements

How To Add CSS

When a browser reads a style sheet, it will format the HTML document according to the information in the style sheet.

Three Ways to Insert CSS

There are three ways of inserting a style sheet:

- External CSS
- Internal CSS
- Inline CS

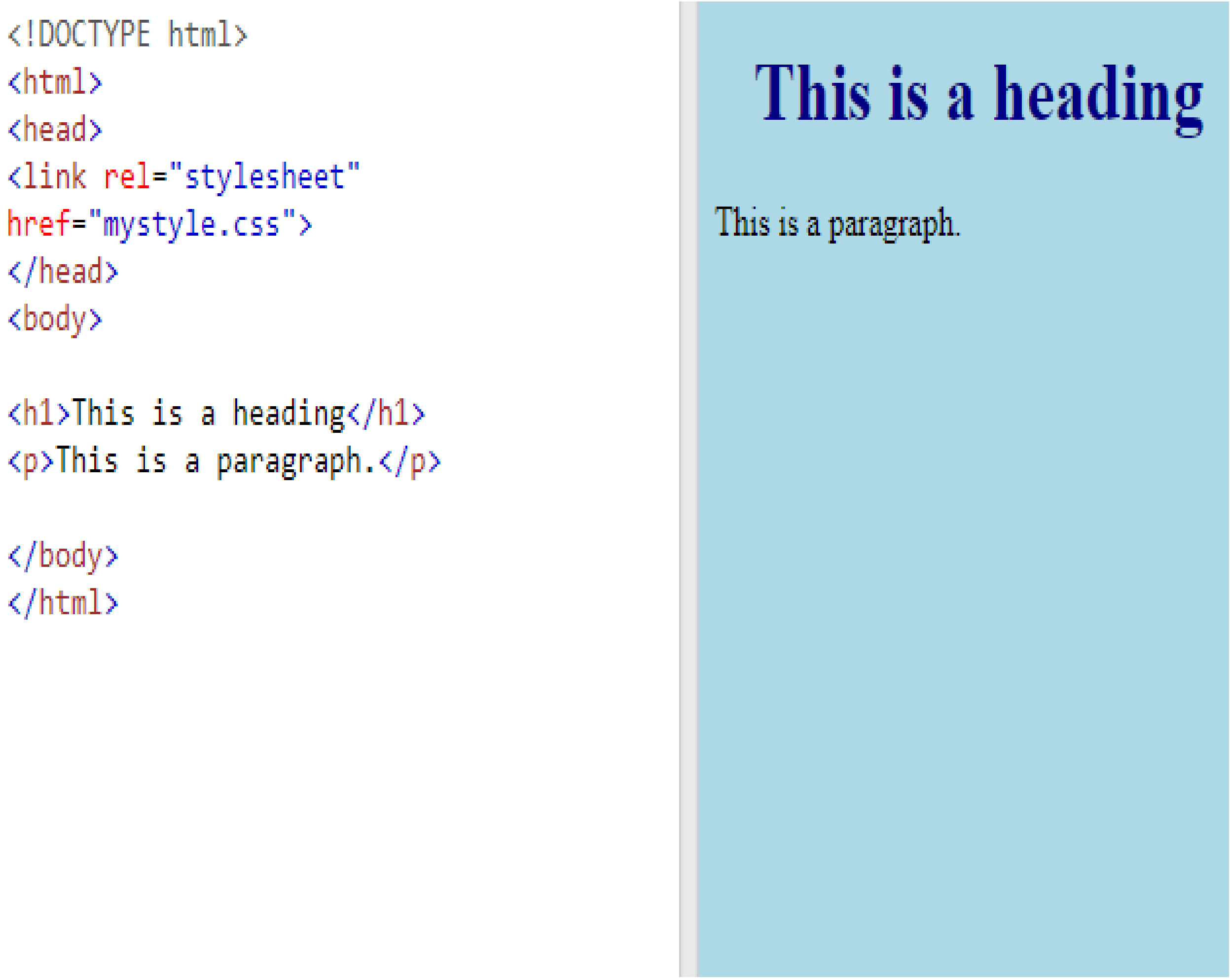
External CSS

With an external style sheet, you can change the look of an entire website by changing just one file!

