

## TP 2 : LES ATTRIBUTS ET LES FEUILLES DE STYLE CSS

---

### 1. MANIPULER LES ATTRIBUTS

Les nombreux sélecteurs de jQuery mis en place, nous entamons dans ce TP l'aspect dynamique de jQuery qui est de modifier les éléments.

#### 1.1. AJOUTER OU SUPPRIMER UNE CLASSE

##### **addClass(classe)**

Ajoute la classe spécifiée à tous les éléments sélectionnés.

`$("#p:last").addClass("selected")` : ajoute la classe `selected` au dernier paragraphe.

Cette méthode retourne un objet jQuery.

Commentaires

Il faut noter que cette méthode ne remplace pas une classe. Elle ajoute simplement une classe.

Il est possible d'ajouter plus d'une classe à la fois. Elles sont notées les unes à la suite des autres, séparées par un espace, soit `addClass(classe1 classe2 classe3)`.

Cette méthode `addClass()` est souvent associée à la méthode `removeClass()` pour créer un effet de commutation.

##### **removeClass(classe)**

Supprime la classe spécifiée à tous les éléments sélectionnés.

`$("#p:last").removeClass("selected")` : supprime la classe `selected` au dernier paragraphe.

Cette méthode retourne un objet jQuery.

*Exercice : Au survol d'une division par le curseur, mettre celui-ci en évidence en le dotant d'un arrièreplan de couleur et d'une bordure. Cet effet se réalise en ajoutant une classe au passage de la souris.*

#### 1.2. VÉRIFIER LA PRÉSENCE D'UNE CLASSE

##### **hasClass(classe)**

Vérifie si la classe spécifiée en argument est présente pour les éléments ciblés. Retourne *true* si la classe spécifiée est présente pour au moins un des éléments

## MMI 1A

---

ciblés, *false* dans le cas contraire.

`$("#p1").hasClass("box")` : vérifie si l'élément identifié par `p1` possède la classe `box`. La méthode renvoie un booléen (*true* ou *false*).

*Exercice* : Soit une série de paragraphes. Au survol de la souris, doter d'un arrière-plan de couleur et d'une bordure uniquement les paragraphes avec la classe `new`.

### 1.3. BASCULER ENTRE DEUX CLASSES

jQuery propose plusieurs méthodes qui permettent de déclencher tantôt une action, tantôt une autre. Cet effet de permutation est repris sous le terme de *toggle*.

Outre des effets spectaculaires, ces méthodes entraînent une économie appréciable de lignes de code.

#### **toggleClass(classe)**

Ajoute la classe spécifiée si elle n'est pas présente, retire la classe spécifiée si elle est présente.

`$(p).toggleClass("classe1")` : applique la classe `classe1` aux paragraphes si elle n'est pas présente. Si elle l'est, enlève la classe `classe1`. Cette méthode retourne un objet jQuery.

*Exercice* : Au clic du lien, faire apparaître ou disparaître une division de la page.

### 1.4. CONNAÎTRE LA VALEUR D'UN ATTRIBUT

#### **attr(nom de l'attribut)**

Accède à la valeur de l'attribut mentionné.

Cette méthode est assez utile pour retrouver la valeur d'un attribut de l'élément sélectionné ou du premier élément sélectionné s'il y en a plusieurs. Si l'élément n'a pas d'attribut répondant à ce nom, la valeur `undefined` est retournée.

`$("#a").attr("title")` : récupère la valeur de l'attribut `title` du premier lien rencontré. Cette méthode retourne un objet jQuery.

*Exercice* : au clic sur le bouton, rechercher le style attaché à la balise `<span>JavaScript</span>`. Le résultat sera affiché dans une division prévue à cet effet.

### 1.5. AJOUTER UN ATTRIBUT ET SA VALEUR

#### **attr(attribut, valeur)**

Assigne une paire attribut/valeur à tous les éléments concernés.

`$("#photo").attr("alt", "Parc éoliennes")` : assigne à l'élément identifié

## MMI 1A

---

par `#photo`, l'attribut `alt="Parc éoliennes"`. Cette méthode retourne un objet jQuery.

*Exercice* : Au clic sur le bouton, un tableau de données sera affiché avec une largeur plus grande pour le rendre ainsi plus lisible.

### 1.6. AJOUTER PLUSIEURS ATTRIBUTS ET LEURS VALEURS

#### **attr({propriétés})**

Permet d'assigner un ensemble de paires attribut/valeur aux éléments sélectionnés. Les différentes propriétés sont séparées par une virgule.

