STYLE SHEETS
CSS
(Cascading Style Sheet)

By : Bhoomi D. Patel

Assistant Professor

DEPARTMENT OF QUALITY ASSURANCE

SARASWATI INSTITUTE OF PHARMACEUTICAL SCIENCES

Outline

- Need for CSS, introduction to CSS
- Basic syntax and structure.
- Using CSS
  - Background images, colors and properties.
  - Manipulating texts
- Using fonts, borders and boxes.
- Margins, padding lists.
- Positioning using CSS
- CSS2
- Overview and features of CSS3

Introduction of CSS

- CSS is used to control the style of a web document in a simple and easy way.
- CSS is "Cascading Style Sheet".
- **C**ascading **S**tyle **S**heets, fondly referred to as CSS, is a simple design language intended to simplify the process of making web pages presentable.
- CSS is easy to learn and understand but it provides powerful control over the presentation of an HTML document.
- Most commonly, CSS is combined with the markup languages HTML or XHTML.

Need for CSS (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.
- The style definitions are normally saved in external .css files.
- With an external style sheet file, you can change the look of an entire website by changing just one file!

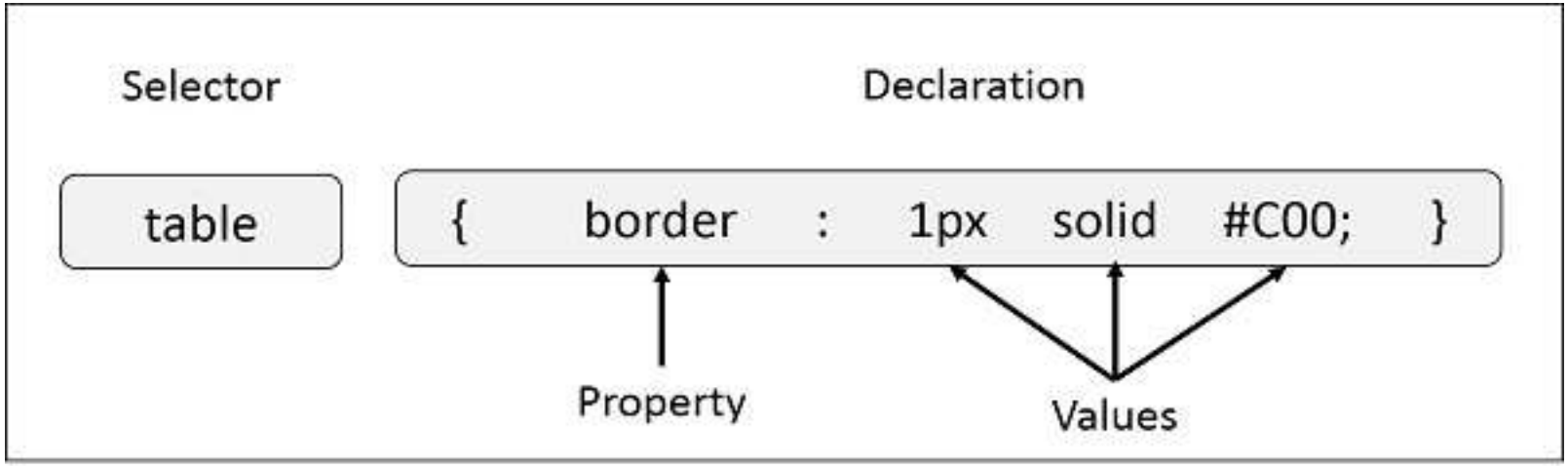
CSS - Syntax

- A style rule is made of three parts −
  - **Selector** − A selector is an **HTML tag** at which a style will be applied. This could be any tag like <h1> or <table> etc.
  - **Property** - A property is a type of attribute of HTML tag. Put simply, all the HTML attributes are converted into CSS properties. They could be *color*, *border* etc.
  - **Value** - Values are assigned to properties. For example, *color* property can have value either *red* or *#F1F1F1* etc.

