Web Programming

Lecturer: Ung Văn Giàu
Email: giau.ung@eiu.edu.vn

CSS Presentation

How to make things shiny?

Contents

1. Text-related Properties

Text-related CSS Properties

- **color** – specifies the color of the text. E.g., #FFF (white), #000 (black), #ccc (border color), rgb(0, 0, 0), rgba(0, 0, 0, 0.5), blue, etc.
- **font-size** – size of font: xx-small, x-small, small, medium, large, x-large, xx-large, smaller, larger or **numeric value** (16px (mobile), 14px (mobile))
- **font-family** – comma separated font names
  - Example: verdana, sans-serif,...
  - The browser loads the **first one** that is **available**
  - There should always be **at least one generic font**:

    "serif", "**sans-serif**", "CURSIVE", "*fantasy*", "monospace"
- **font-weight** can be **normal, bold**, bolder, lighter or a number in range [100 … 900]

CSS Rules for Fonts

- **font-style** – styles the font
  Values: normal, italic

- **text-decoration** – decorates the text
  Values: none, underline, line-through, overline

- **text-align** – defines the alignment of text or other content
  Values: left, right, center, justify

Shorthand Font Property

- **font**
  - Shorthand rule for setting multiple font properties at the same time

    ```
    font: italic normal bold 12px/16px verdana;
    ```

  - is equal to writing this:

    ```
    font-style: italic;
    font-variant: normal;
    font-weight: bold;
    font-size: 12px;
    line-height: 16px;
    font-family: verdana;
    ```

Shorthand Font Property

Shorthand Font Property
Line-height

- The line-height sets the **height of a line box**

- It's commonly used to set the distance between lines of text

- Values:
  - Normal
  - Number: 1, 1.5, 2,…
  - Length: 2em, 3em,…
  - Percentage: 34%, 200%,…

Shorthand Font Property

Font-variant

- Normal

Play with the two different font variants!

- Small-caps

PLAY WITH THE TWO DIFFERENT FONT VARIANTS!

More Fonts

Font Embeds

- Use **@font-face** to **declare** font

- **Point** to font file on server

- **Call** font with font-family

- Use font embedding instead of images


```
@font-face {
    font-family: SketchRockwell;
    src: url('SketchRockwell-Bold.ttf');
}
.my_CSS3_class {
    font-family: SketchRockwell;
    font-size: 3.2em;
}
```

Google Fonts

- **Link:** https://fonts.google.com/
- **Embed Font:**
  - **<link>**

```
<link rel="preconnect" href="https://fonts.googleapis.com">
<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
<link href="https://fonts.googleapis.com/
css2?family=Roboto:wght@400;700&display=swap"
rel="stylesheet">
```

  - **@import**

```
<style>
    @import url('https://fonts.googleapis.com/
        css2?family=Roboto:wght@400;700&display=swap');
</style>
```

Google Fonts

- **Specify in CSS:**

```css
font-family: 'Roboto', sans-serif;
```

- **Note:**
Select the Language Font (Vietnamese)

Text Shadow

- Applies shadow to text

- **Syntax:**

  text-shadow: <horizontal-distance> <vertical-distance> <blur-radius> <shadow-color>;

- Do not alter the size of a box

- Example:

  text-shadow: 2px 2px 7px #000000;

This text is the preview text.

Text Overflow

- Specifies what should happen when text overflows the containing element
- **Syntax: text-overflow:** <value>;
- Possible **values**:
  - **ellipsis** - Display ellipses (…) to represent clipped text
  - **clip** - Default value, clips text
  - **string** - Display the given string to represent the clipped text (only works in Firefox)

> This is some long text that …

> This is some long text that wi

Text Overflow

- Both of the following **properties are required** for text-overflow:

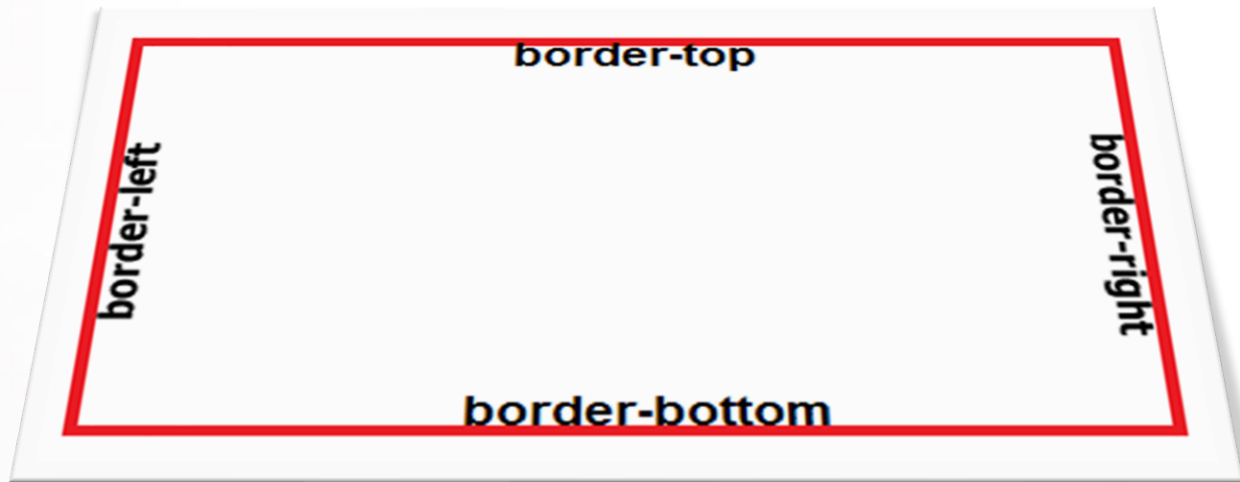 - **white-space**: nowrap;

