

```
1. <!--
2.
3. blink.html
4.
5. Resurrects the blink tag.
6.
7. Computer Science 50
8. David J. Malan
9.
10. -->
11.
12. <!DOCTYPE html>
13.
14. <html>
15.   <head>
16.     <script>
17.
18.       function blinker()
19.       {
20.         var blinks = document.getElementsByTagName("blink");
21.         for (var i = 0; i < blinks.length; i++)
22.         {
23.           if (blinks[i].style.visibility == "hidden")
24.             blinks[i].style.visibility = "visible";
25.           else
26.             blinks[i].style.visibility = "hidden";
27.         }
28.       }
29.
30.       window.setInterval(blinker, 500);
31.
32.     </script>
33.     <title>blink</title>
34.   </head>
35.   <body>
36.     <div style="margin: 240px; text-align: center">
37.       <blink><h1>hello, world</h1></blink>
38.     </div>
39.   </body>
40. </html>
```

```
1. <!--
2.
3. ajax1.html
4.
5. Gets stock quote from quotel.php via Ajax, displaying result with alert().
6.
7. Computer Science 50
8. David J. Malan
9.
10. -->
11.
12. <!DOCTYPE html>
13.
14. <html>
15.   <head>
16.     <script>
17.
18.       // an XMLHttpRequest
19.       var xhr = null;
20.
21.       /*
22.        * void
23.        * quote()
24.        *
25.        * Gets a quote.
26.        */
27.       function quote()
28.       {
29.         // instantiate XMLHttpRequest object
30.         try
31.         {
32.           xhr = new XMLHttpRequest();
33.         }
34.         catch (e)
35.         {
36.           xhr = new ActiveXObject("Microsoft.XMLHTTP");
37.         }
38.
39.         // handle old browsers
40.         if (xhr == null)
41.         {
42.           alert("Ajax not supported by your browser!");
43.           return;
44.         }
45.
46.         // construct URL
47.         var url = "quotel.php?symbol=" + document.getElementById("symbol").value;
48.
```

```
49.         // get quote
50.         xhr.onreadystatechange = handler;
51.         xhr.open("GET", url, true);
52.         xhr.send(null);
53.     }
54.
55.
56.     /*
57.     * void
58.     * handler()
59.     *
60.     * Handles the Ajax response.
61.     */
62.     function handler()
63.     {
64.         // only handle loaded requests
65.         if (xhr.readyState == 4)
66.         {
67.             // display response if possible
68.             if (xhr.status == 200)
69.                 alert(xhr.responseText);
70.             else
71.                 alert("Error with Ajax call!");
72.         }
73.     }
74.
75.     </script>
76.     <title>ajax1</title>
77. </head>
78. <body>
79.     <form onsubmit="quote(); return false;">
80.         Symbol: <input id="symbol" type="text">
81.         <br><br>
82.         <input type="submit" value="Get Quote">
83.     </form>
84. </body>
85. </html>
```

```
1. <!--
2.
3. ajax2.html
4.
5. Gets stock quote from quotel.php via Ajax, embedding result in page itself.
6.
7. Computer Science 50
8. David J. Malan
9.
10. -->
11.
12. <!DOCTYPE html>
13.
14. <html>
15.   <head>
16.     <script>
17.
18.       // an XMLHttpRequest
19.       var xhr = null;
20.
21.       /*
22.        * void
23.        * quote()
24.        *
25.        * Gets a quote.
26.        */
27.       function quote()
28.       {
29.         // instantiate XMLHttpRequest object
30.         try
31.         {
32.           xhr = new XMLHttpRequest();
33.         }
34.         catch (e)
35.         {
36.           xhr = new ActiveXObject("Microsoft.XMLHTTP");
37.         }
38.
39.         // handle old browsers
40.         if (xhr == null)
41.         {
42.           alert("Ajax not supported by your browser!");
43.           return;
44.         }
45.
46.         // construct URL
47.         var url = "quotel.php?symbol=" + document.getElementById("symbol").value;
48.
```

```
49.         // get quote
50.         xhr.onreadystatechange = handler;
51.         xhr.open("GET", url, true);
52.         xhr.send(null);
53.     }
54.
55.
56.     /*
57.     * void
58.     * handler()
59.     *
60.     * Handles the Ajax response.
