

Contents

Topics	Page No.
Unit-3: Introduction to CSS	1
3.0 Structure of Introduction to CSS.....	1
3.1 Learning Outcomes	1
3.2 Introduction to CSS.....	1
3.3 CSS Properties.....	3
3.4 Selectors.....	7
3.5 Types and Levels of CSS.....	13
3.5.1 Inline Style.....	13
3.5.2 Document level Style	15
3.5.3 External CSS	18
3.6 Simple example demonstrating the use of CSS in XHTML...	22
3.7 Self-Assessment Questions.....	27
3.8 Self-Assessment Activities.....	27
3.9 Multiple-Choice Questions.....	27
3.10 Key Answers to Multiple-Choice Questions.....	29
3.11 Summary	29
3.12 Keywords.....	30
3.13 Recommended Resources for Further Reading.....	31

UNIT 3 – INTRODUCTION TO CSS

Structure of Introduction to CSS

- 3.1 Learning Outcomes
- 3.2 Introduction to CSS
- 3.3 CSS Properties
- 3.4 Selectors
- 3.5 Types and Levels of Style Sheets
- 3.6 Simple example demonstrating the use of CSS in an XHTML document
- 3.7 Self-Assessment Questions
- 3.8 Self-Assessment Activities
- 3.9 Multiple-Choice Questions
- 3.10 Key answers to multiple-choice questions
- 3.11 Summary
- 3.12 Keywords
- 3.13 Recommended resources for further reading

3.1 Learning Outcomes:

After the successful completion of this unit, the student will be able to:

- Explain the concept of CSS properties in XHTML.
- Describe the various types of selectors used in CSS.
- Explain the different levels and types of CSS used in Web Pages
- Create presentable web pages using CSS.

3.2 Introduction to CSS

CSS is an acronym for Cascading Style Sheets. The appearance of the web pages and user interface can be controlled by the web developers by specifying how and where the elements should be displayed. CSS allows to add styles to a web document for change of color, font size, background color, foreground color, spacing, and positioning of the elements, sizing of columns and their layout and many more. A Style is a set of formatting instructions that can be applied to a piece of text for better presentation. If no styles are specified the browser's default property values are used. CSS styles change the layout of the web pages. Using CSS saves a lot of work and controls the presentation of multiple web pages. CSS can be used within an XHTML document as well as in external CSS files which have the extension of .css.

The Key aspects of CSS:

- Separation of presentation and content: CSS allows the separation of contents of a web page (XHTML) from the presentation (CSS) making it easier to maintain and update the styles of a web page without changing the contents.
- Selectors and Rules/Properties: Selectors are used to target the XHTML elements and rules/properties are used to define the styles for the elements. For example, selectors like “p”, “h2”, and “div” can be used to define the styling properties like color, font size, font weight, margin, etc. for these selectors.
- Cascading: In CSS, cascading refers to the way the styles are applied. Styles can be inherited from parent elements and overridden by more specific properties.
- External, Internal, and Inline Styles: CSS styles can be applied in different ways. It can be included externally in a separate CSS file, internally using the `<style>` tag within an XHTML document, or inline using the “style” tag within an individual XHTML element.
- Cross-Browser Compatibility: CSS helps ensure that the web pages look and behave consistently across most of the web browsers.
- Responsive Design: CSS is a fundamental tool for creating responsive web pages. Different layouts can be created by the developers to adapt to different screen sizes and devices ensuring that the web pages function well on desktop computers, tablets, and mobile devices.
- Frameworks and Libraries: There are CSS frameworks and libraries, such as Bootstrap, that provide pre-designed CSS components and styles to help developers build responsive and visually appealing websites more quickly.
- Re-usability: The CSS written in external style sheets can be reused in multiple XHTML pages thereby saving time.

CSS is a critical technology in web development that enables designers and developers to control the layout and appearance of web pages, ensuring that they are visually appealing and function as intended on a wide range of devices and browsers.

A group of people within the World Wide Web consortium W3 called the CSS working group maintains CSS specifications. When specifying the styles of the elements in the XHTML document, a set of style rules are followed that are interpreted by the browser and then applied to the corresponding elements in the XHTML document.

The style rule consists of the selector and a declaration block with a property, and a value.

- Selector – A selector is an XHTML tag to which a style can be applied. For example XHTML tags like `<p>`, `<h1>`, `<div>`, `<table>`, `<h2>` etc are the selectors.
- Property – An attribute of an XHTML element is converted to CSS property. For example, color, border, font-size, font-family, etc. are all properties.
- Value – A value is assigned to a property. For example, the font-size property of an element can be set to `18pt`, a font-weight property can be set to “bold” or “italic”, a color property can be set to “red”, “blue” etc.