  - **overflow**: hidden;

- Example:

```css
.text-overflow {
    white-space: nowrap;
    overflow: hidden;
    text-overflow: ellipsis;
    width: 100px;
    height: 18px;
    border: 1px solid #ddd;
}
```

Word Wrapping

- Allows long words to be able to be broken and wrap onto the next line
- **Syntax: word-wrap**: <value>;
- Possible **values**:
  - normal
  - break-word
- Supported in all major browsers

This paragraph has long words thisisaveryverylongwordthatisntreallyoneword and again a longwordwithnospacesinit

This paragraph has long words thisisaveryverylongwordthatisntreallyoneword and again a longwordwithnospacesinit

3. Borders

Borders

- **border-width:** thin, medium, thick or numerical value (e.g. 10px)

- **border-color:** color alias (red, green,…) or RGB value

- **border-style:** none, hidden, dotted, dashed, solid, double, groove, ridge, inset, outset

- **Each property can be defined separately** for left, top, bottom and right

  border-top-style, border-left-color,…

Border Shorthand Property

- **border**: shorthand rule for setting border properties at once:

```
border: 1px solid red;
```

- is equal to writing:

```
border-width:1px;
border-color:red;
border-style:solid;
```

- Specify different borders for the sides via shorthand rules: **border-top, border-left, border-right, border-bottom**

Box shadow

- Allows to easily implement multiple drop shadows (outer or inner) on box elements

- Specifying values for color, size, blur and offset

- **Syntax:**

  box-shadow: [inset] offset-x offset-y [blur-radius] [spread-radius] color

- Example:

```
-moz-box-shadow: 10px 10px 5px #888;
-webkit-box-shadow: 10px 10px 5px #888;
box-shadow: 10px 10px 5px #888;
```

You can also define the color of the shadow. Here the shadow color is grey.

Box shadow

The pulpit rock in Lysefjorden, Norway.

Monterosso al Mare. One of the five villages in Cinque Terre, Italy.

Rounded Corners

- Rounded corners are a part of CSS 3

  - Supported in all major browsers

  - Firefox, IE 9, Chrome, Opera and Safari

- Done by the **border-radius** property

```
border-radius: [<length>|<%>][<length>|<%>]?
```

- Three ways to define corner radius:

```
border-radius: 15px;
border-radius: 15px 20px;
border-radius: 15px 15px 15px 10px;
```
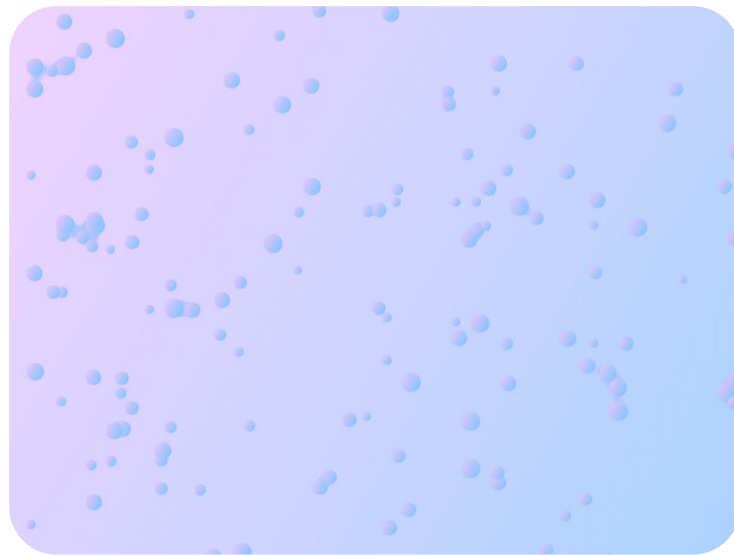
Rounded Corners

Example

border-radius: 25px:

The border-radius property defines the radius of the element's corners.

border-radius: 50px 20px:

If two values are set; the first one is for the top-left and bottom-right corner, the second one for the top-right and bottom-left corner.

