Scan or click here for more resources

DHTML & CSS

Unit - V

DHTML

Dynamic Hyper Text Markup Language (**DHTML**) is a combination of Web development technologies used to create dynamically changing websites. Web pages may include animation, dynamic menus and text effects. The technologies used include a combination of HTML, JavaScript or VB Script, CSS and the document object model (DOM).

- DHTML stands for **D**ynamic **HTML**.
- DHTML is the art of combining
  - HTML
  - CSS
  - JavaScript
  - DOM
- DHTML is NOT a Language
- According to the World Wide Web Consortium (W3C):
  "Dynamic HTML is a term used by some vendors to describe the combination of HTML, style sheets and scripts that allows documents to be animated."
- Designed to enhance a Web user's experience, DHTML includes the following features:

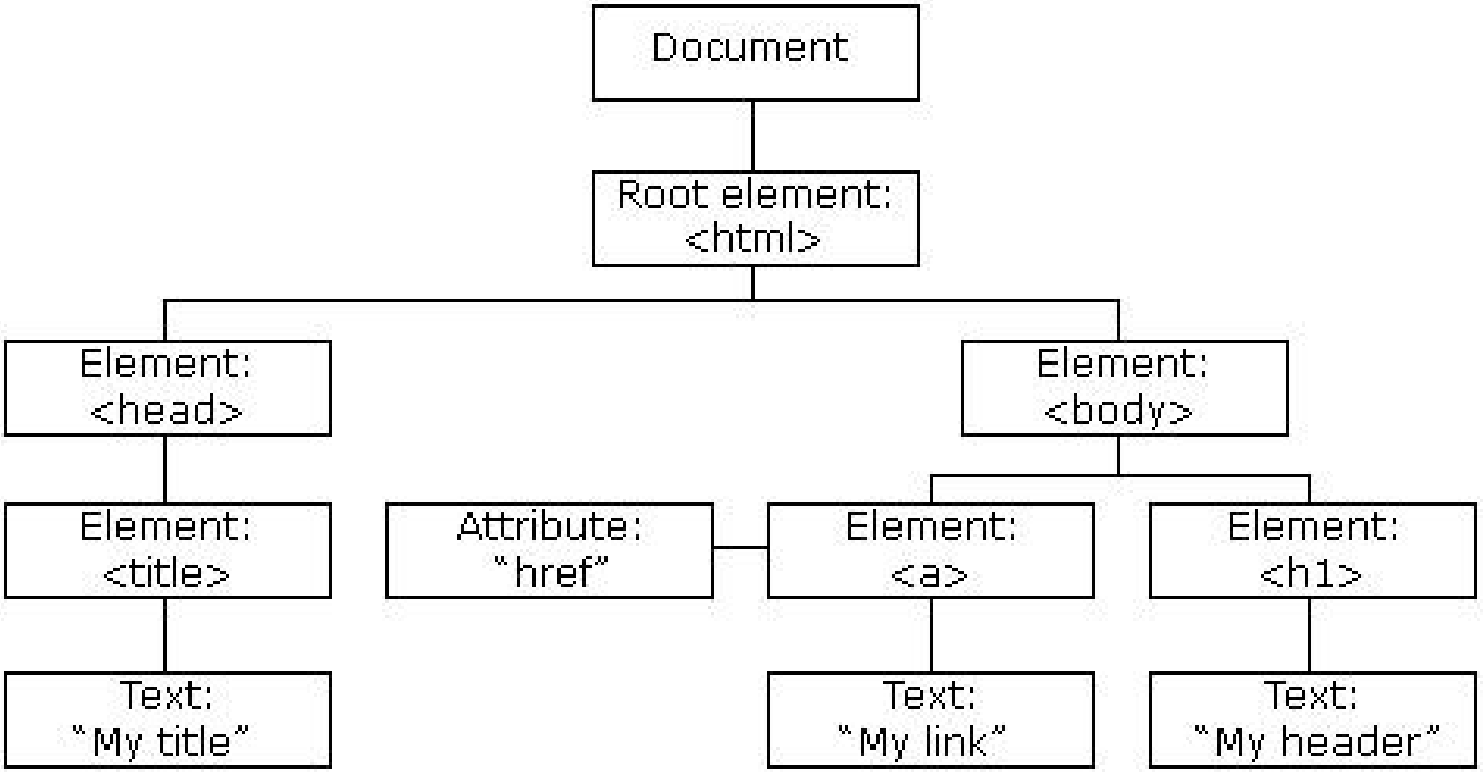
- Dynamic content, which allows the user to dynamically change Web page content
- Dynamic positioning of Web page elements
- Dynamic style, which allows the user to change the Web page's color, font, size or content

