CSS

Chapter 1

CSS Introduction

What is CSS?

CSS stands for Cascading Style Sheets. It is a style sheet language that describes the display of an HTML document, including colors, layouts, and fonts.

Why use CSS?

CSS is used to style HTML components, which makes websites more visually appealing and user-friendly.

How CSS Works

CSS works by selecting HTML components and adding styles to them.

Example:

```
selector {
    property: value;
}
```

Adding CSS

There are three methods to apply CSS to an HTML page:

- Inline CSS
- Internal CSS
- External CSS

Inline CSS

To utilize inline CSS, add the style attribute within the opening tag of an HTML element.

Example:

```
<h1 style="color: blue;"> Tutorials4Coding </h1>
<p> Learn Coding easily! </p>
```

Internal CSS

The internal style is defined within the `<style>` element of the head section.

Example:

```
<!DOCTYPE html>
<html>
<head>
  <style>
    h2 {
     color: red;
    }
  </style>
</head>
<body>
   <h2> Hello Coders! </h2>
   <p> Welcome to this css tutorial. </p>
</body>
</html>
```

External CSS

External CSS is defined in a separate CSS file and linked to the HTML page in the `<head>` section with the `<link>` tag.

CSS Comments

CSS comments are used to provide notes or explanations in the CSS code. Comments have no effect on the style or layout and are ignored by the browser. Comments are enclosed between /* and */.

There are two types of comments in CSS:

- Single-line comments
- Multi-line comments

Single-line comments

Single-line comments are represented by the /* and */. Any information between /* and */ will be treated as a comment and ignored by the browser.

Example:

```
/* This is a single line commnent */
p {
    color: red;
}
```

Multi-line comments

Multi-line comments span several lines and are perfect for detailed explanations.

Example:

```
/* This is a
multi line
commment */
h2 {
    color: red;
    text-align: center;
}
```

Chapter 2

CSS Selectors

A CSS selector is used to select the specific HTML element that you want to style.

CSS selectors are classified into the following types:

- Universal Selector
- Element Selector
- Id Selector
- Class Selector
- Group Selector

Universal Selector

The universal selector selects all HTML elements on the page. It is written with an asterisk (*).

Example:

HTML

```
<!DOCTYPE html>
<html>

<head>
<title> CSS Selectors </title>
</head>

<body>
   <h2> CSS Tutorial </h2>
   <p> This is an simple example of css selecters </p>
</body>
</html>
```

CSS

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

Element Selector

The element selector is used to select HTML elements such as `<h1>`, `<p>`, `<div>` etc.

Example:

HTML

```
<!DOCTYPE html>
<html>

<head>
<head> CSS Selectors </head>
</head>

<body>
   <h1> Tutorials4Coding </h1>
   <p> In this website you can learn: </p>
   <p> C Programming </p>
   <p> Python </p>
   <p> JavaScript and many more... </p>
</body>
</html>
```

CSS

```
* {
    color: blue;
}

p {
    color: violet;
}
```

ID Selector

An ID selector targets a specific HTML element based on its unique identifier. It is defined using the hash "#" symbol followed by the ID name.

HTML

Example:

```
<!DOCTYPE html>
<html>

<head>
<head> CSS Selectors </head>
</head>

<body>
   <h1 id="heading"> Tutorials4Coding </h1>
   <p id="para"> A Code Learning Platform </p>
</body>
</html>
```

CSS

```
#heading {
    color: blue;
}

#para {
    color: red;
}
```

Class Selector

The class selector selects the HTML element based on their class attribute. It is defined using the period “.” character followed by the class name.

HTML

Example:

```
<!DOCTYPE html>
<html>

<head>
<head> CSS Selectors </head>
</head>

<body>
   <h1 class="heading"> CSS ID Selector </h1>
   <p class="para"> This is an example of css class
selectors </p>
</body>
</html>
```

Chapter 3

CSS Backgrounds

The CSS background property is used to set the element's background style and effects.

There are several background properties, including:

Background color

The background-color property defines the background color of an HTML element.

Syntax:

```
selector {
    background-color: color;
}
```

Example:

HTML

```
<!DOCTYPE html>
<html>

<head>
    <title> CSS Backgrounds </title>
</head>

<body>
   <h1> CSS Background Color </h1>
   <p> This is an example of CSS background color. </p>
</body>
</html>
```

CSS

```
h1 {
    background-color: aqua;
}

p {
    background-color: aquamarine;
}
```

Background Position

The background-position property specifies the position of the background image.

