

```
1. <!DOCTYPE html>
2. <html lang="en">
3. <head>
4.     <meta charset="utf-8">
5.
6.     <title>CSS3 Demonstration - CS76 Spring 2012</title>
7.
8.     <style type="text/css">
9.         /*  */
10.
11.         h1 {
12.             position:absolute;
13.             top:50%;
14.             left:0;
15.             text-align:right;
16.             margin-top:-1em;
17.             margin-bottom:1em;
18.             height:2em;
19.             max-height:2em;
20.             line-height:1.01;
21.             font-size:1.9em;
22.             font-style:italic;
23.             font-weight:400;
24.             font-family:Baskerville, Georgia, "Times New Roman", serif;
25.             /* These properties can accept a number of values, one of them
26.              being rotate() which itself accepts numerical input followed
27.              by a unit like deg or rad (for degree and radians respectively). */
28.             -webkit-transform: rotate(-90deg);
29.             -moz-transform: rotate(-90deg);
30.             -ms-transform: rotate(-90deg);
31.             -o-transform: rotate(-90deg);
32.             transform: rotate(-90deg);
33.         }
34.
35.         #name, #burn {
36.             font-size: 2em;
37.             margin: 10px;
38.             padding: 40px 0 10px;
39.             text-align: center;
40.             color:#767;
41.         }
42.
43.         #name {
44.             /* This is the magic. We'll include four different
45.              comma-separated text shadows. Credit: CSS3.info */
46.             text-shadow: 0 0 4px white,
47.                          0 -5px 4px #FFF33,
48.                          2px -10px 6px #FFDD33,</pre></div>
```

```

49.             -2px -15px 11px #FF8800,
50.             2px -25px 18px #FF2200;
51.     }
52.
53.     #round {
54.         border-radius:12px;
55.         -moz-border-radius:12px;
56.         -webkit-border-radius:12px;
57.         -o-border-radius:12px;
58.         -ms-border-radius:12px;
59.         border: 2px solid #FF9900;
60.         width:200px;
61.         padding:8px;
62.         margin:0 auto;
63.         background-color:#FF9900;
64.         color:#FFF;
65.         text-align:center;
66.         cursor:pointer;
67.         /* Here is an example of vendor-specific prefixes as well
68.            as the use of multiple animations/ transition. The easing
69.            method and delay are option (and omitted). */
70.         transition: background-color 2s, border-color 2s;
71.         -moz-transition: background-color 2s, border-color 2s; /* Firefox 4 */
72.         -webkit-transition: background-color 2s, border-color 2s; /* Safari and Chrome */
73.         -o-transition: background-color 2s, border-color 2s; /* Opera */
74.     }
75.
76.     /* These are the values that will be transition to, based on
77.        the properties that we said we would manipulate. If you include
78.        other properties here they will take effect without transition. */
79.     #round:hover {
80.         background-color:#FF2200;
81.         border-color:#FF1100;
82.     }
83.
84.     /* The three CSS3 keyframes below do the same thing just for different
85.        browsers. Now it becomes clear one of the disadvantages, repetition.
86.        For more into see http://inspectelement.com/articles/ring-a-bell-with-css-keyframe-animations/
87.        Tom Kenny wrote a great example/ tutorial. */
88.     @keyframes burn {
89.         0% {
90.             text-shadow: 0 0 4px white,
91.                         0 -5px 4px #FFFF33,
92.                         2px -10px 6px #FFDD33,
93.                         -2px -15px 11px #FF8800,
94.                         2px -28px 18px #FF2200;
95.         }
96.

```

```
97.         25% {
98.             text-shadow: 2px 0 4px white,
99.                         2px -4px 4px #FFFF33,
100.                        4px -12px 6px #FFDD33,
101.                        -4px -14px 11px #FF8800,
102.                        4px -26px 18px #FF2200;
103.         }
104.
