

Web Technology

# Unit - 1

## Cascading Style Sheets (CSS)



# Outline

---

## 1. Introduction to CSS

- What is CSS?
- Importance of CSS

## 2. Basics of CSS

- Basic Syntax & Structure
- Class & ID
- Types of CSS
- Multiple selector, Multilevel selector

## 3. Background

## 4. Fonts & Text

# Outline (Cont.)

---

## 5. Box Model

- Margin, Border, Padding

## 6. List

- List Type, List with Image, List Position

## 7. Links

## 8. CSS Positioning

## 9. CSS Layers

## 10. CSS Floating Property

## 11. Introduction to CSS3

# What is CSS?

---

- Cascading Style Sheets, fondly referred to as **CSS**, is a simple design language intended to **simplify** the process of making web pages **presentable**.
- CSS defines **layout of HTML** documents. For example, CSS covers Fonts, colors, margins, lines, height, width, background images, advanced positions and many other things.

# Importance of CSS

---

- CSS defines HOW HTML elements are to be displayed.
- Styles are normally saved in external .css files.
- External style sheets enable you to change the appearance and layout of all the pages in a Web site, just by editing one single file.
- Advantages :
  - Improves Website Presentation
  - External CSS makes Updates Easier and Smoother
  - External CSS helps Web Pages Load Faster
- Disadvantages :
  - Browser Dependent
  - Difficult to retrofit in old websites

# Basic Syntax of CSS

- A CSS rule has two main parts: a **selector**, and one or more **declarations**

**Selector**

**Declaration 1**

**Declaration 2**

**h1**

```
{color:blue; font-size: 12px;}
```

property

value

property

value

- The **selector** can be HTML element, id or class.
- Each **declaration** consists of a **property** and a **value**.
- The **property** is the style attribute you want to change. Each property has a **value**.

# The “id” selector

---

- The id selector is used to specify a style for a **single, unique** element.
- The id selector uses the id attribute of the HTML element, and is defined with a “#” in css.
- The style rule below will be applied to the element with **id="para1"**:

## HTML

```
<h1 id="para1">  
    Hello Friends  
</h1>  
  
<h1>  
    How are you  
</h1>
```

## CSS

```
#para1{  
    color: blue;  
}
```

## Output

```
Hello Friends  
How are you
```

# The “class” selector

- The class selector is used to specify a style for a **group** of elements.
- The class selector uses the HTML class attribute, and is defined with a “.” in css.

## HTML

```
<h1 class="myClass">  
    Hello Friends  
</h1>  
<h1>  
    How are you  
</h1>  
<h1 class="myClass">  
    How are you  
</h1>
```

## CSS

```
.myClass{  
    color: blue;  
}
```

## Output

```
Hello Friends  
How are you  
How are you
```



# Different ways to write CSS

---

- There are three ways of writing a style sheet:
  1. Inline Style
  2. Internal/Embedded Style sheet
  3. External Style Sheet

# 1) Inline Style

---

- It is possible to place CSS right in your HTML code, and this method of CSS usage is referred to as **inline css**.
- Inline CSS has the **highest priority** out of external, internal, and inline CSS.
- This means that you can **override styles** that are defined in external or internal by using inline CSS.
- If you want to add a style inside an HTML element all you have to do is specify the desired CSS properties with the **style** HTML attribute.
- Example:

## HTML

```
<p style="background: blue; color: white;"> My Inline CSS </p>
```

## 2) Internal Style Sheet

---

- This type of CSS is only for **Single Web Page**.
- When using internal CSS, we must add a new tag, **<style>**, inside the **<head>** tag.
- The HTML code below contains an example of **<style>**'s usage.

### HTML

```
<html><head>  
    <style type="text/css">  
        p{ color: red;}  
    </style>  
</head><body>  
    <p>Your page's content!</p></body>  
</html>
```

# 3) External Style Sheet

---

- When using CSS it is preferable to keep the **CSS separate from your HTML**.
- Placing CSS in a separate file allows the web designer to completely differentiate between content (HTML) and design (CSS).
- External CSS is a file that contains **only CSS** code and is saved with a **".css"** file extension.
- This CSS file is then referenced in your HTML using the **<link>** instead of **<style>**.

# 3) External Style Sheet (Cont.)

- Example :

## Demo.html

```
<html>
<head>
<link rel="stylesheet" type="text/css"
href="test.css">
</head>
<body>

<p> Hello Friends </p>
<p id="para1"> How are you? </p>

</body>
</html>
```

## test.css

```
#para1{
    text-align: center;
}
p
{
    color : blue;
}
```

## Output

```
Hello Friends
How are you?
```

# 3) External Style Sheet (Cont.)

---

- Advantages:
  - It keeps your website design and content separate.
  - It's much easier to reuse your CSS code if you have it in a separate file. Instead of typing the same CSS code on every web page you have, simply have many pages refer to a single CSS file with the "link" tag.
  - You can make drastic changes to your web pages with just a few changes in a single CSS file.

