

JavaScript

M4103C - Programmation Web – client riche

2ème année - S4, cours - 4/4
2018-2019

Marcel.Bosc@iutv.univ-paris13.fr

Département informatique

IUT de Villetaneuse

Université Paris-13

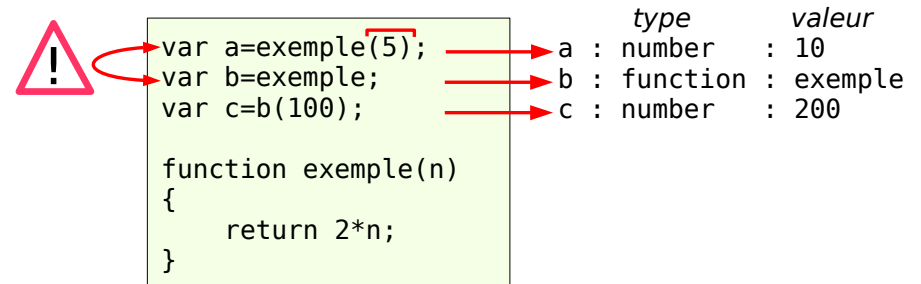
Table des matières

- fonctions JS
- `$(document).ready(...)`
- HTML & texte
- http
- HTML ajouté
- Listes jQuery

1ère partie

Fonction JS

Fonctions : appel / valeur



Fonctions : appel / valeur

1
2
5

```
var x=exemple;  
var r=abcd(x);  
console.log(r);
```

4

```
function exemple(n)  
{  
    return 2*n;  
}
```

3

```
function abcd(f)  
{  
    return f(10);  
}
```

Annotations:

- x : function : exemple
- r : number : 20
- n : number : 10
- f : function : exemple

Gestionnaire d'événements

1
2
3
6
4
5

```
var x=exemple3;  
$('p').click( x );  
console.log("ok");  
function exemple3()  
{  
    console.log("ex3");  
}  
console.log("ok2");
```

Gestionnaire d'événements

click

Fonctions anonymes

```
var b=exemple;  
var c=b(100);  
function exemple(n)  
{  
    return 2*n;  
}
```

1
3
2

```
var b=function(n)  
{  
    return 2*n;  
};  
var c=b(100);
```

Gestionnaire d'événements

```
$('p').click(exemple3);  
function exemple3()  
{  
    console.log("ex3");  
}
```

2

```
$('p').click(function()  
{  
    console.log("ex3");  
});
```

Types en JavaScript

Types « primitifs »

- boolean → true , false
- null
- undefined
- number → 1234 , 3.14159
- string → "bonjour"

Type Object → { nom: "Leïla", score: 123 }
["orange", "banane"]

Array = Objet

typeof

undefined & typeof

```
exemple(10);
exemple();

function exemple(a)
{
  if(typeof a === "undefined")
  {
    a=123;
  }
}
```

Fonctions : paramètres

```
var e=1;
var c="bonjour";
var o={ nom: "Leïla", score: 20 };
exemple(e,c,o);
console.log(e,c,o);

function exemple(x,y,z)
{
  x=100;
  y="au revoir";
  z.nom="Tom";
  z.score=200;
}
```

types « primitifs »

e : 1
c : "bonjour"
o : { nom: "Tom",
score: 200 }

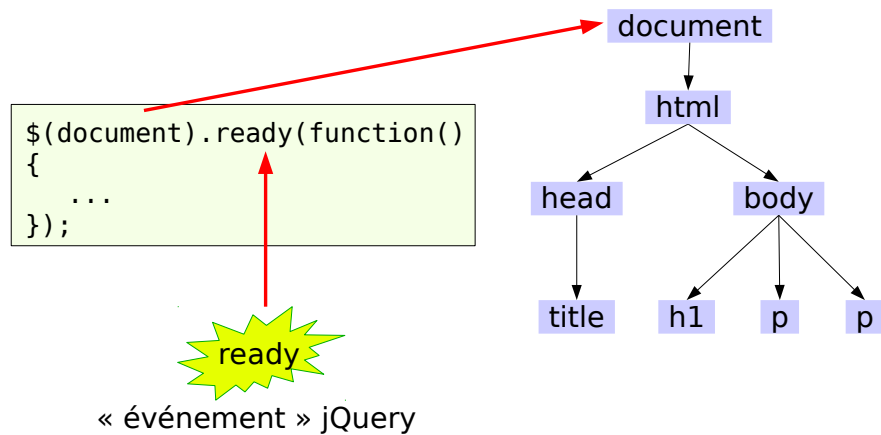
primitifs : par valeur

objets : par référence

2ème partie

\$(document).ready(...)