105.         50% {
106.             text-shadow: 4px 0 4px white,
107.                         4px -4px 4px #FFFF33,
108.                         6px -15px 5px #FFDD33,
109.                         -6px -13px 11px #FF8800,
110.                         6px -23px 17px #FF2200;
111.         }
112.
113.         75% {
114.             text-shadow: 2px 0 4px white,
115.                         2px -4px 4px #FFFF33,
116.                         4px -12px 5px #FFDD33,
117.                         -4px -13px 11px #FF8800,
118.                         4px -26px 16px #FF2200;
119.         }
120.
121.         100% {
122.             text-shadow: 0 0 4px white,
123.                         0 -4px 4px #FFFF33,
124.                         2px -15px 5px #FFDD33,
125.                         -2px -13px 11px #FF8800,
126.                         2px -28px 16px #FF2200;
127.         }
128.     }
129.
130.     @-moz-keyframes burn {
131.         0% {
132.             text-shadow: 0 0 4px white,
133.                         0 -5px 4px #FFFF33,
134.                         2px -10px 6px #FFDD33,
135.                         -2px -15px 11px #FF8800,
136.                         2px -28px 18px #FF2200;
137.         }
138.
139.         25% {
140.             text-shadow: 2px 0 4px white,
141.                         2px -4px 4px #FFFF33,
142.                         4px -12px 6px #FFDD33,
143.                         -4px -14px 11px #FF8800,
144.                         4px -26px 18px #FF2200;
```

```
145.     }
146.
147.     50% {
148.         text-shadow: 4px 0 4px white,
149.                     4px -4px 4px #FFFF33,
150.                     6px -15px 5px #FFDD33,
151.                     -6px -13px 11px #FF8800,
152.                     6px -23px 17px #FF2200;
153.     }
154.
155.     75% {
156.         text-shadow: 2px 0 4px white,
157.                     2px -4px 4px #FFFF33,
158.                     4px -12px 5px #FFDD33,
159.                     -4px -13px 11px #FF8800,
160.                     4px -26px 16px #FF2200;
161.     }
162.
163.     100% {
164.         text-shadow: 0 0 4px white,
165.                     0 -4px 4px #FFFF33,
166.                     2px -15px 5px #FFDD33,
167.                     -2px -13px 11px #FF8800,
168.                     2px -28px 16px #FF2200;
169.     }
170. }
171.
172. @-webkit-keyframes burn {
173.     0% {
174.         text-shadow: 0 0 4px white,
175.                     0 -5px 4px #FFFF33,
176.                     2px -10px 6px #FFDD33,
177.                     -2px -15px 11px #FF8800,
178.                     2px -28px 18px #FF2200;
179.     }
180.
181.     25% {
182.         text-shadow: 2px 0 4px white,
183.                     2px -4px 4px #FFFF33,
184.                     4px -12px 6px #FFDD33,
185.                     -4px -14px 11px #FF8800,
186.                     4px -26px 18px #FF2200;
187.     }
188.
189.     50% {
190.         text-shadow: 4px 0 4px white,
191.                     4px -4px 4px #FFFF33,
192.                     6px -15px 5px #FFDD33,
```

```

193.             -6px -13px 11px #FF8800,
194.             6px -23px 17px #FF2200;
195.         }
196.
197.         75% {
198.             text-shadow: 2px 0 4px white,
199.                         2px -4px 4px #FFFF33,
200.                         4px -12px 5px #FFDD33,
201.                         -4px -13px 11px #FF8800,
202.                         4px -26px 16px #FF2200;
203.         }
204.
205.         100% {
206.             text-shadow: 0 0 4px white,
207.                         0 -4px 4px #FFFF33,
208.                         2px -15px 5px #FFDD33,
209.                         -2px -13px 11px #FF8800,
210.                         2px -28px 16px #FF2200;
211.         }
212.     }
213.
214.     #burn {
215.         -webkit-animation: burn 1s 0s ease-in-out infinite;
216.         -moz-animation: burn 1s 0s ease-in-out infinite;
217.         animation: burn 1s 0s ease-in-out infinite;
218.     }
219.
