



Projet de jeu JavaScript-HTML

Jeu "deviner un nombre"

Rédigé par

David ROUMANET
Professeur BTS SIO

Changement

Date	Révision

Sommaire

A Introduction.....	1
A.1 Objectifs du cours.....	1
B Cahier des charges.....	2
B.1 But.....	2
B.2 Fonctionnement.....	2
B.3 Contraintes.....	2
C Aides pour la conception.....	3
C.1 HTML.....	3
C.1.1 Astuce HTML N°1 : utiliser des identifiants (ID).....	3
C.1.2 Astuce HTML N°2 : utiliser un champ de saisie.....	3
C.1.3 Astuce HTML N°3 : Associer une fonction JavaScript à un bouton.....	3
C.2 JavaScript.....	4
C.2.1 Astuce JS N°1 : Écrire à l'emplacement d'un identifiant HTML.....	4
C.2.2 Astuce JS N°2 : Lire la valeur d'un identifiant HTML.....	4
C.2.3 Astuce JS N°3 : Choisir un chiffre au hasard.....	5
C.2.4 Astuce JS N°4 : utiliser une fonction qui renvoie un texte.....	5
C.2.5 Astuce JS N°5 : appeler une fonction sur un clic d'un bouton HTML.....	5

A Introduction

La programmation nécessite de pratiquer et de réfléchir pour en comprendre la logique.

Ce premier projet est relativement simple, mais doit permettre de prendre de bons réflexes :

- Les éléments de codes donnés ici ne sont que des pistes
- Ce n'est pas le programme complet, vous devez créer le code
- Il faut d'abord réfléchir pour assembler correctement les pièces du puzzle

A.1 Objectifs du cours

Les objectifs de ce cours – classés selon la taxonomie de Bloom – sont :

Mémoriser	Utiliser les notions vues en classe sur JavaScript, HTML et CSS
Comprendre	Comprendre le fonctionnement des éléments fournis
Appliquer	Mettre en œuvre les fonctions étudiées
Analyser	Rechercher les liens entre les morceaux de code Créer son propre code
Évaluer	–
Créer	–

B Cahier des charges

Vous représentez la société JeuCodeFacile.

Un client vous demande de créer un petit jeu pour les enfants en bas âge (entre 4 et 7 ans) qui permet de faire travailler les chiffres de 0 à 9.

B.1 But

Le jeu doit permettre l'apprentissage des chiffres et leur ordre, en utilisant les notions de "plus petit" ou "plus grand".

B.2 Fonctionnement

Le jeu de base doit permettre de saisir un chiffre dans un champ, et lorsque l'enfant clique sur le bouton [Tester], il doit afficher un commentaire sur le chiffre choisit :

- Le chiffre secret est plus grand
- Le chiffre secret est plus petit
- Tu as trouvé le chiffre secret

Le développeur peut utiliser des images libres de droit, des musiques ou des sons, mais surtout, il doit rendre l'affichage visuellement attractif avec un peu de CSS.

B.3 Contraintes

Le jeu sera dans un dossier appelé "NOM_jeu_deviner-chiffre" (NOM est votre nom de famille)

Ce dossier contiendra un dossier "medias" pour les éventuelles images et les effets sonores.

Le jeu sera constitué de 4 fichiers :

- index.html
- aide.html
- jeu.js
- style.css

Le fichier index.html permet de jouer et contient un lien vers le fichier aide.html

Le fichier aide.html contient la notice du jeu (avec une capture d'écran qui permet de comprendre et expliquer le jeu) et un lien de retour vers la page index.html

C Aides pour la conception

C.1 HTML

C.1.1 Astuce HTML N°1 : utiliser des identifiants (ID)

Pour permettre à JavaScript, d'écrire dans une zone précise, on devra indiquer à JavaScript où se trouve cette zone : elle sera repérée par son ID.