# Assign Multiple Classes

- We can apply different class to same html element by giving space separated class names in the class attribute:

## Demo.html

```
<html>
<head>
<link rel="stylesheet" type="text/css"
href="test.css">
</head>
<body>

<h1 class="class1 class2">
    How are you?
</h1>

</body>
</html>
```

## test.css

```
. class1
{
    color : blue;
}
. class2
{
    text-align : center;
}
```

## Output

How are you?

# Multiple Selection

- We can apply same css to multiple selectors using **comma separated** selector list, for example :

## Demo.html

```
<html>
<head>
<link rel="stylesheet" type="text/css"
href="test.css">
</head>
<body>

<p> Hello Friends </p>
<h1> How are you? </h1>

</body>
</html>
```

## test.css

```
p, h1
{
    color : blue;
}
```

## Output

```
Hello Friends
How are you?
```



# Multi-level Selection

- We can use hierarchical path to target html element by **space separated** element/class/id names, for example :

## Demo.html

```
<html>
<head>
<link rel="stylesheet" type="text/css"
href="test.css">
</head>
<body>
<h1>Hello Friends...</h1>
<div>
    <h1>How are you?</h1>
</div>

</body>
</html>
```

## test.css

```
div h1
{
    color : blue;
}
```

## Output

```
Hello Friends...
How are you?
```

# Background Property

---

	Property Name
■ Background Color	(background-color)
■ Background Image	(background-image)
■ Background Image Repeat	(background-repeat)
■ Fixed Background Image	(background-attachment)
■ Background Image Positioning	(background-position)

# Background Color

---

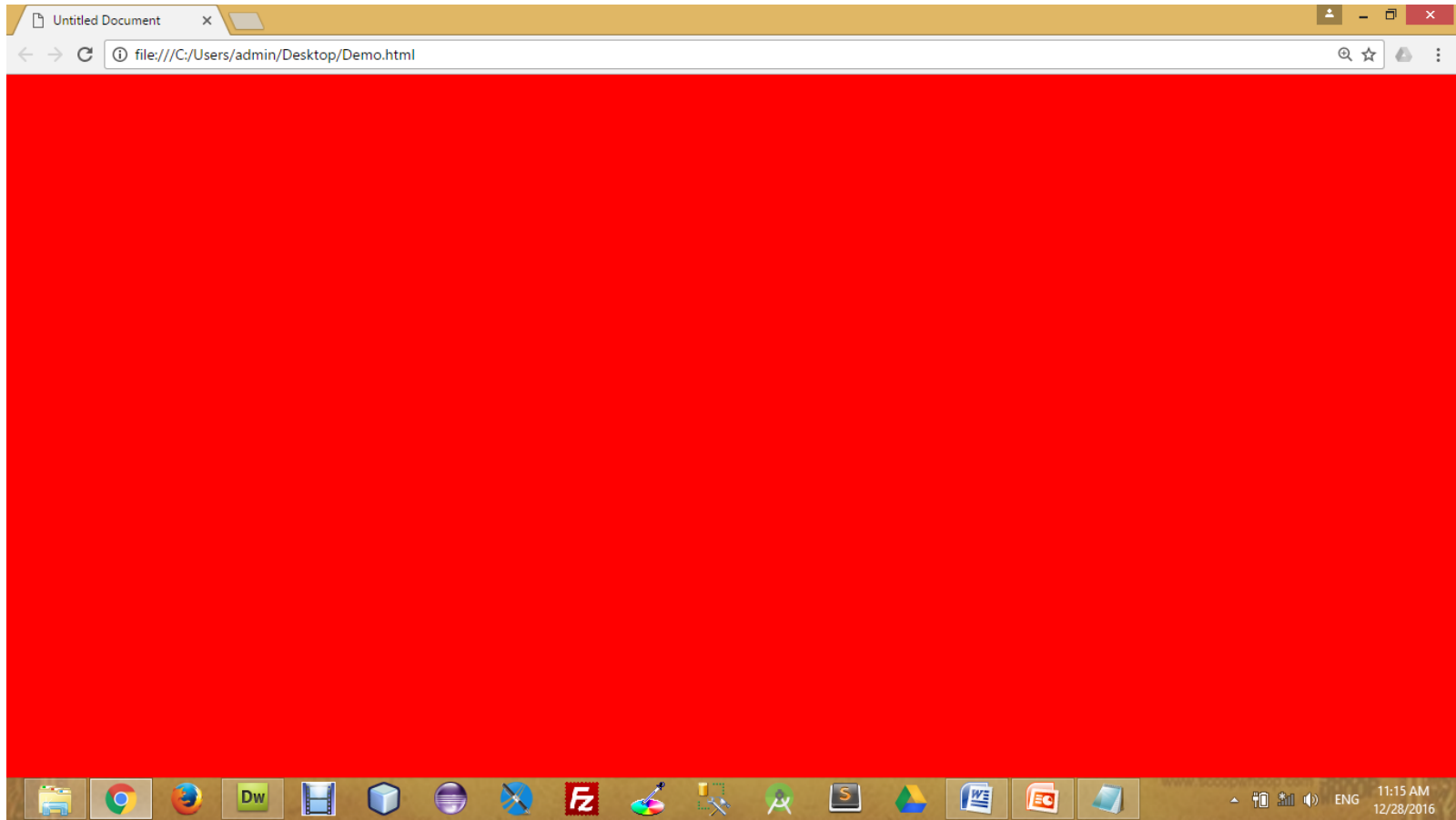
- The **background-color** property specifies the background color of an element.
- The background color of a page is defined in the body selector:
- Below is example of CSS backgrounds