4. Background Properties

Backgrounds

- **background-color**

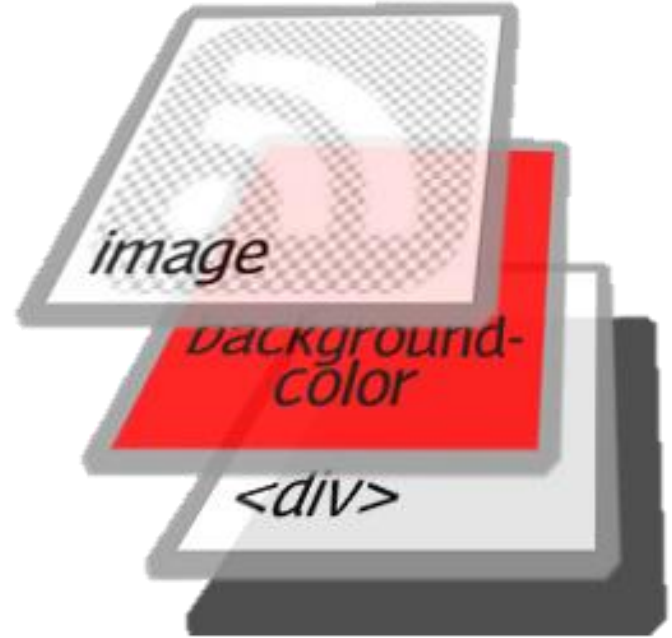   Can use color and image at the same time

- **background-image**
  - URL of image to be used as background
  - Example: background-image: url("back.gif");

- **background-repeat**

    repeat-x, repeat-y, repeat, no-repeat

- **background-attachment**

    fixed / scroll

Backgrounds

- **background-position**

  - **specifies** vertical and horizontal **position of the background image**

  - **Vertical** position: top, center, bottom

  - **Horizontal** position: left, center, right

  - Both can be specified in percentage or other numerical values

  - Examples:

```
background-position: top left;
background-position: -5px 50%;
```

Backgrounds

- **background-size**
  - **specifies** the **size** of the **background images**
  - **Value**: auto | length | contain | cover

auto	Default value. The background image is displayed in its original size
length	Sets the width and height of the background image.
cover	Resize the background image to cover the entire container, even if it has to stretch the image or cut a little bit off one of the edges
contain	Resize the background image to make sure the image is fully visible

  - Examples:
```
background: url(mountain.jpg);
background-repeat: no-repeat;
background-size: 75% 50%;
```

Background Shorthand Property

- **background**: shorthand rule for setting background properties at the same time:

```
background: #FFF0C0 url("back.gif") no-repeat fixed top;
```

- is equal to writing:

```
background-color: #FFF0C0;
background-image: url("back.gif");
background-repeat: no-repeat;
background-attachment: fixed;
background-position: top;
```

- Some browsers will not apply BOTH color and image for background if using shorthand rule

Background-image or <img>?

- **Background images** allow you to **save many image tags** from the HTML

  - Leads to less code

  - More content-oriented approach

- All **images** that are not part of the page content (and are used only **for "beautification"**) should be **moved to** the **CSS**

Gradient Backgrounds

- Gradients are **smooth transitions** between two or more specified colors

- Use of CSS gradients can replace images and reduce download time

  Lots of gradient generators on the WEB

- Create a more flexible layout, and **look better** while **zooming**

- Supported in all major browsers via different keywords

Gradient Backgrounds

Example

```css
background: rgb(34,193,195);
background: -moz-linear-gradient(0deg,
    rgba(34,193,195,1) 0%, rgba(253,187,45,1) 100%);
background: -webkit-linear-gradient(0deg,
    rgba(34,193,195,1) 0%, rgba(253,187,45,1) 100%);
background: linear-gradient(0deg,
    rgba(34,193,195,1) 0%, rgba(253,187,45,1) 100%);
```

Multiple Backgrounds

- CSS3 allows multiple background images

- Simple comma-separated list of images

- Comma separated list for the other properties

```
background-image: url(sheep.png), url(grass.png);
```

5. Opacity

Opacity

- **opacity**

  specifies the opacity of the element

- Floating point number from **0 to 1**

- For old Mozilla browsers use **–moz-opacity**

- For IE use **filter:alpha(opacity=value)** where value is from 0 to 100;
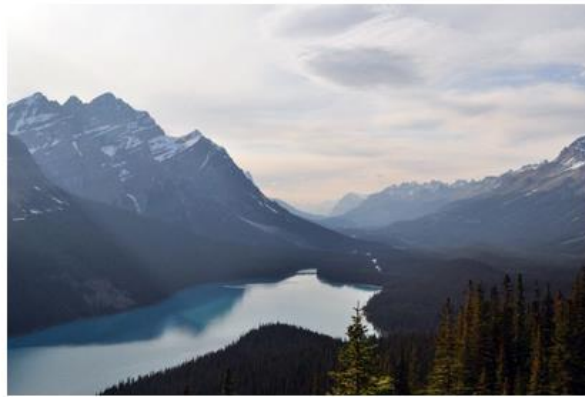
Opacity

Using the opacity property:

10% opacity
30% opacity
60% opacity
opacity 1

Using RGBA color values:

10% opacity
30% opacity
60% opacity
default

Opacity

The **opacity** property is often used together with the **:hover** selector to change the opacity on mouse-over:

Summary

Exercise

Write css for the Lani & Dani webpage.

Q&A