Syntax:

```
selector {
    background-position: property-value;
}
```

HTML

Example:

```
<!DOCTYPE html>
<html>

<head>
    <title> CSS Backgrounds </title>
</head>

<body>
   <h1> Background Position </h1>
   <p> This is an example of CSS background position. </p>
</body>
</html>
```

CSS

```
body {
    background-image: url("random.png");
    background-repeat: no-repeat;
    background-position: top right;
}
```

Chapter 4

CSS Borders

CSS borders are used to create a visual boundary around HTML page elements.

The following are the different properties of a CSS border:

Border Style

The CSS border-style property defines the style of the border.

Example:

HTML

```
<!DOCTYPE html>

<html>
<head>
    <title> CSS Borders </title>
</head>
<body>
   <p class="none"> no border </p>
   <p class="dotted"> Dotted Border </p>
   <p class="dashed"> Dashed Border </p>
   <p class="solid"> Solid border </p>
   <p class="double"> Double Border </p>
   <p class="inset"> inset border </p>
   <p class="outset"> Outset Border </p>
</body>
</html>
```

CSS

```
.none {
    border-style: none;
}

.dotted {
    border-style: dotted;
}

.dashed {
    border-style: dashed;
}

.solid {
    border-style: solid;
}

.double {
    border-style: double;
}

.inset {
    border-style: inset;
}

.outset {
    border-style: outset;
}
```

Border Color

The CSS border-color property specifies the color of the border.

Example:

HTML

```
<!DOCTYPE html>
<html>

<head>
    <title> CSS Borders </title>
</head>

<body>
   <p class="dotted"> Dotted Border </p>
   <p class="dashed"> Dashed Border </p>
   <p class="solid"> Solid border </p>
</body>
</html>
```

CSS

```
.dotted {
    border-style: dotted;
    color: orange;
}

.dashed {
    border-style: dashed;
    color: rgb(39, 39, 255);
}

.solid {
    border-style: solid;
    color: #36e900;
}
```

Border Radius

The CSS border-radius property defines the round corners of the element's border.

Example:

HTML

```
<!DOCTYPE html>
<html>

<head>
    <title> CSS Borders </title>
</head>

<body>
   <p class="border1"> Border 1 </p>
   <p class="border2"> Border 2 </p>
</body>
</html>
```

CSS

```
.border1 {
    border: 4px solid blue;
    border-radius: 6px;
}

.border2 {
    border: 4px solid #1033ff;
    border-radius: 20px;
}
```

Chapter 5

CSS Font

CSS Fonts are an essential component of web design because they allow designers to customize the appearance and style of text on websites.

The following are the different properties of a CSS fonts:

Font size

The font size property sets the size of the text. It can be specified using a variety of units, such as pixels (px), ems (em), and percentages (%).

Example:

HTML

```
<!DOCTYPE html>
<html>

<head>
    <title> CSS Fonts </title>
</head>

<body>
   <h1 class="heading"> This is a heading </h1>
   <p class="para1"> This is a paragraph. </p>
   <p class="para2"> This is a paragraph. </p>
</body>
</html>
```

CSS

```
.heading {
    font-size: 30px;
}

.para1 {
    font-size: 20px;
}

.para2 {
    font-size: 20px;
}
```

Font Style

The font style property sets the style of the font.

There are three types of font styles:

- normal
- italic
- oblique

Example:

HTML

```
<!DOCTYPE html>
<html>

<head>
    <title> CSS Fonts </title>
</head>

<body>
   <p class="para1"> This is a normal paragraph. </p>
   <p class="para2"> This is a italic style paragraph. </p>
   <p class="para3"> This is a oblique style paragraph. </p>
</body>
</html>
```

CSS

```
.para1 {
    font-style: normal;
}

.para2 {
    font-style: italic;
}

.para3 {
    font-style: oblique;
}
```

Font Family

The font-family property specifies the font or collection of fonts for text content.

Example:

HTML

```
<!DOCTYPE html>
<html>

<head>
    <title> CSS Borders </title>
</head>

<body>
   <h1 class="heading1"> This is a heading. </h1>
   <p class="para1"> This is a paragraph. </p>
</body>
</html>
```

CSS

```
.heading1 {
    font-family: 'Times New Roman', Times, serif;
}

.para1 {
    font-family: Arial, Helvetica, sans-serif;
}
```

Chapter 6

CSS FlexBox

Flexible Box Layout, often called CSS Flexbox, is a powerful layout concept that makes creating responsive and flexible layouts simple.

