

Introduction To Events



EVENTS



Eventos – tipo de eventos

UI Events

ocorrem com a interacção do utilizador com a interface
(load, unload, error, resize, scroll)

Keyboard Events

interacção com o teclado
(keydown, keyup, keypress)

Pointer Events

interacção com o rato, trackpad ou touchscreen
(click, dblclick, mousedown, mouseup, mousemove, mouseover, mouseout, contextmenu)



Eventos – tipo de eventos

Focus Events

quanto activamos ou desactivamos um link ou campo de formulário
(focus/focusin, blur/focusout)

Form Events

interacção com um formulário
(input, change, submit, reset, cut, copy, paste, select)



Eventos – event handling

Como é que os eventos são “despoletados” (triggered)

1. Seleccionamos o elemento onde irá decorrer a interacção/acção
2. Definimos o evento que queremos “escutar”
3. Definimos por uma função (método) o código a executar.



Eventos – “Bind” a elementos

HTML “event handlers” (bad practice)
“event handlers” tradicionais com funções
DOM eventListeners



Eventos – “HTML handlers”



```
//.... HTML
<input type="text" id="username" onBlur="checkUsername()" />

//..... JS
function checkUsername(){
    if(username.value.length < 5 ){
        feedback.textContent = 'mais de 5 caracteres...';
    } else {
        feedback.textContent = '';
    }
}
```



Eventos – “Event handlers”



```
username.onblur = checkUsername;

function checkUsername(){
    if(username.value.length < 5 ){
        feedback.textContent = 'mais de 5 caracteres...';
    } else {
        feedback.textContent = '';
    }
}

/// ..... OU .....

username.onblur = function (){
    if(username.value.length < 5 ){
        feedback.textContent = 'mais de 5 caracteres...';
    } else {
        feedback.textContent = '';
    }
}
```