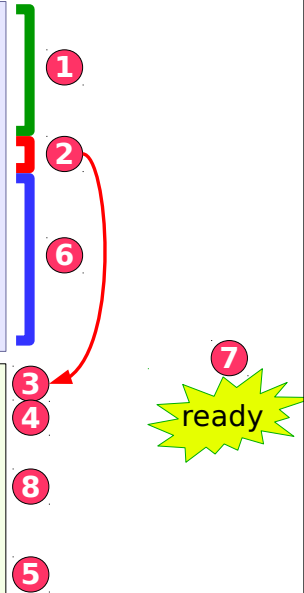
\$(document).ready()



Ordre d'exécution

```
<html>
  <head>
    <title>exemple</title>
    <script src="jquery.js"></script>
    <script src="exemple.js"></script>
  </head>
  <body>
    <h1>Ceci est un titre</h1>
    <p>bla bla bla</p>
  </body>
</html>
```

```
console.log('bonjour');
$(document).ready(function() {
  $('p').css('color', 'yellow');
});
$('p').css('color', 'orange');
vide! ⚠
```



3ème partie

HTML & texte

Entités

>	>	>
<	<	<
"	"	"
'	'	'

HTML:

```
<p> Ceci est un inférieur: &lt; </p>
<p>
  Pour afficher une exemple de html:
  &lt;strong>Un exemple</strong>
</p>
```

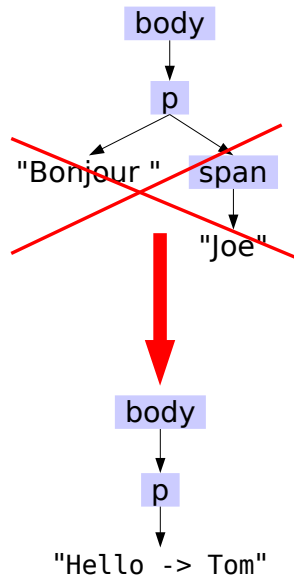
Un exemple

jQuery : .text()

```
<body>
  <p>
    Bonjour <span>Joe</span>
  </p>
</body>
```

```
$('p').text('Hello -> Tom');
```

```
<body>
  <p>
    Hello &gt; Tom
  </p>
</body>
```

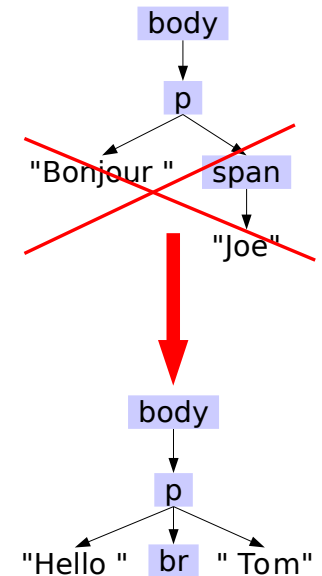


jQuery : .html()

```
<body>
  <p>
    Bonjour <span>Joe</span>
  </p>
</body>
```

```
$('p').html('Hello <br/> Tom');
```

```
<body>
  <p>
    Hello <br/> Tom
  </p>
</body>
```

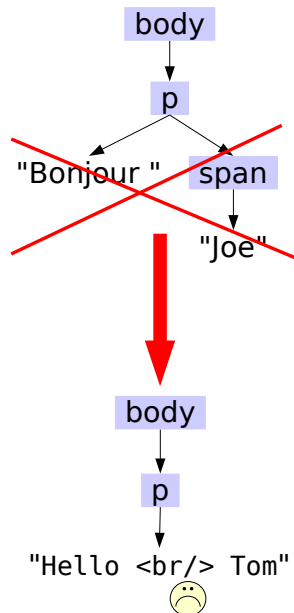


jQuery : .html()

```
<body>
  <p>
    Bonjour <span>Joe</span>
  </p>
</body>
```

```
$('p').text('Hello <br/> Tom');
```

```
<body>
  <p>
    Hello &lt;br/> Tom
  </p>
</body>
```



jQuery : .val()

uniquement formulaire

```
$('#select').val('Chocolat');
```

```
$('#input').val('essai');
```

```
$('#textarea').val('Un exemple...');
```

<select> Chocolat : Chocolat Fraise Vanille	<input type="text"/> essai	<textarea> un exemple de texte
--	-------------------------------	-----------------------------------

