**CHEAT SHEET JQUERY !!! by Kevin Narsicadou :D**

**Les expressions de sélection :**

|  |  |
| --- | --- |
| **SELECTEUR** | **CORRESPONDANCE** |

|  |  |
| --- | --- |
| \* | Tous les éléments |
| #id | L’élément dont l’identifiant est celui indiqué |
| Element | Tous les éléments du type indiqué |
| .classe | Tous les éléments de la classe indiqué |
| *a*, *b* | Les éléments qui correspondent à a ou à b. |
| *a b* | Les éléments b qui sont des descendants de a. |
| *a* > *b* | Les éléments b qui sont des enfants de a. |
| *a* + *b* | Les éléments b qui viennent juste après a. |
| *a* ~ *b* | Les éléments b qui sont des frères de a. |
| :first | Le premier élément dans le jeu résultant. |
| :last | Le dernier élément dans le jeu résultant. |
| :not(*a*) | Tous les éléments du jeu résultant qui ne correspondent pas à a. |
| :even | Les éléments d’indice pair dans le jeu résultant (commence à 0). |
| :odd | Les éléments d’indice impair dans le jeu résultant (commence à 0). |
| :eq(*indice*) | L’élément d’indice indiqué dans le jeu résultant (commence à 0). |
| :gt(*indice*) | Tous les éléments du jeu résultant qui viennent après (supérieurs à)  l’indice indiqué (commence à 0). |
| :lt(*indice*) | Tous les éléments du jeu résultant qui viennent avant (inférieurs à)  l’indice indiqué (commence à 0). |
| :header | Les éléments d’intitulé, comme <h1> ou <h2>. |
| :animated | Les éléments en cours d’animation. |
| :contains(*texte*) | Les éléments qui contiennent le texte indiqué. |
| :empty | Les éléments sans noeuds enfants. |
| :has(*a*) | Les éléments dont un élément descendant correspond à a. |
| :parent | Les éléments qui ont des noeuds enfants. |
| :hidden | Les éléments masqués, que ce soit par CSS ou ceux de type <input  type="hidden" />. |
| :visible | L’inverse de :hidden. |
| [*attr*] | Les éléments qui possèdent l’attribut attr. |
| [*attr*=*valeur*] | Les éléments dont l’attribut attr est égal à valeur. |
| [*attr*!=*valeur*] | Les éléments dont l’attribut attr est différent de valeur. |
| [*attr*^=*valeur*] | Les éléments dont l’attribut attr commence par valeur. |
| [*attr*$=*valeur*] | Les éléments dont l’attribut attr se termine par valeur. |
| [*attr*\*=*valeur*] | Les éléments dont l’attribut attr contient la sous-chaîne valeur. |
| :nth-child(*indice*) | Les éléments qui sont l’enfant d’indice indiqué de leur élément parent  (commence à 1). |
| :nth-child(even) | Les éléments qui sont un enfant d’indice pair de leur élément parent  (commence à 1). |
| :nth-child(odd) | Les éléments qui sont un enfant d’indice impair de leur élément parent  (commence à 1). |
| :nth-child(*formule*) | Les éléments qui sont un enfant d’indice n de leur parent (commence  à 1). La formule est de la forme an+b pour les entiers a et b. |
| :first-child | Les éléments qui sont le premier enfant de leur parent. |
| :last-child | Les éléments qui sont le dernier enfant de leur parent. |
| :only-child | Les éléments qui sont le seul enfant de leur parent. |
| :input | Tous les éléments <input>, <select>, <textarea> et <button>. |
| :text | Les éléments <input> de type text. |
| :password | Les éléments <input> de type password. |
| :radio | Les éléments <input> de type radio. |
| :checkbox | Les éléments <input> de type checkbox. |
| :submit | Les éléments <input> de type submit. |
| :image | Les éléments <input> de type image. |
| :reset | Les éléments <input> de type reset. |
| :button | Les éléments <input> de type button et les éléments <button>. |
| :file | Les éléments <input> de type file. |
| :enabled | Les éléments de formulaire activés. |
| :disabled | Les éléments de formulaire désactivés. |
| :checked | Les cases à cocher et les boutons radios sélectionnés |
| :selected | Les éléments <option> sélectionnés. |

