

TP JavaScript

Exercice 1: Débuter un Script

Testez chacun des scripts suivants et dites pourquoi ils ne fonctionnent pas. Corrigez les erreurs : vous pouvez vous aider d'un débogueur tel que Firebug ou encore le débogueur intégré de google Chrome :

1. buggy1.html/buggy1.js
2. buggy2.html/buggy2.js
3. buggy3.html/buggy3.js
4. buggy4.html/buggy4.js

Exercice 2: Créer un Slideshow

Le but de cet exercice est de développer le slideshow de la capture d'écran suivante :

- Un clique sur le lien **Move Next** affiche l'image suivante dans la galerie d'image en appelant une fonction **MoveNext()**.
- De même, un clique sur le lien **Move Previous** affiche l'image précédente dans la galerie d'image en appelant une fonction **MovePrevious()**.
- Les images **pict1big.jpg**, **pict2big.jpg**, **pict3.jpg** et **pict4big.jpg** sont fournies dans l'archive du TP.
- Les images seront affichées dans le Slideshow sur une taille de 100px sur 100px. Lorsqu'on double clique sur l'image, elle s'ouvre avec sa taille réelle dans une nouvelle fenêtre.

Exercice 3: Création d'un Drop-down menu avec des liens hypertextes

Écrire le code JS qui permet de créer le menu déroulant suivant avec une liste de site WEB :

- Lorsque l'on sélectionne W3C et on clique sur le bouton **GO** le site Web du W3C schools s'ouvre.
- De même pour les autres.
- Astuce: Utilisez la propriété **location de l'objet window et selectedIndex** de la zone **select** du formulaire.

Exercice 4: Création d'un questionnaire

- Construisez un petit formulaire dans une page **QCM.html** qui contiendra en fait certaines questions d'un QCM à réponse unique et qui demandera en plus les informations suivantes :
 - nom (20 caractères au maximum), CIN(8 caractères au maximum)
 - Classe (boutons radio : A et B)
 - Adresse e-mail.
 - Licence (liste déroulante : GL, RT et IAG).
- Le contenu final du formulaire est indiqué ci-dessous.

Les styles à appliquer sont comme suit :

- Modifier la mise en forme (sous forme de tableau de 2 colonnes et 5 lignes ou toutes les cellules ont un arrière plan noir et une bordure solide de couleur blanc et d'épaisseur 3 px) et les styles des champs du formulaire de la page **QCM.html** pour arriver à un formulaire aussi semblable que possible que la figure ci-dessous.
- Pour les libellés des champs : la police de caractère est de type "Times New Roman", Times, serif; de style gras, l'attribut background-color de couleur vaut #CCCCFF, et la bordure en blanc et ayant une épaisseur de 3px et en trait continu, l'alignement du texte est gauche.
- Pour les différentes zones du formulaire (bouton, zone de sélection, zone de saisie etc.), changer l'attribut background-color en #CCCCFF.
- Vous êtes libres d'enrichir les styles selon vos goûts personnels: l'essentiel est que l'application des styles doit se faire de façon **dynamique** dès que l'utilisateur commence à remplir le formulaire.

Validation du formulaire QCM:

- Lorsque l'utilisateur fini de remplir une zone et la zone a été correctement remplie le curseur est placée automatiquement dans la zone qui suit.
- Ajouter en Javascript une fonction qui vérifie les conditions suivantes pour le formulaire simulant le QCM du DS :
 - Tous les champs sont remplis.
 - L'adresse mail contient un @ et un « . »

- Si ces conditions ne sont pas vérifiées, alerter l'utilisateur et lui indiquer le ou les champs vides/faux.

Correction du formulaire QCM:

- Ajouter une fonction qui vérifie si les réponses aux QCM sont correctes :
 - Vous avez le choix entre générer une nouvelle page **reponse.html** ou une boîte de dialogue qui indiquera : "félicitations !!" si toutes les réponses sont correctes sinon, elle indiquera le nombre d'erreur et les bonnes réponses pour chaque question. Par exemple, voici ce que pourrai afficher la nouvelle fenêtre si toutes les réponses sont correctes.

Exercice 5 : Manipulation des propriétés de l'objet Window

- Écrire le squelette d'une page **property.HTML** ayant les propriétés suivantes :
 - Avec la définition du titre dans l'en-tête, et la définition des couleurs dans la balise BODY.
 - Ajouter le code Javascript permettant de visualiser sous forme d'un tableau HTML toutes les propriétés du document : ses couleurs, le titre, la date de dernière modification, et son adresse et du navigateur : sa version, sa langue, l'ensemble des plugin installé,
- Ajouter à cette page HTML un formulaire, lui-même ne contenant qu'une entrée de type button. Pour une telle entrée, nous devons fournir un nom, une valeur (le texte qui apparaît sur le bouton) et associer une action Javascript à l'événement **onClick**. Faire en sorte que cliquer sur ce bouton provoque le changement de la couleur de fond de page.
- Ajouter un lien hypertexte vers la page de votre université sur la page HTML et écrire le code Javascript qui, lorsque la souris passe sur ce lien, cela fait apparaître la destination de ce lien dans le statut de la fenêtre.
- Ajouter au même document HTML un autre formulaire, et y ajouter deux boutons : le premier qui permet de revenir au document précédent, le second d'aller au document suivant.