61.     */
62.     function handler()
63.     {
64.         // only handle loaded requests
65.         if (xhr.readyState == 4)
66.         {
67.             // embed response in page if possible
68.             if (xhr.status == 200)
69.                 document.getElementById("price").innerHTML = xhr.responseText;
70.             else
71.                 alert("Error with Ajax call!");
72.         }
73.     }
74.
75.     </script>
76.     <title>ajax2</title>
77. </head>
78. <body>
79.     <form onsubmit="quote(); return false;">
80.         Symbol: <input id="symbol" type="text">
81.         <br>
82.         Price: <span id="price"><b>to be determined</b></span>
83.         <br><br>
84.         <input type="submit" value="Get Quote">
85.     </form>
86. </body>
87. </html>
```

```
1. <!--
2.
3. ajax3.html
4.
5. Gets stock quote (plus day's low and high) from quote2.php via Ajax,
6. embedding result in page itself after indicating progress with an
7. animated GIF.
8.
9. Computer Science 50
10. David J. Malan
11.
12. -->
13.
14. <!DOCTYPE html>
15.
16. <html>
17.   <head>
18.     <script>
19.
20.         // an XMLHttpRequest
21.         var xhr = null;
22.
23.         /*
24.          * void
25.          * quote()
26.          *
27.          * Gets a quote.
28.          */
29.         function quote()
30.         {
31.             // instantiate XMLHttpRequest object
32.             try
33.             {
34.                 xhr = new XMLHttpRequest();
35.             }
36.             catch (e)
37.             {
38.                 xhr = new ActiveXObject("Microsoft.XMLHTTP");
39.             }
40.
41.             // handle old browsers
42.             if (xhr == null)
43.             {
44.                 alert("Ajax not supported by your browser!");
45.                 return;
46.             }
47.
48.             // construct URL
```

```
49.         var url = "quote2.php?symbol=" + document.getElementById("symbol").value;
50.
51.         // show progress
52.         document.getElementById("progress").style.display = "block";
53.
54.         // get quote
55.         xhr.onreadystatechange = handler;
56.         xhr.open("GET", url, true);
57.         xhr.send(null);
58.     }
59.
60.
61.     /*
62.     * void
63.     * handler()
64.     *
65.     * Handles the Ajax response.
66.     */
67.     function handler()
68.     {
69.         // only handle requests in "loaded" state
70.         if (xhr.readyState == 4)
71.         {
72.             // hide progress
73.             document.getElementById("progress").style.display = "none";
74.
75.             // embed response in page if possible
76.             if (xhr.status == 200)
77.                 document.getElementById("quote").innerHTML = xhr.responseText;
78.             else
79.                 alert("Error with Ajax call!");
80.         }
81.     }
82.
83.     </script>
84.     <title>ajax3</title>
85. </head>
86. <body>
87.     <form onsubmit="quote(); return false;">
88.         Symbol: <input id="symbol" type="text">
89.         <br><br>
90.         <div id="progress" style="display: none">
91.             
92.             <br><br>
93.         </div>
94.         <div id="quote"></div>
95.         <br><br>
96.         <input type="submit" value="Get Quote">
```

---

```
97.     </form>
98. </body>
99. </html>
```

```
1. <!--
2.
3. ajax4.html
4.
5. Gets stock quote from quotel.php via Ajax, displaying result with alert().
6. Implements handler as an anonymous function.
7.
8. Computer Science 50
9. David J. Malan
10.
11. -->
12.
13. <!DOCTYPE html>
14.
15. <html>
16.   <head>
17.     <script>
18.
19.         // an XMLHttpRequest
20.         var xhr = null;
21.
22.         /*
23.          * void
24.          * quote()
25.          *
26.          * Gets a quote.
27.          */
28.         function quote()
29.         {
30.             // instantiate XMLHttpRequest object
31.             try
32.             {
33.                 xhr = new XMLHttpRequest();
34.             }
35.             catch (e)
36.             {
37.                 xhr = new ActiveXObject("Microsoft.XMLHTTP");
38.             }
39.
40.             // handle old browsers
41.             if (xhr == null)
42.             {
43.                 alert("Ajax not supported by your browser!");
44.                 return;
45.             }
46.
47.             // construct URL
48.             var url = "quotel.php?symbol=" + document.getElementById("symbol").value;
```

```
49.
50.     // get quote
51.     xhr.onreadystatechange = function () {
52.         // only handle loaded requests
53.         if (xhr.readyState == 4)
54.         {
55.             // display response if possible
56.             if (xhr.status == 200)
57.                 alert(xhr.responseText);
58.             else
59.                 alert("Error with Ajax call!");
60.         }
61.     };
62.     xhr.open("GET", url, true);
63.     xhr.send(null);
64. }
65.
