Subject Code: CS3 Computer Science 3
Module Code: 7.0 Advanced CSS
Lesson Code: 7.4.1 What's new in CSS 3 (transform)
Time Frame: 30 minutes

TARGET

Time Allocation: 1 min

After completing this module, you are expected to:

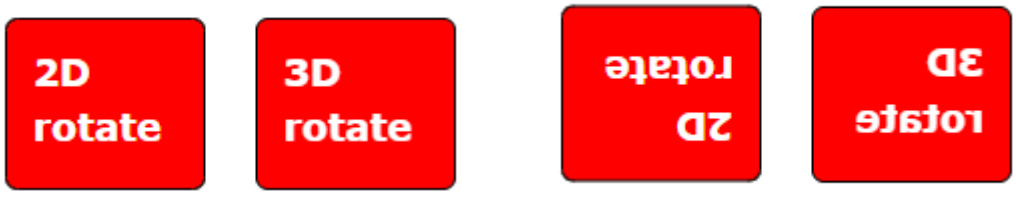
- Learn the basic of transform property in CSS3
- Apply the different 2D and 3D transformation methods

HOOK

Time Allocation: 1 min

In this module, you will be able to learn about the transform property. This transform property can be applied in both 2D and 3D transformation.

Figure 1 (a) shows the state of the elements before a mouse hovers on the element. Figure 1 (b) is the result after a mouse hovers on the element. This illustrates how a 2D transformation and 3D transformation happens.



(b) after rotation

(a) before rotation

Figure 1: Before and after rotation of the element using the transform property

IGNITE

Time Allocation: 22 mins.

What is CSS transform?

CSS transform is a CSS property that allows you to manipulate an element to add visual effects. You can skew, rotate, translate, or scale an element (*Transform*, n.d.-a). The possible values of this property includes the different transformation methods such as rotate(), scale(), skew(), translate(), matrix() and others.

The translate() Method

To translate an element from its current position to another, the translate() method is used. The movement is based on the given X-axis and Y-axis within the given parameter.

```
div {

transform: translate(50px,100px);
}
```

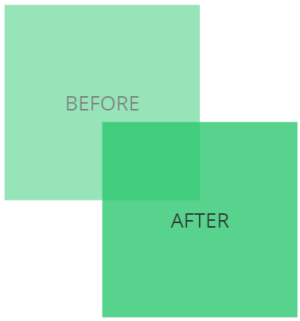


Figure 2: Before and after using the translate() method

The translate() method requires two parameters. The first parameter is for the X-axis, while the second parameter is for the Y-axis. In the given code above, the **`div`** element is moved 50px to the right and 100px going down from its original position. If only 1 parameter is listed (ex. `translate(50px);`), it only moves the element in the X-axis.

The rotate() Method

The rotate() method enables the element to rotate clockwise or counterclockwise. The parameter of this method is the angle of rotation expressed in degrees. The positive values will rotate the element clockwise. However, if this is negative, the element will rotate counter-clockwise.

```
div {
      transform: rotate(-20deg);
}
```

Figure 3: Before and after using the rotate() method

The scale() Method

The scale() method will enable an element to increase or decrease its size based on its given width and height. It requires two parameters. The first parameter will scale the width while the second parameter is for the height. In the given example below, the `div` element will triple its width and decrease its height by half. If only one parameter is used in this method, both the width and height will be scaled according to that value.

```
div {
      transform: scale(3, 0.5);
}
```

Figure 4: Before and after using the scale() method

The scaleX() and scaleY() Method

There are times that you only want the height of an element to increase or decrease. In order to do it, you can use the scaleX() or scaleY() method. The scaleX() and scaleY() method will increase or decrease the width and height, respectively. The parameter will act as a multiplier of the current width or height. For example, **`scaleX(0.5)`** will transform the width of an element 0.5 times or half its original width.

```
div {
      transform: scaleX(0.5);
}
```

```
div {
      transform: scaleX(0.5);
}
```

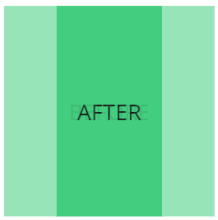


Figure 5: Before and after using the scaleX() method

Figure 6: Before and after using the scaleY() method

The skew(), skewX() and skewY() Method

The skew() method will skew the element along its X-axis and Y-axis. The skewness will be determined by a given angle (in degree), which will be required as parameters of the method. This method will need two parameters which will set the skewness on both X and Y-axis. However, if the second parameter is not specified, it automatically has a zero value. Thus, an element will be skewed along the X-axis only.

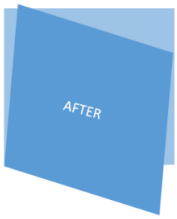


Figure 7: Before and after using the skew() method

```
div {
      transform: skew(5deg, 20deg);
}
```

Figure 8: Before and after using the skewX() method

It is also possible to set skewness on X-axis or Y-axis only. To do this, skewX() or skewY() method is used. Both methods will require only one parameter.

```
div {
      transform: skewX(20deg);
}
```

```
div {
      transform: skewY(20deg);
}
```

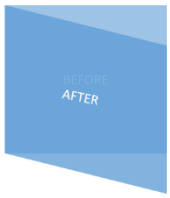


Figure 9: Before and after using the skewY() method

The matrix() Method

To combine all the transformation into one, the matrix() method is used. There are six parameters that this method requires. Using these 6 parameters, the element is scaled, skewed and translated, all at once. The parameters required are the values of scaleX(), skewY(), skewX(), scaleY(), translateX() and translateY(), respectively.

```
div {
      transform: matrix(1, -0.3, 0, 1, 0,
0);
}
```

The rotateX(), rotateY(), rotateZ() Method

The rotate() method allows you to rotate the element clockwise or counterclockwise, depending on the angle of rotation. Using the rotateX() method, you will be able to rotate an element on its X-axis at a given angle of rotation. On the other hand, rotateY() is used to rotate an element around its Y-axis at a given degree. The rotateZ() method will rotate the element in its Z-axis at a given degree.

Figure 10: Before and after using the rotateX() method

```
div {
      transform: rotateX(150deg);
}
```

```
div {
      transform: rotateY(130deg);
}
```

Figure 11: Before and after using the rotateY() method

```
div {
      transform: rotateZ(90deg);
}
```

Figure 12: Before and after using the rotateZ() method

Note: All elements whose layout is ruled by the CSS box model can be transformed. Exceptions to this property include non-replaced inline boxes, table-column boxes, and table-column-group boxes (*Transform*, n.d.-b).

Browser Support

Some browsers may not support the standard syntax of the transform property. To be able to increase compatibility of the property with other browsers, you may add additional syntax for this property, as shown in the example below.

```
div {
      -webkit-transform: translate(50px,100px);
      -ms-transform: translate(50px,100px);
      -mos-transform: translate(50px,100px);
      -o-transform: translate(50px,100px);
      transform: translate(50px,100px);
}
```

The prefix –webkit- is used for lower version of Chrome, Opera, and Safari. From version 9.0 of Internet Explorer going down, the prefix –ms- is used. In Mozilla and Opera, its lower versions may need to have the prefix –moz- and –o-, respectively (*CSS Transform Property*, n.d.).

NAVIGATE

Time Allocation: 5 mins

Let us check your understanding.

Write **T** if the statement is true, otherwise **F** on the space provided before each item.

_____ 1. The transform property will manipulate the visual presentation of an element by scaling, rotating, skewing, and translating.

_____ 2. The unit of the parameter of translate() method is in degree.

_____ 3. An element will rotate counter-clockwise if the value of the rotate() method parameter is positive.

_____ 4. If the scale() method has 1 parameter, only the width will be scaled to the value of that parameter.

_____ 5. The transform method is compatible to all browsers

Note: This is a non-graded assessment.

Time Allocation: 1 mins

In summary,

- CSS transform property will allow you to move, rotate, scale, and skew elements.
- This property will use the transformation methods such as translate(), rotate(), scale(), scaleX(), scaleY(), skew(), skewX(), skewY() and matrix().
- The matrix() method is used to apply transformation all at once.
- Not all browsers are compatible with the transform properties. In order to resolve compatibility issues, prefixes (e.g., mos-, -webkit-) are used together with this property.

REFERENCES

CSS transform property. (n.d.). https://www.w3schools.com/cssref/css3_pr_transform.asp

Transform | css-tricks. (2019). *CSS-Tricks.* https://css-tricks.com/almanac/properties/t/transform/.

Transform. (n.d.). MDN Web Docs. https://developer.mozilla.org/en-US/docs/Web/CSS/transform

Prepared by: JIMMY E. UNILONGO JR.
Position: SST-II
Campus: PSHS-SRC

Reviewed by: ALINE TERESA MENDOZA
Position: SST V
Campus: PSHS- MC