CSS - Syntax

- You can put CSS Style Rule Syntax as follows –

  selector { property: value }



- You can define a table border as follows –

```
Table { border : 1px  solid  #C00; }
```

CSS - Syntax

- 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.
- A CSS declaration always ends with a semicolon, and declaration blocks are surrounded by curly braces

CSS Selectors

- CSS selectors are used to "find" (or select) HTML elements based on their element name, id, class, attribute, and more.
- **The element Selector:**
  - The element selector selects elements based on the element name.
  - You can select all <p> elements on a page like this (in this case, all <p> elements will be center-aligned, with a red text color)
  - ```
    P {
        text-align: center;
        color: red;
      }
    ```
    ```

CSS Selectors

- **The id Selector :**
  - The id selector uses the id attribute of an HTML element to select a specific element.
  - The id of an element should be 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 style rule below will be applied to the HTML element with id="para1":
  - ```
    #para1 {
        text-align: center;
        color: red;
    }
    ```
    ```

CSS Selectors - The class Selector

- The class selector selects elements with a specific class attribute.
- To select elements with a specific class, write a period (.) character, followed by the name of the class.
- In the example below, all HTML elements with class="center" will be red and center-aligned:
- ```
  .center{
      text-align: center;
      color: red;
  }
  ```
  ```

CSS Selectors - The class Selector

- You can also specify that only specific HTML elements should be affected by a class.
- In the example below, only <p> elements with class="center" will be center-aligned:
- ```
  p.center{
      text-align: center;
      color: red;
  }
  ```
  ```

CSS Selectors - The class Selector

- HTML elements can also refer to more than one class.
- In the example below, the <p> element will be styled according to class="center" and to class="large":
- ```
  p.center{
      text-align: center;
      color: red;
  }
  p.large{
      font-size: 300%;
  }
  ```
  ```

CSS Selectors - Grouping Selectors

- If you have elements with the same style definitions, like this:
- ```
  h1{
      text-align: center;
      color: red;
  }

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

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

CSS Selectors - Grouping Selectors

- It will be better to group the selectors, to minimize the code.
- To group selectors, separate each selector with a comma.
- In the example below we have grouped the selectors from the code above:
- ```
  h1, h2, p {
      text-align: center;
      color: red;
  }
  ```
  ```

Using CSS - CSS How To...

- 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.
  1. External style sheet
  2. Internal style sheet
  3. Inline style

External Style Sheet

- With an external style sheet, you can change the look of an entire website by changing just one file!
- Each page must include a reference to the external style sheet file inside the <link> element. The <link> element goes inside the <head> section:
- Example
- ```
  <head>
  <link rel="stylesheet" type="text/css" href="
  mystyle.css">
  </head>
  ```
  ```

External Style Sheet

- An external style sheet can be written in any text editor.
- The file should not contain any html tags.
- The style sheet file must be saved with a .css extension.
- ```
  body{
      background-color: lightblue;
  }

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

Internal Style Sheet

- An internal style sheet may be used if one single page has a unique style.
- Internal styles are defined within the <style> element, inside the <head> section of an HTML page:
- Example

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

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

Inline Styles

- 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.
- The example below shows how to change the color and the left margin of a <h1> element:

- `<h1 style="color:blue;margin-left:30px;">This is a heading.</h1>`

Cascading Order

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

- Generally speaking we can say that all the styles 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

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.
- **Example**
- 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;
  }
  ```

Multiple Style Sheets

- If the internal style is defined after the link to the external style sheet, the <h1> elements will be "orange":
- ```
  <head>
  <link rel="stylesheet" type="text/css" href="
  mystyle.css">
  <style>
  h1 {
      color: orange;
  }
  </style>
  </head>
  ```
  ```

CSS Colors

- Colors in CSS are most often specified by:
  - a valid color name - like "red"
  - an RGB value - like "rgb(255, 0, 0)"
  - a HEX value - like "#ff0000“
- Color names are case-insensitive: "Red" is the same as "red" or "RED".
- **Hexadecimal Colors**
- RGB values can also be specified using **hexadecimal** color values in the form: *#RRGGBB*, where RR (red), GG (green) and BB (blue) are hexadecimal values between 00 and FF (same as decimal 0-255).
- For example, #FF0000 is displayed as red, because red is set to its highest value (FF) and the others are set to the lowest value (00).
- **Note:** HEX values are case-insensitive: "#ff0000" is the same as "FF0000".

CSS Backgrounds

- The CSS background properties are used to define the background effects for elements.
- CSS background properties:
  - background-color
  - background-image
  - background-repeat
  - background-attachment
  - background-position

Background Color

- The background-color property specifies the background color of an element.
- The background color of a page is set like this:
- ```
  body {
      background-color: lightblue;
  }
  ```
- The <h1>, <p>, and <div> elements have different background colors:
  ```