Syntax of CSS rule:

```
selector { declaration block}
```



```
selector { cssproperty: value }
```

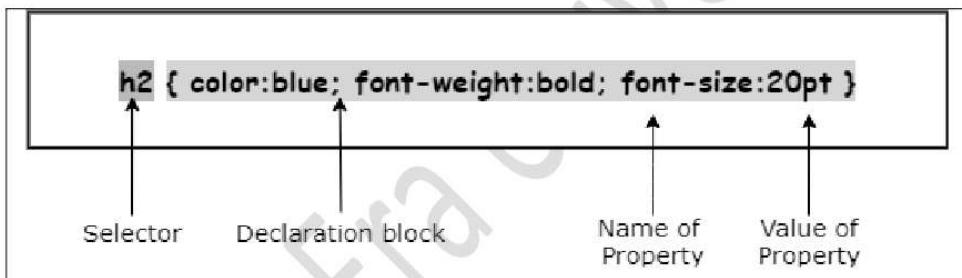


Figure 3.1: Syntax of CSS rule

3.3 CSS Properties

Specifications are followed when setting a value for a property. The values of property can be specified in different forms. It may be an integer, a keyword, url, etc. The syntax for setting a value to a property is given as -

```
cssproperty_1 : value_1;  
cssproperty_2 : value_2;  
cssproperty_3 : value_3;  
...  
cssproperty_n : value_n;
```

Property Value forms:

The property values can be given in various forms –

- Keywords - These are the pre-defined values specified by CSS. For example “bold”, “normal”, “bolder”, “small”, “medium”, “larger”, “italic” etc.

Keyword values are not case-sensitive. Both “bold” and “Bold” mean the same.

- Number values - The value can be an integer or a decimal and can be preceded by a (+) or (-) sign. Units are usually not attached.
- Length values - These are number values followed by unit names.

Units are given as:

px - for pixels

in – for inches

cm - for centimeter

pt – for points (a point is 1/72 inch)

- Percentage values – These values are given by the “%” sign. It is used to provide a measure that is relative to previously used property value. For example “50%”
- URL property - The filename is embedded in parenthesis preceded by the url keyword.

For example: `url(filename.jpg)`

CSS Properties –

There are different types of properties in CSS mainly for setting color, fonts, alignment of text, background, margins, borders, etc. A few are discussed here.

Color:

The color property is used to set the color of the foreground text of the element. It can also be used to set the color of the borders and for showing shading effects.

The color value can be set by using various formats like

- color names → blue, red, green, etc
- hexadecimal code → It is represented by a 6-digit hexadecimal preceded by a “#” sign.

The first two digits(FF) represent a red value - #FF0000,

the next two are a green value(FF) - #00FF00

and the last two digits are the blue values(FF) - #0000FF

- rgb() property → This property takes three values, one each for red, green, and blue. The value can be an integer between 0 and 255 or a percentage.

Red - `rgb(255,0,0)`, Green - `rgb(0, 255, 0)`, and Blue - `rgb(0,0, 255)`,

When elements are nested some property values are inherited by elements while some are not. For example: The background-color property is not inherited whereas the font-size property is inherited from the parent element.

Table 3.1: Examples of setting the foreground color of the text in a paragraph using names of color, hexadecimal code, and rgb() property

Name of the color	Hexadecimal Code	rgb() property
p { color: red }	p { color:#FF0000 }	p { rgb(255,0,0) }
p { color: blue }	p { color:#0000FF }	p { rgb(0,0, 255) }

Background:

The background effects of the element can be set using the background property.

The background property “**background-color**” is used to specify the background color of an element. The setting of the background image and its related properties are explained later.

Example 1: To change the background color of the “body” element, it can be given as

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

Example 2: To change the background color of all “<p>” elements, it can be given as

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

Text:

The text properties are used for formatting the text of the element. Apart from changing the color and font of the text, there are various other properties for aligning text, spacing, decorating, transforming, indenting, etc. A few commonly used properties are given in Table 3.2.