Exemple

```
<input id="saisie" type="text">
<input id="ajouter" type="button" value="Ajouter"/>
<p></p>
```



```
$('#ajouter').click(function()
{
  $('p').html($('#saisie').val());
});
```

.text()



DOM XSS

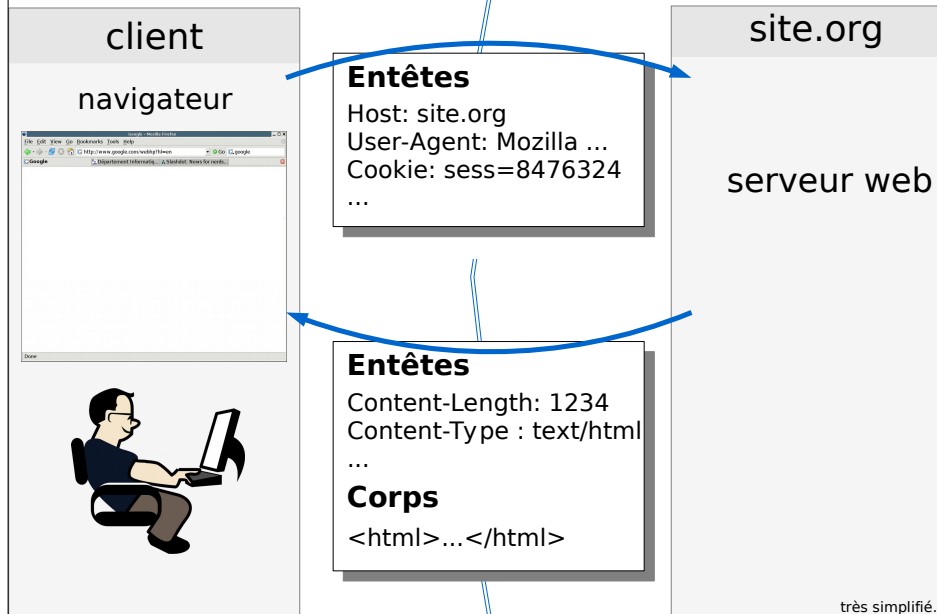


```
<p>a<<joe>></p>
```

4ème partie

http

Requêtes & entêtes http



La requête

Bonjour, je voudrais la page *essai.html* qui se trouve sur le site *musique.org*. Je suis un navigateur *firefox*. J'accepte les pages aux formats

GET,
POST...

- 1: **GET** /essai.html HTTP/1.1
- 2: **Host:** musique.org
- 3: **User-Agent:** Mozilla/5.0 (...) Ubuntu... Firefox/36.0
- 4: **Accept:** text/html,application/xhtml+xml,....
- 5: **Accept-Language:** en-us,en;q=0.5
- 6: **Accept-Encoding:** gzip,deflate
- 7: **Accept-Charset:** UTF-8,*
- 8: **Keep-Alive:** 115
- 9: **Connection:** keep-alive

Console

La réponse

La page demandé a bien été trouvée. Je suis un serveur Apache version... Voici un cookie pour pouvoir vous re-identifier plus tard. Voici la page demandée au format html encodé en UTF-8.

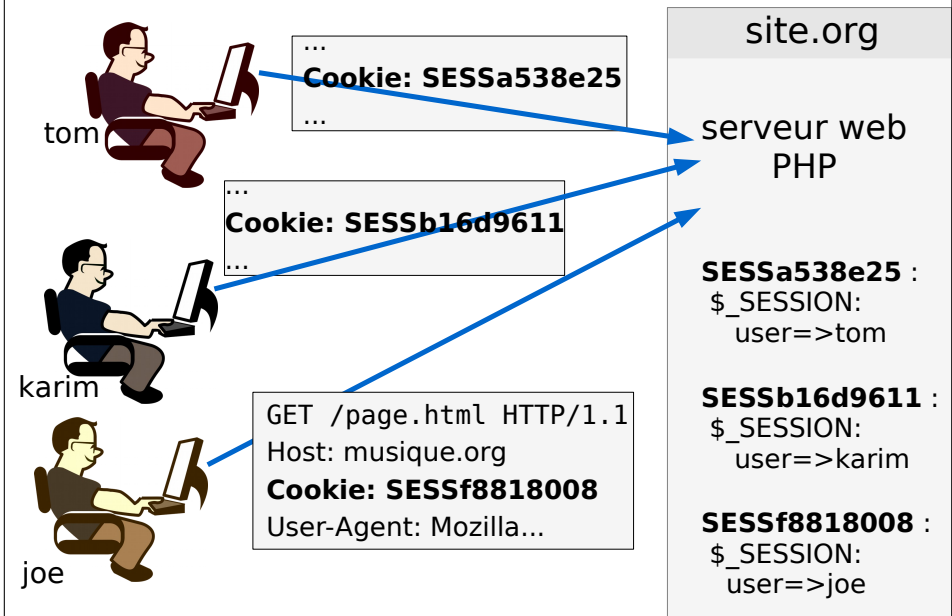
200,404,304, 500...