**test.css**

```
body
{
    background-color : red;
    background-color : #FF0000;
    background-color : rgb(255,0,0);
}
```

# Background Color

---



# Background Image



# Background Image Repeat

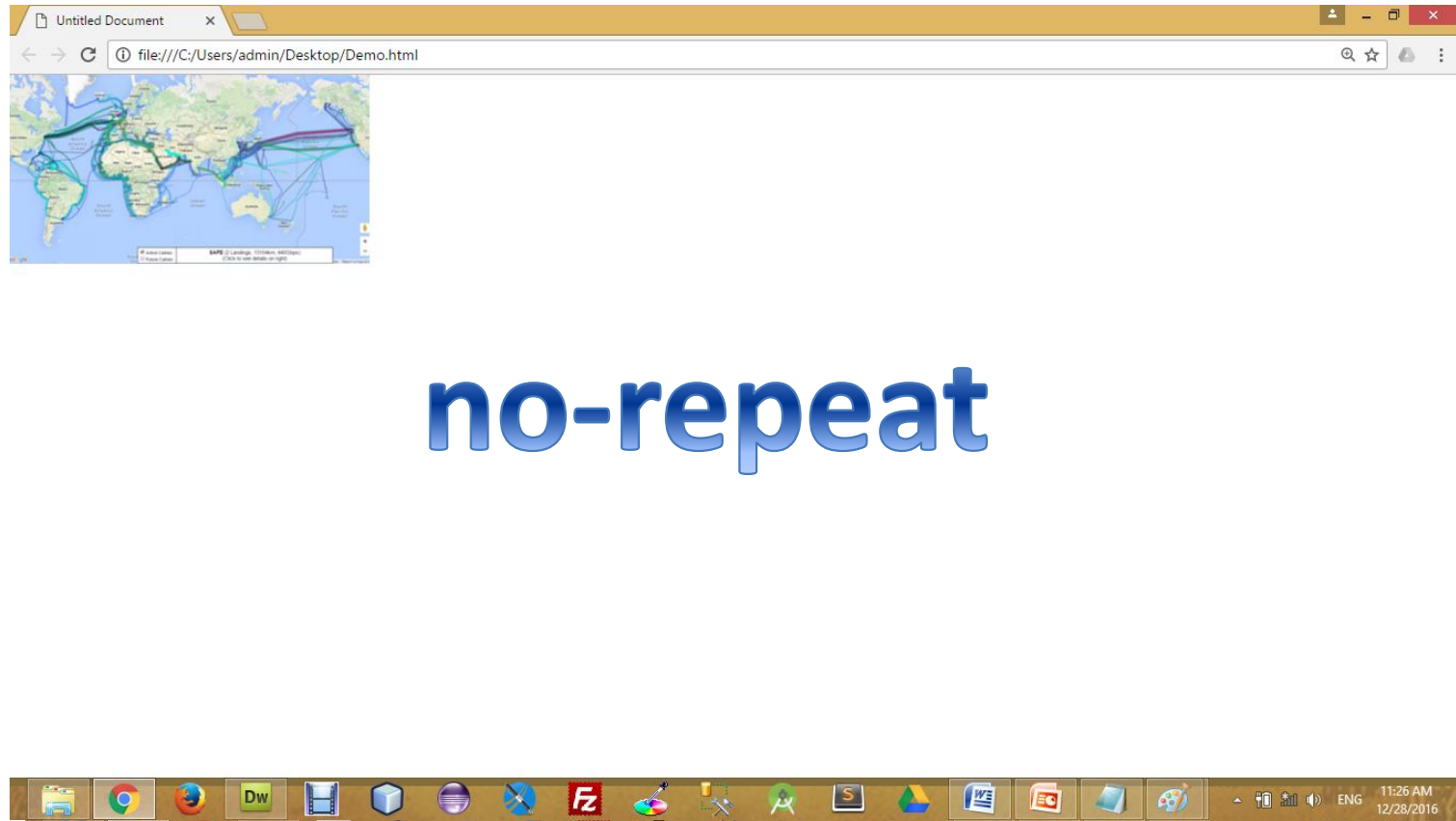
---

- You can have a background image repeat vertically (y-axis), horizontally (x-axis), in both directions, or in neither direction.

**test.css**

```
body
{