Table 3.2: Commonly used CSS properties used with text of an element

Property Name	Description	Values/Example
text-align	Sets the horizontal alignment of text in an element	left, right, center, justify Example: <code>p { text-align: left; }</code>
text-indent	The first line of text in an element is indented	length - px, pt, % Example: <code>p { text-indent: 60px }</code>
text-decoration	It is used to specify some special features for the text.	underline, overline, line-through, etc. Example: <code>p{text-decoration:underline; }</code>
text-transform	It is used to convert text in the element to lowercase, uppercase, or capitalize the first letter of the text.	lowercase, uppercase, capitalize Example: <code>p { text-transform: uppercase; }</code>
direction	It is used to specify the direction of text in an element.	rtl (right to left), ltr (left to right) Example: <code>p { direction: rtl; }</code>
letter-spacing	Space between the characters in a text is set using this property	px, pt, etc Example: <code>p { letter-spacing: 5px; }</code>
word-spacing	Space between the words in a text is set using this property	px, pt, etc Example: <code>p { word-spacing: 5px; }</code>
color	It is used to set the color of the text.	blue, green, red, etc Example: <code>p { color: blue; }</code>

Fonts:

The font properties are the most commonly used for styling. Various properties used with fonts are:

- **font-size:** It defines the size of the font.

Example: **font-size:18pt;** **font-size: medium;**

- **font-family:** It specifies the list of fonts of the text. The first font in the list supported by the browser is used.

Example: **font-family: Arial, Calibri, Helvetica, Consolas**

- **font-weight:** This property takes the values – “normal”, “bold”, “bolder”, “lighter” etc.

Example: **font-weight: bold;**

- **font-style:** This property takes the values – “italic”, and “normal”.

Example: **font-style: italic;**

- **font-variant:** This property takes the values – “small-caps”, “normal”, etc.

Example: **font-variant: small-caps;**

- **font shorthand** – When more than one property has to be specified, the values can be specified in the form of a list. The order in which the properties are specified is of importance. The font names (font-family) should be the last in the list, and font size should be the second last in the list, while other font properties like font style, font weight, font variants, etc. can be in any order but should be preceded by font size and font-family. Example 1: **font: italic 14pt 'Times New Roman' ;**

Example 2: **font: bold 20pt Arial Helvetica;**

3.4 Selectors

CSS Selectors are used to select the content that is to be styled and presented as per the user's specifications instead of using the default browser setting. There are various types of selectors that can be used for setting the styles. They are:

- Simple Selector
- Class Selector
- Generic Selector
- Id Selector
- Universal Selector
- Pseudo Classes

Simple Selector :

This selects XHTML elements by their name. The property values in the rule are applied to all occurrences of the named elements.

Example 1:

Style Specifications:

```
p {  
    color: blue;  
    font-size:20pt;  
}
```

Element/tag declaration:

```
<p> This is my First Paragraph </p>  
<p> This is my Second Paragraph </p>
```

All the content of the paragraphs in the XHTML document will be displayed in blue with a font size of 20pts.

Example 2:

Style Specifications:

```
div {  
    color: red;  
    font-size:24pt;  
    font-weight: bold;  
}
```

Element/tag declaration:

```
<div> This is my First div element </div>  
<div> This is my Second div element </div>
```

All the content of the <div> elements in the XHTML document will be displayed in red with a font size of 24pt and font weight in bold.

Class Selector :

A class can be defined for specific elements that can be used for styling. They are used to allow different occurrences of the same element/tag to use different style specifications. In other words, the tag name in the XHTML can be the same but each tag can belong to a different class for styling.

A class selector is defined with an element/tag name and a dot preceding the class name when declaring the specifications.

Example: Consider the 3 paragraphs declared here, all have the same tag name i.e. <p> but each can have different classes like “mystyle1”, “mystyle2” “mystyle3”, etc. with different styling specifications for presentation.

Styles specifications:

```
p.mystyle1p {  
    color: orange;  
    font-size:24pt  
}  
  
p.mystyle2p {  
    color: red;  
    font-size:22pt  
}  
  
p.mystyle3p {  
    color: green;  
    font-size:20pt  
}
```

Element Declaration:

```
<p class="mystyle1p">This is my First Paragraph </p>  
<p class="mystyle2p">This is my Second Paragraph </p>  
<p class="mystyle3p">This is my Third Paragraph </p>
```

The First paragraph is displayed in orange color with a font size of 24pt.

The Second paragraph is displayed in red color with a font size of 224pt.

The Third paragraph is displayed in green color with a font size of 20pt.

Generic Selector :

A class can be defined for specific elements/tags that can be used for styling. They are used to allow different elements/tags with the same class name to use the same style specifications for presentation. In other words, the tag/element names can be different but more than one element/tag can belong to the same class for styling. A Generic Selector is defined with just a dot and class name when declaring the specifications.

