

TD2

html + Dom + Javascript

Q1 – On propose d'écrire un mini-interpréteur d'une page HTML

- Une fonction doit permettre à l'utilisateur de compter le nombre de balises de chaque type de premier niveau d'arborescence
- Puis une seconde fonction permet de retrouver les balises disposant d'un argument particulier donné dans une fenêtre prompt(). Cette fonction donnera la liste de ces balises et la valeur des arguments recherchés.
- Une troisième fonction permet de retrouver la balise disposant d'un nom particulier donné dans une fenêtre prompt().
- Une quatrième fonction permet de retrouver les balises à partir de leur type donné dans une fenêtre prompt (par exemple h2, p, ...). Cette fonction donnera la liste de ces balises et la valeur des arguments recherchés

Vous fournirez une page html test quelconque.

Exemple

The screenshot shows a web page with the following content:

Exemples d'accès direct aux éléments

Premier titre h2

Un premier paragraphe.

Un deuxième *paragraphe*, portant l'identifiant p12.

Un troisième paragraphe.

Deuxième titre h2

Un quatrième paragraphe.

Un cinquième **paragraphe**, portant le name unpar.

Un sixième paragraphe, portant le name unpar.

Un septième paragraphe.

At the bottom left, there is a dropdown menu with the following options:

- Analyse complète
- Faites votre choix --
- Analyse complète
- Recherche d'identificateurs
- Recherche d'un élément par son nom
- Recherche d'éléments par le nom de balise

At the bottom right, there is another dropdown menu with the following options:

- Faites votre choix --
- Faites votre choix --
- Analyse complète
- Recherche d'identificateurs
- Recherche d'un élément par son nom
- Recherche d'éléments par le nom de balise

Below the second dropdown menu is a button labeled "Terminé".

Q2 – On se propose de créer de toutes pièces des éléments d'une page html.

Pour cela, écrire une fonction qui permette de créer un tableau de n lignes et m colonnes dans un div dont on précisera la position dans la page.

Le texte des cellules du tableau sera, dans un premier temps, automatique : par exemple ligne 1 colonne 1 ...

La possibilité de créer un tableau est offerte par un élément de liste. Un clic sur cet élément de liste permet de faire apparaître un menu ad-hoc dans lequel l'utilisateur a la possibilité de créer une entête et un pied de page (thead et tfoot) ou non, de mettre des bordures..etc. Un bouton "GO" lance la création du tableau.

Exemple

♦ Création d'une table

lignes : colonnes :

entête et pied

OUI : NON :

bordures

OUI : NON :

GO :

Ceci permet de créer la table

ceci est un tableau de 2 lignes et de 2 colonnes

colonne 0	colonne 1
ligne 1, colonne 1	ligne 1, colonne 2
ligne 2, colonne 1	ligne 2, colonne 2
colonne 0	colonne 1

Puis vous améliorerez votre page en permettant à l'utilisateur de créer d'autres éléments comme une liste et un formulaire. Un sous-menu apparaît en fonction du choix opéré par l'utilisateur. Par exemple, pour créer une liste l'utilisateur fera apparaître un nouveau choix qui est de créer un élément ou une sous liste ; le type de liste (ul, ol ou dl) ...etc.

- ♦ Création d'une table
- ♦ Création d'une liste
- ♦ Création d'un formulaire

Q3 : examen Juin 2014

Le but de la question est de permettre à un utilisateur de créer une liste quelconque (par exemple des villes associées au département duquel elles sont préfectures, ou encore des noms d'étudiants associés à l'âge de chacun ...), puis de trier cette liste soit par ordre alphabétique (noms des villes) soit par ordre numérique (numéros des départements).

Au départ, l'utilisateur dispose d'une page html dans laquelle deux grands chapitres apparaissent lui offrant 4 possibilités, comme illustré ci-dessous.

Entrez les éléments de votre liste

Triez les éléments de votre liste

Voici le code html5 de cette page que vous ne pouvez pas modifier.

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml" lang="fr">
<head>
```

```

<title> Appenchild</title>
<meta charset="utf-8" />
<script type="text/javascript" src="Q1bis.js"></script>
</head>
<body>
<h1 id = "ordre">Entrez les &eacute;l&eacute;ments de votre liste</h1>
<table id="liste">
</table>
<form action = "http://www.truc.fr ">
  <p>
    <input type="button" value="ajouter liste" id="ajout" />
    <input type="button" value="supprimer element" id="suppression" />
  </p>
</form>
<h1 id = "ordre">Triez les &eacute;l&eacute;ments de votre liste</h1>
<form action = "http://www.autre.fr ">
  <p>
    <input type="button" id="abc" value="Alphab&eacute;quement" />
    <input type="button" id="geo" value="Num&eacute;riquement" />
  </p>
</form>
</body>
</html>

```

Question 1 : Générer la liste

A chaque clic du bouton "Ajouter liste", une fonction javascript "ajout()" demande à l'utilisateur, par une fen&eatre prompt, la chaêne de caractéres qu'il souhaite générer puis, par une autre fen&eatre prompt, la valeur numérique associée.

Cette m&eame fonction insére les deux informations dans l'élément "liste".

Ecrire la fonction "ajout" et les éventuelles fonctions nécessaires à sa mise en oeuvre. Naturellement, vos fonctions devront vérifier que les informations entr&eées sont "cohérentes", par exemple que la seconde est bien numérique. Vous pouvez autoriser les valeurs en double (entrer plusieurs fois la m&eame donnée).

Exemple apr&eas trois clics sur le bouton "ajouter liste"

Entrez les éléments de votre liste

Dijon 21

Paris 75

Lyon 69

Question 2 : Un clic sur le bouton "supprimer" permet à l'utilisateur, si la liste n'est pas vide, de supprimer un élément (les deux informations alphabétique et numérique) dont il aura fourni le numéro dans la liste.

Ecrire la fonction javascript supprimer() et les fonctions nécessaires à sa mise en place.

Question 3 : tri des éléments

Un clic sur un des deux autres boutons permet à l'utilisateur de trier sa liste non vide, soit par ordre alphabétique, soit par ordre numérique.

Ecrire ces deux fonctions de tri ainsi que les fonctions nécessaires à leur mise en oeuvre. Le résultat du tri est affiché à la place de la liste initiale et le titre dans la page html est changé en conséquence. Attention, les nombres ne sont pas forcément sur 2 chiffres.

Exemple

Tri numérique

Nice	6
Marseille	13
Dijon	21
Lyon	69
Paris	75

Tri alphabétique

Dijon	21
Lyon	69
Marseille	13
Nice	6
Paris	75

Question 4 (ajout par rapport à l'examen). L'utilisateur désire modifier une information déjà fournie. Pour cela, il doit cliquer sur l'élément de la liste à modifier et entrer les nouvelles valeurs. Vous devez associer un événement à chaque élément de la liste et à cet événement, associer la fonction `Modif_element()` qui permet soit de modifier le nom soit de modifier la valeur numérique associée.