    background-image : url('pathToImage.jpg');
    background-repeat : repeat;
    background-repeat : repeat-x;
    background-repeat : repeat-y;
    background-repeat : no-repeat;
}
```

# Background Image Repeat



no-repeat

# Fixed Background Image

---

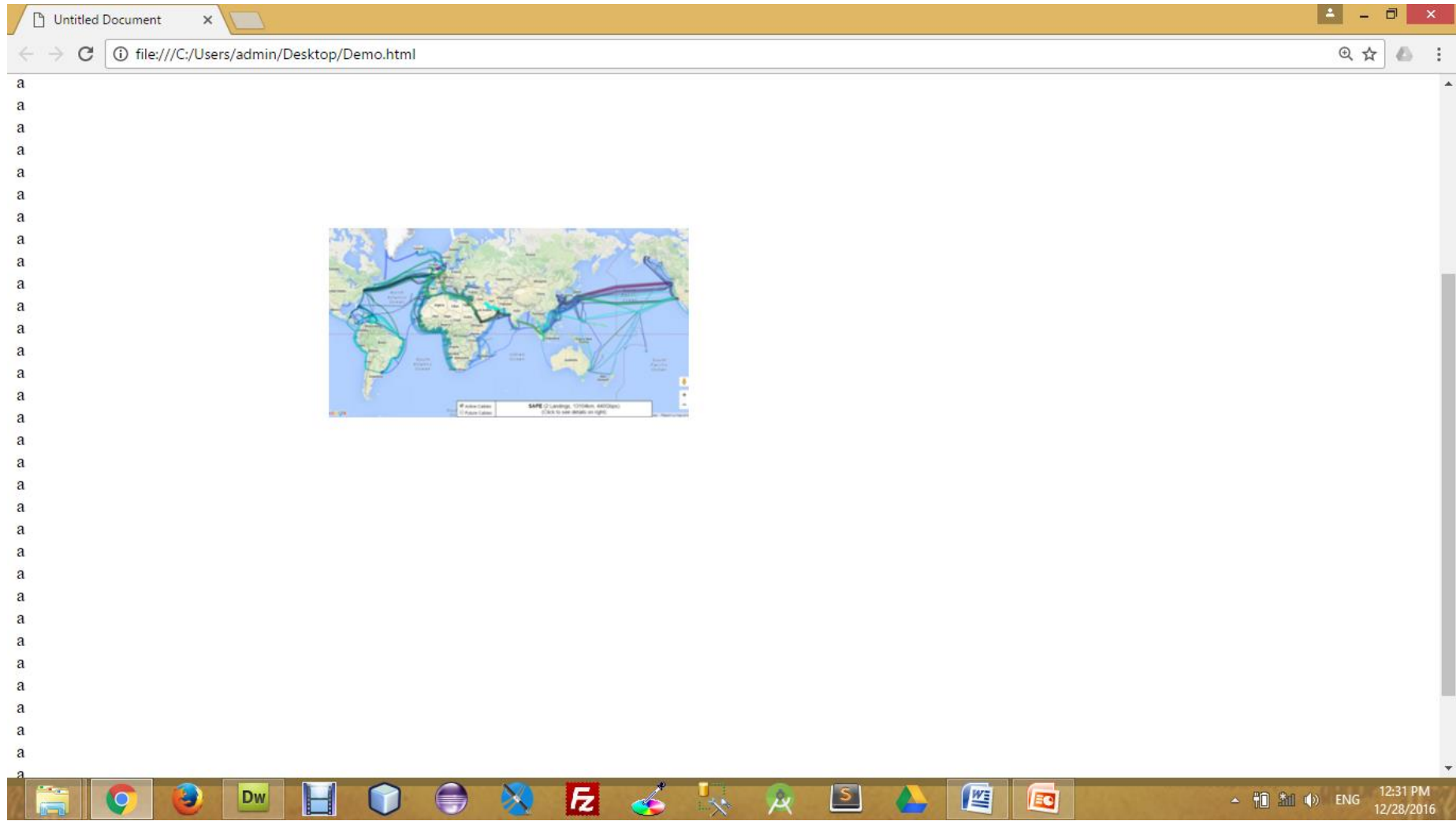
- The background-attachment property sets whether a background image is fixed or scrolls with the rest of the page.
- For Example,

**test.css**

```
body
{
    background-image : url('pathToImage.jpg');
    background-repeat : no-repeat;
    background-attachment : fixed;
}
```



# Background Image Positioning



# CSS Font

---

- CSS font properties define the font family, boldness, size, and the style of a text.

	Property Name
1. Font Color	(color)
2. Font Family	(font-family)
3. Font Size	(font-size)
4. Font Style	(font-style)
5. Font Weight	(font-weight)
6. Font Variant	(font-variant)

# CSS Font (Cont.)

## ■ Font Color

- Set the text-color for different elements

## ■ Font Family

- The font family of a text is set with the font-family property.

## ■ Font Size

- The font-size property sets the size of the text.
  - font-size : 120%
  - font-size : 10px;
  - font-size : x-large;

```
h4{  
    color : red;  
}
```

```
h4{  
    font-family : sans-serif;  
}
```

```
h4{  
    font-size: 120%;  
    font-size : 10px;  
    font-size : small;  
    font-size : smaller;  
    font-size : x-small;  
    font-size : xx-small;  
    font-size : large;  
    font-size : larger;  
    font-size : x-large;  
    font-size : xx-large;  
    font-size : medium;  
}
```

# CSS Font (Cont.)

## ■ Font Style

- The font-style property is mostly used to specify italic text.

```
h4{  
    font-style: italic ;  
}
```

## ■ Font Weight

- The font-weight property sets how thick or thin characters in text should be displayed.

```
h4{  
    font-weight : 300;  
    font-weight : bolder;  
    font-weight : lighter;  
}
```

## ■ Font Variant

- The font-variant property specifies whether or not a text should be displayed in a small-caps font.
  - font-variant : small-caps;

```
h4{  
    font-variant: small-caps;  
}
```

# CSS Text Property

---

- While CSS Font covers most of the traditional ways to format your text, CSS Text allows you to control the spacing, decoration, and alignment of your text.

	Property Name
1. Text Decoration	(text-decoration)
2. Text Indent	(text-indent)
3. Text Align	(text-align)
4. Text Transform	(text-transform)
5. White Space	(white-space)
6. Word Spacing	(word-spacing)
7. Letter Spacing	(letter-spacing)
8. Line Height	(line-height)

# CSS Text Property (Cont.)

## ■ Text Decoration

- The text-decoration property is used to set or remove decorations from text.
- The text-decoration property is mostly used to remove underlines from links for design purposes.

```
h4{
    text-decoration : line-through;
    text-decoration : overline;
    text-decoration : underline;
    text-decoration : none;
}
```

## ■ Text Indent

- The text-indentation property is used to specify the indentation of the first line of a text.

```
h4{
    text-indent : 20px;
    text-indent : 30%;
}
```

## ■ Text Align

- The text-align property is used to set the horizontal alignment of a text.

```
h4{
    text-align : right;
    text-align : justify;
    text-align : left;
    text-align : center;
}
```

# CSS Text Property (Cont.)

## ■ Text Transform

- The text-transform property is used to specify uppercase and lowercase letters in a text.

```
h4{
  text-transform : capitalize;
  text-transform : uppercase;
  text-transform : lowercase;
}
```

## ■ White Space

- The white-space attribute allows you to prevent text from wrapping until you place a break `<br />` into your text.

```
h4{
  white-space : nowrap;
}
```

## ■ Word Spacing

- With the CSS attribute word-spacing you are able to specify the exact value of the spacing between your words. Word-spacing should be defined with exact values.

```
h4{
  word-spacing : 10px;
}
```

# CSS Text Property (Cont.)

---

## ▪ Letter Spacing

- With the CSS attribute letter-spacing you are able to specify the exact value of the spacing between your letters. Letter-spacing should be defined with exact values.

```
h4{  
    letter-spacing : 3px;  
}
```

## ▪ Line Height

- The line-height attribute will set the height of the line in the page.

```
h4{  
    line-height : 10px;  
}
```