Example: Consider the paragraphs and heading tags declared here. Here the first paragraph and `<h1>` elements have the same class name though the elements/tags are different, and the second paragraph and `<h2>` elements have the same class name.

Styles specifications:

```
.mystyle1 {  
    color: orange;  
    font-size:24pt  
}  
.mystyle2 {  
    color: red;  
    font-size:22pt  
}
```

Element Declaration:

```
<p class="mystyle1"> This is my First Paragraph </p>  
<h1 class="mystyle1">This is First Heading </h1>  
<p class="mystyle2">This is my Second Paragraph </p>  
<h2 class="mystyle2">This is Second Heading </h2>
```

Here the content of the first paragraph and `<h1>` is displayed in orange color with a font size of 24pt and the content of the second paragraph and `<h2>` is displayed in red color with a font size of 22pt.

Id Selector:

A unique ID is defined for elements that can be used for styling. The “id” attribute of XHTML is used to define an ID. An ID selector is defined with an “#” preceding the ID when declaring the specifications.

Example: Consider the 2 paragraphs declared here, each has a unique ID. Each `<p>` can be displayed with different styling specifications for presentation.

Styles specifications:

```
#p1 {  
    color: orange;  
    font-size:24pt  
}
```

```
#p2 {  
    color: red;  
    font-size:22pt  
}
```

Element Declaration:

```
<p id="p1">This is my First Paragraph </p>  
<p id="p2">This is my Second Paragraph </p>
```

The First paragraph with id “p1” is displayed in orange color with a font size of 24pt.

The Second paragraph, “p2” is displayed in red color with a font size of 22pt.

Universal Selector:

In CSS, to select all the elements in an XHTML document a Universal Selector (*) is used. The selection applies to all the elements including the elements that are nested inside under the element.

Example: Consider the paragraphs, <div>, and heading tags declared here. Universal selector style specifications apply to all the elements within the XHTML document.

Styles specifications:

```
* {  
    color: orange;  
    font-size:24pt  
}
```

Element Declaration:

```
<p> This is my First Paragraph </p>  
<h1>This is Heading 1 </h1>  
  
<p>This is my Second Paragraph  
    <p> This is nested paragraph </p>  
    </p>  
<h2>This is Heading 2 </h2>
```

All paragraphs, <h1> and <h2> tags will be displayed in orange color with a font size of 24pt.

Pseudo Class:

The pseudo-class, styles specific elements depending on their state or position. For example the “:hover” pseudo-class selects an element and applies the style specifications only when the mouse pointer hovers over it. The “:focus” pseudo-class applies the styles when its associate element has received the focus. The pseudo-class names are defined, preceded by a “:” colon.

Example 1: To change the background color of an `<h1>` element when the mouse hovers over it.

Styles specifications:

```
h1:hover {  
    background-color: blue;  
    color: white  
}
```

Element Declaration:

```
<h1>This is Heading 1 </h1>
```

Example 2: To change the styling once the `<input>` element gets focus and the mouse is hovered over it

Styles specifications:

```
input:focus {  
    color: red  
}  
  
input:hover {  
    color: blue  
}
```

Element Declaration:

```
<form action=" ">  
    Name : <input type="text" />  
</form>
```

3.5 Levels of Style Sheets

In the XHTML document, CSS can be added in 3 ways and they are called levels in CSS. There are 3 types/levels of styles.

- Inline level
 - It is specified by using style attribute inside XHTML elements
- Internal/Document Level
 - It is specified by using <style> element in the <head> section
- External Level
 - It is specified in the <head> by using <link> element and referencing the external CSS file by its href attribute

Inline style has precedence over Internal/document Styles and Internal/document style has precedence over external CSS. In an external CSS file, if the CSS style of a particular property is set for any specified element then its value is used until the value of the same property is reset at document level or inline level. Likewise, the property value set for any specified element at the document level is used until the value is reset at the inline level. The styles applied to a particular element are an accumulation of all the properties that are not repeated and are defined at the external, internal, and inline-level. If the properties are repeated then the precedence rule is applicable.

3.5.1 Inline Style

Inline styles are applied to a single element by using the Inline CSS technique. Inline styles are specified within the opening tag of an XHTML element and apply only to the content of that XHTML element.