Par exemple

```
<p id="zoneReponse">texte initial</p>
```

C.1.2 Astuce HTML N°2 : utiliser un champ de saisie

Les formulaires utilisent fréquemment ce genre de champ. On peut les paramétrer facilement pour accepter un nombre de caractères, pour ne prendre que certains caractères en compte, etc.

La syntaxe est la suivante :

```
<input type="texte" name="reponseJoueur" id="reponseJoueur">
```

Il n'y a pas de balise fermante. Il faut aussi utiliser un identifiant pour JavaScript.

Les paramètres pour un champ de type texte sont :

- placeholder="valeur bidon" pour écrire un texte grisé dans le champ quand il est vide
- maxlength="1" pour n'autoriser qu'un seul caractère
- size="1" pour n'afficher qu'un seul caractère
- pattern="[0-9]" pour n'accepter que les caractères entre 0 et 9

Il existe aussi un champ de saisie pour les nombre, voir la documentation Mozilla : <https://developer.mozilla.org/fr/docs/Web/HTML/Element/input/number>

C.1.3 Astuce HTML N°3 : Associer une fonction JavaScript à un bouton

Le plus simple est de créer un bouton avec un ID pour que JavaScript puisse l'associer :

```
<input type="button" name="boutonTester" id="boutonTester" value="Tester">
```

C.2 JavaScript

C.2.1 Astuce JS N°1 : Écrire à l'emplacement d'un identifiant HTML

Pour pouvoir écrire dans une zone particulière, JavaScript peut utiliser la commande suivante :

```
document.getElementById("zoneReponse").innerHTML = "mon Texte;
```

C.2.2 Astuce JS N°2 : Lire la valeur d'un identifiant HTML

Pour pouvoir écrire dans une zone particulière, JavaScript peut utiliser la commande suivante :

```
let resultat = document.getElementById("zoneReponse").innerHTML.value;
```

C.2.3 Astuce JS N°3 : Choisir un chiffre au hasard

JavaScript peut tirer des nombres au hasard, entre 0 et 1. En multipliant par un nombre, on pourra tirer un nombre entre 0 et une valeur inférieure à ce nombre. Cependant, il faut utiliser la fonction `floor` pour arrondir la valeur à un entier.

```
let nombre = Math.floor(Math.random() * max);
```

Explication : `Math` est une bibliothèque JavaScript qui contient plein de fonctions mathématiques. Nous utilisons :

- `random()` pour générer un nombre à virgule entre 0 et 0.99999
- `floor()` pour arrondir un nombre à l'entier inférieur le plus proche (`floor(5.87)` donne 5)

C.2.4 Astuce JS N°4 : utiliser une fonction qui renvoie un texte

Pour comparer un chiffre avec le chiffre choisi par JavaScript, il est préférable de créer une fonction. Cette dernière contiendra une condition qui renverra le texte voulu (plus petit ou plus grand ou gagné).

```
Function Comparer(x, y) {  
  // test si ... renvoie texte1 sinon renvoie texte2  
  // le mot clé de renvoie = return texte;  
}
```

C.2.5 Astuce JS N°5 : appeler une fonction sur un clic d'un bouton HTML

L'idée est de créer une simple fonction JavaScript qui va s'exécuter lors d'un clic sur le bouton. Il s'agit d'une simple fonction qui doit aller lire le contenu d'un champ par exemple et appeler d'autres fonctions (comme celle de comparaison)

```
Function btnTester() {  
  let resultat = document.getElementById("zoneReponse").innerHTML.value;  
  let texteReponse = Comparer(x, resultat)  
  document.getElementById("zoneReponse").innerHTML = texteReponse;  
  if (texteReponse == "Trouvé") {  
    // tirer un nouveau nombre au hasard dans x  
    // ...  
  }  
}  
  
// création de l'événement dans JavaScript  
// soit let btn = document.getElementById("boutonTester"); //, soit :  
let btn = document.querySelector("boutonTester");  
btn.addEventListener('click', updateBtn);
```