



Aitor Hernández Sánchez  
Adrián Fleitas de la Rosa



# Index

1. How did we get here...
2. What is jQuery?
3. Why using it?
4. Installation and use
5. Syntax
6. Good practices
7. Examples
8. AJAX
9. Plugins
10. Conclusions

1. How did we get here...

# 1. How did we get here...

- 1990-1995: HTML, CSS and JavaScript are invented.

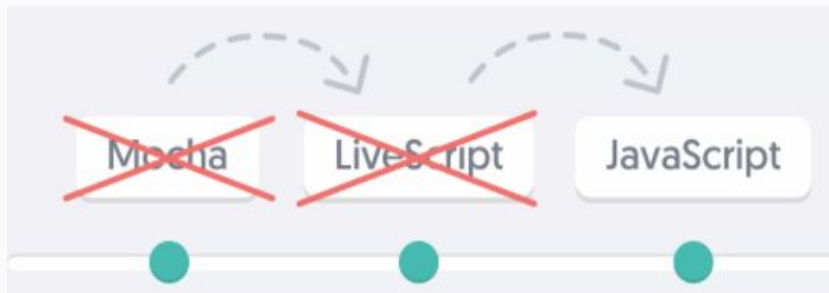
## Cascading HTML style sheets

Date: Tue, 29 Oct 91 10:03:11 GMT+0100  
From: timbl (Tim Berners-Lee)  
Message-Id: <9110290903.AA07413@ nxoc01.cern.ch >  
To: connolly@pixel.convex.com, www-talk  
Subject: Re: status. Re: X11 BROWSER for WWW

Dan,

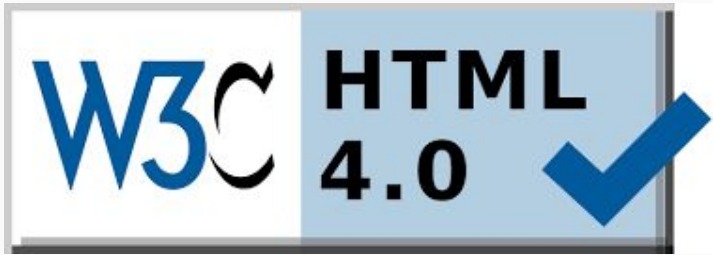
> I've made some tangible progress on the X11 browser, so I though  
> I'd let you know.  
> ...  
> This code is not in any shape to distribute, or even show anybody.  
> But it works, and it's pretty speedy. That's enough to encourage me  
> to polish it off.

[Håkon W Lie](#)  
[howcome@info.cern.ch](mailto:howcome@info.cern.ch)  
10 Oct 1994



# 1. How did we get here...

- 1996–1999 : Standardization efforts begin.



# 1. How did we get here...

- 2000–2006 : CSS frameworks begin to emerge.

jQuery is born.



# 1. How did we get here...

- 2006–2009 : Responsive designs start to urge.



**iPhone  
2007**

# 1. How did we get here...

- 2010–2015: JavaScript Frameworks are born.





## 2. What is jQuery?

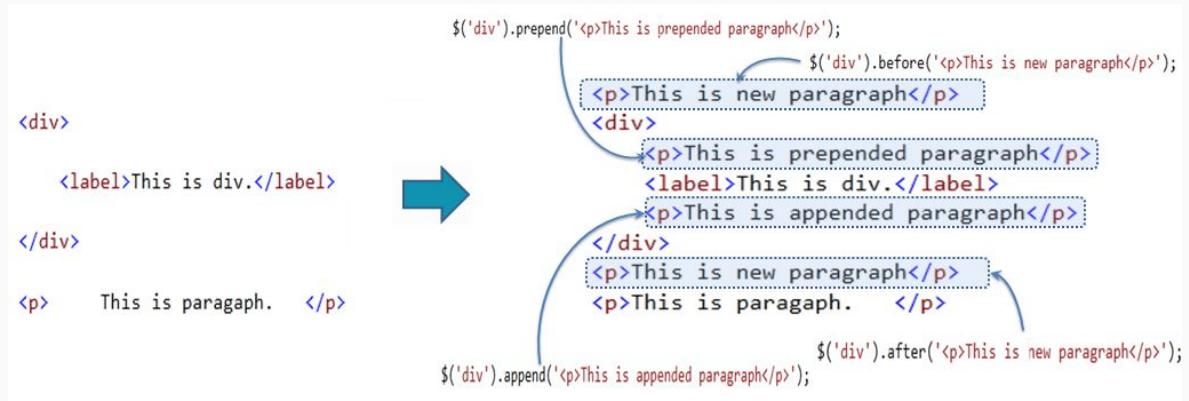
## 2. ¿What is jQuery?