**Les méthode de parcours du DOM :**

|  |  |
| --- | --- |
| **METHODE** | **RETOURNE UN OBJET JQUERY QUI CONTIENT ….** |

|  |  |
| --- | --- |
| .filter(*sélecteur*) | Les éléments sélectionnés qui correspondent au sélecteur indiqué. |
| .filter(*rappel*) | Les éléments sélectionnés pour lesquels la fonction rappel  retourne true. |
| .eq(*indice*) | L’élément sélectionné qui se trouve à l’indice indiqué (commence  à 0). |
| .slice(*début*, [*fin*]) | Les éléments sélectionnés qui se trouvent dans la plage d’indices  indiquée (commence à 0). |
| .not(*sélecteur*) | Les éléments sélectionnés qui ne correspondent pas au sélecteur  indiqué. |
| .add(*sélecteur*) | Les éléments sélectionnés et tout élément supplémentaire qui  correspond au sélecteur indiqué. |
| .find(*sélecteur*) | Les éléments descendants qui correspondent au sélecteur. |
| .contents() | Les noeuds enfants (y compris les noeuds de texte). |
| .children([*sélecteur*]) | Les noeuds enfants, avec filtrage facultatif par le sélecteur indiqué. |
| .next([*sélecteur*]) | Le frère qui vient immédiatement après chaque élément sélectionné,  avec filtrage facultatif par le sélecteur indiqué. |
| .nextAll([*sélecteur*]) | Tous les frères qui suivent chaque élément sélectionné, avec  filtrage facultatif par le sélecteur indiqué. |
| .prev([*sélecteur*]) | Le frère qui vient immédiatement avant chaque élément sélectionné,  avec filtrage facultatif par le sélecteur indiqué. |
| .prevAll([*sélecteur*]) | Tous les frères qui précèdent chaque élément sélectionné, avec  filtrage facultatif par le sélecteur indiqué. |
| .siblings([*sélecteur*]) | Tous les frères, avec filtrage facultatif par le sélecteur indiqué. |
| .parent([*sélecteur*]) | Le parent de chaque élément sélectionné, avec filtrage facultatif  par le sélecteur indiqué. |
| .parents([*sélecteur*]) | Tous les ancêtres, avec filtrage facultatif par le sélecteur indiqué. |
| .closest(*sélecteur*) | Le premier élément qui correspond au sélecteur, en partant de l’élément  sélectionné et en remontant l’arborescence du DOM. |
| .offsetParent() | Le parent positionné (par exemple de manière relative ou absolue)  du premier élément sélectionné. |
| .andSelf() | Les éléments sélectionnés, ainsi que le jeu d’éléments sélectionnés  précédent placé sur la pile interne de jQuery. |
| .end() | Le jeu précédent d’éléments sélectionnés dans la pile interne de  jQuery. |
| .map(*rappel*) | Le résultat de la fonction rappel lorsqu’elle est invoquée sur  chaque élément sélectionné. |