Each HTML page must include a reference to the external style sheet file inside the <link> element, inside the head section.

Example

External styles are defined within the <link> element, inside the <head> section of an HTML page:



```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet"
href="mystyle.css">
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

This is a heading

This is a paragraph.

An external style sheet can be written in any text editor, and must be saved with a .css extension.

The external .css file should not contain any HTML tags.

Here is how the "mystyle.css" file looks:

"mystyle.css"

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

h1 {
  color: navy;
  margin-left: 20px;
}
```

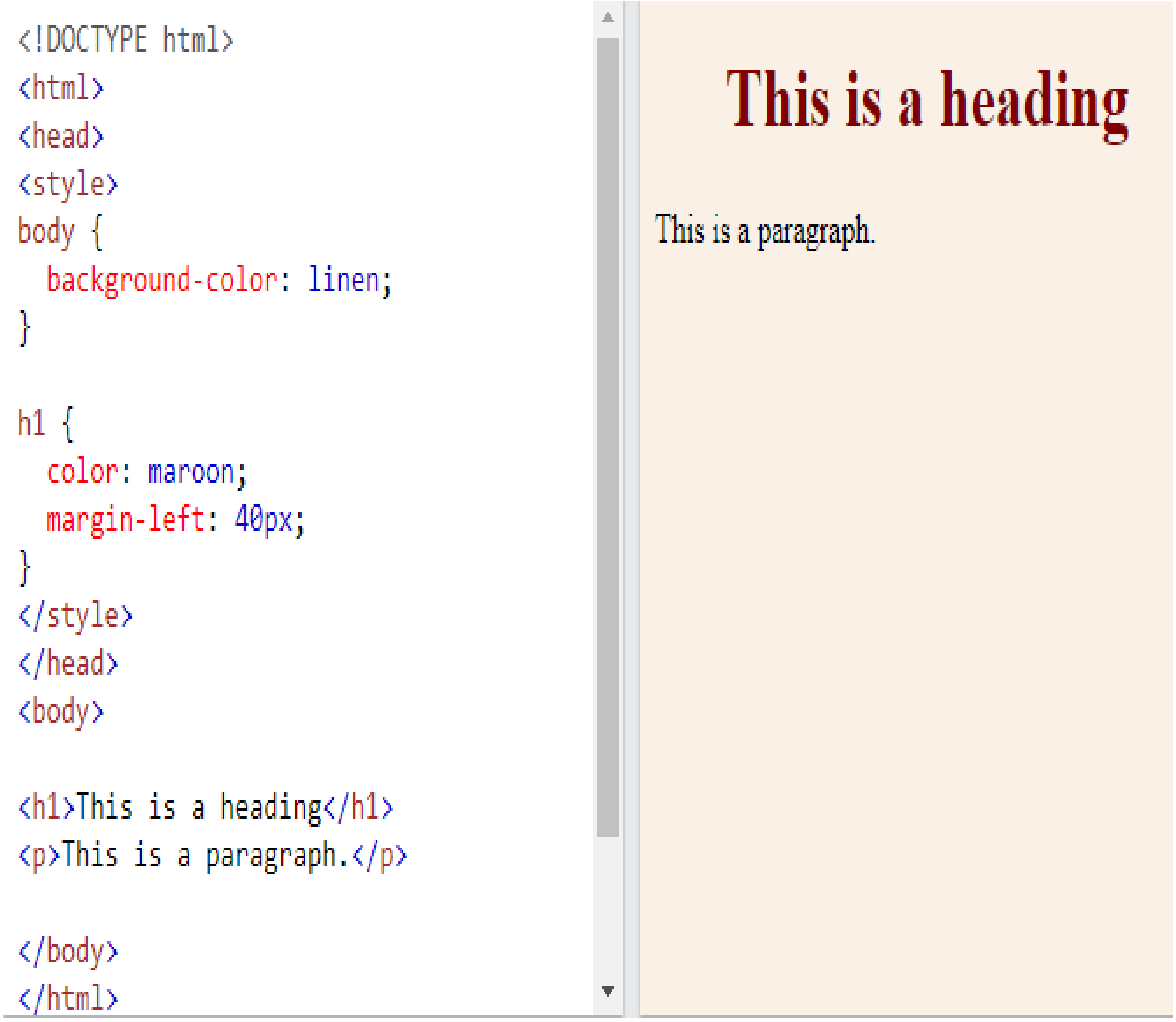
Internal CSS