Syntax :

```
<XHTML-element style="cssproperty1:value; cssproperty2:value;">
    content
</XHTML-element>
```

Example:

```
<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE html>
<html>
<head>
    <title>Example of Inline Style in CSS</title>
```

```
</head>
<body>
    <h2> Graphic Era University </h2>
    <h3> Example of Inline Style in CSS </h3>

    <p style="color:blue; font-size:18pt; background:yellow">
        Inline Styles [color:blue; font-size:18pt;
        background:yellow] used for this first paragraph.
    </p>

    <p style="color:red; font-size:14pt; font-weight:bold">
        Inline Styles [color:red; font-size:14pt;font-weight:bold]
        used for second paragraph.
    </p>

    <p> No Style used for this third paragraph. </p>

</body>
</html>
```

Output:

Graphic Era University

Example of Inline Style in CSS

Inline Styles [color:blue; font-size:18pt; background:yellow]
used for this first paragraph.

Inline Styles [color:red; font-size:14pt; font-weight:bold] used for second
paragraph.

No Style used for this third paragraph.

Figure 3.2: Example of Inline CSS Style with output

Advantages of Inline Style:

- It is quick, easy, and simple to apply CSS styles to a single element in an XHTML document.
- If a unique style is to be applied to a particular/single element, inline styles are very useful.

Disadvantages of Inline Style:

- Styles cannot be re-used by other elements as it is applied to a single element and appear within the opening tag of an XHTML element.
- XHTML markup and inline CSS styles are not separated, they are embedded in the XHTML document hence it becomes difficult to differentiate between the attributes and styles.
- It is difficult to debug/maintain when changes are required.
- It is not possible to use inline CSS with pseudo-classes
- Quotation marks cannot be used in inline style as it is interpreted as end of style by the browser

3.5.2 Document/Internal Level Style

Internal/Document level styles appear in the <head> section within the <style> tag. The styles given here, apply to all the elements within the document that satisfy the selection rules.

Syntax :

```
<head>
    <style type ="text/css">
        Selector  {
            cssproperty1:value1 ;
            cssproperty2:value2 ;
            ...
            csspropertyN:valueN;
        }
    </style>
</head>
```

Example:

```
<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE html>
<html>
<head>
    <title>Example of Internal/document level Style in CSS</title>
    <style type="text/css">

        /* Internal/document level style */
        p {
            color: blue;
            font-weight: bold;
            font-style: italic ;
        }
    </style>
</head>
<body>
    <h2> Graphic Era University </h2>
    <h3> Example of Internal/Document level Style in CSS </h3>
    <p>
        Internal/document Styles used for this first paragraph.
    </p>

    <p style="color:red; font-size:14pt; font-style: normal">
        Inline Style property [font-style:normal] overrides the
        document level for second paragraph.
    </p>

    <h4>No Style used for this heading </h4>
    <p>
        Internal/document Styles used for this third paragraph.
    </p>
</body>
</html>
```

Output:

Graphic Era University

Example of Internal/Document level Style in CSS

Internal/document Styles used for this first paragraph.

Inline Style property [font-style:normal] overrides the document level for second paragraph.

No Style used for this heading

Internal/document Styles used for this third paragraph.

Figure 3.3: Example of Internal/Document CSS Style with output

Advantages of Internal Style:

- All styles appear within the <style> tag in the <head> section of an XHTML document, hence it is easy, and simple to apply CSS styles to all elements in an XHTML document that satisfy the selection rule.
- It ensures that all elements that satisfy the selection rule are uniform in style in look.
- Since internal styles are present in the same XHTML document, there is no need to upload multiple files, thereby saving time.
- Any modification required can be easily made as there is no need to search in multiple files and the styling code is available in one place in the <head> section.

Disadvantages of Internal Style:

- The Internal styles apply only to the elements within the document, they cannot be used across multiple web pages.
- Style cannot be re-used, if the same style is to be used in multiple pages then the internal styling code has to be repeated in every page that requires it.
- When developing large web applications, repeating the code on every page may be messy and time-consuming.

3.5.3 External CSS

External CSS Styles are not embedded in the XHTML document. They are placed in a separate “.css” file.

There are 2 ways of referencing the external style sheets. By using either

- <link> tag
- @import directive

Link tag/element:

The <link> tag/element is used to link the external style sheets. In the XHTML document, the <link> tag/element must appear in the <head> section.

The <link> tag attributes

- **type="text/css"** is used to specify the MIME type
- **rel="stylesheet"** is used to specify the relationship between the external CSS sheet (linked-to) document and XHTML document in which the link appears.
- **href="filename.css"** is used to fetch the external CSS file.