The HTML DOM (Document Object Model)

When a web page is loaded, the browser creates a **D**ocument **O**bject **M**odel of the page.

The **HTML DOM** model is constructed as a tree of **Objects**:

The HTML DOM Tree of Objects



DHTML

- HTML
  - The W3C HTML 4 standard has rich support for dynamic content:
  - HTML supports JavaScript
  - HTML supports the Document Object Model (**DOM**)
  - HTML supports HTML Events
  - HTML supports Cascading Style Sheets (**CSS**)
  - DHTML is about using these features, to create dynamic and interactive web pages.
- JavaScript
  - JavaScript is the most popular scripting language on the internet, and it works in all major browsers.
  - DHTML is about using JavaScript to control, access and manipulate HTML elements.

DHTML

- HTML DOM
  - The HTML DOM is a W3C standard.
  - It describes the **Document Object Model** for HTML.
  - The HTML DOM defines a standard way for accessing and manipulating HTML documents.
  - DHTML is about using the DOM to access and manipulate HTML elements.
- HTML Events
  - HTML events are a part of the HTML DOM.
  - DHTML is about creating web pages that reacts to (user)events.
- CSS
  - CSS defines how to display HTML elements.
  - DHTML is about using JavaScript and the HTML DOM to change the style and positioning of HTML elements.

CASCADING STYLE SHEETS (CSS)

- CSS stands for Cascading Style Sheets. CSS is a standard style sheet language used for describing the presentation (i.e. the layout and formatting) of the web pages.

- CSS was introduced in 1996 by the World Wide Web Consortium (W3C), which also maintains its standard. CSS was designed to enable the separation of presentation and content. Now web designers can move the formatting information of the web pages to a separate style sheet which results in considerably simpler HTML markup, and better maintainability.

- CSS3 is the latest version of the CSS specification. CSS3 adds several new styling features and improvements to enhance the web presentation capabilities.

- **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.

- **CSS**, is a simply designed language intended to simplify the process of making web pages presentable. **CSS** allows you to apply styles to web pages. More importantly, **CSS** enables you to do this independent of the HTML that makes up each web page.
- Styles enable you to define a consistent 'look' for your documents by describing once how headings, paragraphs, quotes, etc. should be displayed.

What You Can Do with CSS

There are lot more things you can do with CSS :-

- You can easily apply same style rules on multiple elements.
- You can control the presentation of multiple pages of a website with a single style sheet.
- You can present the same page differently on different devices.
- You can change the position of an element on a web page without changing the markup.
- You can alter the display of existing HTML elements.
- You can transform elements like scale, rotate, skew, etc. in 2D or 3D space.
- You can create animations and transitions effects without using any JavaScript.
- You can create print friendly version of your web pages.
- The list does not end here, there are many other interesting things that you can do with CSS.

Advantages of Using CSS

The biggest advantage of CSS is that it allows the separation of style and layout from the content of the document. Here are some more advantages -

CSS Save Lots of Time — CSS gives lots of flexibility to set the style properties of an element. You can write CSS once; and then the same code can be applied to the groups of HTML elements, and can also be reused in multiple HTML pages.

Easy Maintenance — CSS provides an easy means to update the formatting of the documents, and to maintain the consistency across multiple documents. Because the content of the entire set of web pages can be easily controlled using one or more style sheets.

Pages Load Faster — CSS enables multiple pages to share the formatting information, which reduces complexity and repetition in the structural contents of the documents. It significantly reduces the file transfer size, which results in a faster page loading.

