

TP de javascript

TP1 : Introduction à javascript

Équipe pédagogique

Mars 2018

Pour rendre les pages accessibles en ligne, il faudra les copier sur le serveur **Forge** dans le dossier **forge_html**. La page **index.html** se trouvera à l'adresse **http://forge.info.univ-angers.fr/~login/index.html**. Toutes les pages, une fois testées localement (c'est-à-dire sur votre ordinateur), devront être validées puis copiées sur **Forge**.

1 Nombre mystère

Le but de cet exercice est de produire une page HTML où une fonction javascript tire un nombre entier aléatoire compris entre 1 et 30 (le nombre mystère), et tente de le faire deviner à l'utilisateur. Pour cela, une zone de texte permet à l'utilisateur de saisir une proposition, et de la soumettre en cliquant sur un bouton. La page web affiche alors des messages comme « c'est plus », ou « c'est moins », ou encore « bravo ! » quand l'utilisateur a trouvé le nombre mystère.

Question 1.1 Produire une première version de la page web où ces messages sont transmis par l'intermédiaire de fenêtres pop up dans un premier temps, puis en générant des paragraphes dans la page web dans un second temps.

Question 1.2 Modifier la page web de manière à ce que les fenêtres pop up ne soient utilisées que lorsque l'utilisateur émet une proposition non valide (une chaîne de caractères, ou un nombre hors de l'intervalle $\{1, \dots, 30\}$). On pourra utiliser la fonction `Number.isInteger()`.

Question 1.3 Modifier la page web de manière à ce que les propositions de l'utilisateur et les réponses les plus récentes de l'ordinateur apparaissent en haut plutôt qu'en bas (avec la fonction `parent.insertBefore()`).

Question 1.4 Faire en sorte que le nombre de tentatives pour deviner le nombre mystère soit inférieur ou égal à 5 (après quoi, le joueur a perdu). En cas de succès du joueur, l'ordinateur affiche le nombre de tentatives qui lui ont été nécessaires.



2 Tic-tac-toe interactif

Question 2.1 Reprendre le fichier HTML du tic-tac-toe du précédent TP, et ajouter manuellement une étiquette à chaque case (comme C00 pour la première case en haut à gauche, et C22 pour la dernière case en bas à droite), ainsi qu'un appel à la fonction javascript `clic('C00')` (pour la case C00). L'argument de la fonction doit être l'étiquette de la case associée. De plus, on supprimera le contenu initial des cases (les X et les O seront supprimés manuellement).

Question 2.2 Écrire la fonction javascript `clic(cellule)` de manière à ce que chaque clic impair dans une case y fasse apparaître un X, et chaque clic pair y fasse apparaître un O.

Question 2.3 Faire en sorte qu'on ne puisse pas modifier le contenu d'une case qui contient déjà X ou O. Le titre du tableau (qui devra apparaître sous le tableau) devra être : « C'est le tour du joueur * », où * est soit X soit O en fonction du prochain caractère qui sera inscrit dans une case.

Question 2.4 Écrire une fonction `gagne(J)` qui renvoie `true` si le joueur J a gagné, et `false` sinon. Cette fonction sera appelée après chaque coup du joueur J, afin que la partie s'arrête s'il y a un gagnant. On détectera également les matches nuls.