Syntax :

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

Example:

“EgExternal.css”

```
p {
    color: blue;
    font-weight: bold;
    font-style: italic;
    font-size: 18pt;
}
```

Figure 3.4: Example Code of External CSS Style

XHTML document

```
<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE html>
<html>
<head>
    <title>Example of External Styles in CSS (Using link tag )</title>

    <!-- Use of link tag -->
    <link rel="stylesheet" href="EgExternal.css" />

</head>
<body>
    <h2> Graphic Era University </h2>
    <h3> Example of External CSS using "link" tag </h3>

    <p>
        External CSS Styles used for this first paragraph.
    </p>

    <p style="color:black; background:cyan; font-style: normal">
        Inline Style properties [font-style, background, color]
        override the external CSS level styles for the second
        paragraph.
    </p>

    <h4>No Style used for this heading </h4>

    <p>
        External CSS Styles used for this third paragraph.
    </p>

</body>
</html>
```

Output:

```
Graphic Era University
Example of External CSS using "link" tag
External CSS Styles used for this first paragraph.

Inline Style properties [font-style,background,color] override the external css level styles for the second paragraph.

No Style used for this heading

External CSS Styles used for this third paragraph.
```

Figure 3.5: Example of External CSS Style (<link> tag) with Output

@import directive

The alternate way of referencing an external style sheet within an XHTML document is by using the @import directive. The @import directive must appear in the <head> section and within the <style> tag.

Syntax :

```
<head>
  <style type="text/css">
    @import url(filename.css)
  </style>
</head>
```

The difference between <link> and @import directive is

1. @import directive should be placed only at the beginning of the content of the <style> tag. (In other words, it should be placed on the first line within the <style> tag)
2. Filename should not be quoted.
3. The imported file can contain CSS styles as well as XHTML markup and even @import directive.

Example

```
<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE html>
<html>
<head>
```

```

<title>Example of External Styles in CSS
    (Using @import directive )
</title>
<!-- Use of @import directive -->
<style>
    @import url(EgExternal.css);
</style>
</head>
<body>
    <h2> Graphic Era University </h2>
    <h3> Example of External CSS using "import directive" </h3>
    <p>
        External CSS Styles used for this first paragraph.
    </p>
    <p style="font-style:normal; background:yellow">
        Inline Style properties [font-style, background]
        override the external CSS level styles for the second
        paragraph.
    </p>
    <h4>No Style used for this heading </h4>
    <p>
        External CSS Styles used for this third paragraph.
    </p>
</body>
</html>

```

Output:

```

Graphic Era University
Example of External CSS using "import directive"
External CSS Styles used for this first paragraph.
Inline Style properties [font-style, background] override the external css level styles for the second paragraph.
No Style used for this heading
External CSS Styles used for this third paragraph.

```

Figure 3.6: Example of External CSS Style (@import directive) with Output

Advantages of External Style Sheet :

- Reusability: The styles defined in an external CSS file can be reused across multiple XHTML pages, promoting consistency in design.
- Separation of concern: Since CSS and XHTML files are separate, markup code can CSS styles be easily identified, making code and styles more organized and maintainable.
- Saves time: When working on large projects, developers can work separately and simultaneously on XHTML as well as on CSS.

Disadvantages of External Style Sheets

- Initial loading is slower: External CSS files require additional HTTP requests, which may lead to an increase in initial loading time.
- Maintenance can be difficult at times: Since external CSS files are used across multiple pages, minor changes required in one of the pages can create problems for others. Editing an external CSS file may be less convenient compared to inline styles in such cases.

3.6 Simple example demonstrating the use of CSS in XHTML

Example 1: Here is an example of using CSS at various levels in an XHTML document and overriding the behavior based on their precedence.

External CSS file -- “**cssexternal.css**”

```
p.p3 {  
    font-size:20pt;  
    font-weight: bold;  
    color: blue;  
}  
#p4 {  
    color: green;  
    font-size:18pt  
}
```

Figure 3.7: Example of External CSS Styles

XHTML document :

```
<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE html>
<html>
    <head>
        <style type="text/css">
            @import url(cssexternal.css);

            .p1 {
                color: blue;
                font-size:16pt;
                font-weight: bold;
            }
            p.p3 {
                font-size:24pt;
                font-style: italic;
                color: red;
                text-decoration: underline;
            }
            #p4 {
                color:#0000FF;
            }
            p.p5 {
                font-size:28pt;
                text-decoration: line-through;
            }
        </style>
    </head>
    <body>
        <p style="color:red;background-color:yellow;font-size:20pt">
            Welcome to GEU (inline style)
        </p>
        <h3 class="p1" style="color:red; font-size:18pt; ">
```

```

Example of style levels in CSS

</h3>

<p class="p1"> Document Level Style used here

</p>

<p class="p3"> External Style used here

</p>

<p id="p4" style="font-size:12pt"class="p5"> Text decoration used here

</p>
</body>
</html>