Superior Styles to HTML — CSS has much wider presentation capabilities than HTML and provide much better control over the layout of your web pages. So you can give far better look to your web pages in comparison to the HTML presentational elements and attributes.

Multiple Device Compatibility — CSS also allows web pages to be optimized for more than one type of device or media. Using CSS the same HTML document can be presented in different viewing styles for different rendering devices such as desktop, cell phones, etc.

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.

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>
This is a paragraph.

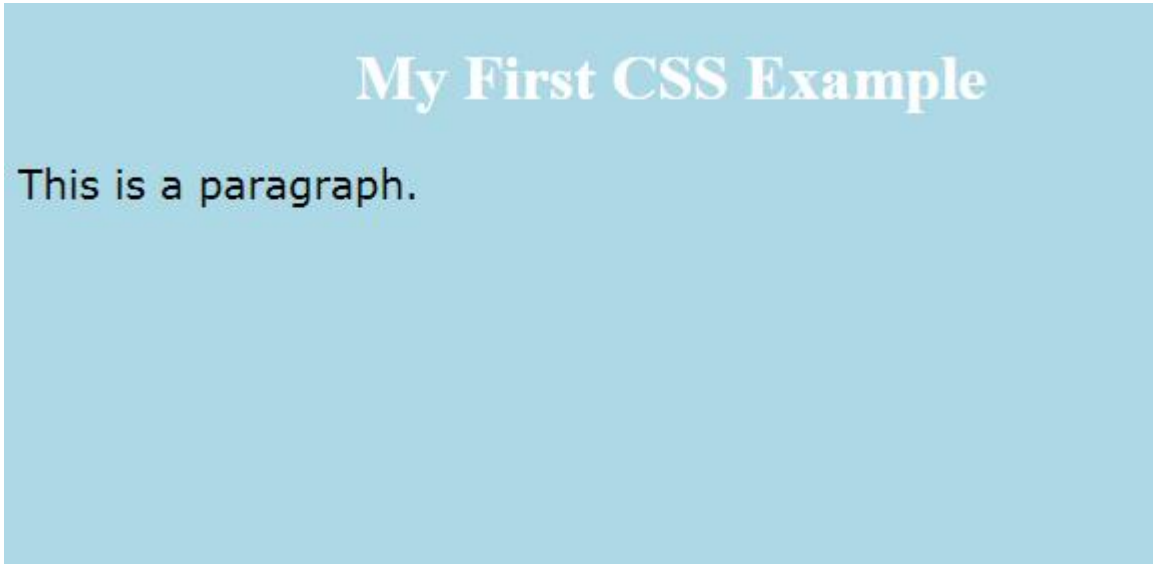
When tags like <font>, and color attributes were added to the HTML 3.2 specification, it started a nightmare for web developers. Development of large websites, where fonts and color information were added to every single page, became a long and expensive process.

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

CSS removed the style formatting from the HTML page!

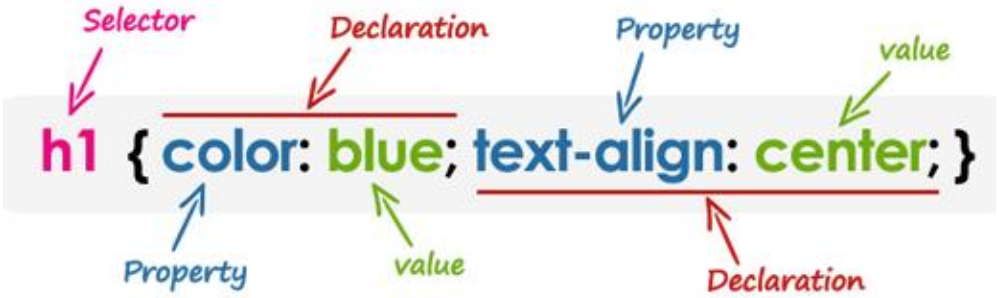
CSS Saves a Lot of Work!
The style definitions are normally saved in external .css files.
With an external stylesheet file, you can change the look of an entire website by changing just one file!

