

Tota la documentació ha de tenir portada, cos del treball i bibliografia (si passa d'una pàgina). El document es lliurará sempre en **format pdf**.

Tarea1. Eventos UI

Entrega

El lliurament serà per mitjà d' un document pdf o en format zip a la plataforma moodle del centre, segons especifiqui el professor. També és possible compartir-ho a través de Google Drive, en aquest cas, el document ha de tenir l'opció de "Compartir amb qualsevol que tingui l'enllaç". NO CAL QUE COMPARTEIXI EL DOCUMENT AMB L' ADREÇA DE CORREU DEL PROFESSOR. S'ha de lliurar el document/s amb la següent nomenclatura: Apellido1_Nombre_U2_Task2. En cas d' especificar-se una altra nomenclatura s' ha de prioritzar la indicada en l' activitat.

1. Qualsevol pràctica que estigui copiada d' un company o d' Internet **automàticament en tindrà un 0**.
2. El lliurament s' ha de lliurar dins del límit temporal fixat, si no és així i sense tenir una deguda justificació **tindrà automàticament un 0** en la nota de l' esmentada pràctica.
3. Qui no lliuri el document amb el format que es demana (nom i estil) **tindrà una penalització del 25%** de la nota final que obtingui.

Valoració

Es valorarà a nivell de presentació:

- El format
- L' ortografia
- La informació aportada

Es valoraran les activitats de la següent manera:

1. Activitat 1: 25%
2. Activitat 2: 25%
3. Activitat 3: 25%
4. Activitat 4: 25%

Es valorarà a nivell de preguntes:

- Resultats
- Explicació

Introducción

Ya hemos trabajado con eventos en las prácticas anteriores, pinchando sobre un botón y programando el onclick de este botón, el concepto de lo que es un evento y para qué sirve ya está adquirido. En este tema se insiste en la forma de utilizar los manejadores de eventos: como atributos, como funciones externas, o semánticamente. Son estructuras diferentes cuyas sintaxis debéis conocer.

El objeto event, lo generan todos los manejadores de eventos, pero se utiliza de forma diferente en Internet Explorer que en resto de navegadores. El objeto event tiene propiedades y métodos, trabajaremos con algunas de ellas, por ejemplo con la propiedad type, pero podéis consultar en internet las propiedades y métodos que tiene este objeto.

Se insiste en el tema de los eventos del teclado y del ratón, porque son bastante utilizados y por la diferencia de uso entre ellos.

Hay que entender el flujo de eventos para comprender la sintaxis del addEventListener. En la práctica pasada.

Activitats

S'han d' adjuntar tots els elements en un arxiu amb la nomenclatura:

Apellido1_Apellido2_Nombre_T2Practica2.zip.

Internament s' han de distingir el codic segons l'activitat, per exemple: a l'activitat 1 → T2Activitat2.* (definim l'extensió segons correspongue).

1. Crear una página HTML en la que se van a añadir una serie de eventos asociados a botones.

- *Botón mágico*: Se debe crear un botón que se denomine “Objeto mágico” que al darle **click** sobre el mismo desaparezca.
- *Botón Pulsador*: Se debe crear un botón que se denomine “Pulsador”, donde al pulsarlo debe mostrarse un mensaje “Has pulsado sobre el botón”.
- *Botones deslizantes*: Se deben crear 2 botones que desplieguen y colapsen un listado de opciones, en este caso:
 - Botón 1: Tendrá como definición “Películas”, donde se deberá incluir un submenú que al pulsar sobre el botón despliegue/colapse dicho submenú con vuestras 3 series favoritas.
 - Botón 2: Tendrá como definición “Series”, donde se deberá incluir un submenú que al pulsar sobre el botón despliegue/colapse dicho submenú con vuestras 3 series favoritas.

2. Gestionar el comportamiento de un *tooltip*:

Escribe JavaScript que muestre un *tooltip* sobre un elemento con el atributo data-tooltip. El valor de este atributo debe convertirse en el texto del tooltip.

