UD5  
Ejercicios Eventos.

short line

[**Ejercicio 1:**](#_5wc91wwk8u4) **1**

[**Ejercicio 2:**](#_8rxeqvrbf3rs) **3**

[**Ejercicio 3:**](#_ou3mdt2rgwl) **4**

**Importante:** Trate de utilizar “addEventListener” para asignar eventos y estaría bien que hiciera uso de la estructura siguiente para asegurarnos que los eventos se añaden cuando el DOM ha sido completamente cargado/construido.

*window.****onload*** *= function() { document.getElementById.addEventListener ( ………..) ………….. }*

## 

## Ejercicio 1:

Dado el siguiente código html:

|  |
| --- |
| <html xmlns="http://www.w3.org/1999/xhtml"> <head>  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />  <title>Ejercicio 1 - DOM básico y atributos HTML</title> </head>  <body>  <div>  <p>  Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Sed mattis enim vitae orci. Phasellus libero. Maecenas nisl arcu, consequat congue, commodo nec, commodo ultricies, turpis. Quisque sapien nunc, posuere vitae, rutrum et, luctus at, pede. Pellentesque massa ante, ornare id, aliquam vitae, ultrices porttitor, pede. Nullam sit amet nisl elementum elit convallis malesuada. Phasellus magna sem, semper quis, faucibus ut, rhoncus non, mi. Duis pellentesque, felis eu adipiscing ullamcorper, odio urna consequat arcu, at posuere ante quam non dolor. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis scelerisque.  </p>  <a href="#">Ocultar contenidos</a>  </div>  <br/>  <div>  <p>  Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Sed mattis enim vitae orci. Phasellus libero. Maecenas nisl arcu, consequat congue, commodo nec, commodo ultricies, turpis. Quisque sapien nunc, posuere vitae, rutrum et, luctus at, pede. Pellentesque massa ante, ornare id, aliquam vitae, ultrices porttitor, pede. Nullam sit amet nisl elementum elit convallis malesuada. Phasellus magna sem, semper quis, faucibus ut, rhoncus non, mi. Duis pellentesque, felis eu adipiscing ullamcorper, odio urna consequat arcu, at posuere ante quam non dolor. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis scelerisque.  </p>  <a href="#">Ocultar contenidos</a>  </div>  <br/>  <div>  <p>  Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Sed mattis enim vitae orci. Phasellus libero. Maecenas nisl arcu, consequat congue, commodo nec, commodo ultricies, turpis. Quisque sapien nunc, posuere vitae, rutrum et, luctus at, pede. Pellentesque massa ante, ornare id, aliquam vitae, ultrices porttitor, pede. Nullam sit amet nisl elementum elit convallis malesuada. Phasellus magna sem, semper quis, faucibus ut, rhoncus non, mi. Duis pellentesque, felis eu adipiscing ullamcorper, odio urna consequat arcu, at posuere ante quam non dolor. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis scelerisque.  </p>  <a href="#">Ocultar contenidos</a>  </div> </body> </html> |

Debes de ser capaz, mediante la asignación de eventos y manejadores de hacer **desaparecer/aparecer** los textos cuando el usuario haga click en los enlaces que les preceden. Por ejemplo, cuando el usuario haga click en el segundo enlace, debe de desaparecer o aparecer el segundo bloque de texto.

El ejercicio debe de ser realizado de 2 formas:

1. Haciendo modificaciones en el código html para que nos resulte más fácil asignar eventos y manejadores.
2. Sin hacer ninguna modificación en el código html (**no se puede añadir ninguna etiqueta**). Aquí la cosa se pone un poco más compleja; a ver que se te ocurre.

**Importante:** Debes de buscar por internet como mostrar/ocultar elementos HTML mediante código javascript. También como no podemos modificar nada del HTML, ¿Cómo vamos a hacer para que desde el código JavaScript podamos hacer referencia a los elementos HTML? Debes de buscar información sobre nuevas formas de buscar elementos (más allá del getElementById).

## 

## Ejercicio 2:

Dado el siguiente código HTML (y CSS):

|  |
| --- |
| <!DOCTYPE html> <html lang="en"> <head>  <meta charset="UTF-8">  <title>Documento</title>  <style>  div {  width: 100px;  height: 100px;  border: 1px solid black;  }    .blanco {  background-color: white;  }    .rojo {  background-color: red;  }    .amarillo {  background-color: yellow;  }  </style> </head> <body>  <div id="prueba" tabindex=1></div> </body> </html> |

Dicho código, como puedes comprobar pinta un recuadro en pantalla. Debes de manejar eventos y asignar manejadores para que cuando el usuario habla click en las teclas “b”, “r”, y “a” pinte el recuadro de “blanco”, “rojo”, y “verde” respectivamente. Para pintar el recuadro, como puedes ver ya existen unas clases en CSS en el código las cuales debes de asignar al elemento “div” cuando alguno de los 3 eventos se produzca.

**Importante:**  Debes de buscar por internet cómo se asignan clases/estilos a un elemento HTML desde código JavaScript, y quizás cómo capturar eventos desde las teclas también.

## 

## Ejercicio 3:

Realiza un código HTML en el que aparezca un texto y 2 botones:

* Sobre el texto vamos a añadirle 2 eventos: **mouseover** y **mouseout** ; pero **solo vamos a utilizar una función manejadora para ambos eventos.** Dentro de esa función manejadora, si el evento es **“mouseover”** el texto se pintará de “rojo”, y con **“mouseout”** volverá a “negro” (color no de la fuente). La dificultad es cómo distinguir dentro de esa función manejadora cual de los 2 eventos se ha producido.
* Sobre los 2 botones (boton1 y boton2) vamos a añadir 1 único evento: **“onclick” y se debe llamar a la misma función “controlarClick()”** . Cuando el usuario cliquea en uno de los botones debe aparecer una ventana/diálogo mostrando el “id” del botón. La dificultad del ejercicio es que solo hay una función manejador para ambos eventos, y dentro de dicha función debemos de saber qué botón está disparando el evento.

**Importante:** Mirar bien en los apuntes en la parte de obtener información de los eventos gracias a la variable “event” ;).