1) Special keys (e.g. CTRL or ALT or SHIFT)

In case if you press any special key, browser will fire only keydown() event but not keypress() event.

Try Yourself

Try pressing some normal keys and then some special keys.

|  |
| --- |
| <html>  <head>  <script type="text/javascript" src="<http://code.jquery.com/jquery-1.10.2.min.js>"></script>  <script type="text/javascript">  var keydownCounter = 0;  var keypressCounter = 0;  $( document ).ready(function(){        $('#textbox').keydown(function(event){          $('#keydownCounter').html(++keydownCounter);      });        $('#textbox').keypress(function(event){          $('#keypressCounter').html(++keypressCounter);      });    });  </script>  </head>  <body>      <h4>jQuery keydown() and keypress() difference</h4>        <label>TextBox : </label><input id="textbox" type="text" size="50" />        <div>          <label>keydown() event fired : </label><span id="keydownCounter">0</span> times.      </div>        <div>          <label>keypress() event fired : </label><span id="keypressCounter">0</span> times.      </div>  </body>  </html> |

2) Capturing keycode or key [ascii](http://www.asciitable.com/" \o "ascii" \t "_blank) value

If you hit the letter a, and A (capital letter), you will find the below behavior of events.

keydown() will display a = 65, A = 65 (case insensitive).

keypresss() will display a= 97, A=65 (case sensitive).

If you want to capture the real character key in, always use keypress() event.

Try Yourself

Try pressing some keys with [Caps Lock] ON and then after turning it OFF.

|  |
| --- |
| <html>  <head>  <script type="text/javascript" src="<http://code.jquery.com/jquery-1.10.2.min.js>"></script>  <script type="text/javascript">  $( document ).ready(function(){        $('#textbox').keydown(function(event){          var keycode = (event.keyCode ? event.keyCode : event.which);          $('#keydownCode').html(keycode);      });        $('#textbox').keypress(function(event){          var keycode = (event.keyCode ? event.keyCode : event.which);          $('#keypressCode').html(keycode);      });    });  </script>  </head>  <body>      <h4>jQuery keydown() and keypress() difference</h4>        <label>TextBox : </label><input id="textbox" type="text" size="50" />        <div>          <label>keydown() detected keycode : </label><span id="keydownCode">0</span>.      </div>        <div>          <label>keypress() detected keycode : </label><span id="keypressCode">0</span>.      </div>  </body>  </html> |

The event.keyCode is not working in FireFox , but work perfect in IE. To get the event.keyCode in Firefox, you should use the event.which instead, and jQuery recommend it as well. So the better way should be

var keycode = (event.keyCode ? event.keyCode : event.which);

3) When any key is long pressed

In this case, according to docs, keydown() event is triggered once, but the keypress() event will keep triggering until the key is released. Anyway, this is browser specific behavior and I will not suggest you on this feature. This should be limited to your knowledge only.

Try Yourself

Press any key and keep it down.

|  |
| --- |
| <html>  <head>  <script type="text/javascript" src="<http://code.jquery.com/jquery-1.10.2.min.js>"></script>  <script type="text/javascript">  var keydownCounter = 0;  var keypressCounter = 0;  $( document ).ready(function(){        $('#textbox').keydown(function(event){          $('#keydownCounter').html(++keydownCounter);      });        $('#textbox').keypress(function(event){          $('#keypressCounter').html(++keypressCounter);      });    });  </script>  </head>  <body>      <h4>jQuery keydown() and keypress() difference</h4>        <label>TextBox : </label><input id="textbox" type="text" size="50" />        <div>          <label>keydown() event fired : </label><span id="keydownCounter">0</span> times.      </div>        <div>          <label>keypress() event fired : </label><span id="keypressCounter">0</span> times.      </div>  </body>  </html> |