

Tipo de Evento	Elemento afectado	Resumen de Accion
Click	<pre> &lt;div class="grid"&gt;   &lt;div class="card c1" id="cardClick"&gt;     &lt;h2&gt;EVENTO AQUÍ&lt;/h2&gt;     &lt;p&gt;Hazme Click (click event)&lt;/p&gt;   &lt;/div&gt; </pre>	<p>Detecta una pulsación simple y lanza una alerta de confirmación.</p>
Mouse Enter	<pre> &lt;div class="card c2" id="cardHover"&gt;   &lt;h2&gt;EVENTO AQUÍ&lt;/h2&gt;   &lt;p&gt;Lleva tu mouse aquí (mouseenter event)&lt;/p&gt; &lt;/div&gt; </pre>	<p>Cambia el color de fondo de la tarjeta cuando el puntero del mouse entra en su área.</p>
Double Click	<pre> &lt;div class="card c3" id="cardDbClick"&gt;   &lt;h2&gt;EVENTO AQUÍ&lt;/h2&gt;   &lt;p&gt;Double-click aquí (dblclick event)&lt;/p&gt; &lt;/div&gt; </pre>	<p>Cambia el texto del encabezado permanentemente al recibir dos clics rápidos.</p>
Keydown	<pre> &lt;div class="card c4"&gt;   &lt;h2&gt;EVENTO AQUÍ&lt;/h2&gt;   &lt;p&gt;Escribe aquí (keydown event)&lt;/p&gt;   &lt;input type="text" id="inputKey"&gt; &lt;/div&gt; </pre>	<p>Registra cada vez que se presiona una tecla dentro del campo de texto</p>
Focus	<pre> &lt;div class="card c5"&gt;   &lt;h2&gt;EVENTO AQUÍ&lt;/h2&gt;   &lt;p&gt;Focus on (focus event)&lt;/p&gt;   &lt;input type="text" id="inputFocus" placeholder="Haz clic dentro"&gt; &lt;/div&gt; </pre>	<p>Resalta el borde del cuadro de texto cuando el usuario hace clic para escribir.</p>



## PREGUNTAS

1. ¿Utilizó listeners para sus eventos o los describió como propiedades de los elementos? Justifique su elección

R// Utilicé Listeners mediante el método `.addEventListener()`. Elegí esta opción por tres razones principales:

Separación de conceptos: Mantiene el HTML limpio de lógica (evita atributos como `onclick`), facilitando la lectura del código.

Múltiples funciones: Permite agregar varios comportamientos al mismo evento sin sobrescribir los anteriores.

Escalabilidad: Es el estándar moderno, lo que hace que el código sea más fácil de mantener y depurar en el futuro.

2. Seleccione uno de sus eventos y explique porque ese tipo de evento era la mejor elección para esa situación y que otro podría haber utilizado

R// Input: fue la mejor opción para validar lo que el usuario escribe en tiempo real, input es superior porque detecta no solo el teclado, sino también si el usuario pega texto con el mouse o borra contenido, asegurando que la interfaz esté siempre sincronizada con el valor real.

Mientras que otro que pude haber utilizado podría haber sido `keyup`, pero tiene la desventaja de que si el usuario mantiene presionada una tecla, el evento solo se dispara una vez al soltarla, perdiendo la actualización fluida de los datos.