```

Figure 3.8: Example of CSS Styles

Output:

Welcome to GEU (inline style)

Example of style levels in CSS

Document Level Style used here

External Style used here

Levels and precedence [inline used here]
 (inline level has precedence over document and document has precedence over external)

Text decoration used here

Figure 3.9: Output of various CSS Styles used in XHTML document

Example 2: Here is an example of using CSS and displaying images and text.

```
<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE html>
<html>
<head>
    <title>Example of Floating images using CSS </title>
    <style type="text/css">
        img {
            float: right;
        }

        .p1 {
            width:400px;
            color:blue;
            font-weight:bold;
            text-align:justify;
        }
    </style>
</head>
<body>
    <h2> Graphic Era University </h2>
    <h3> Example of Floating images using CSS </h3>

    <p class="p1" >
        Image displayed to the right using float property. <br/>
        The aim of this course is to impart the skills and knowledge
        necessary to build simple web applications as well as full-stack
        web applications using modern and scalable web technologies
        and increase employability as a full-stack developer.
        The course includes the basics of HTML, CSS, JavaScript, PHP, and
        the basic components of Full Stack development using MERN stack
        widely used in the industry for developing web pages. The use of
        XHTML and CSS for developing presentable web pages are included.
    </p>
</body>
```

The graduates are expected to be able to create dynamic web pages by applying event-handling mechanisms using JavaScript. Students are taught the concepts of cookies and sessions in PHP for creating large web applications.

</body>

</html>

Output:

Graphic Era University

Example of Floating images using CSS

Image displayed to the right using float property.

The aim of this course is to impart the skills and knowledge necessary to build simple web applications as well as full-stack web applications using modern and scalable web technologies and increase employability as a full-stack developer. The course includes the basics of HTML, CSS, JavaScript, PHP, and the basic components of Full Stack development using MERN stack widely used in the industry for developing web pages. The use of XHTML and CSS for developing presentable web pages are included. The graduates are expected to be able to create dynamic web pages by applying event-handling mechanisms using JavaScript. Students are taught the concepts of cookies and sessions in PHP for creating large web applications.



Figure 3.10: Example of Floating image to the right using CSS Styles

3.7 Self-Assessment Questions

- Q1. What is CSS? Explain the Key aspects of CSS.[8 marks, L2]
- Q2. Discuss the CSS property value forms in brief. [6 marks, L2]
- Q3. Explain the CSS properties – font, color, text, and background. [8 marks, L2]
- Q4. What is a selector in CSS? List and explain the different types of selectors with appropriate examples for each. [8 marks, L2]
- Q5. Differentiate between Class and Generic Selectors in CSS. [6 marks, L2]
- Q6. Explain the levels of CSS with examples for each. [8 marks, L1]
- Q7. Write the required code in XHTML and CSS to demonstrate the levels of CSS and their precedence. [8 marks, L1]
- Q8. Write the required code in XHTML and CSS to demonstrate the use of pseudo-classes In CSS.[6 marks, L1]
- Q9. Explain with a suitable example the use of element selector and ID selector.
[1 mark, L1]
- Q10. Write the required code in XHTML and CSS to demonstrate the use of font properties
In CSS.[6 marks, L1]

3.8 Self-Assessment Activities

- A1. Design a web page to demonstrate the levels of CSS.
- A2. Design a web page to demonstrate the use of various types of selectors in CSS.

3.9 Multiple-Choice Questions

1. CSS in Web development is an acronym for _____. [1 mark, L1]
 - a) Complete Styling System
 - b) Cascading Style Sheet
 - c) Cascading Service Styles
 - d) None of the above
2. The purpose of using CSS in Web development is _____. [1 mark, L1]
 - a) To manage client-side scripts
 - b) To manage server-side scripts
 - c) To store information in the database
 - d) To enhance the user interface with styles
3. The CSS property that is used to change the color of text is _____. [1 mark, L1]
 - a) color