- JavaScript Cross-Browser library



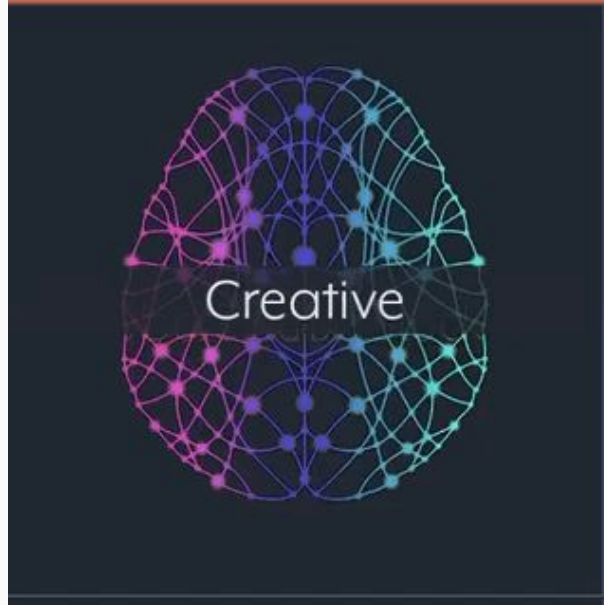
## 2. ¿What is jQuery?

- DOM traversal
- DOM manipulation



## 2. ¿What is jQuery?

- Event handling
- Animation
- AJAX



## 2. ¿What is jQuery?

- Created by John Resig



### 3. Why using it?

### 3. ¿Why using it?

- jQuery is widely used
- Open-Source
- Promotes simplicity



### 3. ¿Why using it?

- Cross-Browser
- Clean syntax
- Lightweight



`$(document).ready();`

`≈ 82 KB`



### 3. ¿Why using it?

- SEO friendly
- Utility features



- jQuery.extend()
- jQuery.grep()
- jQuery.merge()

## 4. Installation and use

# 4. Installation and use

1. `<script> https://code.jquery.com/jquery-3.2.1.min.js </script>`

2. `$ npm i -D jquery`

```
APP.use('/jquery', EXPRESS.static(PATH.join(__dirname, '../node_modules/jquery/dist/')));
```

```
import * as jQuery from './jquery/jquery.js';
```

