

Exercice 1: Mon premier programme JS

Q1 : Etudiez (avec un éditeur de texte comme Notepad++) le code de la page XHTML nommée `exo1.html` qui est contenue dans l'archive [exo1.zip](#).

Q2 : A la lecture du code, notez sur une feuille la trace d'exécution de cette page : évolution des valeurs des variables pour chaque ligne, contenus des boîtes de dialogues affichées.

Ligne	Valeur Variable	Affichage
9	maVar=1	x
11	Ajout de 1 dans cette dernière	x
12	maVar=3	Après ajout de 1, ma variable vaut 2
13	maVar=3	Après incr ma var vaut 3
14	maVar="Quelle valeur veux-tu donner?"	x
15	maVar=[valeur saisie]	Après prompt ma var = [valeur saisie]
14	var="veux-tu naviguer par ici ?"	x
15	x	Modal: Veux-tu naviguer par ici ?
25	x	C'est facile de générer du XHTML dans une page XHTML active
26	x	Il suffit de taper du texte et des balises
27	x	Ca marche même avec des variables : 2 ou des

		fonctions 3
--	--	-------------

Q3 : Ouvrez ce même fichier avec IE, Chrome et/ou Firefox, et comparez la trace que vous aviez prévue avec les informations affichées par le navigateur Web. Essayez d'expliquer les différences qu'il pourrait y avoir entre votre trace sur papier et les affichages du navigateur. Pour vous aider, vous utiliserez le débogueur de Javascript du navigateur choisi (Touche F12).

Les différences visibles peuvent être dues à une incompréhension de la fonction utilisée.

Encore une simple mauvaise interprétation. Exemple: la fonction `incr()` mal interprétée peut ne pas afficher un résultat similaire à celui que l'on aurait pensé.

Q4 : Pouvez-vous maintenant expliquer pourquoi le dernier affichage montre que la fonction `incr()` a fait une concaténation plutôt qu'une addition (alors qu'au début des données affichées dans la page Web, la fonction `incr()` avait bien fait une addition).

L'incrémentation se fait au milieu d'un affichage ce qui fait que l'on a concaténé et non ajouté. Pour modifier , il faut sortir la fonction.

Q5 : Modifiez le code de la page XHTML pour que le dernier affichage force la fonction `incr()` à faire une addition.

<code>document.write('<H3> Ca marche même avec des variables : ' + maVariable + ' ou des fonctions :');</code>
<code>maVariable=incr(maVariable);</code>
<code>document.write(maVariable+'');</code>

Exercice 2 : Centralisation versus Décentralisation des scripts

Q1 : Reprenez l'exercice 1 de façon à ne pas avoir à utiliser le fichier mesfonctions.js.

-> Pouvez-vous placer la fonction dans l'entête de la page XHTML ?

Oui nous pouvons enlever la source et préciser le langage.

Q2 : Quels sont les avantages ainsi que les inconvénients d'utiliser un fichier de script .js ?

Il est déconseillé d'utiliser un script écrit en dur dans la page (bonne pratique), car de nos jours les éléments HTML sont de plus en plus complexes et nous devons être de plus en plus rigoureux. Écrire une fonction en dur pour intercepter un clic sur un élément est pratique mais dès que l'on sort du cadre de la simple interception, il est préférable de faire un fichier .js que l'on appelle. Plus clair et précis.

Exercice 3 : Où placer son code Javascript dans une page web ?

Q1 : Reprenez l'exercice 2 qui n'utilise pas le fichier mesfonctions.js (la déclaration de la fonction incr() se trouve donc dans la page XHTML).

-> Placer la fonction incr() au sein de la balise script du corps de la page XHTML. La page s'exécute-t-elle comme auparavant ?

Oui, elle s'exécute comme auparavant.

-> Est-ce que cela fonctionne si l'on place la fonction incr() après la séquence d'instructions du corps de la page Web ?

Non, cela affiche la fonction.

Q2 : Placer l'ensemble du code Javascript (c'est-à-dire tout le contenu de la balise script) dans le header de la page. La page s'exécute-t-elle toujours correctement ?

Non plus.

Q3 : Placer la fonction `incr()` dans le corps de la page Web en laissant les autres instructions dans le header. La page s'exécute-t-elle correctement ?

Non plus.

Q4 : Quelle est la différence entre placer son code Javascript dans la partie HEAD ou la partie BODY du document ?

Il y a une différence au niveau du chargement. Plus la balise script est à la fin de notre fichier HTML, plus le navigateur va attendre avant de commencer son chargement. Habituellement, quand le script est mis à la fin d'un fichier HTML, c'est pour accélérer le chargement de la page, car le contenu de la page (images, texte, etc.) va être chargé avant les scripts. Ça peut être vraiment intéressant de faire cela, surtout si les scripts qui vont être chargés ne sont pas nécessaires au chargement de la page.

→ En header, on met que ce qui agit sur le rendu visuel de la page. Le reste on le met dans le body en fin.

Q5 : On souhaite que les textes produits par les deux premiers `document.write()` ne soient pas engendrés par du Javascript mais plutôt que ce soit du XHTML en dur dans la page. Modifiez le code de la page web en conséquence.

Exercice 4

Q1 : Ouvrez le fichier exo4.html (contenu dans cette archive) avec un navigateur Internet et observez la portée des variables suivant que celles-ci sont préfixées ou pas par le mot clé var.

En survolant les différentes zones du code avec votre souris, vous verrez quelles sont les variables visibles ou invisibles à un endroit précis du code.

Q2 : Quelle est la différence entre une variable déclarée avec le mot-clé var et une autre déclarée sans le mot-clé var ?

Lorsque la variable est déclarée sans le mot clé var, elle est accessible partout (n'importe quelle fonction du script peut faire appel à cette variable). C'est une variable globale. Mais si on la préfixe du mot-clé var, si la variable est déclarée dans une fonction, seule cette fonction pourra l'utiliser, si elle est déclarée en dehors de toutes fonctions, alors toutes les fonctions du scripts pourront l'utiliser.

Q3 : À votre avis, sans regarder le code source, est-ce que du code Javascript est contenu dans cette page Web ? Expliquer pourquoi.

La code Javascript est contenu sur la page web puisque l'on peut observer qu'au survol de certaines zones avec la souris, la page "change", la page est dynamique, il n'y a pas besoin de charger un élément pour la modifier.

Exercice 5

Exercice 5 joint directement en lien.