220.     /* ]]> */
221. </style>
222. </head>
223. <body>
224.     <!-- Just a boring title -->
225.     <h1>CSS<sub>3</sub> Demonstration - CS76 Spring 2012</h1>
226.
227.     <!-- Flaming text! -->
228.     <p id="name">The roof is on fire!</p>
229.
230.     <!-- Color-animating box -->
231.     <div id="round">
232.         <p>This is a block of text.</p>
233.         <p>It has a nice rounded border.</p>
234.         <p>Hover over and it will animate.</p>
235.     </div>
236.
237.     <!-- Really flaming text! This is a more complicated version of #name
238.          that uses CSS3 keyframes to animate the text-shadow property. -->
239.     <p id="burn">The roof is really on fire!</p>
240. </body>

```

241. `</html>`

```
1. <!DOCTYPE html>
2. <html lang="en">
3. <head>
4.   <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
5.
6.   <title>geolocation Demonstration - CS76 Spring 2012</title>
7.
8.   <script>
9.     // This will run automatically when the browser
10.    // is ready for us to manipulate things.
11.    var init = function()
12.    {
13.      if (typeof(navigator.geolocation) != 'undefined') {
14.
15.        // This will create a popup asking the user
16.        // to allow/ disallow us from accessing their
17.        // current location.
18.
19.        navigator.geolocation.getCurrentPosition(locate, error);
20.      }
21.    };
22.
23.    // Create a function to handle an error.
24.    window.error = function(msg) {
25.
26.      // Retrieve our element.
27.      var mText = document.getElementById('name');
28.
29.      // If it is a string message, print it.
30.      if (typeof(msg) == typeof('string')) {
31.        mText.innerHTML = msg;
32.        return;
33.      }
34.
35.      // Otherwise match against set of constants.
36.      switch(error.code)
37.      {
38.        case error.TIMEOUT:
39.          mText.innerHTML = 'Location check timed out.';
40.          break;
41.        case error.POSITION_UNAVAILABLE:
42.          mText.innerHTML = 'Location is not available.';
43.          break;
44.        case error.PERMISSION_DENIED:
45.          mText.innerHTML = 'Denied access to location.';
46.          break;
47.        case error.UNKNOWN_ERROR:
48.          mText.innerHTML = 'An unknown error has occurred.';
```

```
49.         break;
50.     }
51. };
52.
53. // Create a function to handle successful location retrieval.
54. window.locate = function(position) {
55.
56.     var lat = position.coords.latitude,
57.         long = position.coords.longitude;
58.
59.     // Retrieve our element.
60.     var mText = document.getElementById('name');
61.
62.     mText.innerHTML = 'Your location: (' +
63.         lat + ', ' + long + ').';
64.
65. };
66.
67. // Set what to do when ready.
68. if (typeof(addEventListener) != 'undefined') {
69.     window.addEventListener('load', init, false)
70. } else if (typeof(attachEvent) != 'undefined') {
71.     window.attachEvent('onload', init);
72. } else {
73.     window.onload = init;
74. }
75.
76. </script>
77. </head>
78. <body>
79.     <p id="name">Location loading...</p>
80. </body>
81. </html>
```



```
49.         // Check if the error is caused by
50.         // storing too much data.
51.         if (e == QUOTA_EXCEEDED_ERR) {
52.
53.             mText.innerHTML = 'We\'ve ran out of storage :-(';
54.
55.         }
56.     }
57. }
58. };
59.
60. // Set what to do when ready.
61. if (typeof(addEventListener) != 'undefined') {
62.     window.addEventListener('load', init, false)
63. } else if (typeof(attachEvent) != 'undefined') {
64.     window.attachEvent('onload', init);
65. } else {
66.     window.onload = init;
67. }
68.
69. </script>
70. </head>
71. <body>
72.     <p id="name"></p>
73. </body>
74. </html>
```