



# CSIT884: Web Development

## CSS with JavaScript CSS Transitions, and Transforms



UNIVERSITY  
OF WOLLONGONG  
AUSTRALIA

**School of Computing and Information  
Technology  
University of Wollongong**

# Objectives

- use JavaScript to manipulate CSS property
- producing animated interactive effects with CSS properties

# Change style by JavaScript

- **Step 1:** give the HTML element that we want to change an **ID**

- **Step 2:** use the function

```
var e = document.getElementById("the-id");
```

to get the HTML element that we want to change

- **Step 3:** change the style of the HTML element

```
e.style.[cssProperty] = "the-new-style-value";
```

**for example:**

```
e.style.color = "pink";
```

```
e.style.fontSize = "25px";
```

```
e.style.fontStyle = "italic";
```

```
...
```

# Change style by JavaScript

- Original CSS properties are written in **hyphen convention**, when we translate them to JavaScript code, we need to use **camel case convention**:

Original CSS property	Translate to JavaScript code
font-size	<code>e.style.fontSize</code>
font-style	<code>e.style.fontStyle</code>
background-color	<code>e.style.backgroundColor</code>
border-bottom-style	<code>e.style.borderBottomStyle</code>
border-left-color	<code>e.style.borderColor</code>
padding-right	<code>e.style.paddingRight</code>

# Example: change text style

The web page displays 3 text fields for user to enter **color**, **font size** and **font style**. When the user clicks the button, then we will change the style of the “Hello world” text.

Enter color:  such as orange, pink, blue, ...

Enter font size:  such as 40px

Enter font style:  such as normal, italic, oblique,...

*Hello world*

# Example: change text style

```
<input type="text" id="colorInput" />  
<input type="text" id="fontSizeInput" />  
<input type="text" id="fontStyleInput" />  
<button onClick="changeStyle();" >Change style</button>  
<span id="hello">Hello world</span>
```

# Example: change text style

```
<script>
  function changeStyle() {
    // get user input from text fields
    var colorTf = document.getElementById("colorInput");
    var colorValue = colorTf.value;

    var fontSizeTf = document.getElementById("fontSizeInput");
    var fontSizeValue = fontSizeTf.value;

    var fontStyleTf = document.getElementById("fontStyleInput");
    var fontStyleValue = fontStyleTf.value;

    // get the Hello World span
    var helloSpan = document.getElementById("hello");

    // change color, font size and font style
    helloSpan.style.color = colorValue;
    helloSpan.style.fontSize = fontSizeValue;
    helloSpan.style.fontStyle = fontStyleValue;
  }
</script>
```

# Example: change image style

The web page displays a text field for user to enter **image opacity** value. When the user clicks the button, then we will change the image style.

Enter image opacity value (from 0 to 1):





# Example: change image style

```
<input type="text" id="opacity" />
```

```
<button onClick="changeOpacity()" >
```

Change Image Opacity

```
</button>
```

```
<img id= "cat" src= "cat.png" />
```

# Example: change image style

```
<script>
    function changeOpacity() {
        // get the opacity value
        var opacityField = document.getElementById("opacity");
        var opacityValue = Number(opacityField.value);

        // set the image opacity
        var image = document.getElementById("cat");
        image.style.opacity = opacityValue;
    }
</script>
```

# Example: change paragraph style

The web page displays text fields for user to enter **font size, padding, text color, background color, border style**. When the user clicks the button, then we will change the paragraph style.

Wollongong 17 ° - 25 °

Enter font size:   
Enter padding:   
Enter text color:   
Enter background color:   
Enter border style:

Change style

# Example: change paragraph style

```
<p id="city">Wollongong 17 &deg; - 25 &deg;</p>  
<input type="text" id="fontSizeInput" />  
<input type="text" id="paddingInput" />  
<input type="text" id="colorInput" />  
<input type="text" id="bgInput" />  
<input type="text" id="borderInput" />  
<button onClick="changeStyle();">Change style</button>
```

# Example: change paragraph style

```
<script>
  function changeStyle() {
    // get user input from text fields
    var fontSize = document.getElementById("fontSizeInput").value;
    var padding = document.getElementById("paddingInput").value;
    var color = document.getElementById("colorInput").value;
    var bgColor = document.getElementById("bgInput").value;
    var border = document.getElementById("borderInput").value;

    // change style
    var cityPar = document.getElementById("city");
    cityPar.style.fontSize = fontSize;
    cityPar.style.padding = padding;
    cityPar.style.color = color;
    cityPar.style.backgroundColor = bgColor;
    cityPar.style.borderStyle = border;
  }
</script>
```

