reset   CSS Resets   CSS Tutorials

Make Beautiful Hyperlinks Using CSS

by **CSS Reset** – follow on Twitter **here**

Make Beautiful Hyperlinks Using CSS

A beautiful website design addresses every single aspect and element of the website, whether big or small. Small animations and beautifully designed elements can immensely improve the overall user experience and can take the whole design to a new level. Hyperlinks – or anchor tags – are an important element in any website and it's the only way to display internal or external links on the website. By default, without any CSS underlines and blue coloring will be there for anchor tag and a hand pointer mouse icon on hover. Traditionally, the default design has been followed with some additional CSS to apply different styles for underlining or different colors. Things are not the same now with CSS3 as it offers a lot more to explore in terms of new designs and animation elements.

With a little help of CSS3, the hover effect on hyperlinks can be more beautiful, interactive, and eye-catching. This post provides CSS3 solutions for applying different hover effects on the hyperlink element. Let's begin!

Smoothly Change the Color

CSS3 has a property called "transition" which allows you to change property values smoothly from one value to another over a given duration. It animates the property changes rather than applying them immediately. Consider the following HTML:

Color changes smoothly

The traditional way to change color on a hover event is:

```
#color:hover {
 color: purple;
 }
```

This would change the color immediately on mouse over. But the CSS3 transition property makes this whole process smooth. The following CSS class will apply a color transition over 0.9

seconds.

```
#color:hover {
 transition: color .9s;
 color: purple;
 }
```

You can check out a working demo here to see the difference.

Likewise, you can also change the background color smoothly using the following CSS:

```
#backColor {
 text-decoration: none;
 padding: 0 5px;
 transition: all .9s;
 }
#backColor:hover {
 color: #fff;
 background-color: green;
 }
```

In the above example, both the background and color property will transition due to the value "all" specified for the transition-property. You can check out a working demo here.

Increase the Font Size

To add an increasing font-size animation, you don't have to set the font-size CSS property. You can leverage CSS3's transform property to scale out the text to an extent, resulting in an increase in font size effect. The following CSS code uses transition and transform properties to scale out:

```
a {
 display: inline-block;
 transition: .3s;
 font-weight:bold;
 text-decoration:none;
 }
a:hover {
 -webkit-transform: scale(1.2);
 transform: scale(1.2);
 }
```

You can check out a working demo here.

Flip the Anchor Link Text

Flipping the text on hover is one of my favorite animation effects. The flip effect looks good on a text element when it gets completed quickly. To make this flip animation smooth, put your anchor text in a span element like:

Flip me…

The following CSS code will implement the flip effect on the anchor tag hover event. Assign a rotate value to transform property to rotate the text by 360 degrees to create the flip effect.

```
a,
 a span {
 display: inline-block;
 }
 a span {
 transition: .5s;
 }
 a:hover span {
 -webkit-transform: rotateX(360deg);
 transform: rotateX(360deg);
 }
```

You can check out a working demo here.

Add Border Around the Anchor Text

Adding a border around the anchor text and descaling the text so that it can fit inside the box is another cool way to have a hover animation. The following CSS code attaches a hover event to the anchor tag elements and using the ::before CSS selector, it attaches a box around the anchor text. The ::before selector inserts something before the content of each selected element(s). You need to use the content property to specify the content to insert. For example:

```
a {
 color: #741c7c;
 font-weight: 500;
 -webkit-transition: -webkit-transform 0.2s;
 transition: transform 0.2s;
 display: inline-block;
 text-decoration: none;
 position: relative;
 }
 a:hover {
 -webkit-transform: scale(0.9);
 transform: scale(0.9);
```

```
}
a::before {
position: absolute;
top: -2px;
left: -7px;
box-sizing: content-box;
padding: 0 5px;
width: 100%;
height: 100%;
border: 2px solid #741c7c;
content: '';
opacity: 0;
-webkit-transition: opacity 0.2s, -webkit-transform 0.2s;
transition: opacity 0.2s, transform 0.2s;
-webkit-transform: scale(0.9);
transform: scale(0.9);
}
a:hover::before {
opacity: 1;
-webkit-transform: scale(1.2);
transform: scale(1.2);
}
```

You can see the above code in action here.

Put Anchor Text Inside an Animated Rectangle

The previous CSS code creates a border around the anchor text, but it can also be enhanced to put the anchor text inside an animated rectangle. The animated rectangle appears as a zoom-in effect and covers the anchor text. The following CSS code uses the ::before selector to create a solid rectangle around the anchor element which becomes visible using the hover effect. The transition property adds the transform effect to scale the rectangle which makes a zoom-in effect:

```
a {
  color: #e74c3c;
  font-weight: 500;
  -webkit-transition: -webkit-transform 0.2s;
  transition: transform 0.2s;
  display: inline-block;
  text-decoration: none;
  position: relative;
}
a::before {
position: absolute;
top: 0;
left: -10px;
z-index: -1;
```

```
box-sizing: content-box;
padding: 0 10px;
width: 100%;
height: 100%;
border-radius: 5px;
background-color: #fff;
content: '';
opacity: 0;
-webkit-transition: -webkit-transform 0.4s, opacity 0.3s;
transition: transform 0.4s, opacity 0.3s;
-webkit-transform: scale(0);
transform: scale(0);
}
a:hover::before
{
opacity: 1;
-webkit-transform: scale(1);
transform: scale(1);
}
```

You can see the above code in action here.

Create Square Brackets Around the Anchor Text

The following CSS code adds square brackets around the anchor text on hover. The square brackets appear as a slide-in effect. To append the square bracket the CSS uses ::before and ::after selectors; inside those selectors it defines the position of the bracket from the anchor text. You can easily change the square brackets to any other characters or different type of brackets by modifying the ::before and ::after selector.

```
a {
 color: #213999;
 font-weight: 500;
 -webkit-transition: color 0.2s;
 transition: color 0.2s;
 transition: transform 0.2s;
 display: inline-block;
 text-decoration: none;
 position: relative;
 padding: 0 0.25em;
 }
 a:hover {
 color: #d04c3f;
 }
 a::before,
 a::after {
 position: absolute;
```

```
top: 0;
font-weight: 100;
font-size: 150%;
line-height: 1;
opacity: 0;
-webkit-transition: opacity 0.2s, -webkit-transform 0.2s;
transition: opacity 0.2s, transform 0.2s;
}
a::before {
left: -0.1em;
content: '[';
-webkit-transform: translateX(-100%);
transform: translateX(-100%);
}
a::after {
right: -0.1em;
content: ']';
-webkit-transform: translateX(100%);
transform: translateX(100%);
}
a:hover::before,
a:hover::after {
opacity: 1;
-webkit-transform: translateX(0);
transform: translateX(0);
}
```

You can see the above code in action here.

Tilt the Anchor Text

Tilting the anchor text on hover can easily get a user's attention. To create the tilt effect, all that's required is to rotate the link text to some degree using CSS3 transform property. The transition property can define the duration for completing the tilting effect. The smaller the value, the slower it will be. The following CSS code does exactly the same thing. Based on your requirements, you can change the degree value to manipulate the tilt impact:

```
a {
display: inline-block;
transition: .3s;
text-decoration: none;
}
a:hover {
transform: rotate(15deg);
color: green;
}
```

You can see the above code in action here.

Fade-in the Underline

When no CSS is applied to the anchor tag element an underline will appear. You can manipulate this default behavior and make it more appealing by applying a fade-in effect for showing the underline. The following CSS code initially hides the underline by setting "none" to the "text-decoration" property. It also uses the ::after selector to create an underline below the anchor text and then on hover the effect changes its opacity to 1 to display it on screen.

```
a {
 position: relative;
 display: inline-block;
 transition: .3s;
 text-decoration: none;
 }
 a::after {
 position: absolute;
 bottom: .3em;
 left: 0;
 content: '';
 width: 100%;
 height: 1px;
 background-color: green;
 opacity: 0;
 transition: .3s;
 }
 a:hover::after {
 bottom: 0;
 opacity: 1;
 }
 a:hover {
 color: green;
 }
```

You can see the above code in action here. If you want that underline to appear on top of the anchor text, then you need to change the ::after selector to a ::before selector. Remove the ::after selector CSS classes and use the following CSS classes to put the line on top of the anchor text instead:

```
a::before {
 position: absolute;
 top: .3em;
 left: 0;
 content: '';
```

```
width: 100%;
height: 1px;
background-color: green;
opacity: 0;
transition: .3s;
}
a:hover::before {
top: 0;
opacity: 1;
}
```

You can see the above code in action here.

To show top and bottom both at the same time on hover, use this next block of CSS code. This code makes use of ::after and ::before selectors both to create a line on the top and the bottom of the anchor text.

```
a {
  position: relative;
  display: inline-block;
  transition: .3s;
  text-decoration: none;
}
a::after,
a::before {
position: absolute;
left: 0;
content: '';
width: 100%;
height: 1px;
background-color: green;
opacity: 0;
transition: .3s;
}
a:hover::before {
top: 0;
opacity: 1;
}
a:hover::after {
bottom: 0;
opacity: 1;
}
a:hover {
color: green;
}
```

You can see a demo here.

Conclusion

To conclude, this post provides different CSS3 solutions to animate or change the UI design of the anchor tag element on hover event. Using these solutions, you can easily change colors, increase the font size, flip the text, create a border around the text, put a rectangle or square brackets around the anchor, tilt the text, and manipulate the underline. These solutions are self-explanatory and easily understandable with a little knowledge of CSS3. These solutions can also be used as a reference or guide to create new hover behaviors.

About • Contact • Terms & Conditions • Privacy