

Programmation Web – client riche

M4103 - TD n° 2 (séance 3)



1 Objectifs

Ce TD illustre la partie du cours sur la programmation web et en particulier les concepts liés à la conception d'application riches coté client, c'est-à-dire dans le navigateur. Les concepts principaux abordés seront :

- Les principes avancés du langage JavaScript
- L'exploitation du DOM

Un compte rendu donnant la réponse à chacune des questions posées devra être rendu à votre responsable de TD par email à la fin de chacune des séances. L'objet (sujet) de l'email devra être le suivant :

[S4T][JS][Gx][TDx]Nom-Prénom / Nom-Prénom

Avec Gx le numéro de votre groupe (exemple G1 pour le groupe 1) et TDx le numéro du TD inscrit sur le sujet du TD (exemple TD1 pour le premier TD).

Vous ne pouvez pas faire 2 séances avec le même binôme. De même, vous ne pouvez pas faire plus de la moitié des séances seul sauf si votre groupe de TD est assez petit pour que chaque étudiant soit sur une machine de l'IUT.

Veillez, sauf indication contraire de votre responsable de TD, faire l'ensemble des exercices marqués « Obligatoire » dans l'ordre de la feuille de TD. Les exercices « Conseillés » sont à faire si vous avez terminé les précédents ou lors de vos révisions. Les exercices « optionnels » sont là pour les plus passionnés d'entre vous.

2 Documentation (obligatoire)

Pour ce TD mais également pour les TD suivant, sauf mention contraire dans l'énoncé ou si un lien vous est fourni, vous ne devez pas aller chercher des réponses sur internet !! Les seuls sites suivants sont autorisés :

- Le site contenant les [supports de cours et de TD](#).
- Le site de Mozilla : [Mozilla Developer Network](#).
- Les sites du W3C : pour la validation [HTML5](#) ou du [CSS](#). Mais aussi pour avoir la documentation sur le [DOM](#).

3 Prise en main de l'environnement (obligatoire)

Pour réaliser ce TD, vous aurez également besoin d'un éditeur de texte pour écrire le code de vos pages. Vous pouvez par exemple utiliser un des logiciels [Notepad++](#), [ConText](#), [Quanta+](#), [WebExpert](#),... Microsoft vous propose de télécharger gratuitement [Expression Web](#).

Pensez à sauvegarder régulièrement votre travail et à commenter votre code. N'hésitez pas à sauvegarder les différentes versions, évolutions d'un même exercice de manière à pouvoir facilement réviser ou le réutiliser par la suite.

4 Exercice 1 : Les évènements (obligatoire)

Pour cet exercice vous devez partir d'une page HTML5 (valide ça va de soi) qui contiendra au moins une balise h1, deux balises h2 quelques balises div, p, ul ou ol, li, a et span.

Un objet devra avoir une couleur d'arrière-plan spécifiée par son id, d'autres par des classes CSS et d'autres encore par leur type de balises.

Au besoin, vous pouvez utiliser « [Lorem Ipsum](#) » pour générer de faux contenu pour votre page web.



4.1 Sélection d'un objet

L'objectif final de notre exemple sera de pouvoir sélectionner un objet (HTML`Element`) dans notre page via un simple clic souris. La sélection de l'objet sera indiquée à l'utilisateur par un changement de la couleur de son arrière-plan (en rouge par exemple).

Une façon simple de faire cela est d'abonner l'élément de plus haut niveau de notre page à l'évènement « click ».

Ecrivez une fonction (`initSelection()`) qui aura pour but de faire cela. L'ajout d'un écouteur d'évènements « click » qui déclenchera une fonction `selection()` devra se faire en JavaScript.

➤ Comment avez-vous ajouté l'écouteur d'évènement et sur quel objet ?

Maintenant on souhaite écrire la fonction `selection()` qui va mettre en couleur le fond de notre lors du premier clic, et le supprimer lors du clic suivant. Petit indice, sur le paramètre de type évènement de votre fonction de « *callback* », la propriété `target` vous indique l'objet qui a reçu l'évènement [MDN]. Attention, à ne pas confondre avec l'objet qui à l'écouteur d'évènement. Je vous rappelle qu'on avait déjà vu cette différence lors d'un TP de C# en S2.

➤ Que se passe-t-il si vous utilisez `currentTarget` en lieu et place de `target` ?

Astuces :

- pour tester si une variable est déjà définie, on peut comparer son type avec la chaîne de caractères « undefined ».
- pour supprimer une variable de la mémoire (elle sera alors de type non définie), on peut utiliser la fonction `delete(variable)`.

4.2 Insertion d'objets

Ajoutez la div suivante au début du body de votre page au chargement de votre page (donc depuis votre JavaScript).

```
<div id="frameInsert">
  <input type="button" value="DIV" onclick="insertB('div')"/>
  <input type="button" value="P" onclick="insertB('p')"/>
  <input type="button" value="H2" onclick="insertB('h2')"/>
  <input type="text" id="toInsert"/>
</div>
```

Ecrivez une fonction `selection2()`, qui fonctionne comme la fonction `selection()` sauf que

1. `selection2()` ne permet de sélectionner d'un seul élément à la fois. On mettra le fond en bleu pour différencier les 2 fonctions.
2. `selection2()` ne permet pas de sélectionner la div ci-dessus ainsi que les éléments qu'elle contient.

Ecrivez maintenant la fonction `insertB(string)` qui insert avant l'élément sélectionné (par la fonction `selection2()`) un élément dont le type est passé en paramètre sous forme d'une chaîne de caractères et qui contient le texte fourni dans l'input de type « text ».

Les éléments ainsi ajoutés à votre page, pourront ensuite eux même être sélectionnés ...

➤ Comment avez-vous ajouté l'élément ?

➤ Comment avez-vous fait pour que la fonction `selection2()` ignore les éléments de la div donnée ci-dessus ?

5 Exercice 2 : parcours de l'objet document (conseillé)

Puisque vous avez fini la fiche de TD, n'hésitez pas à revenir sur la fiche précédente et principalement le dernier exercice. Et si vous pensez l'avoir déjà fini, ajoutez les balises ci-dessous à votre programme et essayez de rechercher le texte



« ornare ». Si l'ensemble des occurrences n'est pas bien sélectionné (il y en a 8 dans les balises ci-dessous), corrigez votre programme !

```
<div> Proin ornare nunc sodales <span>ornare</span> lorem rhoncus ornare fringilla</div>  
<div> Proin ornare nunc sodales or<span>nare</span> lorem rhoncus ornare fringilla</div>  
<div> Proin ornare nunc sodales ornare lorem rhoncus ornare fringilla</div>
```

6 Exercice 3 : parcours de l'objet document (optionnel)

En repartant de votre exercice 1 (premier partie), ajoutez les fonctionnalités suivantes :

1. Quand je sélectionne un HTMLÉlement via un clic, l'arrière-plan de l'objet passe en rouge, celui de son père en orange.
2. Si je sélectionne un obj O_1 , puis son père PO_1 , O_1 et PO_1 ont un arrière-plan rouge et l'arrière-plan du père de PO_1 est en orange. Maintenant, si je resélectionne PO_1 , son arrière-plan doit revenir orange.

7 Conclusion

J'espère que ce deuxième TD vous a permis de découvrir ou redécouvrir les bases du parcours de JavaScript et vous a donné une idée des possibilités qu'offre ce langage. Dans le prochain TD nous explorerons d'autres propriétés, nous développerons des exemples un peu plus complexes et nous irons un peu plus loin dans les subtilités du langage.