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

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

p {
  font-family: verdana;
  font-size: 20px;
}
</style>
</head>
<body>
<h1>My First CSS Example</h1>
<p>This is a paragraph.</p>
</body>
</html>
```



CSS Syntax

A CSS stylesheet consists of a set of rules that are interpreted by the web browser and then applied to the corresponding elements such as paragraphs, headings, etc. in the document.
A CSS rule have two main parts, a selector and one or more declarations:



The selector points to the HTML element you want to style.
The declaration block contains one or more declarations separated by semicolons.
Each declaration consists of a property and a value separated by a colon (:) and ending with a semicolon (;), and the declaration groups are surrounded by curly braces {}.

The selector specifies which element or elements in the HTML page the CSS rule applies to.

Whereas, the declarations within the block determines how the elements are formatted on a webpage.
The property is the style attribute you want to change; they could be font, color, background, etc. Each property has a value, for example color property can have value either blue or #0000FF etc.

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

To make the CSS more readable, you can put one declaration on each line, like this:

Example

```
H1
{
color: blue;
text-align: center;
}
```

In the example above h1 is a selector in CSS (it points to the HTML element you want to style: <H1>),
color is a property, and red is the property value
text-align is a property, and center are the corresponding values of these properties.

Note: A CSS declaration always ends with a semicolon ";", and the declaration groups are always surrounded by the curly brackets "{}".

Case Sensitivity in CSS

CSS property names and many values are not case-sensitive. Whereas, **CSS selectors are usually case-sensitive**, for instance, the class selector .maincontent is not the same as .mainContent.
Therefore, to be on safer side, you should assume that all components of CSS rules are case-sensitive.

CSS Getting Started

A CSS file is simply a plain text file saved with the .css extension. 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

CSS can either be attached as a separate document or embedded in the HTML document itself. There are three ways of inserting a style sheet:

- External CSS
- Internal CSS
- Inline CSS

External style sheets —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.

An external style sheet is ideal when the style is applied to many pages of the website.

An external style sheet holds all the style rules in a separate document that you can link from any HTML file on your site. External style sheets are the most flexible because with an external style sheet, you can change the look of an entire website by changing just one file.

```
<!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>
```

"mystyle.css"

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

Note: Do not add a space between the property value and the unit (such as margin-left: 20 px;). The correct way is: margin-left: 20px;

Linking To An External CSS

- Do not put <style></style> tags in the .CSS file; this will prevent it from working
- Add CSS rules as needed; break lines where necessary; format as desired
- Save as *filename*.css
- Reference .CSS in <head> of .HTML(s)

```
<head>
<link rel="stylesheet" type="text/css"
      href="mystyles.css">
</head>
```

Link to External CSS

External style sheets can be referenced with a full URL or with a path relative to the current web page.

Example
This example uses a full URL to link to a style sheet:
<link rel="stylesheet" href="https://www.w3schools.com/html/styles.css">

Example
This example links to a style sheet located in the html folder on the current web site:
<link rel="stylesheet" href="/html/styles.css">

Example
This example links to a style sheet located in the same folder as the current page:
<link rel="stylesheet" href="styles.css">

Internal or Embedded 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.

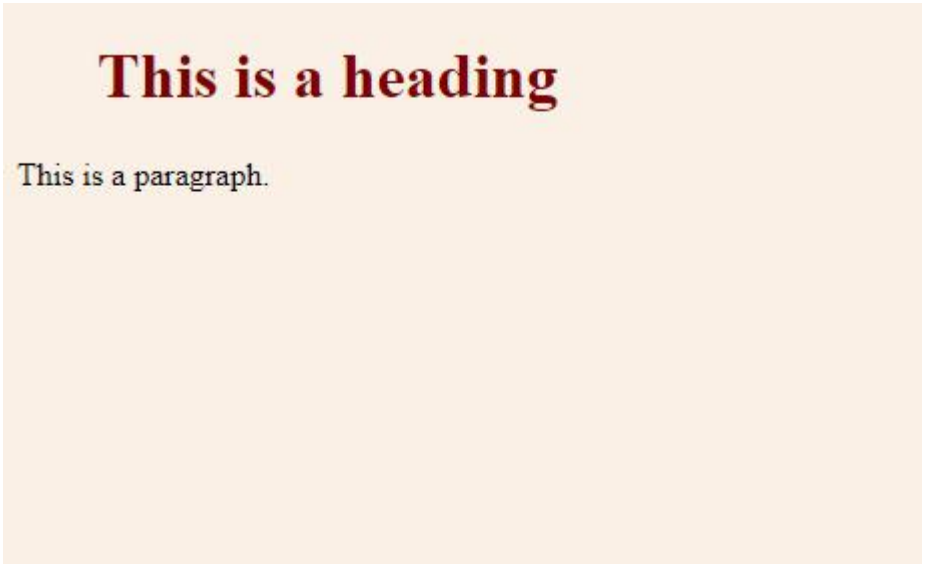
Embedded style sheets are defined in the <head> section of an HTML document using the <style> element. You can define any number of <style> elements in an HTML document but they must appear between the <head> and </head> tags.

Embedded or internal style sheets only affect the document they are embedded in.

```
<html lang="en">
<head>
 <title>My HTML Document</title>
<style>
body { background-color: YellowGreen; }
p { color: #fff; }
</style>
</head>
<body>
<h1>This is a heading</h1>
<p>This is a paragraph of text.</p>
</body>
 </html>
```

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;
}
p{color: navy;
  margin-left: 20px;
</style>
</head>
<body>
<h1>This is a heading</h1>
<p>This is a paragraph.</p>
</body>
</html>
```



Inline Styles

Inline styles are used to apply the unique style rules to an element by putting the CSS rules directly into the start tag. It can be attached to an element using the style attribute.

The style attribute includes a series of CSS property and value pairs. Each "property: value" pair is separated by a semicolon (;), just as you would write into an embedded or external style sheets. But it needs to be all in one line i.e. no line break after the semicolon, as shown here:

```
<!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>
```

Note : Using the inline styles are generally considered as a bad practice. As style rules are embedded directly inside the HTML tag, it causes the presentation to become mixed with the content of the document; which makes the code hard to maintain and negates the purpose of using CSS.

```
<html lang="en">
<head>
        <meta charset="utf-8">
   <title>Example of CSS Inline Styles</title>
</head>
<body>
        <h1 style="color:red; font-size:30px;">This is a heading</h1>
        <p style="color:green; font-size:22px;">This is a paragraph.</p>
        <div style="color:blue; font-size:14px;">This is some textcontent.</div>
</body>
</html>
```

Cascading Order

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.

CSS Comments

Comments are used to explain the code, and may help when you edit the source code at a later date. Comments are ignored by browsers. A CSS comment is placed inside the <style> element, and starts with /* and ends with */:

Example

```
<!DOCTYPE html>
<html>
<head>
<style>
/* This is a single-line comment */

p {color: red;}

</style>
</head>
<body>

<p>Hello World!</p>
<p>This paragraph is styled with CSS.</p>
<p>CSS comments are not shown in the
output.</p>
</body>
</html>
```

Hello World!

This paragraph is styled with CSS.

CSS comments are not shown in the output.

HTML and CSS Comments

From the HTML tutorial, you learned that you can add comments to your HTML source by using the <!--...--> syntax.

In the following example, we use a combination of HTML and CSS comments:

```
<!DOCTYPE html>
<html>
<head>
<style>
p {
  color: red; /* Set text color to red */
}
</style>
</head>
<body>
<h2>My Heading</h2>
<!-- These paragraphs will be red -->
<p>Hello World!</p>
<p>This paragraph is styled with CSS.</p>
<p>HTML and CSS comments are not shown in the
output.</p>
</body>
</html>
```

CSS Colors, Fonts and Sizes

Here, we will commonly used CSS properties.

The CSS color property defines the text color to be used.
The CSS font-family property defines the font to be used.
The CSS font-size property defines the text size to be used.

