

# Gestión de Eventos II

# Eventos de Ratón

Información adicional...

- Se accede a ella mediante `event.propiedad` (en FireFox) o `window.event.propiedad` en IE.

Propiedad del objeto Event	Descripción
<code>altKey, ctrlKey, shiftKey</code>	Valor lógico que indica si se pulsó la tecla alt, ctrl o shift
<code>button</code>	Indica el botón que se ha pulsado; 0 para el botón izquierdo, 1 para el derecho y 2 para el central. En el modelo de eventos de Internet Explorer los valores son 1, 2 y 4.
<code>clientX, clientY</code>	Coordenadas en pixel de dónde se ha producido el evento dentro del área cliente.
<code>relatedTarget</code>	En <code>mouseover</code> indica el nodo que ha dejado el ratón, en <code>mouseout</code> indica el nodo al que ha salido. En Internet Explorer la información se guarda en las propiedades <code>toElement</code> y <code>fromElement</code>
<code>screenX, screenY</code>	Coordenadas en pixel dónde se ha producido el evento dentro de la pantalla.

# Eventos de Ratón

## Click

- Información adicional proporcionada a través del **objeto event**:
- `screenX`, `screenY`, `clientX`, `clientY`, `altKey`, `ctrlKey`, `shiftKey`, `button`

.

# Eventos de Ratón

## mousedown y mouseup

- ❑ `mousedown` (se corresponde con la propiedad `html onmousedown`).
  - Información adicional proporcionada a través del objeto event:
    - ✓ `screenX, screenY, clientX, clientY, altKey, ctrlKey, shiftKey, button.`
- ❑ `mouseup` (se corresponde con la propiedad `html onmouseup`).
  - Información adicional proporcionada a través del objeto event:
    - ✓ `screenX, screenY, clientX, clientY, altKey, ctrlKey, shiftKey, button.`

# Eventos de Ratón

- ❑ `mouseover` (se corresponde con la propiedad `html onmouseover`).
  - Información adicional proporcionada a través del objeto `event`:
    - ✓ `screenX`, `screenY`, `clientX`, `clientY`, `altKey`, `ctrlKey`, `shiftKey`, `button`, `relatedTarget`.
- ❑ `mousemove` (se corresponde con la propiedad `html onmousemove`).
  - Información adicional proporcionada a través del objeto `event`:
    - ✓ `screenX`, `screenY`, `clientX`, `clientY`, `altKey`, `ctrlKey`, `shiftKey`, `button`.
- ❑ `mouseout` (se corresponde con la propiedad `html onmouseout`).
  - Información adicional proporcionada a través del objeto `event`:
    - ✓ `screenX`, `screenY`, `clientX`, `clientY`, `altKey`, `ctrlKey`, `shiftKey`, `button`, `relatedTarget`.
      - `relatedTarget` hace referencia al control dónde sale el ratón en `mouseover` y a dónde va en `mouseout`.

# Eventos de Teclado

Se producen en el orden keydown, keypress y keyup respectivamente.

- keydown y keyup permiten detectar teclas especiales (alt,ctrl, retroceso, etc.).
- Información adicional sobre el evento.
- altKey, ctrlKey, shiftKey devuelven un valor lógico que indica si se ha pulsado la tecla alt, ctrl o shift.

# Eventos de Teclado

Detección de la tecla pulsada.

- En Firefox.

En el evento keypress, charCode y which devuelven el código de la tecla pulsada.

- En los eventos keyup y keydown, keyCode y which devuelven el código de la **mayúscula de la tecla pulsada (si se ha pulsado “a” devuelve 65).**

- En Internet Explorer (hasta la versión 8).

En el evento keypress, keyCode devuelve el código de la tecla pulsada.

En los eventos keyup y keydown, keyCode devuelve el código de la mayúscula de la tecla pulsada.

- Distinguir entre IE y FireFox.

En Firefox, se manda el evento en forma de un objeto event a la función que lanza el evento.

o Si el argumento es nulo, es que el script se está ejecutando en IE.

# Notas

- Evento **keydown**:
  - Mismo comportamiento en todos los navegadores:
    - Propiedad `keyCode`: código interno de la tecla
    - Propiedad `charCode`: no definido
- Evento **keypress**:
  - Internet Explorer:
    - Propiedad `keyCode`: el código del carácter de la tecla que se ha pulsado
    - Propiedad `charCode`: no definido
  - Resto de navegadores:
    - Propiedad `keyCode`: para las teclas normales, no definido. Para las teclas especiales, el código interno de la tecla.
    - Propiedad `charCode`: para las teclas normales, el código del carácter de la tecla que se ha pulsado. Para las teclas especiales, 0.
- Evento **keyup**:
  - Mismo comportamiento en todos los navegadores:
    - Propiedad `keyCode`: código interno de la tecla
    - Propiedad `charCode`: no definido
- Para convertir el código de un carácter (no confundir con el código interno) al carácter que representa la tecla que se ha pulsado, se utiliza la función `String.fromCharCode()`.



# Eventos de Teclado

```
function eventoKeyPress(event){  
  //Si event no es nulo (estamos en Firefox), la condición del if es  
  verdadera  
  . if(event)  
    alert("keypress \ncharCode:" + event.charCode +  
    "\nkeyCode: " + event.keyCode + "\nwhich: " + event.which);  
  else  
    //Si event es falso estamos en FireFox  
    alert("keypress \nkeyCode: " + window.event.keyCode)
```

# preventDefault

- Para anular la acción de un evento utilizamos este método del objeto event.
- Sintaxis:

```
e.preventDefault();
```

# Eventos de html

Los eventos html sí son estándar y están recogidos en la especificación del W3C.

- **Load.**
  - Se produce cuando se ha acabado de cargar el documento.
- **unload.**
  - Se produce cuando se elimina el documento de una ventana.
  - Aplicable a los elementos body y frameset.
- **abort.**
  - Se produce cuando se cancela la carga de un documento.
- **error.**
  - Se produce cuando no se carga correctamente un documento, objeto o marco.
  - Aplicable a los elementos body, object y frameset.
- **select.**
  - Se produce cuando el usuario selecciona texto.
  - Aplicable a los elementos input y textarea.
- **change.**
  - Se produce cuando un elemento pierde el foco y se ha modificado su contenido.
  - Aplicable a los elementos input, select y textarea.

# Eventos html

- **submit.**
  - Se produce cuando se envía un formulario.
  - Sólo se aplica el elemento form.
- **reset.**
  - Se produce cuando inicializa un formulario.
  - Sólo se aplica el elemento form.
- **focus.**
  - Se produce cuando un elemento entra en foco.
  - Aplicable a los elementos label, input, select, textarea, y button.
- **blur.**
  - Se produce cuando un elemento pierde el foco.
  - Aplicable a los elementos label, input, select, textarea, y button.
- **resize.**
  - Se produce cuando el documento se redimensiona.
- **scroll.**
  - Se produce cuando se hace un desplazamiento del documento.