- b) text-style
 - c) text-color
 - d) font-color
4. The CSS property that is used to control the size of the text in an element is _____. [1 mark, L1]
- a) font-style
 - b) font-size
 - c) font-family
 - d) text-size
5. The style level in CSS that is used for applying styles directly within an HTML element is _____. [1 mark, L1]
- a) External
 - b) Internal
 - c) Inline
 - d) Intrinsic
6. The primary advantage of using an external style sheet in CSS is _____. [1 mark, L1]
- a) It allows for faster page rendering.
 - b) It simplifies HTML code.
 - c) It provides greater control over individual elements.
 - d) It promotes code reusability across multiple pages.
7. In CSS, a selector is used for _____. [1 mark, L1]
- a) Selecting a programming language
 - b) Selecting HTML elements to apply styles
 - c) Selecting a font for the webpage
 - d) Selecting a color palette for the website
8. The CSS selector that is used to target an element by its unique identifier is _____. [1 mark, L1]
- a) Class Selector
 - b) Generic Selector
 - c) ID Selector
 - d) Element Selector

9. The purpose of the “:hover” pseudo-class in CSS is, that _____. [1 mark, L1]
- a) It selects elements when the mouse pointer is over them.
 - b) It selects elements with a specific ID.
 - c) It selects elements when they are clicked.
 - d) It selects the first element of a specific type.
10. _____ of the following is NOT a valid use of CSS. [1 mark, L1]
- a) Styling the background color of a webpage
 - b) Controlling the behavior of JavaScript functions
 - c) Adjusting the spacing between paragraphs
 - d) Changing the font size of text elements

3.10 Key Answers to Multiple-Choice Questions

1. CSS in Web development is an acronym for Cascading Style Sheets. [b]
2. The purpose of using CSS in Web development is to enhance the user interface with styles. [d]
3. The CSS property that is used to change the color of text is color. [a]
4. The CSS property that is used to control the size of the text in an element is font-size. [b]
5. The style level in CSS that is used for applying styles directly within an HTML element is inline. [c]
6. The primary advantage of using an external style sheet in CSS is it promotes code reusability across multiple pages. [d]
7. In CSS, a selector is used for selecting HTML elements to apply styles. [b]
8. The CSS selector that is used to target an element by its unique identifier is ID Selector. [c]
9. The purpose of the “:hover” pseudo-class in CSS is, that it selects elements when the mouse pointer is over them. [a]
10. Controlling the behavior of JavaScript functions is NOT a valid use of CSS. [b]

3.11 Summary

CSS is an acronym for Cascading Style Sheets. The appearance of the web pages and user interface can be controlled by the web developers by specifying how and where the elements should be displayed. CSS allows to add styles to a web document for change of color, font size, background color, foreground color, spacing, and positioning of the elements, sizing of columns and their layout and many more.

A Style is a set of formatting instructions that can be applied to a piece of text for better presentation. If no styles are specified the browser's default property values are used.

The Key aspects of CSS: - Separation of presentation and content, Selectors and Rules/Properties, Cascading, External, Internal, and Inline Styles, Cross-Browser Compatibility, Responsive Design, Frameworks and Libraries, Re-usability, etc.

The property values can be given in various forms – keywords, number values, length values, percentage values, url properties.

CSS Selectors are used to select the content that is to be styled and presented as per the user's specifications instead of using the default browser setting. There are various types of selectors that can be used for setting the styles. They are - Simple Selectors, Class Selectors, Generic Selectors, ID Selectors, Universal Selectors, Pseudo Classes, etc.

In the XHTML document, CSS can be added in 3 ways and they are called levels in CSS.

There are 3 types/levels of styles. Inline, Document, and External levels

Inline level - It is specified by using `<style>` attribute inside XHTML elements.

Internal/Document Level - It is specified by using the `<style>` element in the `<head>` section.

External Level - It is specified in the `<head>` by using `<link>` element and referencing the external CSS file by its href attribute.

3.12 Keywords

- Cascading Style Sheet (CSS)
- Inline
- Internal/Document level
- External level
- Selectors
- Class Selector
- Generic Selector
- ID Selector
- Universal Selector
- Pseudo class
- Fonts
- Style tag
- Link tag
- Properties

3.13 Recommended resources for further reading

a. Essential Reading

1. Sebesta, R. W. (2010). Programming the World Wide Web (6th ed.), Pearson education.
2. Subramanian, V. (2019). Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node (2nd Ed.), Apress.

b. Recommended Reading

1. DT Editorial Services. (2016). HTML 5: Covers CSS3, JavaScript, XML, XHTML, AJAX, PHP & jQuery: Black Book, Dreamtech Press.
2. Koroliova, E. W. I., (2018). MERN Quick Start Guide: Build Web applications with MongoDB, Express.js, React and Node, Packt.

--*--