

TUTORIALS • ARTICLES • PLAYGROUND • BLUEPRINTS • COLLECTIVE •



In TUTORIALS by SERGIO CAMALICH • January 30, 2012 • 89 COMMENTS

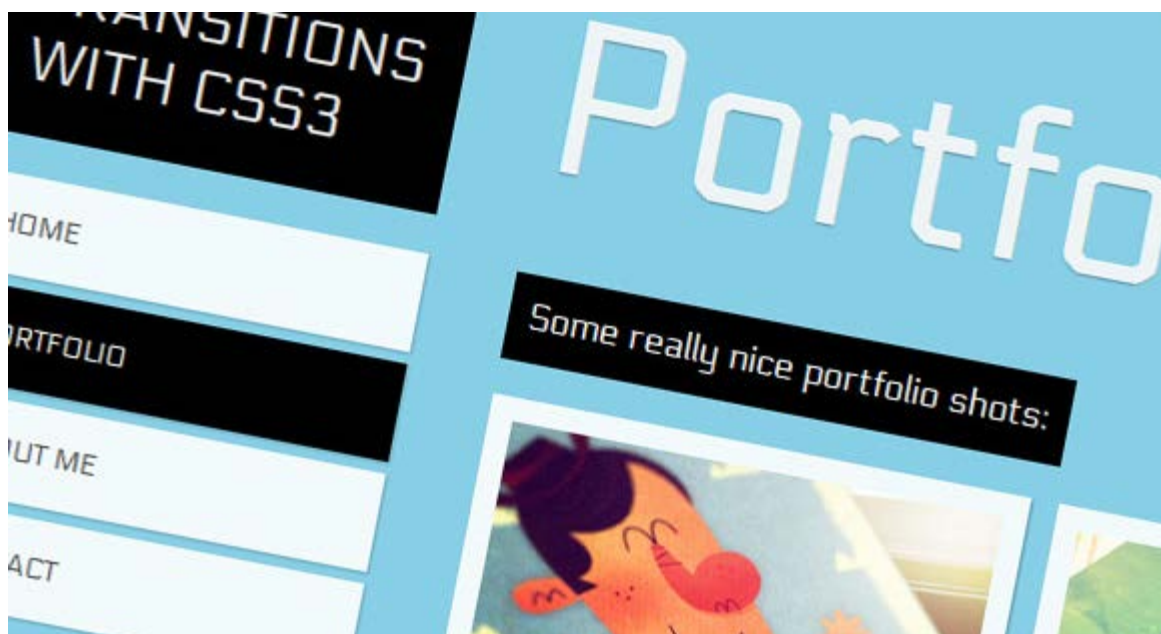
PAGE TRANSITIONS WITH CSS3

In the last few years, we've seen a lot of single page websites lying around the internet, most of them using JavaScript for some transitions effect. Well, now I'm gonna teach you how you can have your own, but instead I'll be using CSS Transitions and the :target property to do all the magic.

ADVERTISEMENT

A nicer way to learn HTML & CSS.




[VIEW DEMO](#)
[DOWNLOAD SOURCE](#)

In the last few years, we've seen a lot of single page websites lying around the internet, most of them using JavaScript for some transitions effect. Well, now I'm gonna teach you how you can have your own, but instead I'll be using CSS Transitions and the `:target` property to do all the magic.

The Dribbble shots used in the demos are by Matt Kaufenberg. Check out his [Dribbble profile](#) or visit his [An Illustration-a-Day Blog](#).

ADVERTISEMENT

WordPress Themes
Awesomeness by *mythemeshop*



FOR ONLY
\$35.00

[Browse Now](#)

MARKUP

The HTML will contain five main divisions: a header and the four content sections. Each of the content sections is going to have an ID and the class **panel**. Moreover, we will add another division inside which will have the class **content**. The first content section which is `#home` will only have the **content** class and will not require an extra division:

```
1 <!-- Home -->
```

```

2  <div id="home" class="content">
3      <h2>Home</h2>
4      <p>Some content</p>
5      <!-- ... -->
6  </div>
7  <!-- /Home -->
8
9  <!-- Portfolio -->
10 <div id="portfolio" class="panel">
11     <div class="content">
12         <h2>Portfolio</h2>
13         <p>Some content</p>
14         <!-- ... -->
15     </div>
16 </div>
17 <!-- /Portfolio -->
18
19 <!-- About -->
20 <div id="about" class="panel">
21     <div class="content">
22         <h2>About</h2>
23         <p>Some content</p>
24         <!-- ... -->
25     </div>
26 </div>
27 <!-- /About -->
28
29 <!-- Contact -->
30 <div id="contact" class="panel">
31     <div class="content">
32         <h2>Contact</h2>
33         <p>Some content</p>
34         <!-- ... -->
35     </div>
36 </div>
37 <!-- /Contact -->

```

In the header we will have the main heading and the navigation:

```

1  <!-- Header with Navigation -->
2  <div id="header">
3      <h1>Page Transitions with CSS3</h1>
4      <ul id="navigation">
5          <li><a id="link-home" href="#home">Home</a></li>
6          <li><a id="link-portfolio" href="#portfolio">Portfolio</a></li>
7          <li><a id="link-about" href="#about">About Me</a></li>
8          <li><a id="link-contact" href="#contact">Contact</a></li>
9      </ul>
10 </div>

```

The reason for having such an “unordered” structure by adding the header to the end, is that we make the navigation “reachable” using the general sibling selector (~), so that we can color the respective items differently.

Now, the main idea is to use the pseudo-class :target in order to add a transition to the current section. In this example, we will be sliding up and down our sections.

CSS

First we will give style to our header and the navigation. We want to keep these on the same spot at all the time, even though everything else will be moving.

```

1  #header{
2      position: absolute;
3      z-index: 2000;
4      width: 235px;
5      top: 50px;
6  }
7  #header h1{
8      font-size: 30px;
9      font-weight: 400;
10     text-transform: uppercase;
11     color: rgba(255,255,255,0.9);
12     text-shadow: 0px 1px 1px rgba(0,0,0,0.3);
13     padding: 20px;
14     background: #000;
15 }

```

```

16  #navigation {
17      margin-top: 20px;
18      width: 235px;
19      display: block;
20      list-style: none;
21      z-index: 3;
22  }
23  #navigation a{
24      color: #444;
25      display: block;
26      background: #fff;
27      background: rgba(255,255,255,0.9);
28      line-height: 50px;
29      padding: 0px 20px;
30      text-transform: uppercase;
31      margin-bottom: 6px;
32      box-shadow: 1px 1px 2px rgba(0,0,0,0.2);
33      font-size: 14px;
34  }
35  #navigation a:hover {
36      background: #ddd;
37  }

```

All the sections except #home have the **panel** class. Here we will be using the transition whenever an element with this class gets “targeted”. The trick is to use a negative margin in the “normal” class and no margin at all in the :target pseudo-class. Adding a transitions will make the panel slide from the top whenever one is “selected”:

```

1  .panel{
2      min-width: 100%;
3      height: 98%;
4      overflow-y: auto;
5      overflow-x: hidden;
6      margin-top: -150%;
7      position: absolute;
8      background: #000;
9      box-shadow: 0px 4px 7px rgba(0,0,0,0.6);
10     z-index: 2;
11     -webkit-transition: all .8s ease-in-out;

```

```

12     -moz-transition: all .8s ease-in-out;
13     -o-transition: all .8s ease-in-out;
14     transition: all .8s ease-in-out;
15 }
16 .panel:target{
17     margin-top: 0%;
18     background-color: #ffcb00;
19 }

```

Next, let's style the **content** class that all our sections have:

```

1  .content{
2      right: 40px;
3      left: 280px;
4      top: 0px;
5      position: absolute;
6      padding-bottom: 30px;
7  }
8  .content h2{
9      font-size: 110px;
10     padding: 10px 0px 20px 0px;
11     margin-top: 52px;
12     color: #fff;
13     color: rgba(255,255,255,0.9);
14     text-shadow: 0px 1px 1px rgba(0,0,0,0.3);
15 }
16 .content p{
17     font-size: 18px;
18     padding: 10px;
19     line-height: 24px;
20     color: #fff;
21     display: inline-block;
22     background: black;
23     padding: 10px;
24     margin: 3px 0px;
25 }

```

In order to change the color of the current item in the navigation, we'll use the `:target` pseudo-class with the general sibling selector to "get to" the respective navigation item:

```
1 #home:target ~ #header #navigation #link-home,  
2 #portfolio:target ~ #header #navigation #link-portfolio,  
3 #about:target ~ #header #navigation #link-about,  
4 #contact:target ~ #header #navigation #link-contact{  
5     background: #000;  
6     color: #fff;  
7 }
```

And that's all you need. Check out the demos and you'll see other examples of how you can do this.

Hope you liked this tutorial and maybe you can experiment with your own effects.

¡Hasta la próxima!

VIEW DEMO

DOWNLOAD SOURCE

PREVIOUS:



[Modular front-end development with LESS](#)

NEXT:



[How to create animated tooltips with CSS3](#)

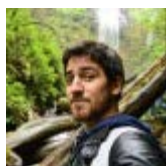
Tagged with:

CSS3

GENERAL SIBLING SELECTOR

PSEUDO-ELEMENT

TRANSITION



SERGIO CAMALICH

Sergio Camalich is a mexican graphic designer, specialized in web design and front-end development, working at [Element Studios](#) and freelancing. He loves camping and is addicted to root beer.

View all contributions by [Sergio Camalich](#)