# CSS Transitions

- Often called tweening
- Smooth out changes to property values between two states over time by filling in the frames in between
- CSS Transitions can enrich interfaces and improve usability
- Transitions require a beginning state and an end state
  - the end state needs to be triggered by a state change such as :hover, :focus, or :active

# CSS Transitions

- When applying a transition, needs to be taken into account:
  - transition-property: which CSS property to change (required)
  - transition-duration: duration of change (required)
  - transition-timing-function: how transition accelerates (ease, linear, ease-in, ease-out, ease-in-out)
  - transition-delay: is there a pause before it starts
- Shorthand transition property combines all of these properties into one declaration
  - transition: *property duration timing-function delay*;

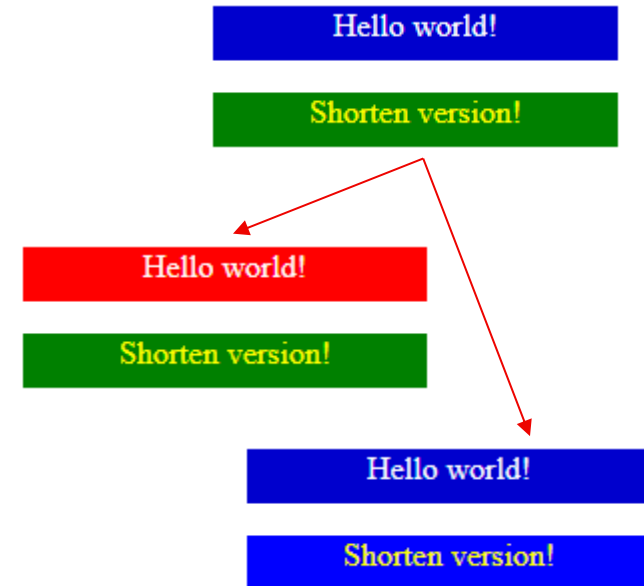
# CSS Transitions

## *what is the output?*

```
<style>
  .trans1 {
    text-align: center;
    padding: 1px;
    height: 25px;
    width: 200px;
    color: white;
    background-color: mediumblue;
    transition-property: background-color;
    transition-duration: 3s;
    transition-timing-function: ease-in-out;
    transition-delay: 1s;
  }
  .trans1:hover {
    background-color: red;
  }

  .trans2 {
    text-align: center;
    padding: 1px;
    height: 25px;
    width: 200px;
    color: yellow;
    background-color: green;
    transition: background-color 3s ease-in-out 1s;
  }
  .trans2:hover, .trans2:focus {
    background-color: blue;
  }
</style>
```

```
. . .
<p class="trans1">Hello world!</p>
<p class="trans2">Shorten version!</p>
. . .
```





# CSS Transforms

- The CSS3 Transforms allows to rotate, relocate, resize, and skew HTML elements in two- and three-dimensional space
- Transforms can be applied
  - to the normal state of an element (so it appears in its transformed state when the page loads), or
  - when users interact with the element (for example via :hover or a JavaScript event)
- The two-dimensional transform functions include: rotate(), translate(), scale(), and skew()

This a normal div element.

This div element is skewed 20 degrees.



# CSS Transforms

*what is the output?*

```
<div>
This a normal div element.
</div>
```

```
<div id="myDiv">
This div element is skewed
20 degrees.
</div>
```

```

<p>cat that rotate</div>
```

```
<style>
div {
    margin-left:100px;
    width: 300px;
    height: 50px;
    vertical-align: middle;
    background-color: yellow;
    border: 1px solid black;
}

div#myDiv {
    transform: skew(20deg);
}

div#myDiv:hover {
    transform: skew(20deg) scale(1.5);
}

img {
    margin-left: 100px;
    height: 100px;
    width: 100px;
}

img:hover {
    transform: rotate(-50deg);
}
</style>
```

# References

- Jennifer Niederst Robbins, Learning Web Design - A Beginner's guide to HTML, CSS, JavaScript and Web Graphics, 5th edition, O'Reilly Media, 2018.
- <http://www.w3schools.com/css>
- <http://www.w3schools.com/js>
- <https://developer.mozilla.org/en-US/docs/Web/CSS/Reference>
- <http://developer.mozilla.org/en-US/docs/Web/JavaScript>
- [https://developer.mozilla.org/en-US/docs/Web/CSS/CSS\\_Transitions/Using\\_CSS\\_transitions](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Transitions/Using_CSS_transitions)