**Les méthodes d’évènement :**

|  |  |
| --- | --- |
| **METHODE** | **DESCRIPTION** |

|  |  |
| --- | --- |
| .ready(*gestionnaire*) | gestionnaire sera invoqué lorsque le DOM et CSS seront  intégralement chargés. |
| .bind(*type*, [*données*],  *gestionnaire*) | gestionnaire sera invoqué lorsque l’événement type sera  envoyé à l’élément. |
| .one(*type*, [*données*],  *gestionnaire*) | gestionnaire sera invoqué lorsque l’événement type sera  envoyé à l’élément. La liaison est supprimée après l’appel de  gestionnaire. |
| .unbind([*type*], [*gestionnaire*]) | Supprime les liaisons sur l’élément (pour l’événement type, le  gestionnaire particulier ou toutes les liaisons). |
| .live(*type*, *gestionnaire*) | gestionnaire sera invoqué lorsque l’événement type sera  envoyé à l’élément, avec utilisation de la délégation d’événement. |
| .die(*type*, [*gestionnaire*]) | Supprime les liaisons sur l’élément précédemment lié avec  .live(). |
| .blur(*gestionnaire*) | gestionnaire sera invoqué lorsque l’élément perdra le focus  du clavier. |
| .change(*gestionnaire*) | gestionnaire sera invoqué lorsque l’élément perdra le focus  du clavier et que sa valeur aura été changée. |
| .click(*gestionnaire*) | gestionnaire sera invoqué lors d’un clic sur l’élément. |
| .dblclick(*gestionnaire*) | gestionnaire sera invoqué lors d’un double-clic sur l’élément. |
| .error(*gestionnaire*) | gestionnaire sera invoqué lorsque l’élément recevra un  événement d’erreur (dépend du navigateur). |
| .focus(*gestionnaire*) | gestionnaire sera invoqué lorsque l’élément recevra le focus  du clavier. |
| .keydown(*gestionnaire*) | gestionnaire sera invoqué lorsqu’une touche sera enfoncée  alors que l’élément possède le focus du clavier. |
| .keypress(*gestionnaire*) | gestionnaire sera invoqué lorsqu’une touche sera appuyée  alors que l’élément possède le focus du clavier. |
| .keyup(*gestionnaire*) | gestionnaire sera invoqué lorsqu’une touche sera relâchée  alors que l’élément possède le focus du clavier. |
| .load(*gestionnaire*) | gestionnaire sera invoqué lorsque le chargement de l’élément  sera terminé. |
| .mousedown(*gestionnaire*) | gestionnaire sera invoqué lorsque le bouton de la souris sera  enfoncé au-dessus de l’élément. |
| .mouseenter(*gestionnaire*) | gestionnaire sera invoqué lorsque le pointeur de la souris  entrera dans l’élément. Le bouillonnement n’a pas d’effet. |
| .mouseleave(*gestionnaire*) | gestionnaire sera invoqué lorsque le pointeur de la souris  quittera l’élément. Le bouillonnement n’a pas d’effet. |
| .mousemove(*gestionnaire*) | gestionnaire sera invoqué lorsque le pointeur de la souris se  déplacera au-dessus de l’élément. |
| .mouseout(*gestionnaire*) | gestionnaire sera invoqué lorsque le pointeur de la souris  quittera l’élément. |
| .mouseover(*gestionnaire*) | gestionnaire sera invoqué lorsque le pointeur de la souris  entrera dans l’élément. |
| .mouseup(*gestionnaire*) | gestionnaire sera invoqué lorsque le bouton de la souris sera  relâché au-dessus de l’élément. |
| .resize(*gestionnaire*) | gestionnaire sera invoqué lorsque l’élément sera redimensionné. |
| .scroll(*gestionnaire*) | gestionnaire sera invoqué lorsque la position de défilement de  l’élément changera. |
| .select(*gestionnaire*) | gestionnaire sera invoqué lorsque le texte dans l’élément sera  sélectionné. |
| .submit(*gestionnaire*) | gestionnaire sera invoqué lorsque l’élément (un formulaire)  sera envoyé. |
| .unload(*gestionnaire*) | gestionnaire sera invoqué lorsque l’élément sera retiré de la mémoire. |
| .hover(*entrée*, *sortie*) | entrée sera invoqué lorsque la souris entrera dans l’élément et  sortie lorsque la souris en sortira. |
| .toggle(*gestionnaire1*,  *gestionnaire2*,...) | gestionnaire1 sera invoqué lors du premier clic de souris sur  l’élément, gestionnaire2 sera invoqué lors du deuxième clic,  et ainsi de suite pour les clics suivants. |
| .trigger(*type*, [*données*]) | Invoque l’événement type sur les éléments et exécute l’action  par défaut pour l’événement. |
| .triggerHandler(*type*,  [*données*]) | Invoque l’événement type sur le premier élément, sans exécuter  les actions par défaut ni les événements par délégation. |
| .blur() | Déclenche l’événement blur. |
| .change() | Déclenche l’événement change. |
| .click() | Déclenche l’événement click. |
| .dblclick() | Déclenche l’événement dblclick. |
| .error() | Déclenche l’événement error. |
| .focus() | Déclenche l’événement focus. |
| .keydown() | Déclenche l’événement keydown. |
| .keypress() | Déclenche l’événement keypress. |
| .keyup() | Déclenche l’événement keyup. |
| .select() | Déclenche l’événement select. |
| .submit() | Déclenche l’événement submit. |

Les méthode d’effet :

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |