Web Designing
Overview of CSS

Control the Display of an Element

- CSS allows you to control the display of an HTML element by using the display and the visibility property.
- The property used to control the display of an element using CSS given below.
  - **Display** -- The display property specifies how to display an element.
  - **Visibility** -- The visibility property specifies whether the element should be visible or hidden.
- The following code snippet shows the use of display property:

  h1 { display: none; }
- The following code snippet shows the use of visibility property:

  p { visibility: visible; }
- CSS also allows you to display the content of an HTML element as inline or blocks.

2

Display Property

- When you display the content of an element as block, then it takes the full width of a Web page and is preceded and followed by a line break.

Value	Description
none	Does not display an element.
block	Generates a block box, which means a line break before and after an element.
inline	Generates an inline box, which means no line break before and after an element. This is the default value.
inline-block	Generates a block box, laid out as an inline box.
inline-table	Generates an inline table element without any line break before and after the element.
list-item	Generates an element as an item of a list element.
table	Generates a table element with a line break before and after the element.

3

Display Property

Value	Description
table-caption	Displays an element as a table caption.
table-cell	Displays an element as a table cell.
table-column	Displays an element as a table column.
table-column-group	Displays an element as a group of table columns.
table-footer-group	Displays an element as a group of table footer.
table-header-group	Displays an element as a group of table headers.
table-row	Displays an element as a table row.
table-row-group	Displays as element as a group of rows.
inherit	Inherits the value of the display property from the parent element.

4

Display

```
<!DOCTYPE HTML>
<HTML>
        <HEAD>
                <TITLE>display</TITLE>
                <STYLE type="text/css">
                        p{display:inline; }
                        span{display:block;    }
                        h1{display:none; }
                </STYLE>
        </HEAD>
        <BODY>
                <H2>Hello all</H2>
                <P>This is </P>
```

5

Display

```
                <P>my paragraph</P>
                <H2>Hello all</H2>
                <SPAN>This is </SPAN>
                <SPAN>my paragraph</SPAN>
                <H1>Hello World</H1>

        </BODY>
</HTML>
```

6

Visibility Property

- The visibility property specifies whether an element is visible on a Web page or not. It takes four values, visible, hidden, collapse, and inherit.

Value	Description
visible	Makes an element visible
hidden	Hides an element
collapse	Represent a value that is applied only to a table element and removes its rows and columns
inherit	Inherits the value of visibility property from the parent element

- You can hide an element either by setting its display property as none or by setting its visibility property as hidden. The difference is that the visibility property hides the element, but it preserves the space required by the element on the document. However, the display property does not preserve any space.

7

Visibility

```
<!DOCTYPE HTML>
<HTML>
    <HEAD>
        <TITLE>visibility</TITLE>
        <STYLE>
            .visible {visibility: visible; }
            .hidden {visibility: hidden; }
        </STYLE>
    </HEAD>
    <BODY>
        <H1> Working with visibility Property </H1>
        <P class="hidden">My paragraph is hidden</P>
        <P class="visible">My paragraph is visible</P>
```

8

Visibility

```
            The first paragraph is hidden and the second
            paragraph is visible
        </BODY>
</HTML>
```

9

Hiding an Element

```
<!DOCTYPE HTML>
<HTML>
        <HEAD>
                <TITLE>display and visibility</TITLE>
                <STYLE type="text/css">
                        h2{ display:none;       }
                        h3{ visibility:hidden;  }
                </STYLE>
        </HEAD>
        <BODY>
                <H1>Hiding Elements using display and visibility
                 Properties</H1>
                <P>my paragraph 1</P>
```

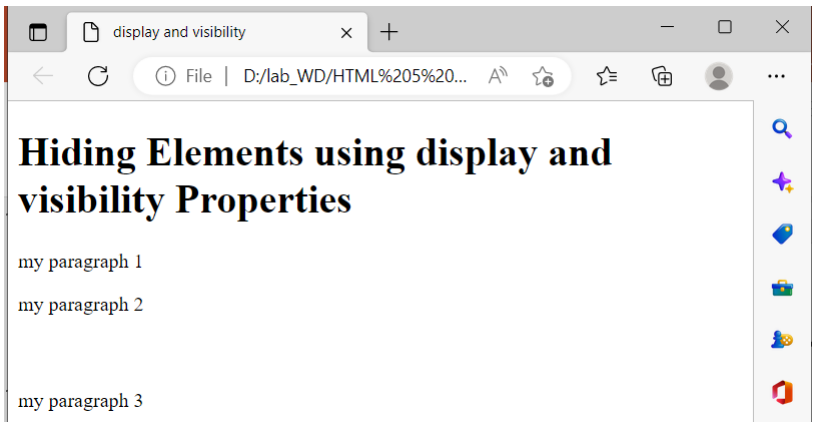
10

Hiding an Element

```
                <H2>Hello World</H2>
                <P>my paragraph 2</P>
                <H3>Hello world</H3>
                <P>my paragraph 3</P>
        </BODY>
</HTML>
```

11

output



Positioning an Element

- CSS provides a property, position, which controls the position of elements with respect to the normal flow of the content on a Web page.
- You can apply the position property on any HTML element, such as P, DIV, TABLE, FORM, and TEXTAREA.
- The syntax to use the position property is given as follows:

  position: [value];

13

Positioning an Element

- The position property takes values described in the table given below:

Value	Description
relative	Specifies relative position of an element with respect to the normal flow the content
absolute	Specifies the position of a block element with respect to the normal flow of the content
fixed	Fixes the position of an element with respect to the normal flow of the content
static	Specifies the normal position of an element
inherit	Specifies that an element uses the same settings of position as of its parent element

14

Positioning an Element

- The following code fragment shows how to use the position property:

  p { position: fixed; }

- CSS also provides some properties that specify the offset position of an element with respect to the normal flow of the content of a Web page.
- These properties are described in the following table:

Property	Description
top	Offsets an element in the top direction of a Web page
bottom	Offsets an element in the bottom direction of a Web page
left	Offsets an element in the left direction of a Web page
right	Offsets an element in the right direction of a Web page

Positioning an Element

- The following code fragment shows how to set the offset positions for an element:

  p { position: fixed; top: 10px; right: 5px; }

- In the preceding code fragment, the offset position of the P element is top-right and the position is fixed with respect to the normal flow of the Web page content.
- Elements can be positioning as follows:
  - Fixed positioning
  - Static positioning
  - Absolute positioning
  - Relative positioning

16

Static Positioning

- HTML elements are positioned static by default. A static positioned element is always positioned according to the normal flow of the web page.
- Static positioned elements are not affected by the top, bottom, left, and right properties.

17

Fixed Positioning

- It set the fixed position for an element.
- It keeps an element fixed with respect to the remaining content of the Web page.
- An element assigned with fixed positioning does not change its position when the Web page is scrolled.
- This type of positioning is often used when a fixed header or footer needs to be specified in each page of a website irrespective of the scrolling of the page.
- This is also used to create a frame whose header and side bar are kept constant and the remaining content keeps changing on scrolling.

18

Fixed Positioning

- The following code fragment shows how to fix the position of the P element:

```
<style type="text/css">
p{
   position: fixed;
   bottom: 10px;
   right: 5px;
}
</style>
```

19

Fixed Position

```
<!DOCTYPE HTML>
<HTML>
        <HEAD>
                <TITLE>fixedposition</TITLE>
                <STYLE type="text/css">
                        .pos_fixed{position:fixed; bottom:10px;
                                right:5px;      }
                </STYLE>
        </HEAD>
        <BODY>
                <H1> Fixing the Position of the content  </H1>
                <A href="feedback.html">
```

20

Fixed Position

```
                    <IMG class="pos_fixed" src="go.png"
                      style="height:60px;width:120" /> </A>
                <P> In this Web page, we have fixed the position of
                 the feedback image to the bottom right corner of
                 the Page. The image remain fixed at this position
                 even if you scroll the page.</P>
                <IMG src="desert.jpg"/>
                <P> We are at the bottom of the Web page and the
                 image, showing an arrow, displays at the same
                 position of the page.</P>
                <BR/><BR/>
        </BODY>
</HTML>
```

21

Absolute Positioning

- The absolute value of the position property is used to set the absolute position of an element with respect to the content of its parent element.
- An absolutely positioned element and its parent element are placed independently.
- The layout of each absolutely positioned element is independent of other elements.
- By default, an absolutely positioned element is placed just above (in z-space coordinate) its parent element.
- Example:

```
<style type="text/css">
   .absolute {position: absolute; left: 100px; top: 150px;  }
   .absolute2 {position: absolute; left: 10px; top: 50px;   }
</style>
```

22

Absolute Positioning

- **Note** - Using CSS, you can also overlap the content of two elements. To overlap two elements, you should specify the position of one element as absolute.
- CSS provides the z-index property to overlap the content of two elements. This property specifies a stack order, that positions one element over another.
- The following code fragments overlaps the content of the P element over the content of the parent element:

<style type="text/css">

p { position: absolute; left: 0px; top: 0px; z-index: -1; } </style>

23

Absolute Positioning

```
<!DOCTYPE HTML>
<HTML>
        <HEAD>
                <TITLE>Absolute Positioning</TITLE>
                <STYLE type="text/css">
                        .absolute{ position:absolute;  left:100px;
                                top:150px;              }
                        .absolute2{position:absolute; left:10px;
                                top:50px;               }
                </STYLE>
        </HEAD>
        <BODY>
```

24

Absolute Positioning

```
<H1 class="absolute">This is a heading with an
 absolute position</h1>
<IMG class="absolute2"
 style="height:100px;width:100px;" src="Car.jpg" />
<P>
        This is my paragraph<BR/>
        This is my paragraph<BR/>
        This is my paragraph<BR/>
        This is my paragraph<BR/>
        This is my paragraph<BR/>
        This is my paragraph<BR/>
        This is my paragraph<BR/>
        This is my paragraph<BR/>
```

25

Absolute Positioning

```
        This is my paragraph<BR/>
        This is my paragraph<BR/>
        This is my paragraph<BR/>
        This is my paragraph<BR/>
        This is my paragraph<BR/>
        This is my paragraph<BR/>
        This is my paragraph<BR/>
        This is my paragraph<BR/>
        This is my paragraph<BR/>
</P>
    </BODY>
</HTML>
```

26

Relative Positioning

- The relative value of the position property is used to set the relative position of an element with respect to the content of its parent element.
- This type of positioning allows an element to retain its natural formatting.
- Relative positioning provides the following advantages :
  - Adjusts the z-order coordinate relative to the other elements that may occupy the same area.
  - Positions the child elements that are underlying or overlaying the content of the parent element.
  - Moves or hides an element dynamically using a scripting language.
- Relative positioning defines a new coordinate system for child elements, with the original located at the position where the first child element is rendered.

27

Relative Positioning

- The following code fragment shows how to set relative position using the CSS position property:

```
<style type="text/css">
    .pos_left { position: relative; left: -20px; }
    .pos_right { position: relative; left:20px; }
</style>
```

28

Relative Position

```
<!DOCTYPE HTML>
<HTML>
    <HEAD>
        <TITLE>relative</TITLE>
        <STYLE type="text/css">
            .pos_left{position:relative;  left:-20px;}
            .pos_right{position:relative; left:20px;}
        </STYLE>
    </HEAD>
    <BODY>
        <H1>Showing Relative Position of the Text </H1>
        <P>Normal positioning</P>
```

29

Relative Position

```
        <P class="pos_left">paragraph is moved 20 px left
         w.r.t. the normal positioning</p>
        <P class="pos_right">paragraph is moved 20 px
         right w.r.t. the normal positioning</P>
    </BODY>
</HTML>
```

30

Floating an Element

- Sometimes you need to use the next wrap feature in a document to properly arrange the position of images with respect to their description.
- For instance, you need to set an image as floated to right side and rest of the content in the left side.
- CSS allows you to implement the text wrap feature in a Web page by using the float property.
- Float property makes an HTML element as a floated element and defines the side where other elements are displayed.
- The syntax to use the float property is given as follows :

  float: [value];

31

Floating an Element

- The float property supports values described in the table given here:

Value	Description
left	Floats an element to the left with respect to the content.
right	Floats an element to the right with respect to the content.
none	Does not float an element.
inherit	Floats an element using the same float settings as specified for its parent element.

32

Floating an Element

- The following code fragment floats an image in the right side :

```
<p>
<img src="image.gif" alt="image" style="float: right;" /> The text is in left direction.
</p>
```

- You can disable the effect of the float property by using the clear property. This means that you can turn off the effect of the float property by using the clear property.
- The syntax to use the clear property is given as follows:

```
clear: [value];
```

33

Floating an Element

- The clear property supports values descried in the following table:

Value	Description
left	Disables the effect of a left floated element.
right	Disables the effect of a right floated element.
both	Disables the effect of both left and right floated.
none	Does not disable the effect of a floated element.
inherit	Uses the same clear settings as specified for its parent element.

- An example of using the clear property is given as follows:
- img { clear: both; }

34

Float an element to left

```
<!DOCTYPE HTML>
<HTML>
        <HEAD>
                <TITLE>float to left</TITLE>
                <STYLE type="text/css">
                        .foo1{  float:left; }
                </STYLE>
        </HEAD>
        <BODY>
                <P>
                <H1>Floating an Element to Left </H1>
                <IMG class="foo1" src="Car.jpg" />
```

35

Float an element to left

My paragraph. My paragraph. My paragraph. My paragraph.
My paragraph. My paragraph. My paragraph. My paragraph.
My paragraph. My paragraph. My paragraph. My paragraph.
My paragraph. My paragraph. My paragraph. My paragraph.
My paragraph. My paragraph. My paragraph. My paragraph.
My paragraph. My paragraph. My paragraph. My paragraph.
My paragraph. My paragraph. My paragraph. My paragraph.
My paragraph. My paragraph. My paragraph. My paragraph.
My paragraph. My paragraph. My paragraph. My paragraph.
My paragraph. My paragraph. My paragraph. My paragraph.
My paragraph. My paragraph. My paragraph. My paragraph.

36

Float an element to left

```
        My paragraph. My paragraph. My paragraph. My paragraph.
    </P>
  </BODY>
</HTML>
```

37

Float an element to right

```
<!DOCTYPE HTML>
<HTML>
        <HEAD>
                <TITLE>float to the right</TITLE>
                <STYLE type="text/css">
                        .foo1{  float:right;       }
                </STYLE>
        </HEAD>
        <BODY>
                <H1>Floating an Element to Right </H1>
                <P>
                        <IMG class="foo1" src="Car.jpg" />
```

38

Float an element to right

My paragraph. My paragraph. My paragraph. My paragraph.
My paragraph. My paragraph. My paragraph. My paragraph.
My paragraph. My paragraph. My paragraph. My paragraph.
My paragraph. My paragraph. My paragraph. My paragraph.
My paragraph. My paragraph. My paragraph. My paragraph.
My paragraph. My paragraph. My paragraph. My paragraph.
My paragraph. My paragraph. My paragraph. My paragraph.
My paragraph. My paragraph. My paragraph. My paragraph.
My paragraph. My paragraph. My paragraph. My paragraph.
My paragraph. My paragraph. My paragraph. My paragraph.
My paragraph. My paragraph. My paragraph. My paragraph.
My paragraph. My paragraph. My paragraph. My paragraph.

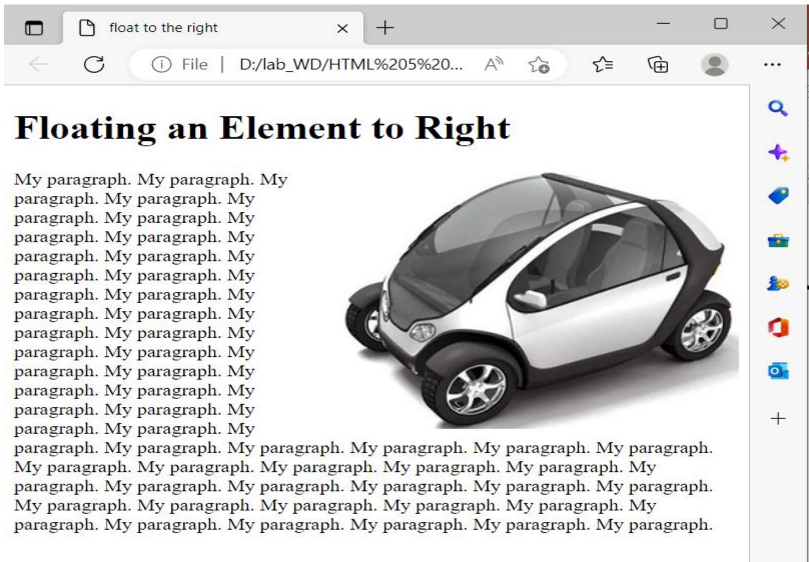
39

Float an element to right

```
        My paragraph. My paragraph. My paragraph. My paragraph.
    </P>
  </BODY>
</HTML>
```

40

output



Turning-off Float with Clear

```
<!DOCTYPE HTML>
<HTML>
        <HEAD>
                <TITLE>clear</TITLE>
                <STYLE type="text/css">
                        .off{    float:left clear:left;     }
                </STYLE>
        </HEAD>
        <BODY>
                <H1>Turning off Floating of the Element </H1>
                <P>
                <IMG class="off" src="Car.jpg" />
```

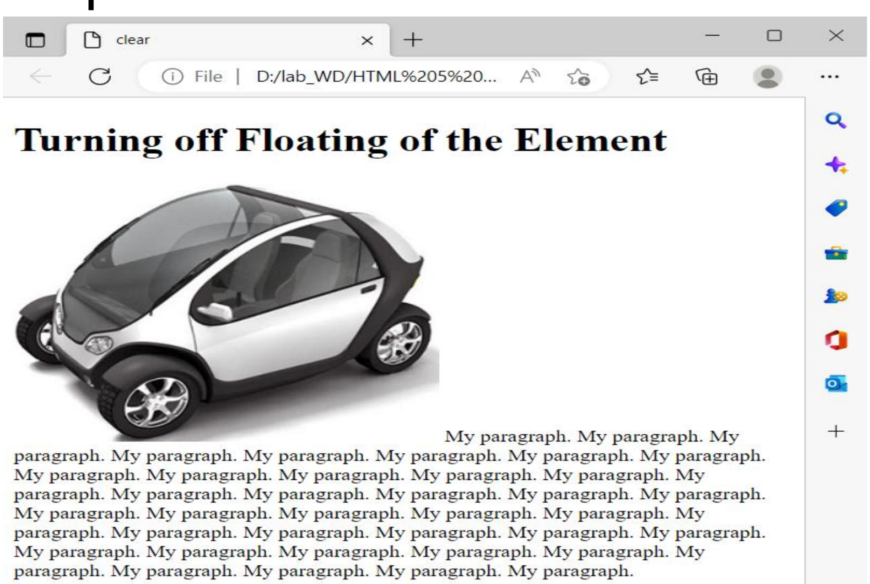
42

Turning-off Float with Clear

```
        My paragraph. My paragraph. My paragraph. My paragraph.
        My paragraph. My paragraph. My paragraph. My paragraph.
        My paragraph. My paragraph. My paragraph. My paragraph.
        My paragraph. My paragraph. My paragraph. My paragraph.
        My paragraph. My paragraph. My paragraph. My paragraph.
        My paragraph. My paragraph. My paragraph. My paragraph.
        My paragraph. My paragraph. My paragraph. My paragraph.
        My paragraph. My paragraph. My paragraph. My paragraph.
        My paragraph. My paragraph. My paragraph. My paragraph.
        My paragraph. My paragraph. My paragraph. My paragraph.
    </P>
  </BODY>
</HTML>
```

43

output



45