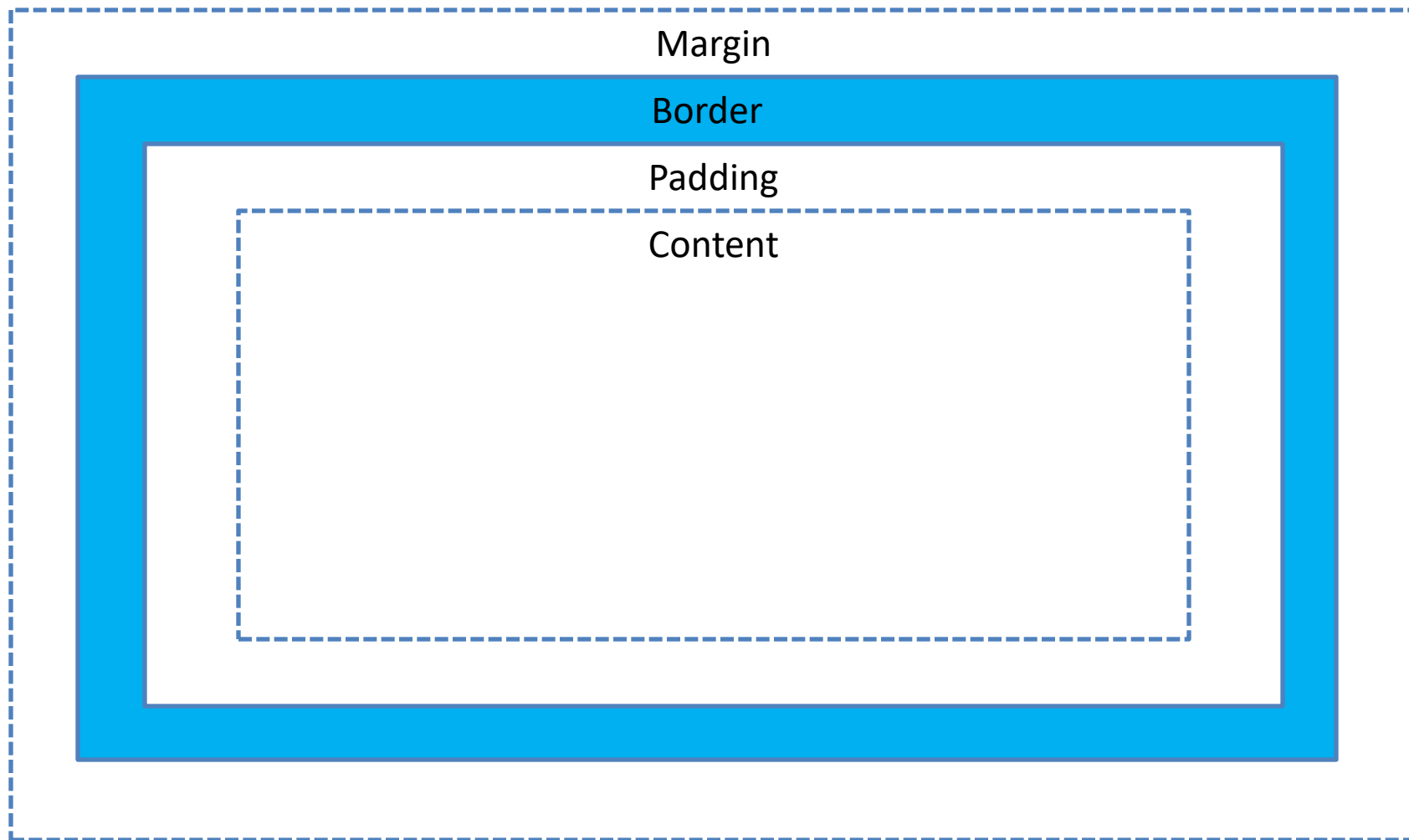
# The Box Model

---

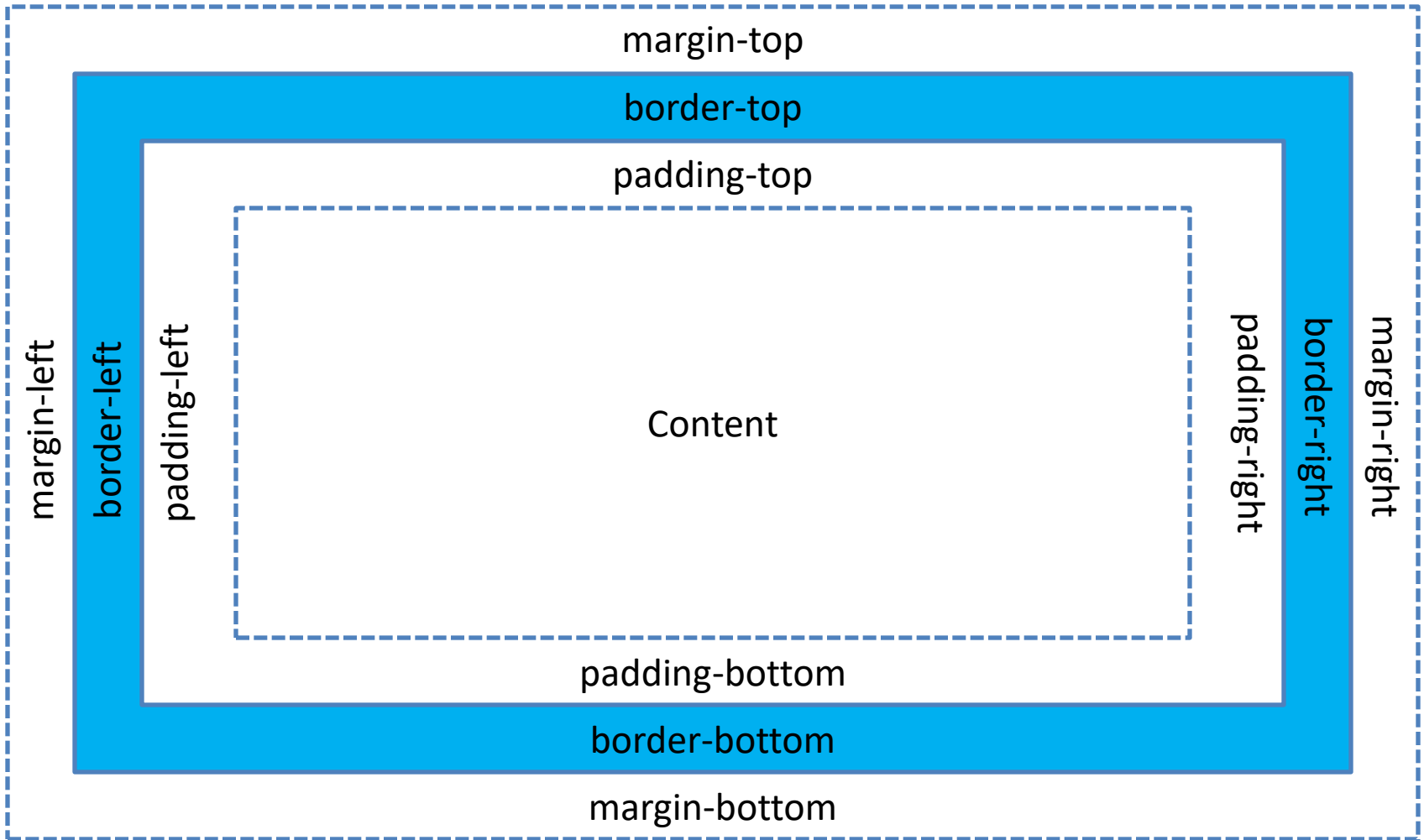
- All HTML elements can be considered as boxes. In CSS, the term "box model" is used when talking about design and layout.
- The CSS box model is essentially a box that wraps around HTML elements, and it consists of: **margins**, **borders**, **padding**, and the actual **content**.
- The box model allows us to place a border around elements and space elements in relation to other elements.

# The Box Model (Cont)

- The image below illustrates the box model:



# The Box Model (Cont)



# CSS Padding

- The CSS padding properties define the space between the element border and the element content.
- The top, right, bottom, and left padding can be changed independently using separate properties.
- A shorthand padding property can also be used, to change all padding at once.

```
h4{  
    padding : 10px;  
}
```

```
h4{  
    padding-top : 10px;  
    padding-right : 20px;  
    padding-bottom : 30 px;  
    padding-left : 40 px;  
}
```

```
h4{  
    padding : 10px 20px 30px 40px;  
}
```

# CSS Border

- The CSS border properties allow you to specify the style and color of an element's border.
- Border Style Types
  - The border-style property specifies what kind of border to display.
- Border Width
  - The border-width property is used to set the width of the border.
- Border Color
  - The border-color property is used to set the color of the border.
  - Border colors can be any color defined by RGB, hexadecimal, or key terms. Below is an example of each of these types.
- The top, right, bottom, and left border can be changed independently using separate properties.

```
h4{  
    border : 1px solid red;  
}
```

```
h4{  
    border-style : solid;  
    border-style : dotted;  
    border-style : double;  
}
```

```
h4{  
    border-width : 7px;  
}
```

```
h4{  
    border-color : red;  
}
```

```
h4{  
    border-top : 1px solid red;  
}
```