# 5. Syntax

# 5. Syntax

SELECTORS			ATTRIBUTES / CSS		MANIPULATION	TRAVERSING
<b>Basics</b> * .class element #id selector1, selectorN, ...	<b>Visibility Filters</b> :hidden :visible  <b>Attribute</b> [name]="value" [name*]="value" [name~]="value" [name\$]="value" [name]="value" [name^]="value" [name]="value"[name2]="value2"]	<b>Forms</b> :button :checkbox :checked :disabled :enabled :focus :file :image :input :password :radio :reset :selected :submit :text	<b>Attributes</b> .attr() .prop() .removeAttr() .removeProp() .val()  <b>CSS</b> .addClass() .css() jQuery.cssHooks jQuery.cssNumber jQuery.escapeSelector() .hasClass() .removeClass() .toggleClass()	<b>Dimensions</b> .height() .innerHeight() .innerWidth() .outerHeight() .outerWidth() .width()  <b>Offset</b> .offset() .offsetParent() .position() .scrollLeft() .scrollTop()  <b>Data</b> jQuery.data() .data() jQuery.hasData() jQuery.removeData() .removeData()	<b>Copying</b> .clone()  <b>DOM Insertion, Around</b> .wrap() .wrapAll() .wrapInner()  <b>DOM Insertion, Inside</b> .append() .appendTo() .html() .prepend() .prependTo() .text()  <b>DOM Insertion, Outside</b> .after() .before() .insertAfter() .insertBefore()  <b>DOM Removal</b> .detach() .empty() .remove() .unwrap()  <b>DOM Replacement</b> .replaceAll() .replaceWith()	<b>Filtering</b> .eq() .filter() .first() .has() .is() .last() .map() .not() .slice()  <b>Miscellaneous Traversing</b> .add() .addBack() .andSelf() .contents() .each() .end()  <b>Tree Traversal</b> .children() .closest() .find() .next() .nextAll() .nextUntil() .parent() .parents() .parentsUntil() .prev() .prevAll() .prevUntil() .siblings()