`$("img").attr({ src: "hat.gif", alt: "Logo jQuery!" })` : assigne les attributs `src` et `alt` aux images. Cette méthode retourne un objet jQuery.

*Exercice* : Passer d'une image à une autre par un simple clic sur un lien.

### 1.7. SUPPRIMER UN ATTRIBUT

#### **removeAttr(nom de l'attribut)**

Supprime un attribut des éléments concernés.

`$("#div1").removeAttr("disabled")` : supprime l'attribut `disabled` de l'élément identifié par `div1`. Cette méthode retourne un objet jQuery.

*Exercice* : Supprimer l'attribut de style de la division afin de rendre la lecture plus aisée.

### 1.8. CONNAÎTRE L'ATTRIBUT VALUE

Ce point a trait aux formulaires et leur attribut `value`.

**val()** Récupère sous forme de chaîne de caractères, le contenu de l'attribut `value` du premier élément de la sélection.

`$("input").val()` : récupère le contenu de l'attribut `value` du premier champ de formulaire de type `<input>`. Cette méthode retourne une chaîne de caractères.

*Exercice* : Récupérer la valeur du bouton radio coché.

### 1.9. MODIFIER L'ATTRIBUT VALUE

Variante de `val()` étudié au point précédent. Ici, jQuery permet de modifier l'attribut `value`.

**val(valeur)** Assigne une nouvelle valeur à l'attribut `value` de l'élément sélectionné.

`$("#input").val("Test")` : ajoute la valeur `Test` à une ligne de texte.

---

*MMI 1A*

---

Cette méthode retourne un objet jQuery.

*Exercice* : Après une première soumission du formulaire, le texte du bouton d'envoi est modifié.

## 2. LA MANIPULATION DES FEUILLES DE STYLE CSS

Avec jQuery, la modification dynamique des propriétés de style CSS se révèle facile à implémenter.

### 2.1. ACCÉDER À UNE PROPRIÉTÉ DE STYLE

La méthode `css()` se décline de trois façons. La première permet uniquement d'accéder à la propriété de style CSS d'un élément donné.

#### **css(nom)**

Permet d'accéder à une propriété de style du premier élément trouvé. Le nom est une chaîne de caractères qui reprend la propriété de style à accéder.

```
$( "p" ).css( "color" );
```

Cette méthode renvoie une chaîne de caractères (String).

*Exercice* : Soit une page avec 3 éléments de type bloc. Au clic sur un élément de type bloc, afficher respectivement, de gauche à droite, la couleur d'arrièreplan, la couleur de la bordure et la visibilité.

### 2.2. MODIFIER LES PROPRIÉTÉS DE STYLE

La fonction `css()`, dotée de certains paramètres, permet également de modifier des propriétés de style d'éléments de la page.

#### **css({propriété de style})**

Modifie les propriétés de style d'un élément donné en utilisant la notation CSS clé/valeur pour les propriétés de style à transformer.

```
$( "p" ).css( { color: "red", background: "blue" } );
```

Il faut remarquer la présence des accolades habituelles pour les déclarations de style. Si la clé contient un trait d'union comme par exemple `background-color`, celle-ci doit être placée entre des guillemets (soit `"background-color"`). La notation JavaScript (CamelCase) peut également être adoptée soit `backgroundColor` au lieu de `background-color`. Cette méthode renvoie un objet jQuery.

*Exercice* : Au survol d'un paragraphe, celui-ci est doté d'un arrièreplan et affiche la police de caractères en italique.

## MMI 1A

---

**2.3. ATTRIBUER UNE PROPRIÉTÉ DE STYLE**

La fonction `css()` de jQuery propose une dernière façon de noter les transformations des propriétés de style.

**css(clé,valeur)**

Modifie les propriétés de style d'un élément donné en utilisant la notation clé/valeur pour les propriétés de style à transformer.

- clé (chaîne de caractère) correspond au nom de la propriété de style à modifier.
- valeur (chaîne de caractère ou nombre) est la nouvelle valeur de la propriété. Si un nombre est spécifié, jQuery le converti automatiquement en pixel.

`$("#p").css("color","red");` Cette méthode renvoie un objet jQuery.

*Exercice : Prendre deux divisions superposées, modifier l'ordre de celles-ci au survol de la souris.*

**2.4. LE DIMENSIONNEMENT**

jQuery propose une série de méthodes relatives à la dimension des éléments.

**height()**

Renvoie la hauteur, exprimée en pixels, d'un élément.

