

Lecture 14

CSS



CSS Backgrounds

The CSS background properties are used to add background effects for elements.

- **background-color**
- **background-image**
- **background-repeat**
- **background-attachment**
- **background-position**
- **background-size**



CSS 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.

```
body {  
  background-image: url("paper.gif");  
}
```

When using a background image, use an image that does not disturb the text.

The background image can also be set for specific elements, like the **<p>** element.

CSS background-repeat

By default, the background-image property repeats an image both horizontally and vertically. Some images should be repeated only horizontally or vertically. If the image is repeated only horizontally (**background-repeat: repeat-x;**), the background will look better. To repeat an image vertically, set **background-repeat: repeat-y;**

Showing the background image only once is also specified by the background-repeat property:

```
body {  
    background-image: url("img_tree.png");  
    background-repeat: no-repeat;  
}
```

CSS background-position

The **background-position** property is used to specify the position of the background image.

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



CSS Background Attachment

The **background-attachment** property specifies whether the background image should scroll or be fixed (will not scroll with the rest of the page):

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

Opacity / Transparency

The **opacity** property specifies the opacity/transparency of an element. It can take a value from **0.0 - 1.0**. The lower value, the more transparent:

```
div {  
  background-color: green;  
  opacity: 0.3;  
}
```



When using the opacity property to add transparency to the background of an element, all of its child elements inherit the same transparency. This can make the text inside a fully transparent element hard to read. In addition to RGB, you can use an RGB color value with an alpha channel (RGBA) - which specifies the opacity for a color.

CSS Margins

Margins are used to create space around elements, outside of any defined borders. With CSS, you have full control over the margins. There are properties for setting the margin for each side of an element (**top, right, bottom, and left**).

```
div {  
  margin: 70px;  
  border: 1px solid #4CAF50;  
}
```


Margin - Individual Sides

CSS has properties for specifying the margin for each side of an element:

- **margin-top**
- **margin-right**
- **margin-bottom**
- **margin-left**

All the margin properties can have the following values:

- **auto** - the browser calculates the margin
- **length** - specifies a margin in px, pt, cm, etc.
- **%** - specifies a margin in % of the width of the containing element
- **inherit** - specifies that the margin should be inherited from the parent element

Negative values are allowed.

CSS Padding

Padding is used to create space around an element's content, inside of any defined borders. With CSS, you have full control over the padding. There are properties for setting the padding for each side of an element (**top, right, bottom, and left**). CSS has properties for specifying the padding for each side of an element:

- **padding-top**
- **padding-right**
- **padding-bottom**
- **padding-left**

if an element has a specified width, the padding added to that element will be added to the total width of the element. This is often an undesirable result.

you can use the box-sizing property. This causes the element to maintain its actual width; if you increase the padding, the available content space will decrease.

CSS Height/Width

The CSS **height** and **width** properties are used to set the **height** and **width** of an element. The height and width properties do not include padding, borders, or margins. It sets the **height/width** of the area inside the padding, border, and margin of the element.

```
div {  
  height: 200px;  
  width: 50%;  
  background-color: powderblue;  
}
```



CSS Height/Width

The **max-width** can be specified in length values, like **px, cm, etc.**, or in percent (%) of the containing block, or set to none (this is default. Means that there is no maximum width). The problem with the **<div>** above occurs when the browser window is smaller than the **width** of the element (500px). The browser then adds a horizontal scrollbar to the page. Using **max-width** instead, in this situation, will improve the browser's handling of small windows.

```
div {  
    max-width: 500px;  
    height: 100px;  
    background-color: powderblue;  
}
```

CSS Box Model

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 every HTML element. It consists of: **margins, borders, padding, and the actual content.**

- **Content** - The content of the box, where text and images appear
- **Padding** - Clears an area around the content. The padding is transparent
- **Border** - A border that goes around the padding and content
- **Margin** - Clears an area outside the border. The margin is transparent

CSS Box Model

Margin

Border

Padding

Content