# 5. Syntax

EVENTS		EFFECTS	AJAX	CORE		
<b>Browser Events</b> <code>.error()</code> <code>.resize()</code> <code>.scroll()</code>		<b>Basics</b> <code>.hide()</code> <code>.show()</code> <code>.toggle()</code>	<b>Global Ajax Event Handlers</b> <code>.ajaxComplete()</code> <code>.ajaxError()</code> <code>.ajaxSend()</code> <code>.ajaxStart()</code> <code>.ajaxStop()</code> <code>.ajaxSuccess()</code>	<b>jQuery Object</b> <code>jQuery()</code> <code>jQuery.noConflict()</code> <code>jQuery.sizzle()</code> <code>jQuery.holdReady()</code> <code>jQuery.when()</code>		
<b>Document Loading</b> <code>.load()</code> <code>.ready()</code> <code>.unload()</code>		<b>Custom</b> <code>.animate()</code> <code>.clearQueue()</code> <code>.delay()</code> <code>.dequeue()</code> <code>jQuery.dequeue()</code> <code>.finish()</code> <code>jQuery.fx.interval</code> <code>jQuery.fx.off</code> <code>jQuery.speed</code> <code>.queue()</code> <code>jQuery.queue()</code> <code>.stop()</code>	<b>Helper Functions</b> <code>jQuery.param()</code> <code>.serialize()</code> <code>.serializeArray()</code>	<b>Utilities</b> <code>jQuery.boxModel</code> <code>jQuery.browser</code> <code>jQuery.contains()</code> <code>jQuery.each()</code> <code>jQuery.extend()</code> <code>jQuery.globalEval()</code> <code>jQuery.grep()</code> <code>jQuery.inArray()</code> <code>jQuery.isArray()</code> <code>jQuery.isEmptyObject()</code> <code>jQuery.isFunction()</code> <code>jQuery.isNumeric()</code> <code>jQuery.isPlainObject()</code> <code>jQuery.isWindow()</code> <code>jQuery.isXMLDoc()</code> <code>jQuery.makeArray()</code> <code>jQuery.map()</code> <code>jQuery.merge()</code> <code>jQuery.noop()</code> <code>jQuery.now()</code> <code>jQuery.parseHTML()</code> <code>jQuery.parseJSON()</code> <code>jQuery.parseXML()</code> <code>jQuery.proxy()</code> <code>jQuery.support</code> <code>jQuery.trim()</code> <code>jQuery.type()</code> <code>jQuery.unique()</code> <code>jQuery.uniqueSort()</code>		
<b>Event Handler Attachment</b> <code>.bind()</code> <code>.delegate()</code> <code>.die()</code> <code>.live()</code> <code>.off()</code> <code>.on()</code> <code>.one()</code> <code>.trigger()</code> <code>.triggerHandler()</code> <code>.unbind()</code> <code>.undelegate()</code>		<b>Event Object</b> <code>event.currentTarget</code> <code>event.delegateTarget</code> <code>event.data</code> <code>event.isDefaultPrevented()</code> <code>event.isImmediatePropagationStopped()</code> <code>event.isPropagationStopped()</code> <code>event.metaKey</code> <code>event.namespace</code> <code>event.pageX</code> <code>event.pageY</code> <code>event.preventDefault()</code> <code>event.relatedTarget</code> <code>event.result</code> <code>event.stopImmediatePropagation()</code> <code>event.stopPropagation()</code> <code>event.target</code> <code>event.timeStamp</code> <code>event.type</code> <code>event.which</code>	<b>Low-Level Interface</b> <code>jQuery.ajax()</code> <code>jQuery.ajaxPrefilter()</code> <code>jQuery.ajaxSetup()</code> <code>jQuery.ajaxTransport()</code>	<b>Deferred Object</b> <code>jQuery.Deferred()</code> <code>deferred.always()</code> <code>deferred.done()</code> <code>deferred.fail()</code> <code>deferred.isRejected()</code> <code>deferred.isResolved()</code> <code>deferred.notify()</code> <code>deferred.notifyWith()</code> <code>deferred.pipe()</code> <code>deferred.progress()</code> <code>deferred.promise()</code> <code>deferred.reject()</code> <code>deferred.rejectWith()</code> <code>deferred.resolve()</code> <code>deferred.resolveWith()</code> <code>deferred.state()</code> <code>deferred.then()</code> <code>.promise()</code>		
<b>Form Events</b> <code>.blur()</code> <code>.change()</code> <code>.focus()</code> <code>.focusin()</code> <code>.focusout()</code> <code>.select()</code> <code>.submit()</code>		<b>Fading</b> <code>.fadeIn()</code> <code>.fadeOut()</code> <code>.fadeTo()</code> <code>.fadeToggle()</code>	<b>Shorthand Methods</b> <code>jQuery.get()</code> <code>jQuery.getJSON()</code> <code>jQuery.getScript()</code> <code>jQuery.post()</code> <code>.load()</code>	<b>DOM Element Methods</b> <code>.get()</code> <code>.index()</code> <code>.size()</code> <code>.toArray()</code>		
<b>Keyboard Events</b> <code>.keydown()</code> <code>.keypress()</code> <code>.keyup()</code>		<b>Sliding</b> <code>.slideDown()</code> <code>.slideToggle()</code> <code>.slideUp()</code>		<b>Internals</b> <code>.jquery</code> <code>.context</code> <code>jQuery.error()</code> <code>.length</code> <code>.pushStack()</code> <code>.selector</code>		
				<b>Callbacks Object</b> <code>jQuery.Callbacks()</code> <code>callbacks.add()</code> <code>callbacks.disable()</code> <code>callbacks.disabled()</code> <code>callbacks.empty()</code> <code>callbacks.fire()</code> <code>callbacks.fired()</code> <code>callbacks.fireWith()</code> <code>callbacks.has()</code> <code>callbacks.lock()</code> <code>callbacks.locked()</code> <code>callbacks.remove()</code>		

# 5. Syntax

- Html tag -> \$('p')

```
let $paragraph = $('p');  
$paragraph.text('Jquery is easy!');  
$paragraph.css('margin-left', 50);  
$paragraph.css('color', 'orange');
```

Jquery is easy!

# 5. Syntax

- By class -> \$('.someclass')

Jquery is simple!