An internal style sheet may be used if one single HTML page has a unique style.

The internal style is defined inside the <style> element, inside the head section.

Example

Internal styles are defined within the <style> element, inside the <head> section of an HTML page:



```
<!DOCTYPE html>
<html>
<head>
<style>
body {
  background-color: linen;
}

h1 {
  color: maroon;
  margin-left: 40px;
}
</style>
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

This is a heading

This is a paragraph.

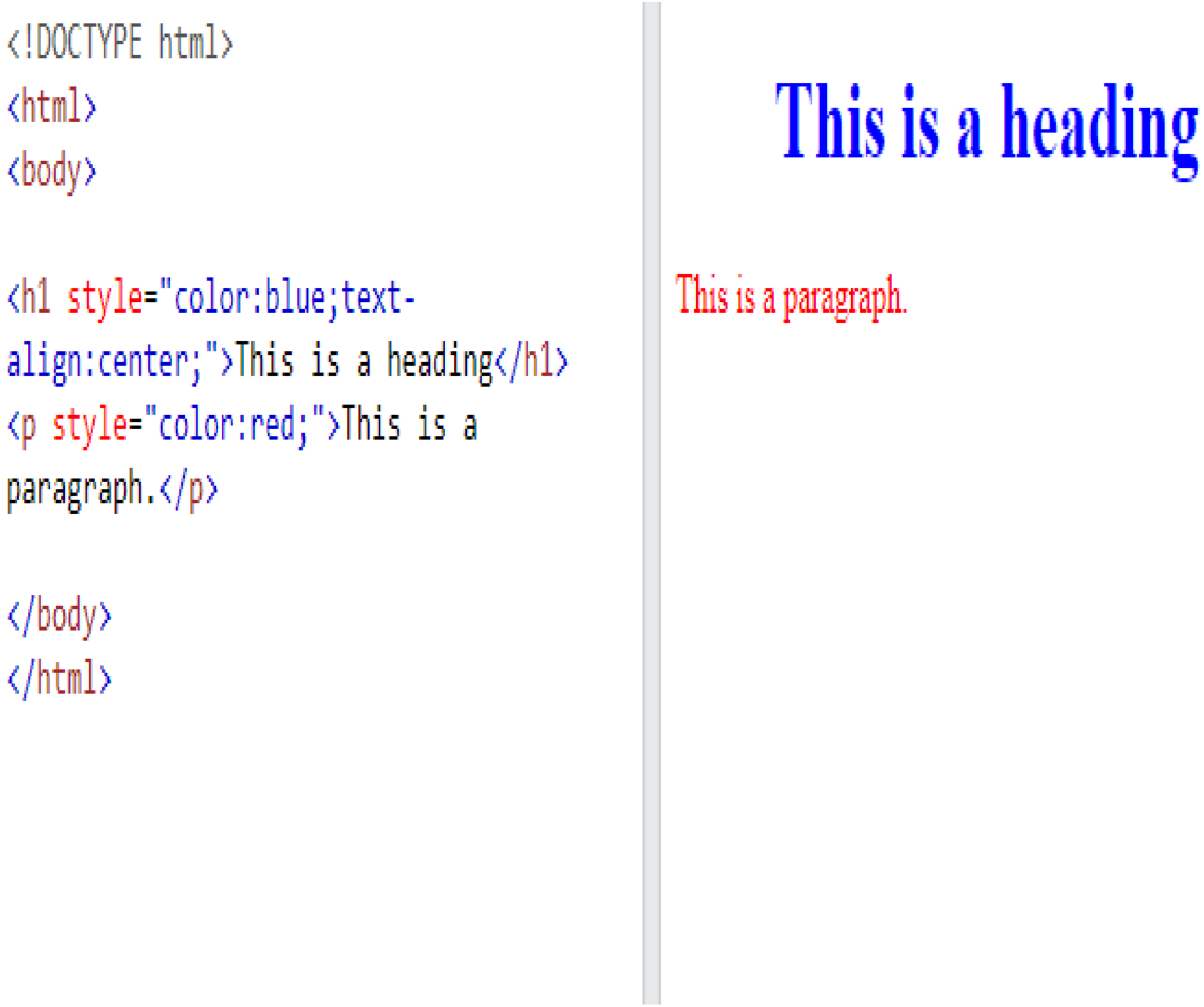
Inline CSS

An inline style may be used to apply a unique style for a single element.

To use inline styles, add the style attribute to the relevant element. The style attribute can contain any CSS property.

Example

Inline styles are defined within the "style" attribute of the relevant element:



```
<!DOCTYPE html>
<html>
<body>

<h1 style="color:blue;text-
align:center;">This is a heading</h1>
<p style="color:red;">This is a
paragraph.</p>

</body>
</html>
```

This is a heading

This is a paragraph.

Tip: An inline style loses many of the advantages of a style sheet (by mixing content with presentation). Use this method sparingly.

Multiple Style Sheets

If some properties have been defined for the same selector (element) in different style sheets, the value from the last read style sheet will be used.

Assume that an **external style sheet** has the following style for the <h1> element:

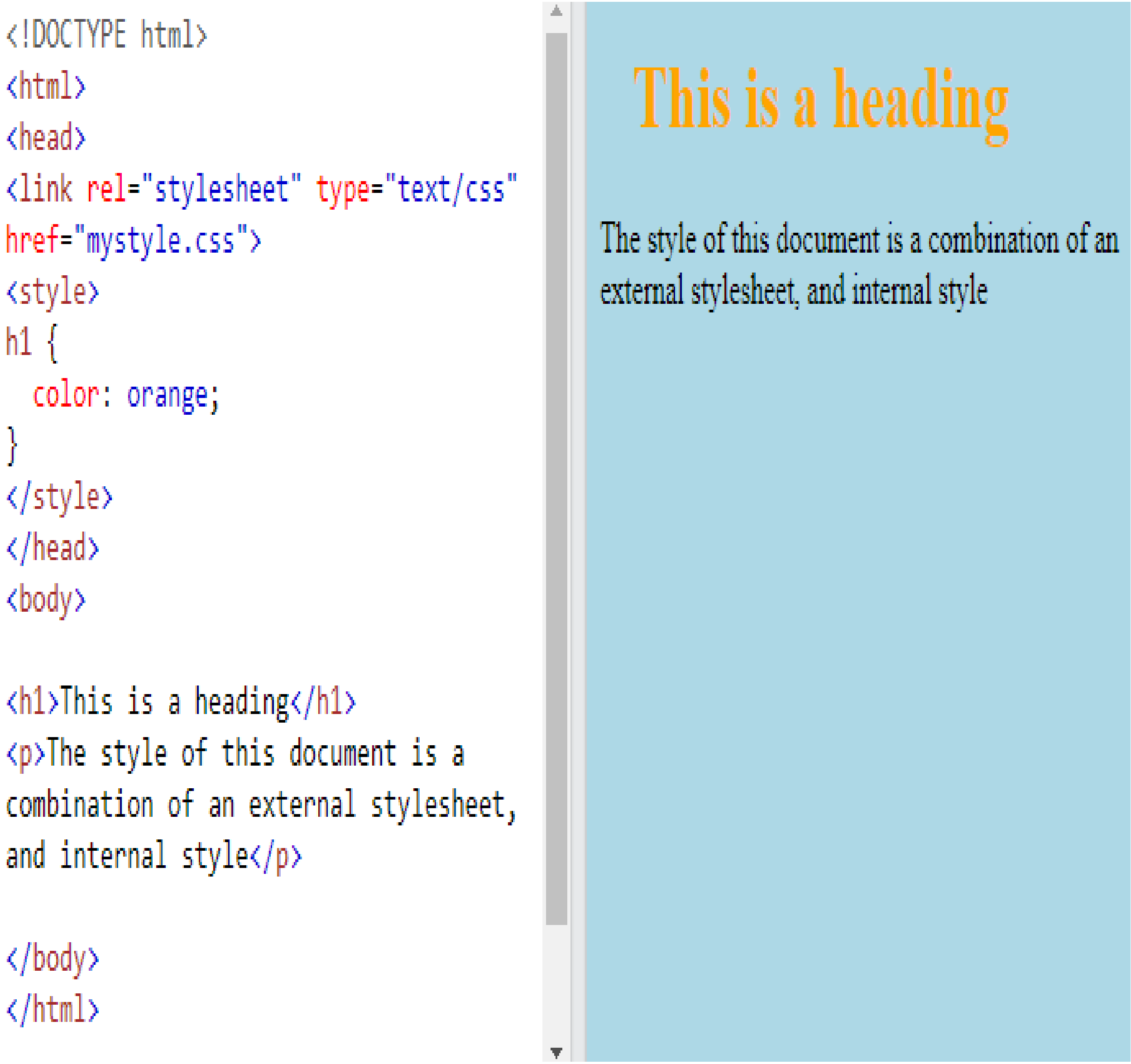
```
h1 {
  color: navy;
}
```

Then, assume that an **internal style sheet** also has the following style for the <h1> element:

```
h1 {
  color: orange;
}
```

Example

If the internal style is defined **after** the link to the external style sheet, the <h1> elements will be "orange":



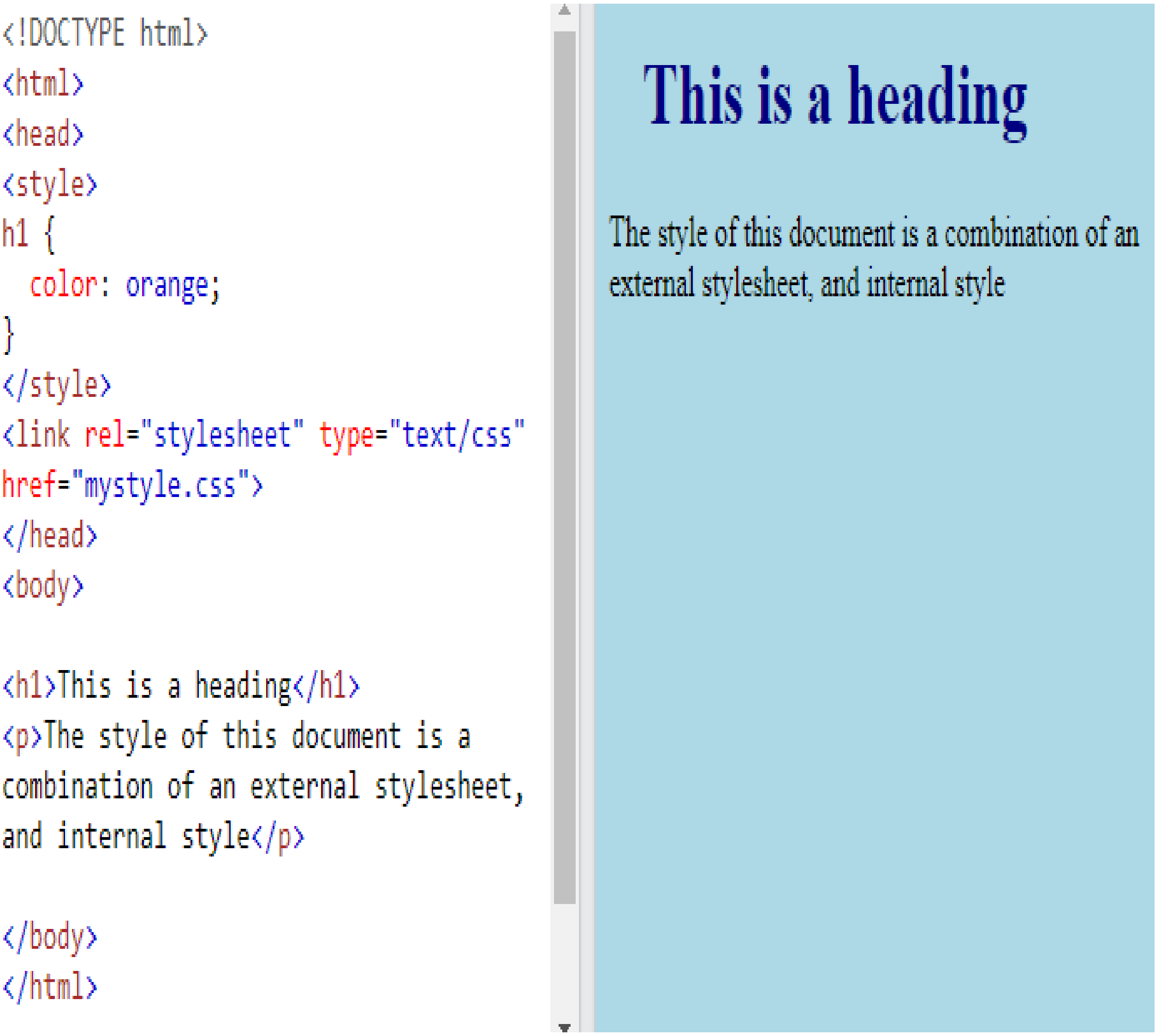
```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" type="text/css"
href="mystyle.css">
<style>
h1 {
  color: orange;
}
</style>
</head>
<body>