66. </script>
67. <title>ajax4</title>
68. </head>
69. <body>
70.     <form onsubmit="quote(); return false;">
71.         Symbol: <input id="symbol" type="text">
72.         <br><br>
73.         <input type="submit" value="Get Quote">
74.     </form>
75. </body>
76. </html>
```

```
1. <!--
2.
3. ajax5.html
4.
5. Gets stock quote (plus day's low and high) from quote3.php via Ajax,
6. embedding (JSON) result in page itself.
7.
8. Computer Science 50
9. David J. Malan
10.
11. -->
12.
13. <!DOCTYPE html>
14.
15. <html>
16.   <head>
17.     <script>
18.
19.         // an XMLHttpRequest
20.         var xhr = null;
21.
22.         /*
23.          * void
24.          * quote()
25.          *
26.          * Gets a quote.
27.          */
28.         function quote()
29.         {
30.             // instantiate XMLHttpRequest object
31.             try
32.             {
33.                 xhr = new XMLHttpRequest();
34.             }
35.             catch (e)
36.             {
37.                 xhr = new ActiveXObject("Microsoft.XMLHTTP");
38.             }
39.
40.             // handle old browsers
41.             if (xhr == null)
42.             {
43.                 alert("Ajax not supported by your browser!");
44.                 return;
45.             }
46.
47.             // construct URL
48.             var url = "quote3.php?symbol=" + document.getElementById("symbol").value;
```

```
49.
50.     // get quote
51.     xhr.onreadystatechange = function() {
52.
53.         // only handle requests in "loaded" state
54.         if (xhr.readyState == 4)
55.         {
56.             // embed response in page if possible
57.             if (xhr.status == 200)
58.             {
59.                 var quote = eval("(" + xhr.responseText + ")");
60.                 document.getElementById("price").innerHTML = quote.price;
61.                 document.getElementById("high").innerHTML = quote.high;
62.                 document.getElementById("low").innerHTML = quote.low;
63.             }
64.             else
65.                 alert("Error with Ajax call!");
66.         }
67.
68.     };
69.     xhr.open("GET", url, true);
70.     xhr.send(null);
71. }
72.
73. </script>
74. <title>ajax5</title>
75. </head>
76. <body>
77.     <form onsubmit="quote(); return false;">
78.         Symbol: <input id="symbol" type="text">
79.         <br><br>
80.         Price: <span id="price"></span>
81.         <br>
82.         High: <span id="high"></span>
83.         <br>
84.         Low: <span id="low"></span>
85.         <br>
86.         <br><br>
87.         <input type="submit" value="Get Quote">
88.     </form>
89. </body>
90. </html>
```

```
1. <!--
2.
3. ajax6.html
4.
5. Gets stock quote (plus day's low and high) from quote3.php via Ajax
6. using jQuery, embedding (JSON) result in page itself.
7.
8. Computer Science 50
9. David J. Malan
10.
11. -->
12.
13. <!DOCTYPE html>
14.
15. <html>
16.   <head>
17.     <script src="http://code.jquery.com/jquery-latest.js"></script>
18.     <script>
19.
20.         $(document).ready(function() {
21.             $("#form").submit(function() {
22.                 $.ajax({
23.                     url: "quote3.php",
24.                     data: {
25.                         symbol: $("#symbol").val()
26.                     },
27.                     success: function(data) {
28.                         $("#price").html(data.price);
29.                         $("#high").html(data.high);
30.                         $("#low").html(data.low);
31.                     }
32.                 });
33.                 return false;
34.             });
35.         });
36.
37.     </script>
38.     <title>ajax6</title>
39. </head>
40. <body>
41.   <form id="form">
42.     Symbol: <input id="symbol" type="text">
43.     <br><br>
44.     Price: <span id="price"></span>
45.     <br>
46.     High: <span id="high"></span>
47.     <br>
48.     Low: <span id="low"></span>
```

```
49.     <br>
50.     <br><br>
51.     <input type="submit" value="Get Quote">
52. </form>
53. </body>
54. </html>
```

```
1. <?
2.
3.  /**
4.   * quote1.php
5.   *
6.   * Outputs price of given symbol as text/html.
7.   *
8.   * Computer Science 50
9.   * David J. Malan
10.  */
11.
12. // get quote
13. $handle = @fopen("http://download.finance.yahoo.com/d/quotes.csv?s={$_GET["symbol"]}&f=e1l1", "r");
14. if ($handle !== FALSE)
15. {
16.     $data = fgetcsv($handle);
17.     if ($data !== FALSE && $data[0] == "N/A")
18.         print($data[1]);
19.     fclose($handle);
20. }
21. ?>
```

```
1. <?
2.
3.  /**
4.   * quote2.php
5.   *
6.   * Outputs price, low, and high of given symbol as text/html, after
7.   * inserting an artificial delay.
8.   *
9.   * Computer Science 50
10.  * David J. Malan
11.  */
12.
13.
14.  // pretend server is slow
15.  sleep(5);
16.
17.  // try to get quote
18.  $handle = @fopen("http://download.finance.yahoo.com/d/quotes.csv?s={$_GET["symbol"]}&f=e1llhg", "r");
19.  if ($handle !== FALSE)
20.  {
21.      $data = fgetcsv($handle);
22.      if ($data !== FALSE && $data[0] == "N/A")
23.      {
24.          print("Price: {$data[1]}");
25.          print("<br>");
26.          print("High: {$data[2]}");
27.          print("<br>");
28.          print("Low: {$data[3]}");
29.      }
30.      fclose($handle);
31.  }
32. ?>
```

```
1. <?
2.
3.  /**
4.   * quote3.php
5.   *
6.   * Outputs price, low, and high of given symbol as JSON.
7.   *
8.   * Computer Science 50
9.   * David J. Malan
10.  */
11.
12. // try to get quote
13. $quote = array();
14. $handle = @fopen("http://download.finance.yahoo.com/d/quotes.csv?s={$_GET["symbol"]}&f=e1l1hg", "r");
15. if ($handle !== FALSE)
16. {
17.     $data = fgetcsv($handle);
18.     if ($data !== FALSE && $data[0] == "N/A")
19.     {
20.         $quote["price"] = $data[1];
21.         $quote["high"] = $data[2];
22.         $quote["low"] = $data[3];
23.     }
24.     fclose($handle);
25. }
26. header("Content-type: application/json");
27. print(json_encode($quote));
28.
29. ?>
```

```
1. <?
2.
3.     /**
4.      * dump.php
5.      *
6.      * Dumps HTTP requests.
7.      *
8.      * Computer Science 50
9.      * David J. Malan
10.     */
11.
12. ?>
13.
14. <!DOCTYPE html>
15.
16. <html>
17.     <head>
18.         <title>dump</title>
19.     </head>
20.     <body>
21.         <pre><? print_r($_GET); ?></pre>
22.     </body>
23. </html>
```

```
1. <!--
2.
3. form1.html
4.
5. A form without client-side validation.
6.
7. Computer Science 50
8. David J. Malan
9.
10. -->
11.
12. <!DOCTYPE html>
13.
14. <html>
15.   <head>
16.     <title>form1</title>
17.   </head>
18.   <body>
19.     <form action="dump.php" method="get">
20.       Email: <input name="email" type="text">
21.       <br>
22.       Password: <input name="password1" type="password">
23.       <br>
24.       Password (again): <input name="password2" type="password">
25.       <br>
26.       I agree to the terms and conditions: <input name="agreement" type="checkbox">
27.       <br><br>
28.       <input type="submit" value="Submit">
29.     </form>
30.   </body>
31. </html>
```

```
1. <!--
2.
3. form2.html
4.
5. A form with client-side validation.
6.
7. Computer Science 50
8. David J. Malan
9.
10. -->
11.
12. <!DOCTYPE html>
13.
14. <html>
15.   <head>
16.     <script>
17.
18.       function validate()
19.       {
20.         if (document.forms.registration.email.value == "")
21.         {
22.           alert("You must provide an email address.");
23.           return false;
24.         }
25.         else if (document.forms.registration.password1.value == "")
26.         {
27.           alert("You must provide a password.");
28.           return false;
29.         }
30.         else if (document.forms.registration.password1.value != document.forms.registration.password2.value)
31.         {
32.           alert("You must provide the same password twice.");
33.           return false;
34.         }
35.         else if (!document.forms.registration.agreement.checked)
36.         {
37.           alert("You must agree to our terms and conditions.");
38.           return false;
39.         }
40.         return true;
41.       }
42.
43.     </script>
44.     <title>form2</title>
45.   </head>
46.   <body>
47.     <form action="dump.php" method="get" name="registration" onsubmit="return validate();">
48.       Email: <input name="email" type="text">
```

```
49.     <br>
50.     Password: <input name="password1" type="password">
51.     <br>
52.     Password (again): <input name="password2" type="password">
53.     <br>
54.     I agree to the terms and conditions: <input name="agreement" type="checkbox">
55.     <br><br>
56.     <input type="submit" value="Submit">
57. </form>
58. </body>
59. </html>
```

```
1. <!--
2.
3. form3.html
4.
5. A form with client-side validation demonstrating "this" keyword.
6.
7. Computer Science 50
8. David J. Malan
9.
10. -->
11.
12. <!DOCTYPE html>
13.
14. <html>
15.   <head>
16.     <script>
17.
18.       function validate(f)
19.       {
20.         if (f.email.value == "")
21.         {
22.           alert("You must provide an email address.");
23.           return false;
24.         }
25.         else if (f.password1.value == "")
26.         {
27.           alert("You must provide a password.");
28.           return false;
29.         }
30.         else if (f.password1.value != f.password2.value)
31.         {
32.           alert("You must provide the same password twice.");
33.           return false;
34.         }
35.         else if (!f.agreement.checked)
36.         {
37.           alert("You must agree to our terms and conditions.");
38.           return false;
39.         }
40.         return true;
41.       }
42.
43.     </script>
44.     <title>form3</title>
45.   </head>
46.   <body>
47.     <form action="dump.php" method="get" onsubmit="return validate(this);">
48.       Email: <input name="email" type="text">
```

```
49.     <br>
50.     Password: <input name="password1" type="password">
51.     <br>
52.     Password (again): <input name="password2" type="password">
53.     <br>
54.     I agree to the terms and conditions: <input name="agreement" type="checkbox">
55.     <br><br>
56.     <input type="submit" value="Submit">
57. </form>
58. </body>
59. </html>
```

```
1. <!--
2.
3. form4.html
4.
5. A form with client-side validation demonstrating disabled property.
6.
7. Computer Science 50
8. David J. Malan
9.
10. -->
11.
12. <!DOCTYPE html>
13.
14. <html>
15.   <head>
16.     <script>
17.
18.       function toggle()
19.       {
20.         if (document.forms.registration.button.disabled)
21.           document.forms.registration.button.disabled = false;
22.         else
23.           document.forms.registration.button.disabled = true;
24.       }
25.
26.       function validate()
27.       {
28.         if (document.forms.registration.email.value == "")
29.         {
30.           alert("You must provide an email address.");
31.           return false;
32.         }
33.         else if (document.forms.registration.password1.value == "")
34.         {
35.           alert("You must provide a password.");
36.           return false;
37.         }
38.         else if (document.forms.registration.password1.value != document.forms.registration.password2.value)
39.         {
40.           alert("You must provide the same password twice.");
41.           return false;
42.         }
43.         else if (!document.forms.registration.agreement.checked)
44.         {
45.           alert("You must agree to our terms and conditions.");
46.           return false;
47.         }
48.         return true;
```

```
49.     }
50.
51.     </script>
52.     <title>form4</title>
53. </head>
54. <body>
55.     <form action="dump.php" method="get" name="registration" onsubmit="return validate();">
56.         Email: <input name="email" type="text">
57.         <br>
58.         Password: <input name="password1" type="password">
59.         <br>
60.         Password (again): <input name="password2" type="password">
61.         <br>
62.         I agree to the terms and conditions: <input name="agreement" onclick="toggle();" type="checkbox">
63.         <br><br>
64.         <input disabled="disabled" name="button" type="submit" value="Submit">
65.     </form>
66. </body>
67. </html>
```

```
1. <!--
2.
3. form5.html
4.
5. A form with client-side validation demonstrating regular expressions.
6.
7. Computer Science 50
8. David J. Malan
9.
10. -->
11.
12. <!DOCTYPE html>
13.
14. <html>
15.   <head>
16.     <script>
17.
18.       function validate()
19.       {
20.         if (!document.forms.registration.email.value.match(/.+@.+\.edu$/))
21.         {
22.           alert("You must provide a .edu email address.");
23.           return false;
24.         }
25.         else if (document.forms.registration.password1.value == "")
26.         {
27.           alert("You must provide a password.");
28.           return false;
29.         }
30.         else if (document.forms.registration.password1.value != document.forms.registration.password2.value)
31.         {
32.           alert("You must provide the same password twice.");
33.           return false;
34.         }
35.         else if (!document.forms.registration.agreement.checked)
36.         {
37.           alert("You must agree to our terms and conditions.");
38.           return false;
39.         }
40.         return true;
41.       }
42.