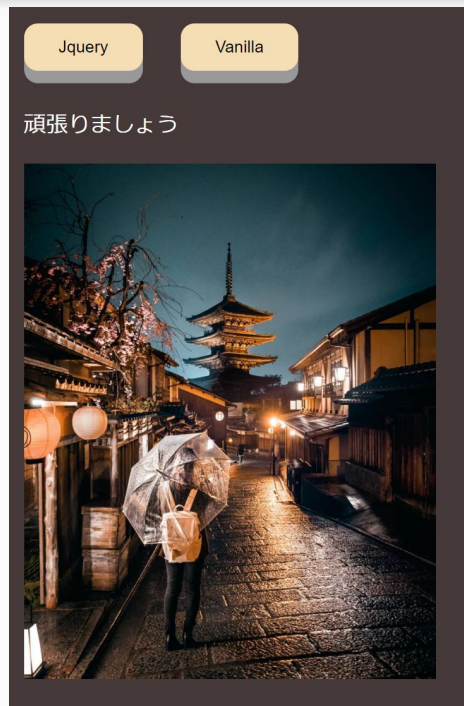
```
$('.paragraph').text('Jquery is simple!');  
$('.paragraph').css(['margin-left', 80]);  
$('.paragraph').css('color', 'red');
```



# 5. Syntax

- By id -> \$('#someid')

```
/** Jquery way to toggle between divs */  
#jqueryToggleDiv() {  
  $('#jquery').on('click', function() {  
    $('#content').toggle();  
    $('#alternative').toggle();  
  });  
}
```



## 6. Good practices

## 6. Good practices

- Naming with dollar sign \$img = \$('#img')

```
let $before = $('#before');  
let $after = $('#after');  
let $img = $('#img');
```



## 6. Good practices

- Avoid conflict using `jQuery.noConflict()`
- You can call `$jq = jQuery.noConflict()`

```
$jq(function () {  
  $jq('a').click(function (event) {  
    event.preventDefault();  
    alert('Hello world!');  
  });  
});
```



## 6. Good practices

- Cache your jQuery object in a variable
- Use chaining to reduce repetition

Don't do this:

```
$('#p').text('jQuery is easy!');  
$('#p').css('margin-left', 50);  
$('#p').css('color', 'orange');
```

Instead, do this:

```
let $paragraph = $('#p');  
$paragraph.text('jQuery is easy!');  
$paragraph.css('margin-left', 50);  
$paragraph.css('color', 'orange');
```

# 7. Examples

# 7. Examples

- Toggle

<http://10.6.130.141:8080/toggle.html>

```
/** Vanilla javascript way to toggle between divs */
#toggleDiv() {
  if(document.getElementById('content').style.display === 'block') {
    document.getElementById('content').style.display = 'none';
    document.getElementById('alternative').style.display = 'block';
  } else {
    document.getElementById('alternative').style.display = 'none';
    document.getElementById('content').style.display = 'block'
  }
}
```

# 7. Examples

- Fade

<http://10.6.130.141:8080/fade.html>

```
/** JQuery way to fade toggle an image and change button text */
#jqueryFade() {
  this.$jquery.on('click', () => {
    $('#alternative').fadeToggle(600);
    if (this.$jquery.text() === 'Jquery fade in') {
      this.$jquery.text('Jquery fade out');
    } else {
      this.$jquery.text('Jquery fade in');
    }
  });
}
```

```
/** Vanilla way the using css fade in an element
 * @param {Object} element
 */
#fadeIn(element){
  element.classList.add('show');
  element.classList.remove('hide');
}

/** Vanilla way that using css fade out an element
 * @param {Object} element
 */
#fadeOut(element){
  element.classList.add('hide');
  element.classList.remove('show');
}

/** Vanilla way to fade toggle an image and change button text */
#vanillaFade() {
  let img = document.getElementById('alternative');
  let btn = document.getElementById('vanilla');
  btn.addEventListener('click', () => {
    if (img.className.indexOf('hide') !== -1) {
      this.#fadeIn(img);
      btn.innerHTML = 'Vanilla fade out';
    } else {
      this.#fadeOut(img);
      btn.innerHTML = 'Vanilla fade in';
    }
  });
}
```