Eventos – “Event Listeners”



```
username.addEventListener('blur', check, false);

function check(){
    if(username.value.length < 5 ){
        feedback.textContent = 'mais de 5 caracteres...';
    } else {
        feedback.textContent = '';
    }
};

/// .... OU .....

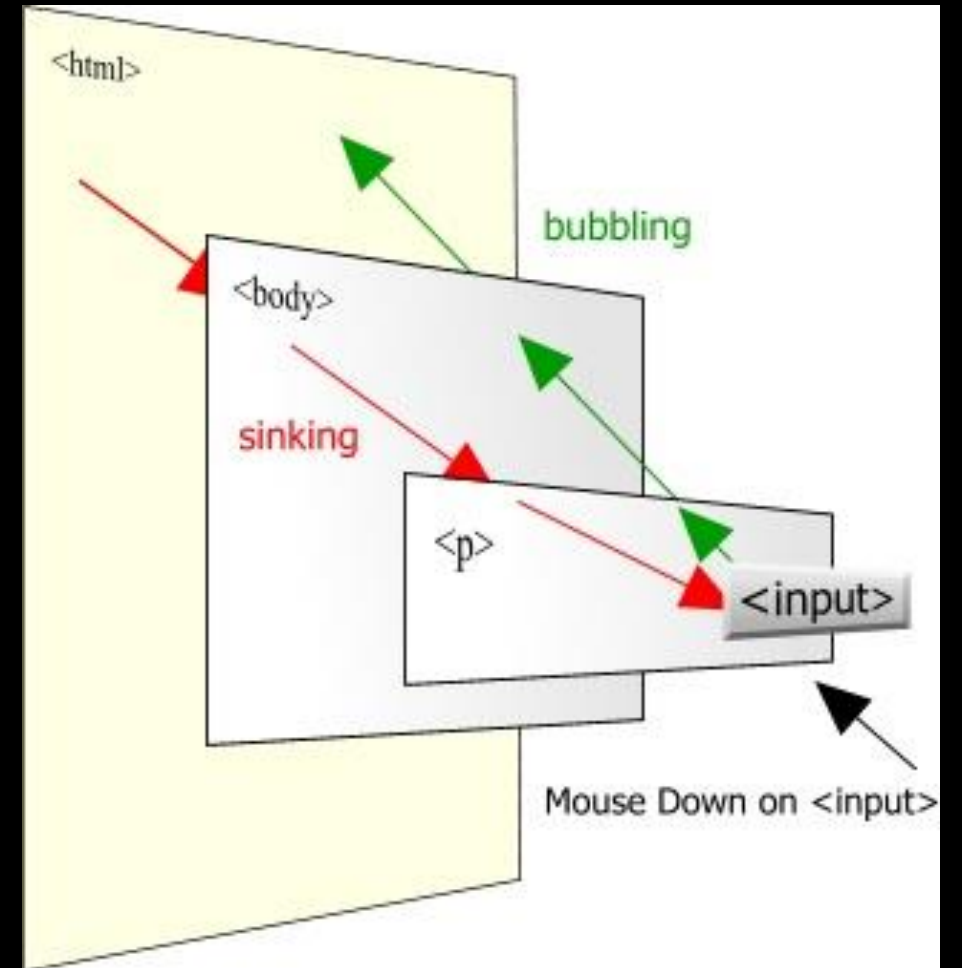
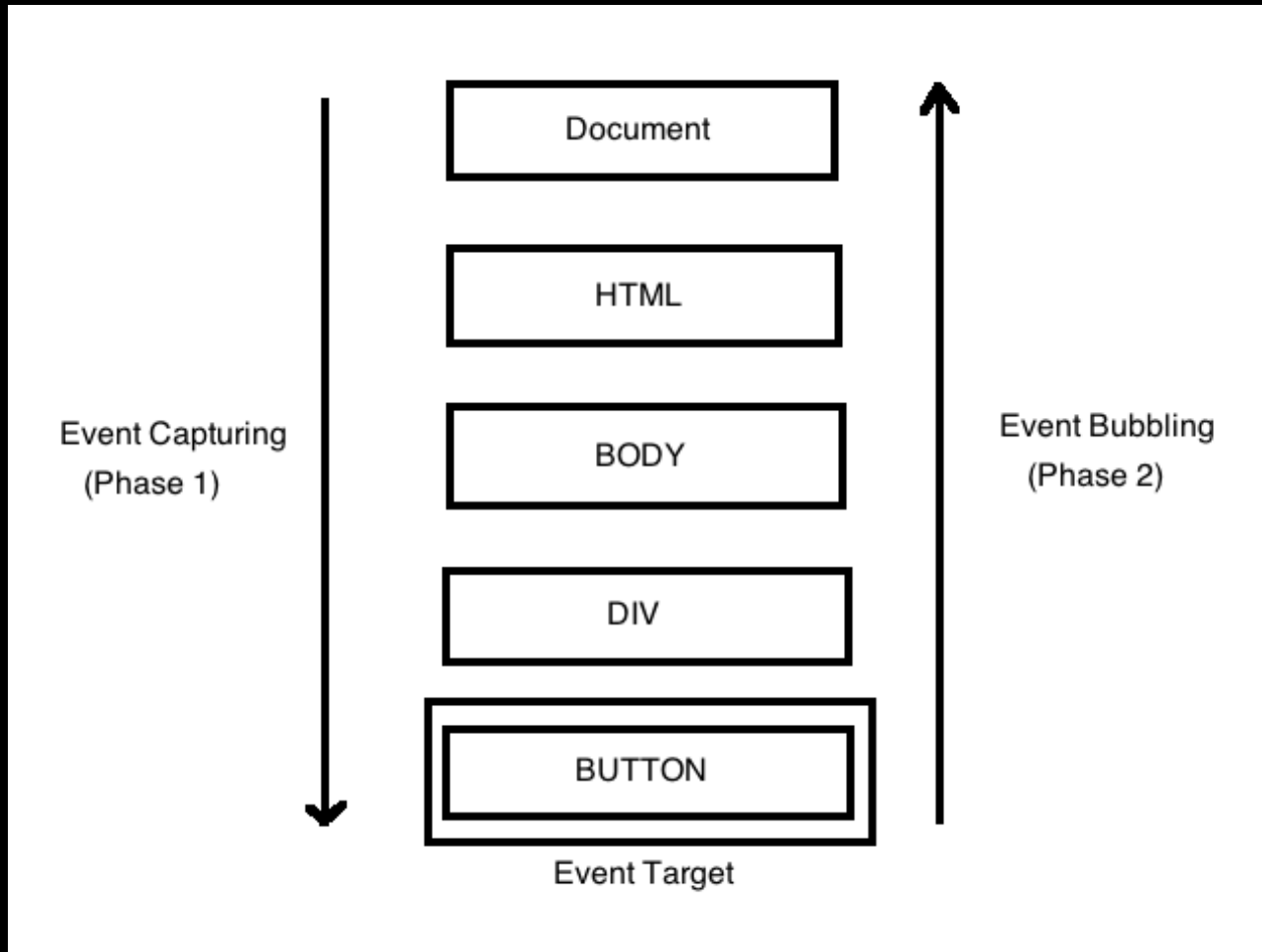
username.addEventListener('blur', function(){
    if(username.value.length < 5 ){
        feedback.textContent = 'mais de 5 caracteres...';
    } else {
        feedback.textContent = '';
    }
}, false);
```



Ver exemplos...



Event Flow



Exemplos...



Event Object

Cada ocorrência de um evento gera um objecto com propriedades e métodos a que chamamos “Event Object”

As propriedades diferem de acordo com o objecto onde ocorre o evento

Propriedades (ex: target, type, cancelable, ...)

Métodos (ex: preventDefault(), stopPropagation(), ...)



Event Object



```
let username = document.querySelector('#username');

username.addEventListener('blur', check, false);

function check(event){
  var target = e.target;
}

///// Com parametros

let username = document.querySelector('#username');

username.addEventListener('blur', function(e){check(e, 5)}, false);

function check(event, minLength){
  var target = e.target;
}
```



Event Delegation

Criar demasiados “Event Listeners” acaba por poder influenciar na performance de uma página.

Podemos (baseados no “flow” do evento) definir um event listener a um elemento pai, e captar o “target” dentro deste onde ocorreu o evento.



exemplo...



Exercicio...