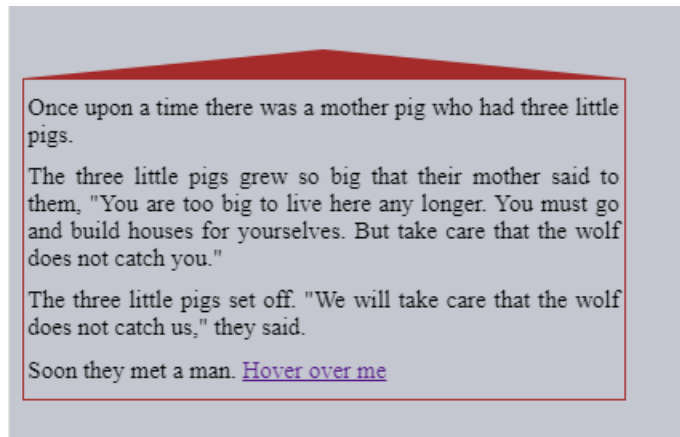
Solamente un tooltip puede aparecer a la vez.

Por ejemplo:

```
<div data-tooltip="Aquí – está el interior de la casa" id="house">
  <div data-tooltip="Aquí – está el techo" id="roof"></div>
  ...
  <a href="https://en.wikipedia.org/wiki/The_Three_Little_Pigs" data-tooltip="Continúa leyendo...">Colócate sobre mí</a>
</div>
```

Un ejemplo de resultado en el iframe (el contenido del mensaje es de elección libre por

parte del alumno):



A continuación el código html, a partir del cual se debe resolver el ejercicio usando código javascript:

Index.html

```
<!DOCTYPE HTML>
<html>
<head>
  <meta charset="utf-8">
  <style>
    body {
      height: 2000px;
      /* El tooltip debe funcionar también despues de hacer scroll */
    }
    .tooltip {
      position: fixed;
      z-index: 100;
      padding: 10px 20px;
      border: 1px solid #b3c9ce;
      border-radius: 4px;
      text-align: center;
      font: italic 14px/1.3 sans-serif;
      color: #333;
      background: #fff;
      box-shadow: 3px 3px 3px rgba(0, 0, 0, .3);
    }
    #house {
      margin-top: 50px;
      width: 400px;
      border: 1px solid brown;
    }
    #roof {
      width: 0;
      height: 0;
      border-left: 200px solid transparent;
      border-right: 200px solid transparent;
      border-bottom: 20px solid brown;
      margin-top: -20px;
    }
  </style>
</head>
<body>
  <div class="house">
    <div class="roof">
      <div class="text">
        <p>Once upon a time there was a mother pig who had three little pigs.
        The three little pigs grew so big that their mother said to them, "You are too big to live here any longer. You must go and build houses for yourselves. But take care that the wolf does not catch you."
        The three little pigs set off. "We will take care that the wolf does not catch us," they said.
        Soon they met a man. Hover over me</p>
      </div>
    </div>
  </div>
</body>
</html>
```

```

    p {
      text-align: justify;
      margin: 10px 3px;
    }
  </style>
</head>

<body>
  <div data-tooltip="Aquí esta el interior de la casa" id="house">
    <div data-tooltip="Aquí está el techo" id="roof"></div>
    <p>El contenido queda a elección del alumno</p>
    <p><a href="https://en.wikipedia.org/wiki/The_Three_Little_Pigs" data-
      tooltip="Continúa leyendo...">Colócate sobre mí</a></p>
  </div>
  <script>
    // ...tu código...
  </script>
</body>
</html>

```

3. Crear un control deslizante:

A continuación, se muestra un ejemplo de referencia para ver cómo quedaría:



Arrastra el pasador azul con el ratón y muévelo. Detalles importantes:

- Cuando el botón del ratón es presionado, durante el arrastrado del ratón puedes ir por arriba o debajo de la barra deslizante. Ésta seguirá funcionando (es lo conveniente para el usuario).
- Si el ratón se mueve muy rápido hacia la izquierda o la derecha, el pasador se detiene exactamente en el borde.