# 7. Examples

- Slide

<http://10.6.130.141:8080/slide.html>

```
/** Vanilla javascript way to slide an element */
#slideDiv() {
  const slided = document.getElementById('alternative').offsetHeight;
  if(slided === 0) {
    document.getElementById('alternative').style.height = '400px';
  } else {
    document.getElementById('alternative').style.height = '0px';
  }
}
```

```
/** Jquery way to slide an element */
#jquerySlideDiv() {
  $('#jquery').on('click', function() {
    $('#content').slideToggle('slow');
  });
}
```

# 7. Examples

- Shake and mouse over image switch

<http://10.6.130.141:8080/index.html>

```
/** On mouseover an image it switch to another one */
#over() {
  let imageNumber = 3;
  this.#img.on('mouseover', () => {
    this.#img.toggle('drop');
    let imageString = '../img/japan' + imageNumber + '.jpg';
    imageNumber = imageNumber < 6 ? imageNumber : 2;
    imageNumber++;
    this.#img.attr('src', imageString).toggle('drop');
  });
}
```

```
/** First shake the image then replace it with the after version
 * If the after is clicked apply a pulsate effect
 */
#shake() {
  this.#before.on('click', () => {
    this.#before.effect('shake', 'left', () => {
      this.#before.hide();
      this.#after.show();
    });
  });
  this.#after.on('click', () => {
    this.#after.effect('pulsate', 4);
  });
  $('#reset').on('mousedown', () => {
    this.#after.hide();
    this.#before.show();
  });
}
```

## 8. AJAX

# 8. AJAX



- Asynchronous JavaScript And XML
- Gmail, Google Maps, Youtube, Facebook tabs...

# 8. AJAX

GET	304	document	Other
GET	200	script	<a href="#">ajax.html</a>
GET	304	stylesheet	<a href="#">ajax.html</a>
GET	304	stylesheet	<a href="#">ajax.html</a>
GET	304	script	<a href="#">ajax.html</a>
GET	304	xhr	<a href="#">jquery.min.js:2</a>
POST	200	xhr	<a href="#">jquery.min.js:2</a>
POST	200	xhr	<a href="#">jquery.min.js:2</a>

XML Http Request (XHR)

# 8. AJAX

\$.ajax()

\$.get()

\$.post()

```
$.ajax({
  type: 'GET', // POST
  url: 'http://10.6.128.208:8080/api/orders.json',
  success: function(data) {
    console.log(data);
    $.each(data, function(i, order) {
      $orders.append('<li>Name: ' + order.name + ', Drink: ' + order.drink + '</li>');
    });
  },
  error: function() {
    alert('Error loading orders!');
  }
});
```

## 8. AJAX

```
1. let response;  
2. $.get( 'foo.txt', function(data) {  
3.     response = data;  
4. }  
5. console.log(response);
```

## 8. AJAX

```
1. $.get( 'foo.php', function(response) {  
2.     console.log(response);  
3. });
```



## 9. Plugins

# 9. Plugins

[plugins.jquery.com](https://plugins.jquery.com)



The screenshot shows the homepage of The jQuery Plugin Registry. At the top, there is a dark blue header with a search bar. Below the header, the main title "The jQuery Plugin Registry" is displayed in large white letters. A large, rounded search bar is centered below the title. A prominent message in the center states: "The jQuery Plugin Registry is in read-only mode. New plugin releases will not be processed." Below this message, a text block explains the recommendation to move to npm, using "jquery-plugin" as a keyword. On the left side, there is a section titled "Popular Tags" with a list of tags and their counts.

Search

## The jQuery Plugin Registry

Search

**The jQuery Plugin Registry is in read-only mode.**  
**New plugin releases will not be processed.**

We recommend moving to npm, using "jquery-plugin" as the keyword in your package.json. The npm blog has [instructions for publishing your plugin to npm](#).

**Popular Tags**

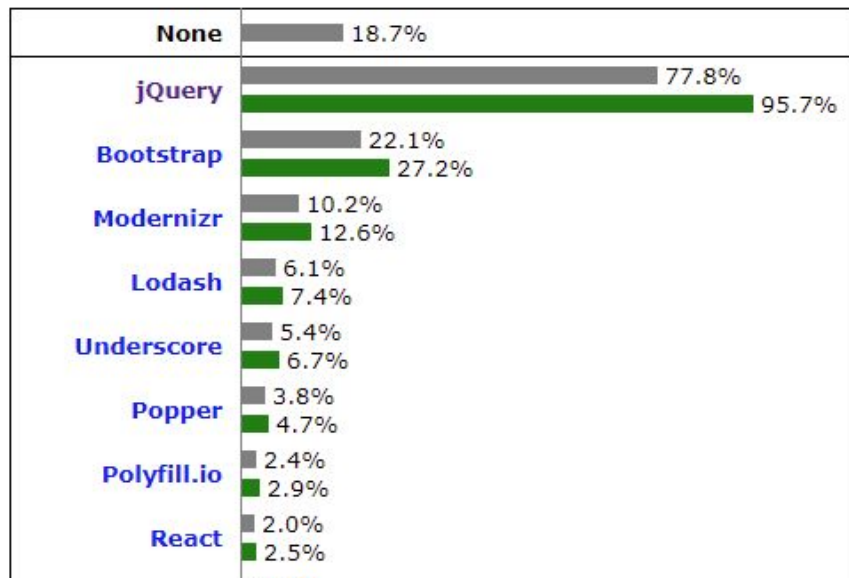
- [ui](#) (542)
- [jquery](#) (482)
- [form](#) (285)
- [animation](#) (273)
- [input](#) (252)
- [image](#) (210)
- [responsive](#) (184)
- [slider](#) (172)
- [ajax](#) (154)
- [scroll](#) (140)

# 10. Conclusions

# 10. Conclusions

## w3techs usage of jQuery

18.7% of the websites use none of the JavaScript libraries that we monitor. jQuery is used by 77.8% of all the websites, that is a JavaScript library market share of 95.7%.



# 10. Conclusions

you might not  
need jquery

## EVENTS

### Delegate

JQUERY

```
$(document).on(eventName, elementSelector, handler);
```

IE8+

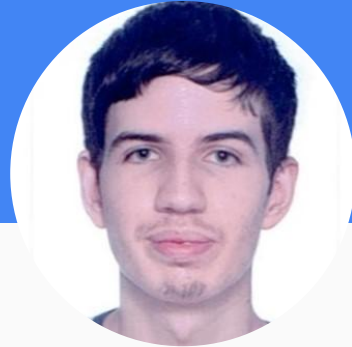
```
document.addEventListener(eventName, function(e) {  
    // loop parent nodes from the target to the delegation  
    for (var target = e.target; target && target !== this; target = target.parentNode) {  
        if (target.matches(elementSelector)) {  
            handler.call(target, e);  
            break;  
        }  
    }  
}, false);
```

# Us



Aitor Hernández Sánchez

alu0100987522@ull.edu.es



Adrián Fleitas de la Rosa

alu0101024363@ull.edu.es

# Bibliography:

<https://stackoverflow.com/tags/jquery/info> SO Post info and best practices jQuery

<https://oscarotero.com/jquery/> Cheat Sheet

<http://youmightnotneedjquery.com/> Vanilla JavaScript vs jquery methods

<https://learn.jquery.com/> Learn jQuery

<https://tekslate.com/11-benefits-jquery-every-web-designers-know> Benefits of jquery

[https://w3techs.com/technologies/overview/javascript\\_library](https://w3techs.com/technologies/overview/javascript_library) Usage of jQuery chart

<https://plugins.jquery.com/> jQuery plugins registry