Background Image

- The background-image property specifies an image to use as the background of an element.
- By default, the image is repeated so it covers the entire element.
- The background image for a page can be set like this:
- ```
  body{
      background-image: url("digishala.jpg");
  }
  ```
  ```

Background Image - Repeat Horizontally or Vertically

- Some images should be repeated only horizontally or vertically, or they will look strange,
- repeated only horizontally (background-repeat: repeat-x;),
-
```
body{
    background-image: url(" digishala.jpg ");
    background-repeat: repeat-x;
}
```

All CSS Background Properties

Property	Description
background	Sets all the background properties in one declaration
background-attachment	Sets whether a background image is fixed or scrolls with the rest of the page
background-color	Sets the background color of an element
background-image	Sets the background image for an element
background-position	Sets the starting position of a background image
background-repeat	Sets how a background image will be repeated

CSS Border Properties

- The CSS border properties allow you to specify the style, width, and color of an element's border.

- **Border Style:**
- The border-style property specifies what kind of border to display.

- The following values are allowed:

Border Style

- dotted - Defines a dotted border
- dashed - Defines a dashed border
- solid - Defines a solid border
- double - Defines a double border
- groove - Defines a 3D grooved border. The effect depends on the border-color value
- ridge - Defines a 3D ridged border. The effect depends on the border-color value
- inset - Defines a 3D inset border. The effect depends on the border-color value
- outset - Defines a 3D outset border. The effect depends on the border-color value
- none - Defines no border
- hidden - Defines a hidden border

Border Width

- The border-width property specifies the width of the four borders.
- The width can be set as a specific size (in px, pt, cm, em, etc) or by using one of the three pre-defined values: thin, medium, or thick.
- The border-width property can have from one to four values (for the top border, right border, bottom border, and the left border).
- **The "border-width" property does not work if it is used alone. Always specify the "border-style" property to set the borders first.**

Border Width

```
p.one {
    border-style: solid;
    border-width: 5px;
}

p.two {
    border-style: solid;
    border-width: medium;
}

p.three {
    border-style: solid;
    border-width: 2px 10px 4px 20px;
}
```

Border Color

- The border-color property is used to set the color of the four borders.
- The border-color property can have from one to four values (for the top border, right border, bottom border, and the left border).
- If border-color is not set, it inherits the color of the element.

Border - Individual Sides

- It is possible to specify a different border for each side.
- In CSS, there is also properties for specifying each of the borders (top, right, bottom, and left):
- ```
  p {
      border-top-style: dotted;
      border-right-style: solid;
      border-bottom-style: dotted;
      border-left-style: solid;
  }
  ```
  ```

CSS Border

- If the border-style property has **four** values:
- **border-style: dotted solid double dashed;**
  - top border is dotted
  - right border is solid
  - bottom border is double
  - left border is dashed
- If the border-style property has **three** values:
- **border-style: dotted solid double;**
  - top border is dotted
  - right and left borders are solid
  - bottom border is double

CSS Border

- If the border-style property has two values:
- **border-style: dotted solid;**
  - top and bottom borders are dotted
  - right and left borders are solid

Rounded Borders

- The border-radius property is used to add rounded borders to an element.
- Example
- ```
  p {
      border: 2px solid red;
      border-radius: 5px;
  }
  ```
  ```

CSS Layout - The position Property

- The position property specifies the type of positioning method used for an element.
- There are four different position values:
  - static
  - relative
  - fixed
  - Absolute
- Elements are then positioned using the top, bottom, left, and right properties.
- However, these properties will not work unless the **position property is set first.** They also work differently depending on the position value.

CSS Dropdowns

- Create a hover able dropdown with CSS.
- Basic Dropdown
  - Create a dropdown box that appears when the user moves the mouse over an element.

CSS3 Modules

- CSS3 has been split into "modules". It contains the "old CSS specification" (which has been split into smaller pieces). In addition, new modules are added.
- Some of the most important CSS3 modules are:
  - Selectors
  - Box Model
  - Backgrounds and Borders
  - Image Values and Replaced Content
  - Text Effects
  - 2D/3D Transformations
  - Animations
  - Multiple Column Layout
  - User Interface


Thank-You