

## TP 1

### Exercices TP-1

En utilisant l'inspecteur, quel est le nom de la balise HTML qui contient le texte "XYZ" ci-dessous:

XYZ

Votre réponse

Envoyer

Vous avez 4 essais.

```
... tml#yui_3_17_2_1_1643374778266_53.yui3-js-enabled body#page-mod-lesson-view.format-topics.path-mor ...
::after
Styles Computed Layout Event Listeners DOM Breakpoints Properties Accessibility
Filter :hov .cls + [X]
element.style {
}
h4, .h4 { all:1
font-size: 1.40625rem;
}
h1, h2, h3, h4, h5, h6, .h1, .h2, .path-calendar .maincalendar .calendar-controls all:1
.current, .h3, .h4, .h5, .h6 {
margin-bottom: 0.5rem;
font-weight: 300;
line-height: 1.2;
}
h1, h2, h3, h4, h5, h6 { all:1
margin-top: 0;
margin-bottom: 0.5rem;
}
*, ::before, ::after { all:1
-webkit-box-sizing: border-box;
```

En utilisant l'inspecteur, quel est la taille de la police de caractères (font-size) du texte "XYZ" ci-dessous:

XYZ

✓ Votre réponse :

1.40625rem

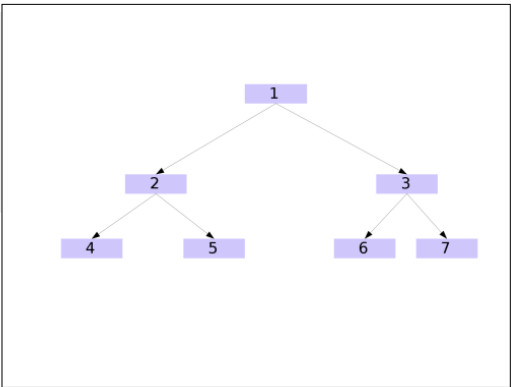
Feedback:

C'est une réponse correcte

# Révision: arbre simple

```
<body>
  <p>
    Un exemple de page en html.<br/>
    
  </p>
  <ul>
    <li>pomme</li>
    <li>fraise</li>
  </ul>
</body>
```

Remarque: dans ces schémas, on ne tient pas compte du texte.

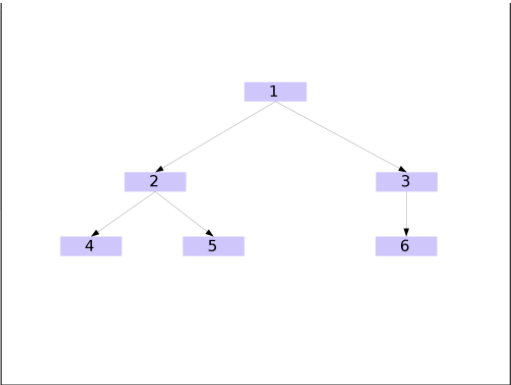


✓ *Votre réponse :*

```
1 = body
2 = p
3 = ul
4 = br
5 = img
6 = li
7 = li
```

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8"/>
    <title>Exemple</title>
  </head>
  <body>
    <h1>Bonjour!</h1>
  </body>
</html>
```

(<!DOCTYPE> n'est pas dans l'arbre)



✓ *Votre réponse :*

```
1 = html
2 = head
3 = body
4 = meta
5 = title
6 = h1
```

*Feedback:*  
C'est une réponse correcte

✓ *Votre réponse :*

XX1 = h1

XX2 = p

XX3 = a

XX4 = h2

XX5 = ul

XX6 = li

XX7 = span

XX8 = div

XX9 = img

*Feedback:*

C'est une réponse correcte

Certains éléments HTML prennent au moins une ligne entière. On les appelle des éléments "block". Si vous mettez plusieurs éléments block d'affilé, ils seront affichés sur des lignes différentes.

D'autres éléments HTML s'affichent à l'intérieur d'une ligne, de gauche à droite (inline), comme du texte.

Parmi les éléments suivants, lesquels prennent au moins toute une ligne (block) ?

**BLOCK:**



**INLINE:**



✓ *Votre réponse :*

div

p

h1

HTML:

```
<p>
  La souris se trouve à la position
  x=<AAA class="coord">123</AAA>,
  y=<AAA class="coord">432</AAA>
</p>
```

Dans cet exemple, quelle balise conviendrait le mieux, à la place de "AAA" ?

✓ *Votre réponse :*

span

*Feedback:*

Oui, la balise span sert à regrouper du texte, ce qui permet de le designer en CSS ou en JS.

```
<p>
  <span id="velo">Vélo</span>,
  <span id="voiture">Voiture</span>,
  <span id="avion">Avion</span>
</p>
```

Que faut-il écrire dans le fichier CSS pour afficher "Voiture" en rouge ?  
(réponse en une seule ligne)

✓ *Votre réponse :*

```
#voiture{color: red;}
```

*Feedback:*

C'est une réponse correcte

## Révision CSS

HTML:

```
<p>
  <span AAA="aliment">Chocolat</span>,
  <span AAA="vehicule">Vélo</span>,
  <span AAA="aliment">Fraise</span>,
  <span AAA="vehicule">Voiture</span>,
  <span AAA="vehicule">Avion</span>
  <span AAA="aliment">Poire</span>
</p>
```

CSS:

```
.aliment
{
  color: red;
}
```

Que faut-il écrire à la place de "AAA" pour afficher "Chocolat" "Fraise" et "Poire" en rouge ?

✓ *Votre réponse :*

class

*Feedback:*

C'est une réponse correcte

## Révision CSS

```
<h2>Ingrédients</h2>
<p>
  <span>Chocolat</span>, <span>Fraise</span>
</p>
```

Que faut-il écrire dans le fichier CSS pour afficher "Ingrédients" en rouge ?  
(réponse en une seule ligne)

✓ *Votre réponse :*

```
h2{color: red;}
```

*Feedback:*

C'est une réponse correcte

## Révision CSS

```
<h2>Ingrédients</h2>
<p>
  <span>Chocolat</span>, <span AAA>Fraise</span>
</p>
```

Que faut-il écrire à la place de "AAA" pour afficher "Fraise" en rouge ?  
(réponse en une seule ligne)

✓ *Votre réponse :*

```
style='color:red'
```

*Feedback:*

C'est une réponse correcte

Ouvrez la console des outils de dév.

Rappel : `$('.xyz')` est une liste jQuery contenant tous les éléments correspondant au sélecteur xyz

Dans la console tapez une liste jQuery contenant tous les paragraphes et appuyez sur entrée:.

La console affiche les deux paragraphes comme ceci:

```
>> $('p')
← ▶ Object { 0: p, 1: p, length: 2, prevObject: {...}, context: HTMLDocument http://localhost
/test/js/tp-1/exo-1.html, selector: "p" }
```

Une liste jQuery est comme un tableau. Celle-ci contient deux éléments: 0: p et 1: p.

Dans la console, passez la souris (sans cliquer) sur les petits carrés désignés par les flèches rouges.

Vous devrez voir les paragraphes surlignés dans la fenêtre principale.

Cliquez sur un des petits carrés dans la console.

Vous arrivez sur le paragraphe dans l'onglet HTML des outils de dév.

Revenez sur l'onglet console.

Essayez d'autres listes jQuery, avec des sélecteurs différents.

Que faut-il taper pour obtenir une liste jQuery contenant juste le bouton ?

(réponse la plus simple)

✓ *Votre réponse :*

`$('.input')`

*Feedback:*

C'est une réponse correcte

## Exercices TP-1

### ▼ Réagir bouton

En vous inspirant des exemples vus précédemment, écrivez le code nécessaire pour afficher dans la console "Le bouton a été cliqué" quand l'utilisateur clique sur le bouton.

Continuer

```
exo-1.js > ready() callback
$(document).ready(function () {
  console.log("Le document est prêt");

  $("p").click(function () {
    console.log('Le bouton a été cliqué');
  });
});
```

On veut cacher l'image quand l'utilisateur clique sur le bouton.

Il faut donc :

- 1) une liste jQuery avec l'image
- 2) Une fonction jQuery permettant de cacher un élément

Dans la documentation de jQuery

<http://api.jquery.com/category/effects/>

cherchez le nom de la fonction permettant de cacher un élément.

Attention, il y en a plusieurs, on veut la plus simple (sans effets spéciaux).

(on demande juste le nom de la fonction)

✓ *Votre réponse :*

```
.hide()
```

*Feedback:*

C'est une réponse correcte



## ▼ Cacher l'image

Ajoutez la ligne permettant de cacher l'image lorsqu'on clique sur le bouton.

Vérifiez que ça marche.

**Indication / rappel** : la syntaxe est : `listejQuery.fonctionjQuery()`;

La `listejQuery` contient l'image

La `fonctionjQuery` cache l'image

Effets plus sophistiqués :

Dans [la documentation](#), vous avez vu d'autres fonctions (`.fadeOut()` , `.slideUp()` ) permettant de cacher un élément.

Essayez ces fonctions à la place de `.hide()`.

Explications:

`.hide()` ajoute simplement `style="display:none"` à l'élément `<input>`

Par contre `.fadeOut()` et `.slideUp()` font des choses beaucoup plus compliquées (animations).

```
$("#p").click(function () {  
    console.log("Le bouton a été cliqué");  
    $("#img").hide();  
});  
console.log("La mise en place est finie. En attente d'événements...");
```

Le bouton affiche "cacher"

Une fois l'image cachée on voudrait qu'il affiche "montrer".

On va utiliser la fonction jQuery `.val()`

Documentation: <http://api.jquery.com/val/>

Comme d'autres fonction jQuery ( `.html()`, `.text()` ...), elle a deux variantes:

`.val()` : permet de lire la valeur (pas d'argument)

`.val('exemple')` : permet de modifier la valeur (argument)

Vérifiez que vous arrivez à afficher "montrer" dans le bouton.

```

$("p").click(function () {
    console.log("Le bouton a été cliqué");
    $("img").hide();
    $('input').val('montrer');
});

```

## ▼ Cacher / montrer

Lorsque l'utilisateur clique sur le bouton, on voudrait montrer ou cacher l'image, en fonction de ce qui est affiché dans le bouton.

Quand l'utilisateur clique, on doit donc distinguer deux cas:

- le bouton affiche actuellement "montrer" : on doit montrer l'image et changer l'affichage du bouton à "cacher"
- le bouton affiche actuellement "cacher" : on doit cacher l'image et changer l'affichage du bouton à "montrer"

Précisions:

`if` et `else` s'utilisent comme en C/Java/PHP.

pour comparer deux chaînes de caractères, utilisez `===`

Trouvez la fonction pour montrer l'image ici : <http://api.jquery.com/category/effects/>

```

$(document).ready(function () {
    console.log("Le document est prêt");

    $("p").click(function () {
        console.log("Le bouton a été cliqué");
        if ($("input").val() === "cacher") {
            $("img").hide();
            $("input").val("montrer");
        } else {
            $("img").show();
            $("input").val("cacher");
        }
    });
    console.log("La mise en place est finie. En attente d'événements...");
});

```

Ensuite créez un fichier `exo-2.js` :

```
$(document).ready(function()
{
    console.log("Le document est prêt");
    $('AAA').BBB(function(event)
    {
        $('CCC').DDD(EEE);
        $('FFF').DDD(GGG);
    });
    console.log("La mise en place est finie. En attente d'événements...");
});
```

Dans les pages suivantes, on va compléter, **ensemble**, les informations manquantes (AAA, BBB, CCC...).

*Comme souvent en JS, on veut réagir à un événement qui survient sur un élément.*

Ici l'événement qui nous intéresse est le mouvement de la souris. L'élément sur lequel il survient c'est l'image.

Notre réaction c'est ce qui se passe dans la fonction: `function(event){...}`

Que faut-il écrire à la place de AAA ?

✓ Votre réponse :

img

*Feedback:*

C'est une réponse correcte

```
$(document).ready(function()  
{  
  console.log("Le document est prêt");  
  $('AAA').BBB(function(event)  
  {  
    $('CCC').DDD(EEE);  
    $('FFF').DDD(GGG);  
  });  
  console.log("La mise en place est finie. En attente d'événements...");  
});
```

Maintenant il faut trouver la fonction jQuery qui permet de réagir au mouvement de la souris.

Cherchez dans la documentation, ici:

<http://api.jquery.com/category/events/mouse-events/>

Que faut-il écrire à la place de BBB:

✓ *Votre réponse :*

mousemove

*Feedback:*

C'est une réponse correcte

Continuer

```
$(document).ready(function()  
{  
  console.log("Le document est prêt");  
  $('AAA').BBB(function(event)  
  {  
    $('CCC').DDD(EEE);  
    $('FFF').DDD(GGG);  
  });  
  console.log("La mise en place est finie. En attente d'événements...");  
});
```

Maintenant il faut trouver les informations sur la position de la souris.

Vous avez remarqué que dans cet exemple on utilisait `function(event){...}` au lieu de `function(){...}`

Notre fonction est appelée avec le paramètre "event" qui décrit l'événement "mouvement de souris" qui a provoqué cet appel.

Cet objet "event" contient plusieurs propriétés décrivant l'événement.

Elles sont décrites ici :

<http://api.jquery.com/category/events/event-object/>

Cherchez dans cette documentation quelle propriété donne la position de la souris.

Que faut-il écrire à la place de EEE:

✓ *Votre réponse :*

event.pageX

*Feedback:*

C'est une réponse correcte

```
$(document).ready(function()
{
    console.log("Le document est prêt");
    $('AAA').BBB(function(event)
    {
        $('CCC').DDD(EEE);
        $('FFF').DDD(GGG);
    });
    console.log("La mise en place est finie. En attente d'événements...");
});
```

Maintenant il faut afficher ces positions sur la page.

Vous avez remarqué que dans le premier paragraphe deux éléments sont prévus à cet effet.

Que faut-il écrire à la place de CCC:

✓ *Votre réponse :*

#affiche-x

*Feedback:*

C'est une réponse correcte

Continuer

```
$(document).ready(function()
{
    console.log("Le document est prêt");
    $('AAA').BBB(function(event)
    {
        $('CCC').DDD(EEE);
        $('FFF').DDD(GGG);
    });
    console.log("La mise en place est finie. En attente d'événements...");
});
```

Maintenant il nous manque juste le nom de la fonction jQuery qui permet de changer le contenu de ces spans.

Que faut-il écrire à la place de DDD:

✓ *Votre réponse :*

text

*Feedback:*

C'est une réponse correcte

Continuer

```
$(document).ready(function () {
    console.log("Le document est prêt");
    $("img").mousemove(function (event) {
        $("#affiche-x").text(event.pageX);
        $("#affiche-y").text(event.pageY);
    });
    console.log("La mise en place est finie. En at
});
```

```
console.log("Ce programme JS vient d'être chargé");
$(document).ready(function()
{
    console.log("Le document est prêt");
    $('img').mousemove(function(event)
    {
        $('#affiche-x').text(event.pageX);
        $('#affiche-y').text(event.pageY);
    });
    console.log("La mise en place est finie. En attente d'événements...");
});
```

Décortiquons ce qui vient d'être écrit :

✓ *Votre réponse :*

**\$('#affiche-x')** = Liste jQuery

**#affiche-x** = Sélecteur CSS

**'#affiche-x'** = Chaîne de caractères

*Feedback:*

C'est une réponse correcte

```

console.log("Ce programme JS vient d'être chargé");
$(document).ready(function()
{
    console.log("Le document est prêt");
    $('img').mousemove(function(event)
    {
        $('#affiche-x').text(event.pageX);
        $('#affiche-y').text(event.pageY);
    });
    console.log("La mise en place est finie. En attente d'événements...");
});

```

Lors de l'utilisation normale de ce programme, quelle(s) portions de code sont appelées plusieurs fois ?

✓ *Votre réponse :*

`$('#affiche-x').text(event.pageX);`

*Feedback:*

C'est une réponse correcte

```

[A] console.log("Le document est prêt");
    $('img').mousemove(function(event)
    {
[B]     $('#affiche-x').text(event.pageX);
        $('#affiche-y').text(event.pageY);
    });
[C] console.log("La mise en place est finie. En attente d'événements...");
});

```

Quel est l'ordre d'exécution ?

Attention, réfléchissez bien. C'est une question délicate et importante. On vous conseille d'avoir les pages de cours sur la programmation événementielle sous la main ([page1](#), [page2](#)).

Ce problème est important. C'est probablement ce qui pose le plus de difficultés quand on apprend le JavaScript.

✓ *Votre réponse :*

1 = A

2 = C

3 = B

*Feedback:*

Oui! La fonction [B] est un "gestionnaire d'événements".

La ligne "`$('img').mousemove(function(){...});`" veut dire:

"appeler fonction(){...} **plus tard**, quand l'événement `mousemove` surviendra."



## ▼ exo-3 : position absoute

En modifiant uniquement le fichier exo-3.css, expérimentez les propriétés position: absolute, left, et top .

Si vous arrivez à positionner l'image verte entre les yeux du pingouin, vous pouvez passer à la question suivante !

En cas de difficultés:

<https://developer.mozilla.org/fr/docs/Web/CSS/position>

<https://developer.mozilla.org/fr/docs/Web/CSS/left>

```
exo-3.css > .splat
body {
  background-color: red;
}

.splat {
  border: 2px solid black;
  position: absolute;
  right: 42.45em;
  top: 80px;
}

.tux{
  position: absolute;
}
```

```
$( 'img' ).click(function(event)
{
    console.log("Click sur l'image");
    var splat=$( '' );
    splat.BBB(CCC);
    splat.BBB(DDD);
    $( 'EEE' ).FFF(splat);
});
```

On veut positionner l'image à l'endroit où l'utilisateur a cliqué.

Tout à l'heure, on l'avait fait à la main, dans le fichier CSS, en utilisant "left: 123px; top: 123px".

En JavaScript on ne modifie pas le fichier CSS.

On va donc modifier la propriété style de l'élément img (splat).

On a vu en cours qu'il existait une fonction jQuery permettant de modifier cette propriété "style".

Comment s'appelle BBB ?

(dans le doute, regardez votre cours ou cherchez ici: <http://api.jquery.com/category/manipulation/style-properties/> )

✓ *Votre réponse :*

CSS

La fonction jQuery .css() s'utilise comme ceci:

```
splat.css(nom, valeur);
```

Rappel : dans un fichier CSS on écrirait: "left: 123px;".

Ici nom serait "left" et la valeur "123px"

Pour la valeur associée à "left" on veut utiliser la position X de la souris.

N'oubliez pas non plus le "px".

En JavaScript on peut coller ensemble deux chaînes de caractères en utilisant le "+" (comme en Java).  
(concaténation)

"abc"+"def" => "abcdef"

Ça marche aussi pour coller ensemble un nombre et une chaîne.

Que faut-il écrire à la place de "valeur" ?

✓ *Votre réponse :*

```
event.pageX + "px"
```

*Feedback:*

C'est une réponse correcte