# CSS Margin

- The CSS margin properties define the space around elements
- The top, right, bottom, and left margin can be changed independently using separate properties.
- A shorthand margin property can also be used, to change all margins at once.

```
h4{  
    margin: 10px;  
}
```

```
h4{  
    margin -top : 10px;  
    margin -right : 20px;  
    margin -bottom : 30 px;  
    margin -left : 40 px;  
}
```

```
h4{  
    margin : 10px 20px 30px 40px;  
}
```

# CSS List

- The CSS list properties allow you to:
  - Set different list item markers for ordered & unordered lists
  - Set an image as the list item marker
  - Set the position of the marker
- CSS List Style Type
- CSS List with Image
- CSS List Position

```
ul{  
list-style-type: circle;  
list-style-type: disc;  
list-style-type: square;  
list-style-type: armenian;  
list-style-type: cjk-ideographic;  
list-style-type: decimal-leading-zero;  
list-style-type: georgian;  
list-style-type: hebrew;  
list-style-type: katakana;  
list-style-type: lower-greek;  
}
```

```
ol{  
list-style-image : url('imgPath');  
}
```

```
ol{  
list-style-position : outside;  
list-style-position : inside;  
}
```

# Styling Links

## ■ Anchor/Link States

- The **four** links states are:
  1. a:link - a normal, unvisited link
  2. a:visited - a link the user has visited
  3. a:hover - a link when the user mouse over it
  4. a:active - a link the moment it is clicked

```
a:link{  
    color:#FF0000;  
    /*unvisited link*/  
}
```

```
a:visited{  
    text-decoration : none;  
    /*visited link*/  
}
```

```
a:hover{  
    color:#00FF00;  
    /*mouse over link*/  
}
```

```
a:active{  
    color:#0000FF;  
    /*selected link*/  
}
```



# CSS Positioning

- Absolute Positioning
  - With absolute positioning, you define the exact pixel value where the specified HTML element will appear.
  - The point of origin is the top-left of the browser's viewable area, so be sure you are measuring from that point.
- Relative Positioning
  - Relative positioning changes the position of the HTML element relative to where it normally appears.
- Fixed Positioning
  - The element is positioned relative to the browser window, in fixed position, element will be in the same place even we scroll the screen.

```
h1{  
    position : absolute;  
    left : 50px;  
    top : 100px;  
}
```

```
h1{  
    position : relative;  
    left : 50px;  
    top : 100px;  
}
```

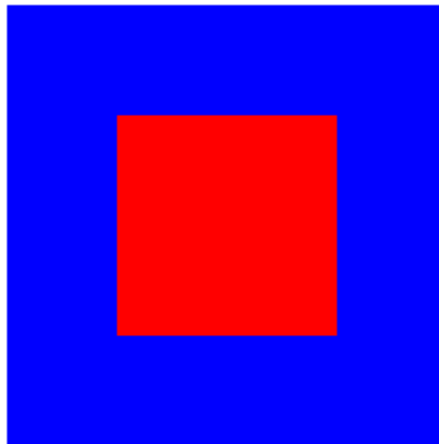
```
h1{  
    position : fixed;  
    top : 50px;  
    left : 100px;  
}
```

# CSS Layers

- CSS allows you to control which item will appear on top with the use of layers.
- In CSS, each element is given a priority.
- If there are two overlapping CSS positioned elements, the element with the higher priority will appear on top of the other.
- To manually define a priority, set the z-index value. The larger the value, the higher the priority the element will have.

## HTML

```
<div id="division1">  
</div>  
<div id="division2">  
</div>
```



## CSS

```
#division1{  
    position : absolute;  
    height : 100px;  
    width : 100px;  
    left : 100px;  
    top : 150px;  
    background-color : red;  
    z-index : 5;  
}  
#division2{  
    position : absolute;  
    height : 200px;  
    width : 200px;  
    left : 50px;  
    top : 100px;  
    background-color : blue;  
    z-index : 2;  
}
```

# CSS Float Property

- The CSS float property defines that an element should be taken out of the normal flow of the document and placed along the left or right side of its containing block.
- Text and inline elements will then wrap around this element.

## HTML

```
<div id="division1">  
    ABC Content  
</div>  
<div id="division2">  
    XYZ Content  
</div>
```

## CSS

```
#division1{  
    background-color : red;  
    float : left;  
    width: 40%;  
}  
#division2{  
    background-color : blue;  
    float : right;  
    width: 40%;  
}
```

# CSS Float Property

---

