

Examen du module Programmation Internet

Jeudi 19 Juin 2008

Durée : 1h30mn

Tous les documents de cours, TD et TP sont autorisés.
Le barème est donné à titre indicatif.

Exercice 1 (4 points).

Définir une feuille de style pour que tous les titres h1 dans une page HTML aient un texte :

- de couleur rouge,
- qui soit aligné à droite,
- qui s'affiche en gras.

Définir une feuille de style pour que tous les titres h2 dans une page HTML aient un texte :

- de couleur verte,
- qui soit centré,
- qui s'affiche en italique.

Exercice 2 (6 points).

Ecrire un code HTML permettant d'afficher un texte, un champ de texte et deux boutons comme sur la figure ci-dessous :

Compteur :

39	
Lancer	Stopper

Ajouter du code JavaScript permettant de :

1. Démarrer ou redémarrer un compteur (dont la valeur s'affiche dans le champ de texte) quand on clique sur le bouton « Lancer ».
2. Arrêter le compteur quand on clique sur le bouton « Stopper ».

Quand on lance le compteur, il est régulièrement incrémenté de 1 et ce processus ne peut s'interrompre qu'en appuyant sur le bouton « Stopper » (ou en fermant le navigateur).

Au chargement de la page, le compteur est à 0 et est arrêté.

Quand on redémarre le compteur après l'avoir arrêté, il repart de sa valeur courante (Il n'est donc pas réinitialisé à 0).

On ne tiendra pas compte des problèmes de débordement numérique pouvant affecter la valeur du compteur.

Exercice 3 (3 points).

Ecrire une fonction JavaScript qui calcule le produit des éléments d'un tableau qu'elle reçoit en paramètre. Si le tableau est vide, la fonction retourne la valeur spéciale « undefined » (Pour cela, on utilise la construction syntaxique suivante : « return; »). La fonction n'a pas à contrôler le fait que les éléments du tableau soient des nombres ou pas.

Exercice 4 (7 points).

1) Une société vendant différents produits met en place sur son serveur Web une page permettant à un internaute de passer une commande.

Ecrire le code HTML et JavaScript permettant d'obtenir le formulaire de saisie représenté et décrit ci-dessous :

Nom :

Prénom :

Adresse e-mail :

Produit sélectionné :

Prix du produit : euros

Le produit sélectionné est un élément de la liste suivante :

Aucun, Livre, CD, DVD

Le champ correspondant au prix doit être mis à jour automatiquement par un code JavaScript quand l'internaute choisit un élément de la liste.

La société a choisi les tarifs suivants :

Aucun → 0

Livre → 10

CD → 15

DVD → 22

Indication : Quand l'élément sélectionné dans la liste change, un événement « Change » est engendré par le navigateur. On dispose de l'attribut « onChange » pour associer un gestionnaire à cet événement.

La soumission du formulaire doit amener l'exécution d'un script CGI référencé par l'URI suivante : /cgi-bin/stocker.pl

Quand on clique sur le bouton de soumission du formulaire (le bouton « Commander »), du code JavaScript doit vérifier que le nom, le prénom et l'adresse e-mail ont été saisis par l'internaute. Si ce n'est pas le cas, une boîte de message signalant **précisément** l'information manquante doit apparaître avant que l'internaute ne recommence sa saisie.

2) On vous demande à présent d'écrire le code Perl du script CGI « stocker.pl ».

Remarque : Vous pouvez faire appel, si besoin, à la bibliothèque « cgi-lib.pl » pour écrire les scripts CGI.

- Dans un premier temps, le script CGI effectue les vérifications suivantes :

- Le nom n'est constitué que des caractères suivants : lettres, espace, apostrophe, tiret.
Exemples de noms valides : D'Arsonval, Saint-Just, De Vinci, a-b'
Exemples de noms invalides : ZX81, 10\$, Molière
- Le prénom n'est constitué que des caractères suivants : lettres et tiret.
Exemples de prénoms corrects : Louis-Ferdinand, Stephanie, -x
Exemples de prénoms incorrects : Pierre-François, Stéphanie, D2R2
- Le produit choisi n'est pas « Aucun ».

Si l'une des trois vérifications échoue, le script retourne une page HTML avec un texte signalant une erreur dans les données envoyées.

- Dans un second temps (si les vérifications ont réussi), le script sauvegarde dans un fichier « commandes.txt » - que l'on suppose dans le même répertoire que le script – la commande de l'internaute sous la forme :

numéro_commande/nom/prénom/adresse_e_mail/produit

« numéro_commande » est un entier positif incrémenté de 1 à chaque nouvelle commande.

Si le fichier « commandes.txt » n'existe pas, le numéro de la première commande sera 1.

Exemple d'un fichier « commandes.txt » possible :

521/Dupont/Eric/dupond.eric@machin.org/Livre
522/Durand/Herve/durandh12@machin.fr/DVD
523/Dupont/Emilie/dupond.eric@machin.org/DVD

On remarquera que le fichier ne commence pas nécessairement par un numéro de commande égal à 1 (Quand une commande a été traitée par la société, elle l'efface de ce fichier pour la stocker dans une base de données).

Après la sauvegarde dans le fichier « commandes.txt », le script CGI renvoie finalement une page HTML indiquant que la commande de l'internaute a été enregistrée.