Flexbox is made up of flex containers and flex items:

Flex Container Properties

The properties of the flex container are:

Flex Direction

It specifies which direction the flex elements will be shown. It accepts values such as row, column, row-reverse, and column-reverse.

Example:

HTML

```
<!DOCTYPE html>
<html>

<head>
    <title> CSS FlexBox </title>
</head>

<body>
   <div class="flex-container">
     <div> 1 </div>
     <div> 2 </div>
     <div> 3 </div>
   </div>
</body>
</html>
```

CSS

```
.flex-container {
    display: flex;
    flex-direction: column;
    background-color: #234EDDFF;
}

.flex-container div {
    background-color: white;
    margin: 10px;
    text-align: center;
    line-height: 75px;
    font-size: 30px;
}
```

Flex Wrap

The flex-wrap property determines whether flex items should wrap or not.

Example:

HTML

```
<!DOCTYPE html>
<html>

<head>
    <title> CSS FlexBox </title>
</head>

<body>
   <div class="flex-container">
     <div> 1 </div>
     <div> 2 </div>
     <div> 3 </div>
     <div> 4 </div>
     <div> 5 </div>
     <div> 6 </div>
     <div> 7 </div>
     <div> 8 </div>
     <div> 9 </div>
     <div> 10 </div>
     <div> 11 </div>
     <div> 12 </div>
   </div>
</body>
</html>
```

CSS

```
.flex-container {
    display: flex;
    flex-wrap: wrap;
    background-color: #234EDDFF;
}

.flex-container div {
    background-color: white;
    margin: 10px;
    text-align: center;
    line-height: 75px;
    font-size: 30px;
}
```

Justify Content

The justify-content property is used to align the flex items along the main axis.

Example:

HTML

```
<!DOCTYPE html>
<html>

<head>
    <title> CSS FlexBox </title>
</head>

<body>
   <div class="flex-container">
     <div> 1 </div>
     <div> 2 </div>
     <div> 3 </div>
   </div>
</body>
</html>
```

CSS

```
.flex-container {
    display: flex;
    justify-content: center;
    background-color: #234EDDFF;
}

.flex-container div {
    background-color: white;
    margin: 10px;
    text-align: center;
    line-height: 75px;
    font-size: 30px;
}
```

Align Items

The align-items property is used to align the flex items along the cross axis.

Example:

HTML

```
<!DOCTYPE html>
<html>

<head>
    <title> CSS FlexBox </title>
</head>

<body>
   <div class="flex-container">
     <div> 1 </div>
     <div> 2 </div>
     <div> 3 </div>
   </div>
</body>
</html>
```

CSS

```
.flex-container {
    display: flex;
    align-items: center;
    background-color: #234EDDFF;
}

.flex-container div {
    background-color: white;
    margin: 10px;
    text-align: center;
    line-height: 75px;
    font-size: 30px;
}
```

Chapter 7

CSS Grid

CSS Grid is a powerful layout system that allows developers to design two-dimensional layouts using rows and columns.

Example:

HTML

```
<!DOCTYPE html>
<html>

<head>
    <title> CSS Grid </title>
</head>

<body>
   <div class="grid-container">
     <div class="grid-item"> 1 </div>
     <div class="grid-item"> 2 </div>
     <div class="grid-item"> 3 </div>
     <div class="grid-item"> 4 </div>
     <div class="grid-item"> 5 </div>
     <div class="grid-item"> 6 </div>
   </div>
</body>
</html>
```

CSS

```
.grid-container {
    display: grid;
    grid-template-columns: auto auto auto;
    background-color: #234edd;
}

.grid-item {
    color: white;
    padding: 20px;
    border: 1px solid black;
    font-size: 30px;
    text-align: center;
}
```

Chapter 8

CSS Media Queries

CSS media queries are an essential component of responsive web design. They allow webpages to adapt to various screen sizes, resolutions, and device capabilities.

Example:

HTML

```
<!DOCTYPE html>
<html>

<head>
    <title> CSS Media Queries </title>
</head>

<body>
    <h1> CSS Media Queries </h1>
    <p> Resize the browser window to see the effect! </p>
</body>
</html>
```

CSS

```
body {
    background-color: #234edd;
}

@media screen and (min-width: 480px) {
  body {
    background-color: aqua;
  }
}
```