

# Eventos

# Definiciones

- **Evento** – Algo que sucede.
  - Interacción del usuario (click, scroll,...)
  - Error en la carga de imágenes
  - Modificación del HTML
  - Fin de la carga de la página
  - Cambio de estado de una petición al servidor
  - ...
- **Manejador de evento (handler)** – Función que se ejecuta ante un evento.
- **Registrar el evento** – Asociar un evento a un manejador

# Registro en línea de eventos

- En la propia etiqueta HTML incluimos el código a ejecutar

```
<a href="pagina.html" onclick="f1();f2();f3();">Pulsa aqui</a>
```

- Inconveniente - Mezclamos código con el HTML

# Registro de eventos tradicional

- Se realiza desde el código.

- Se necesita una referencia al objeto que genera el evento.
  - objetoDeLaPagina.onNombreEvento = handler;

- Registrar el evento

```
oBoton = document.getElementById("btnAceptar");  
oBoton.onclick = procesarAceptar; // Sin paréntesis
```

- Borrar el registro de un evento

```
oBoton.onclick = null;
```

- Forzar que se ejecute el manejador

```
oBoton.onclick();
```

- Más de un manejador de evento

```
oBoton.onclick = f1;
```

```
oBoton.onclick = f2; // Sobrescribe el manejador
```

- Solución:

```
oBoton.onclick = function (){ f1(); f2(); };
```

# Modelo de registro de eventos según W3C

- Se realiza desde el código.

- Se necesita una referencia al objeto que genera el evento.

```
objeto.addEventListener('evento', manejador, true | false);
```

- Existe un método para borrar el registro del evento.

```
objeto.removeEventListener('evento', manejador, true | false);
```

- Fase de tratamiento del evento

- Tercer parámetro false --> Fase de bubbling o burbujeo

- Tercer parámetro true --> Fase de captura

- Ejemplo:

```
oBoton = document.getElementById("btnAceptar");
```

```
oBoton.addEventListener('click', f1, false);
```

```
oBoton.addEventListener('click', f2, false);
```

```
oBoton.removeEventListener('click', f1, false);
```

# Propagación de eventos

- Existe un orden de disparo de los eventos

```
<body>  
  <div>  
    <h2>Hola</h2>  
  </div>  
</body>
```

- Si hago click en `<h2>` → también hago click sobre `<div>` → también hago click sobre `<body>`
- El evento se propaga desde el elemento contenedor más externo hacia el elemento más interno donde se produce el evento.

# Propagación de eventos (II)

- Fase de captura – el evento desciende hasta el elemento más interno. (eventPhase = 1)

1º Click en body

2º Click en div

- Fase onTarget – el evento se encuentra (eventPhase = 2 // objeto target)