<h1>This is a heading</h1>
<p>The style of this document is a
combination of an external stylesheet,
and internal style</p>

</body>
</html>
```

This is a heading

The style of this document is a combination of an external stylesheet, and internal style

Example

However, if the internal style is defined **before** the link to the external style sheet, the <h1> elements will be "navy":



```
<!DOCTYPE html>
<html>
<head>
<style>
h1 {
  color: orange;
}
</style>
<link rel="stylesheet" type="text/css"
href="mystyle.css">
</head>
<body>

<h1>This is a heading</h1>
<p>The style of this document is a
combination of an external stylesheet,
and internal style</p>

</body>
</html>
```

This is a heading

The style of this document is a combination of an external stylesheet, and internal style

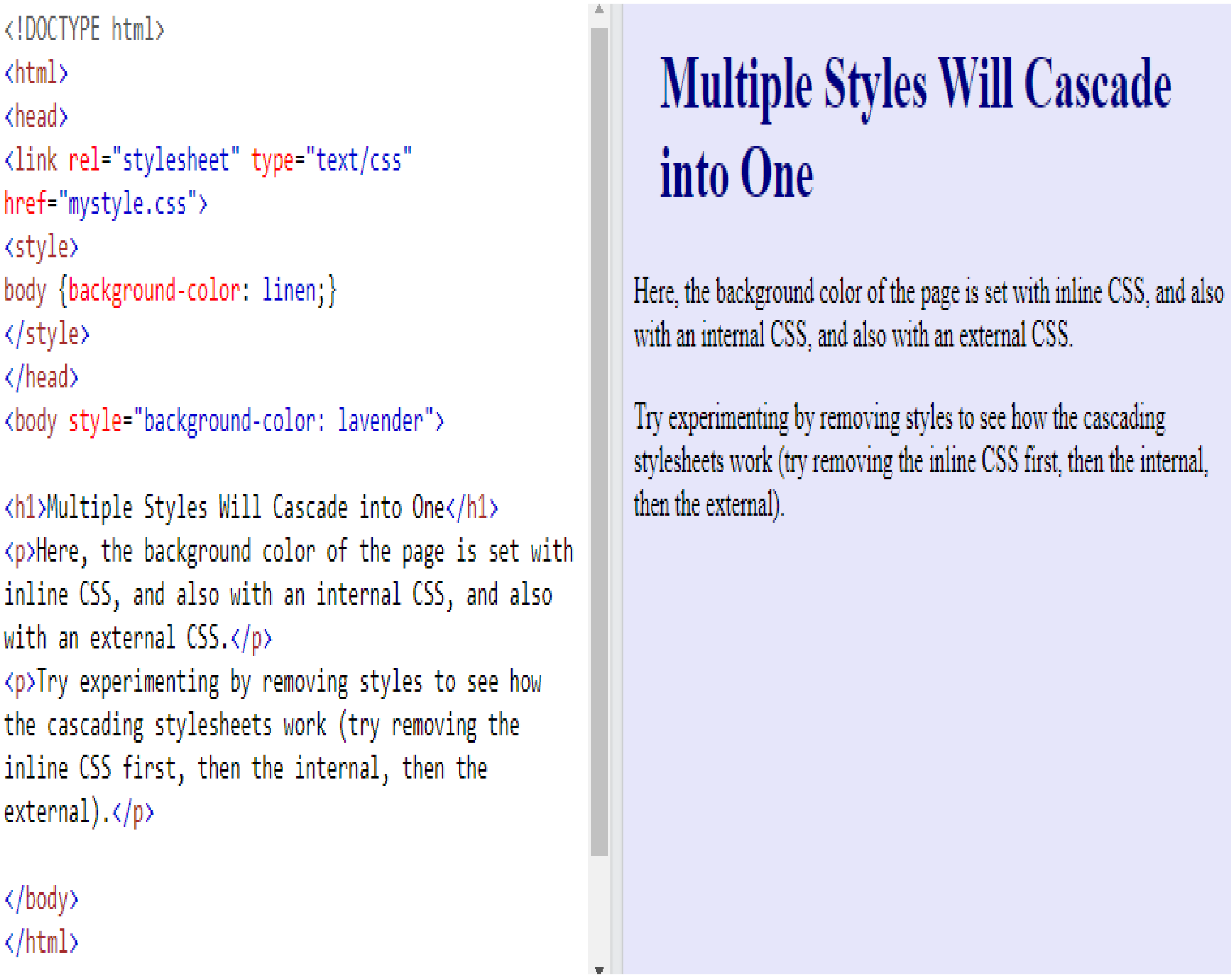
Cascading Order

What style will be used when there is more than one style specified for an HTML element?

All the styles in a page will "cascade" into a new "virtual" style sheet by the following rules, where number one has the highest priority:

1. Inline style (inside an HTML element)
2. External and internal style sheets (in the head section)
3. Browser default

So, an inline style has the highest priority, and will override external and internal styles and browser defaults.



```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" type="text/css"
href="mystyle.css">
<style>
body {background-color: linen;}
</style>
</head>
<body style="background-color: lavender">

<h1>Multiple Styles Will Cascade into One</h1>
<p>Here, the background color of the page is set with
inline CSS, and also with an internal CSS, and also
with an external CSS.</p>
<p>Try experimenting by removing styles to see how
the cascading stylesheets work (try removing the
inline CSS first, then the internal, then the
external).</p>

</body>
</html>
```

Multiple Styles Will Cascade into One

Here, the background color of the page is set with inline CSS, and also with an internal CSS, and also with an external CSS.

Try experimenting by removing styles to see how the cascading stylesheets work (try removing the inline CSS first, then the internal, then the external).