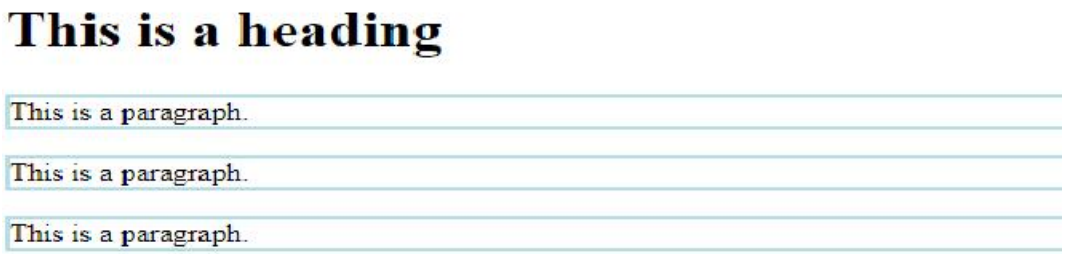
```
<!DOCTYPE html>
<html><head>
<style>
h1 {
  color: blue;
  font-family: verdana;
  font-size: 300%;
}
p {
  color: red;
  font-family: courier;
  font-size: 160%;
}
</style>
</head><body>
<h1>This is a heading</h1>
<p>This is a paragraph.</p>
</body></html>
```

CSS Border

The CSS border property defines a border around an HTML element.

You can define a border for nearly all HTML elements.

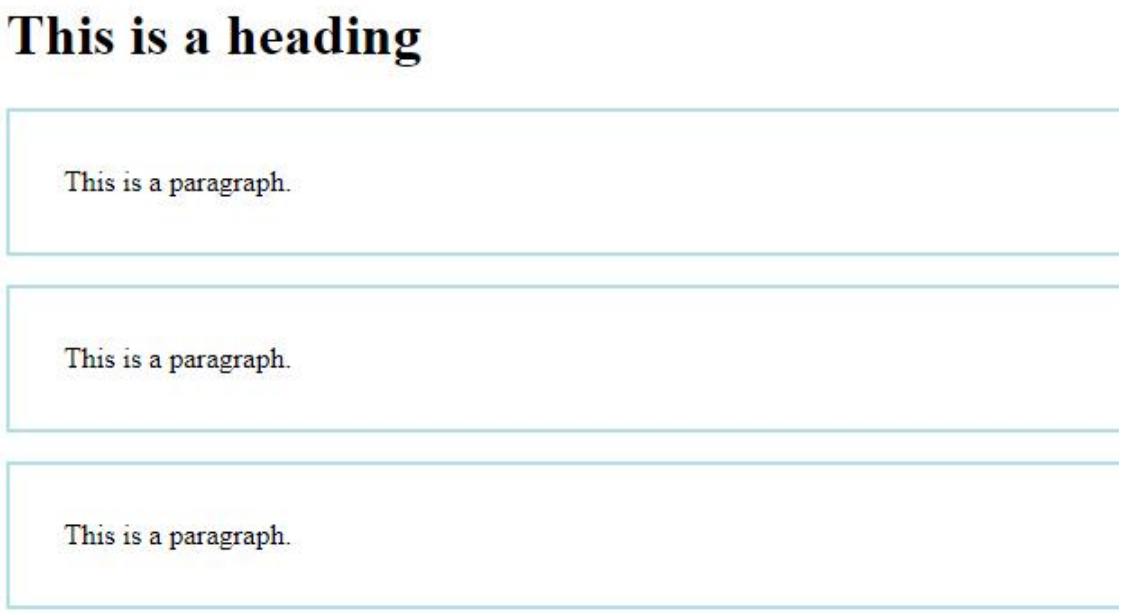
```
<!DOCTYPE html>
<html>
<head>
<style>
p {
  border: 2px solid powderblue;
}
</style>
</head>
<body>

<h1>This is a heading</h1>

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

</body>
</html>
```



CSS Padding

The CSS padding property defines a padding (space) between the text and the border.

```
<!DOCTYPE html>
<html>
<head>
<style>
p {
  border: 2px solid powderblue;
  padding: 30px;
}
</style>
</head>
<body>

<h1>This is a heading</h1>

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

</body>
</html>
```



CSS Margin

The CSS margin property defines a margin (space) outside the border.

```
<!DOCTYPE html>
<html>
<head>
<style>
p {
  border: 2px solid powderblue;
  margin: 50px;
}
</style>
</head>
<body>

<h1>This is a heading</h1>

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

</body>
</html>
```