`$("#p").height();`

Cette méthode renvoie une chaîne de caractères (String).

**height(valeur)**

Assigne une hauteur aux éléments spécifiés. Si aucune unité n'est spécifiée (comme `em` ou `%`), l'unité `px` sera choisie par défaut.

- valeur (chaîne de caractères ou entier) : valeur de la hauteur assignée.

`$("#p").height(120);`

Cette méthode renvoie un objet jQuery. Après la hauteur, la largeur... **width()**  
Renvoie la largeur, exprimée en pixels, d'un élément

`$("#p").width();`

Cette méthode renvoie une chaîne de caractères (String).

**width(valeur)**

Assigne une largeur aux éléments spécifiés. Si aucune unité n'est spécifiée (comme `em` ou `%`), l'unité `px` sera choisie par défaut.

- valeur (chaîne de caractères ou entier) : valeur de la largeur assignée.

## MMI 1A

---

```
$("#p").width(120);
```

Cette méthode renvoie un objet jQuery. Citons encore les méthodes :

← **innerHeight()** : retourne la hauteur intérieure (la bordure exclue mais le padding inclus) du premier élément trouvé dans la sélection.

← **innerWidth()** : récupère la largeur intérieure (la bordure exclue mais le padding inclus) du premier élément trouvé répondant à la sélection.

**outerHeight(options)** : retourne la hauteur extérieure (la bordure incluse et le padding par défaut) pour le premier élément trouvé répondant à la sélection.

**outerWidth(options)** : retourne la largeur extérieure (inclut la bordure et le padding par défaut) pour le premier élément trouvé répondant à la sélection. Ces méthodes fonctionnent pour les éléments visibles et non visibles.

*Exercice :* Au clic sur un élément boîte, détecter la largeur et la hauteur. Ces dimensions sont alors doublées.

## 2.5. LE POSITIONNEMENT

Parallèlement aux méthodes jQuery sur la dimension des éléments, jQuery propose des méthodes pour déterminer la position de ceux-ci.

### position()

Renvoie la valeur `top` et `left` de la position d'un élément relative à son élément parent. `$("#p:first").position();`

Cette méthode renvoie un objet de type `object{top,left}`.

### offset()

Renvoie la valeur `top` et `left` de la position d'un élément relative au document.

`$("#p:first").offset();` Cette méthode renvoie un objet de type `object{top,left}`.

Plus particulières sont les méthodes `scrollTop(valeur)` et `scrollLeft(valeur)` qui permettent de modifier pour un élément le décalage par rapport au bord supérieur ou au bord gauche. D'une certaine façon, ces méthodes permettent de contrôler l'amplitude de la barre de défilement verticale et horizontale.

### scrollTop(valeur)

Modifie le décalage (en pixels) entre le bord supérieur du document (`top`) et l'élément sélectionné, en prenant la valeur passée en argument.

valeur : nombre positif représentant le nouveau décalage désiré.

## MMI 1A

---

```
$( "div" ).scrollTop( 300 );
```

Cette méthode renvoie un objet jQuery.

### **scrollLeft(valeur)**

Modifie le décalage (en pixels) entre le bord gauche du document (left) et l'élément sélectionné, en prenant la valeur passée en argument.

valeur : nombre positif représentant le nouveau décalage désiré.

```
$( "div" ).scrollLeft( 300 );
```

Cette méthode renvoie un objet jQuery.

*Exercice : Par le clic sur un bouton, permettre à l'utilisateur d'atteindre directement le second paragraphe d'un texte.*

## 2.6. APPLICATIONS

### 2.6.1 Redimensionner la taille des caractères

Nous allons permettre à l'utilisateur de modifier, à son gré, la taille de la police de caractères pour un meilleur confort de lecture.

*Exercice : permettre au clic d'un premier bouton d'agrandir la taille des caractères. Un second bouton est prévu pour diminuer celle-ci.*

### 2.6.2 Zoom sur image avec une légende

*Exercice : soit une page qui présente des images miniatures. Au survol de la souris, l'image est affichée dans sa taille réelle. Le script offre la possibilité de joindre une légende à celle-ci. La légende qui apparaît dans l'image agrandie est reprise dans l'attribut title du lien. Grâce au lien vers l'image dans sa dimension initiale, la page reste accessible aux utilisateurs qui auraient désactivé le JavaScript.*

### 2.6.3 Une infobulle avec jQuery

*Exercice : Afficher une légende au survol des mots JavaScript, Dom et Ajax.*