entêtes

```
1: HTTP/1.1 200 OK
2: Date: Mon, 16 Mar 2015 11:24:21 GMT
3: Server: Apache/2.2.22 (Debian)
4: X-Powered-By: PHP/5.4.36-0+deb7u3
5: Set-Cookie: SESSf8818008...
6: Last-Modified: Mon, 16 Mar 2015 10:41:34 GMT
7: Content-Type: text/html; charset=utf-8
8:
9: 12d79
10: <!DOCTYPE html ...>
11: <html ...>
12: <head>
13: <title>Le site Musique.org</title>
```

corps

Cookies



5ème partie

HTML ajouté

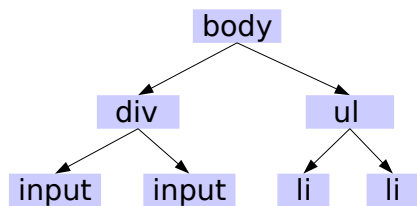
jQuery : .is()

```
$('#liste').is('ul') → true
$('#l2').is('#liste li') → true
$('#l2').is('ul') → false
$('body').mousedown(function()
{
    if($(this).is('#l1')){ ... }
});
```

```
<ul id="liste">
  <li id="l1">Tom</li>
  <li id="l2">Joe</li>
</ul>
```

Ajout DOM : HTML

```
<body>
  <div>
    <input id="saisie" type="text">
    <input id="ajouter" type="button" value="Ajouter"/>
  </div>
  <ul id="liste">
    <li>Tom</li>
    <li>Joe</li>
  </ul>
</body>
```

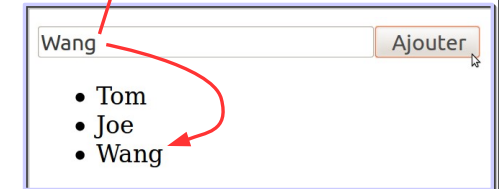


Ajout DOM : JS

```
$('#ajouter').click(function()
{
  var ligne=$('#<li></li>');
  var texte=$('#saisie').val();
  ligne.text(texte);
  $('#liste').append(ligne);
});

$('li').mousedown(function(){
  alert('mousedown!');
});
```

```
<div>
  <input id="saisie"
  <input id="ajouter"
</div>
<ul id="liste">
  <li>Tom</li>
  <li>Joe</li>
  <li>Wang</li>
</ul>
```



Ajout DOM : JS

```
1 $('#ajouter').click(function()
{
  var ligne=$('#<li></li>');
  var texte=$('#saisie').val();
  ligne.text(texte);
  4 $('#liste').append(ligne);
});

2 $('li').mousedown(function(){
  alert('mousedown!');
});
```

```
<div>
  <input id="saisie"
  <input id="ajouter"
</div>
<ul id="liste">
  <li>Tom</li>
  <li>Joe</li>
  4 <li>Wang</li>
</ul>
```

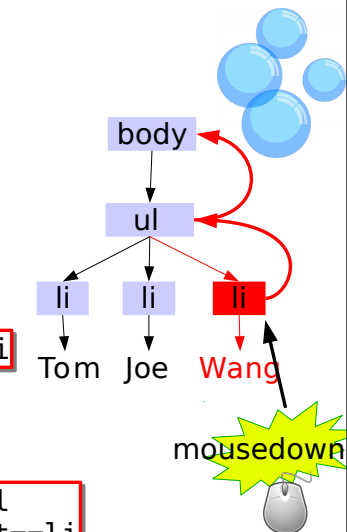


Bubbling

```
$('#ajouter').click(function()
{
  var ligne=$('#<li></li>');
  var texte=$('#saisie').val();
  ligne.text(texte);
  $('#liste').append(ligne);
});

$('li').mousedown(function(){
  alert('mousedown!');
}); this==li

$('ul').mousedown(function(e){
  if($(e.target).is('li'))
  {
    alert('mousedown!');
  }
  this==ul
  e.target==li
});
```