43.     </script>
44.     <title>form5</title>
45.   </head>
46.   <body>
47.     <form action="dump.php" method="get" name="registration" onsubmit="return validate();">
48.       Email: <input name="email" type="text">
```

```
49.     <br>
50.     Password: <input name="password1" type="password">
51.     <br>
52.     Password (again): <input name="password2" type="password">
53.     <br>
54.     I agree to the terms and conditions: <input name="agreement" type="checkbox">
55.     <br><br>
56.     <input type="submit" value="Submit">
57. </form>
58. </body>
59. </html>
```

```
1. <!--
2.
3. form6.html
4.
5. A form with client-side validation demonstrating jQuery.
6.
7. Computer Science 50
8. David J. Malan
9.
10. -->
11.
12. <!DOCTYPE html>
13.
14. <html>
15.   <head>
16.     <script src="http://code.jquery.com/jquery-latest.js"></script>
17.     <script>
18.
19.       $(document).ready(function() {
20.         $("#registration").submit(function() {
21.           if (!$("#email").val().match(/.+@.+\.edu$/))
22.             {
23.               alert("You must provide a .edu email address.");
24.               return false;
25.             }
26.           else if ($("#password1").val() == "")
27.             {
28.               alert("You must provide a password.");
29.               return false;
30.             }
31.           else if ($("#password1").val() != $("#password2").val())
32.             {
33.               alert("You must provide the same password twice.");
34.               return false;
35.             }
36.           else if (!$("#agreement").attr("checked"))
37.             {
38.               alert("You must agree to our terms and conditions.");
39.               return false;
40.             }
41.           return true;
42.         });
43.       });
44.
45.     </script>
46.     <title>form6</title>
47.   </head>
48.   <body>
```

```
49. <form action="dump.php" id="registration" method="get">
50.   Email: <input id="email" name="email" type="text">
51.   <br>
52.   Password: <input id="password1" name="password1" type="password">
53.   <br>
54.   Password (again): <input id="password2" name="password2" type="password">
55.   <br>
56.   I agree to the terms and conditions: <input id="agreement" name="agreement" type="checkbox">
57.   <br><br>
58.   <input type="submit" value="Submit">
59. </form>
60. </body>
61. </html>
```

```
1. <!-- http://docs.jquery.com/Plugins/Validation -->
2.
3. <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
4.     "http://www.w3.org/TR/html4/loose.dtd">
5. <html>
6. <head>
7.     <script src="http://code.jquery.com/jquery-latest.js"></script>
8.     <script type="text/javascript" src="http://dev.jquery.com/view/trunk/plugins/validate/jquery.validate.js"></script>
9.     <style type="text/css">
10. * { font-family: Verdana; font-size: 96%; }
11. label { width: 10em; float: left; }
12. label.error { float: none; color: red; padding-left: .5em; vertical-align: top; }
13. p { clear: both; }
14. .submit { margin-left: 12em; }
15. em { font-weight: bold; padding-right: 1em; vertical-align: top; }
16. </style>
17.     <script>
18.     $(document).ready(function(){
19.         $("#commentForm").validate();
20.     });
21.     </script>
22.
23. </head>
24. <body>
25.
26.
27.     <form class="cmxform" id="commentForm" method="get" action="">
28.     <fieldset>
29.         <legend>A simple comment form with submit validation and default messages</legend>
30.         <p>
31.             <label for="cname">Name</label>
32.             <em>*</em><input id="cname" name="name" size="25" class="required" minlength="2" />
33.         </p>
34.         <p>
35.             <label for="cemail">E-Mail</label>
36.             <em>*</em><input id="cemail" name="email" size="25" class="required email" />
37.         </p>
38.         <p>
39.             <label for="curl">URL</label>
40.             <em> </em><input id="curl" name="url" size="25" class="url" value="" />
41.         </p>
42.         <p>
43.             <label for="ccomment">Your comment</label>
44.             <em>*</em><textarea id="ccomment" name="comment" cols="22" class="required"></textarea>
45.         </p>
46.         <p>
47.             <input class="submit" type="submit" value="Submit"/>
48.         </p>
```

---

```
49. </fieldset>
50. </form>
51. </body>
52. </html>
```