A continuación se muestra el “html” y el “css” sobre el que se te tendrá que resolver el ejercicio:

Index.html

```

<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <link rel="stylesheet" href="style.css">
  </head>
  <body>
    <div id="slider" class="slider">
      <div class="thumb"></div>
    </div>
    <script>
      // ... tu código...
    </script>
  </body>
</html>

```

```
</script>
</body>
</html>
```

Style.css

```
.slider {
  border-radius: 5px;
  background: #E0E0E0;
  background: linear-gradient(left top, #E0E0E0, #EEEEEE);
  width: 310px;
  height: 15px;
  margin: 5px;
}
.thumb {
  width: 10px;
  height: 25px;
  border-radius: 3px;
  position: relative;
  left: 10px;
  top: -5px;
  background: blue;
  cursor: pointer;
}
```

4. Arrastrar objetos por el campo

Esta tarea te puede ayudar a comprobar tu entendimiento de varios aspectos de Arrastrar y Soltar, y del DOM. Hacer que todos los elementos con clase draggable sean arrastrables.

Requerimientos:

- Usa delegación de eventos para detectar el inicio del arrastrado: un solo manejador de eventos en el document para mousedown.
- Si los elementos son arrastrados a los bordes superior/inferior de la ventana: la página se desliza hacia arriba/abajo para permitir dicho arrastre.
- Sin desplazamiento horizontal (esto hace la tarea un poco más simple, añadirlo es fácil).
- Los elementos arrastrables o sus partes nunca deben dejar la ventana, incluso después de movimientos rápidos del ratón.

A continuación, se muestra el código html y css, a partir del cual se debe generar el código javascript necesario para su resolución (se recomienda incluir dicho código en un archivo js).

Index.html

```

<!DOCTYPE HTML>
<html>

<head>
  <meta charset="utf-8">
  <link rel="stylesheet" href="heroes.css">
</head>
<body>
  <h2>Ubica los superhéroes por el campo.</h2>
  <p>Los superhéroes y la pelota son elementos con la clase "draggable". Haz que sean realmente arrastrables.</p>
  <p>Importante: limita el arrastre a la ventana. Si el arrastrable alcanza el borde inferior o superior de la ventana, la página debe desplazarse para permitir seguir arrastrando.</p>
  <p>Si en tu monitor cabe el documento entero, haz la ventana del navegador más pequeña para que aparezca la barra de desplazamiento vertical y así puedas probarlo.</p>
  <p>En esta tarea es suficiente manejar solo el desplazamiento vertical. No suele usarse el desplazamiento horizontal, y el manejo es similar si se necesita.</p>
  <p>Y algo más: los héroes nunca pueden dejar la página. Si alcanzan el borde del documento, no pueden arrastrarse fuera de él.</p>
  <div id="field">
  </div>
  <div class="hero draggable" id="hero1"></div>
  <div class="hero draggable" id="hero2"></div>
  <div class="hero draggable" id="hero3"></div>
  <div class="hero draggable" id="hero4"></div>
  <div class="hero draggable" id="hero5"></div>
  <div class="hero draggable" id="hero6"></div>
  
  <div style="clear:both"></div>
  <script src="heroes.js"></script>
</body>
</html>

```

Heroes.css

```

html, body {
  margin: 0;
  padding: 0;
}
#field {
  background: url(field.svg);
  width: 800px;
  height: 500px;
  float: left;
}
/* heroes and the ball (dragables) */
.hero {
  background: url(https://js.cx/drag-heroes/heroes.png);
  width: 130px;
  height: 128px;
}

```

```
    float: left;
}
#hero1 {
    background-position: 0 0;
}
#hero2 {
    background-position: 0 -128px;
}
#hero3 {
    background-position: -120px 0;
}
#hero4 {
    background-position: -125px -128px;
}
#hero5 {
    background-position: -248px -128px;
}
#hero6 {
    background-position: -244px 0;
}
.draggable {
    cursor: pointer;
}
```