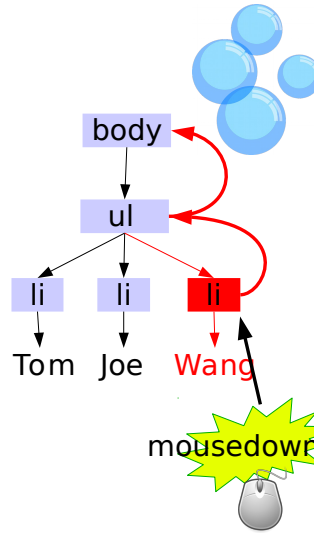


.on()

```
$('#ajouter').click(function()
{
  var ligne=$('#<li></li>');
  var texte=$('#saisie').val();
  ligne.text(texte);
  $('#liste').append(ligne);
});

$('#li').mousedown(function(){
  alert('mousedown'); this==li
});

$('#ul').on('mousedown','li',
  function(){
    alert('mousedown!'); this==li
  });
```



6ème partie

Listes jQuery

Listes jQuery

à partir d'un sélecteur

affectée à variable

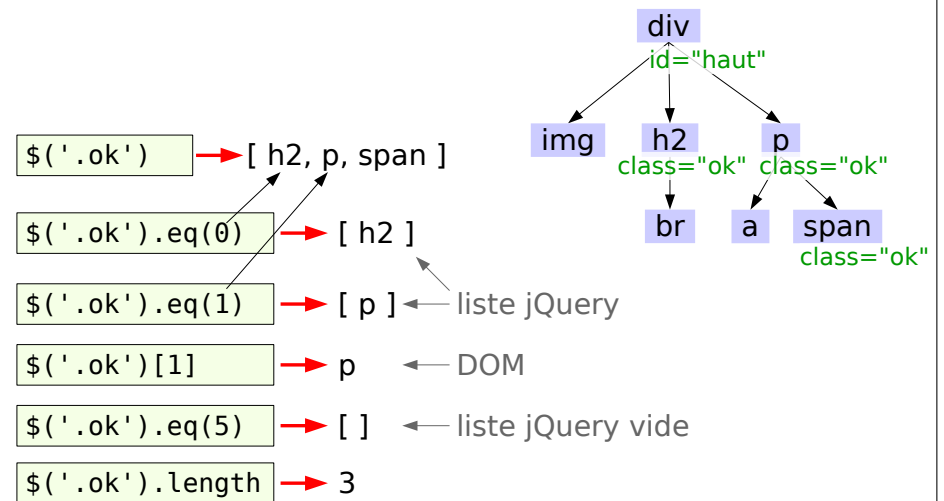
```
$('#ajouter').click(function()
{
  var ligne=$('#<li></li>');
  var texte=$('#saisie').val();
  ligne.text(texte);
  $('#liste').append(ligne);
});

$('li').mousedown(function(){
  $(this).addClass('ok');
});
```

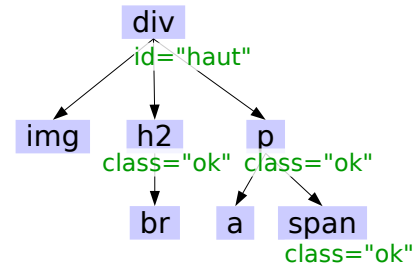
à partir él. DOM

à partir d'HTML

Listes jQuery



Parcours d'arbre



`$('p')` → [p]

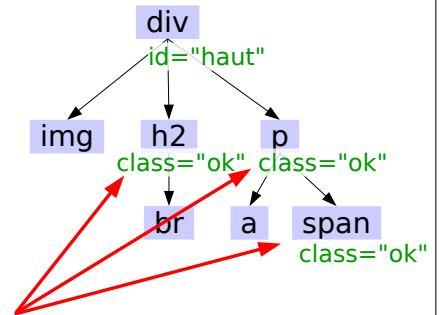
`$('p').parent()` → [div]

`$('p').parent().children()` → [img, h2, p]

`$('p').parent().children().eq(1)` → [h2]

opération sur liste => nouvelle liste

.click()



`$('.ok')` → [h2, p, span]

`$('.ok').click(function(){ ... });`

Ce document est distribué librement.

Sous licence GNU FDL :

<http://www.gnu.org/copyleft/fdl.html>

Les originaux sont disponibles au format LibreOffice

<http://www-info.iutv.univ-paris13.fr/~bosc>