

Développement Front-End

LP – SIMO / Bigdata

TP 3 (JS-jQuery)

L'objectif du TP est de vous faire travailler sur les aspects suivants

- Structures de contrôles, boucles et tableaux
- Gestion d'évènements, modification du DOM, ...
- Transition Javascript / jQuery
- Structures de données orientées objets

Partie 1

Structures de contrôle, boucles, objets tableaux, évènements

Dans votre répertoire de travail (qui porte votre nom et prénom), créer un nouveau répertoire nommé TP3. Dans ce répertoire, créer un deuxième répertoire nommé partie 1 dans lequel vous aller travailler.

Dans la suite, utilisez les supports du cours à partir de la diapo 75 pour vous aider.

Structures de contrôle et boucles

- Écrire un programme qui choisit un nombre au hasard entre 0 et 10 et qui demande à l'utilisateur de le deviner à travers une boîte de dialogue (il ne s'agit pas de la balise Dialog de HTML5).
- Le programme doit ainsi orienter l'utilisateur en lui indiquant si son chiffre est au dessus ou en dessous du nombre de trophées choisi.
- Dans un deuxième temps, limiter le nombre d'essais possible à trois.

Tableaux et gestion d'évènements

- Dans votre page js, créer 4 tableaux, correspondant à 4 pays de votre choix. Chaque tableau doit contenir au moins 4 villes du pays correspondant. (Exemple : tableau France, contient les valeurs Paris, Marseille, Lyon, Grenoble).
- Maintenant, votre page doit demander à l'utilisateur le nom de la ville où il se trouve, à travers un formulaire contenant un unique champ de saisie.
- Dans le même formulaire, créer un bouton "submit" qui, génère une alerte souhaitant la bienvenue au pays correspondant à la ville saisie.
- Ajouter un contrôle de saisie vérifiant que le champ texte contient une ville parmi celles qui sont prédéfinies dans votre fichier js. (Pensez aussi à bloquer la propagation d'évènement pour la validation du formulaire).

Partie 2

Modification du DOM – transition à jQuery

- Utiliser le code de l'exercice précédent pour créer un système de suggestion dynamique sur votre page web. Ainsi, à travers un bouton « Choisir une destination », votre page web demande à l'utilisateur le nom d'une ville où il souhaite passer ses vacances (parmi les noms prédéfinis dans vos quatre tableaux). Une fois la ville choisie, la page web affichera le message suivant : « Bienvenue à ... (pays correspondant à la ville), vous pouvez également visiter les lieux suivants : (les autres villes du même tableau)
 - ville 1
 - ville 2
 - ville 3
 - ville 4 »
- Les villes affichées correspondent aux valeurs préenregistrées dans les tableaux et doivent apparaître sous forme de liste.
- Une fois les villes affichées, le bouton « Choisir une destination » devient « Ajouter une autre destination » et permet de rajouter infiniment d'autres suggestions du même type. (Attention, une confirmation doit précéder chaque ajout de nouvelle destination.)

jQuery

- Toujours dans votre répertoire TP3, créer un nouveau fichier JS qui ne contiendra que du jQuery !
- Refaire le même système de suggestions en utilisant jQuery, tout en adaptant votre fichier HTML à sa prise en compte.
- Il est recommandé d'utiliser la fiche de transition JS-jQuery disponible sur Chamilo.

Partie 3

POO - (application à votre site)

Votre chef de projet est ravi de votre travail esthétique sur les pages du site. Néanmoins, il trouve que cela manque d'interactivité. Il n'apprécie pas trop l'idée que vous devez changer de page à chaque fois que vous cliquez sur un onglet. Vous allez donc faire évoluer votre site, tout en ayant la liberté d'utiliser du js natif ou jQuery.

- Dans votre répertoire TP3, créer un répertoire "TP3_projet" et recopiez y votre projet existant (html-css). (Si vous travaillez sur Github, n'oubliez pas de faire un nouveau commit à la fin de chaque partie de ce TP pour enregistrer votre travail.)

Modélisation (objet json)

- Votre catalogue (d'animaux, destinations touristiques, ou joueurs) a un inconvénient. Les informations ne s'affichent pas d'une manière dynamique. Ce n'est pas très pratique pour ajouter des éléments. Vous allez donc tout d'abord
 - Définir une structure de données permettant de représenter vos éléments : un(e) Destination / Animal / Joueur ...
 - Créer une variable dans votre code, contenant les représentations des destinations / animaux / joueurs ... que vous avez pour l'instant dans votre tableau.

Affichage de la liste à partir du modèle

- Maintenant, vous devez utiliser votre structure de données pour afficher la liste des animaux / destinations / joueurs ... dans la page correspondante.

Affichage du menu à partir du modèle

- Évidemment, un menu en dur alors qu'on gère les données qu'il doit afficher, ça ne marche pas... Vous sauriez améliorer ça ?

Insertion / suppression dynamique des données

- Votre patron vous remercie et souhaite avoir la main pour rajouter lui même des données dans votre page catalogue (le tableau destination/animaux/joueurs).
- Ce sera plus pratique pour lui d'avoir un champ de saisie et un bouton pour rentrer des données (nom, description, image, etc.).
- Également, un bouton "modifier" au niveau de chaque élément permettra de modifier les données de ce dernier.
- Pour finir, un bouton "supprimer" permettra de supprimer des éléments existant.