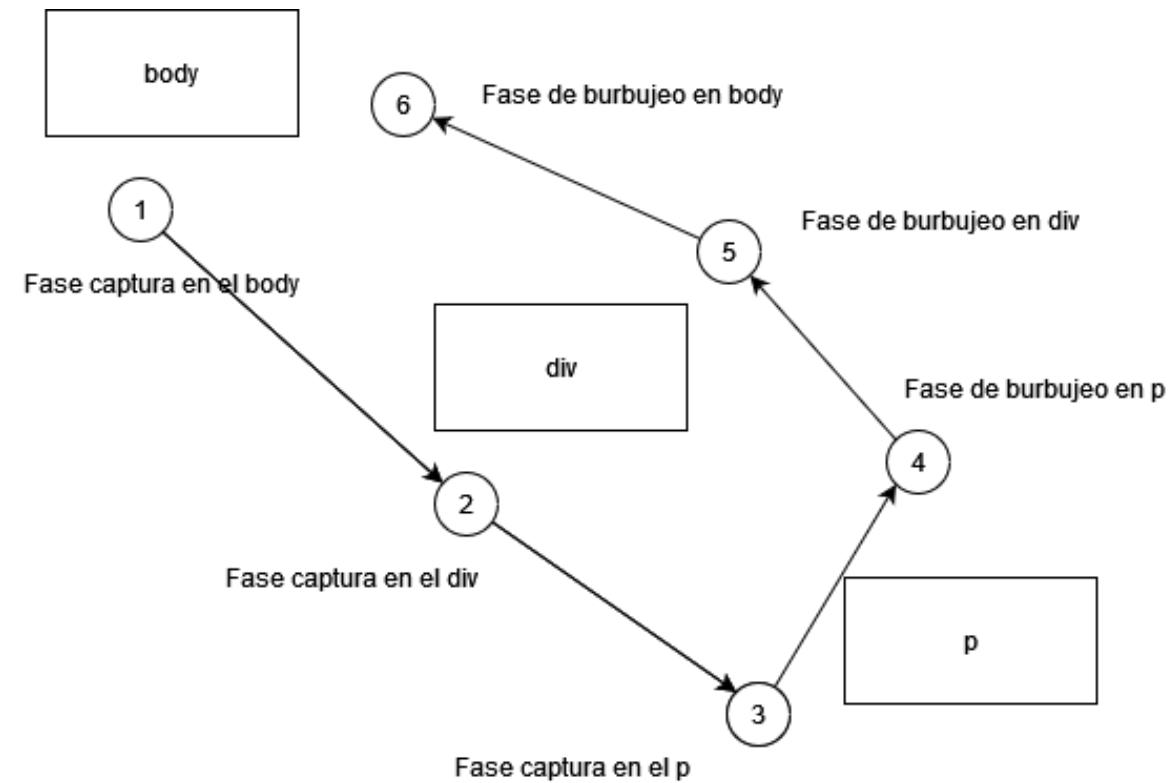
3º Click en h2 (bajando)

4º Click en h2 (subiendo)

- Fase de burbujeo o bubbling – el evento asciende hasta el elemento más externo (eventPhase = 3)

5º Click en div

6º Click en body



# Objeto asociado al evento

- Cuando se produce un evento hay información de contexto relacionada. Hay un objeto que contiene esa información (objeto Event).
  - Por ejemplo, si hago click con el ratón, puedo saber en qué posición de la pantalla se ha producido mediante las coordenadas en pixeles.
  - Otro ejemplo, si se pulsa una tecla, puedo saber si estaba pulsado el botón de mayúsculas a la vez.
- El objeto del evento se recoge en la función manejadora como parámetro de entrada.

```
function manejadora(oEvento){  
    let oE = oEvento || window.event;  
  
    // Acceso a propiedades y métodos de oE  
}
```

# Propiedades del objeto del evento

- **target** - Indica en que objeto más interno que se produce el evento
- **currentTarget** - Indica el objeto para el que estamos procesando el evento
- **eventPhase** - Indica la fase de propagación del evento
  - Valores
    - 1 - Captura
    - 2 - onTarget
    - 3 - Burbujeo

# Eventos de ratón

- **click** - botón izquierdo
- **dblclick** - click dos veces
- **contextmenu** - botón derecho
- **mousedown** - se pulsa una tecla del ratón
- **mouseup** - se libera una tecla del ratón
- **mouseenter** - cuando el cursor entra en una zona
- **mouseout** - cuando el cursor sale de una zona
- **mouseover** - cuando el cursor entra en un elemento y también cuando entra en un elemento hijo
- **mousemove** - cuando el cursor se mueve (se repite)

# Eventos de teclado

- **keydown**—Se produce al presionar la tecla (pulsación de bajada) y de forma repetida durante el tiempo que la tecla permanece presionada.
- **keypress**—Se produce repetidamente durante el tiempo que la tecla permanece presionada. No se genera este evento ante teclas modificadoras: Ctrl, Mayúsculas (shift), Alt o AltGr.
- **keyup**—Se produce en el momento de soltar la tecla.