

Introduction au développement informatique

Matthieu Falce & Jordan Bouchoucha

November 29, 2017

Le but de ce TP est de vous faire réaliser une page web interactive en 2h.

Vous allez créer un mini-jeu dans lequel l'utilisateur va devoir retrouver un nombre choisi par l'ordinateur.

Cela vous permettra de découvrir le Javascript, HTML et CSS. Pour les points de syntaxes ou pour tous points de blocage, vous référez à la "feuille de triche", si ce n'est pas clair, n'hésitez pas à nous poser des questions.

1 Javascript

Le Javascript contient "l'intelligence" de votre jeu. Il va permettre au navigateur d'indiquer à l'utilisateur si le nombre donné est plus petit, plus grand ou si c'est le bon.

1.1 Implémenter l'algorithme

Nous avons trouvé l'algorithme à implémenter durant la présentation.

À vous de l'implémenter.

Notions à utiliser : les types de variables, les boucles, les conditions et tests ainsi que les fonctions `input`, `console.log`, `alert`.

1.2 Mettre du code dans une fonction

Pour pouvoir faciliter la suite, vous allez **encapsuler** la logique du jeu dans une fonction. Elle agira comme une boîte noire qui fait ce qu'il faut.

Pour un nombre donné en paramètre, la fonction va écrire si c'est plus, moins ou si c'est bon.

N'hésitez pas à vous servir de ce que vous avez fait précédemment.

Squelette de la fonction :

```
function tour(nombre){  
    // ici c'est à vous  
    ...  
}
```

Contrat d'utilisation :

```
// imaginons que le nombre  
// à trouver soit 32  
tour(13); // écrit "c'est plus"  
tour(43); // écrit "c'est moins"  
tour(32); // écrit "c'est bon"
```

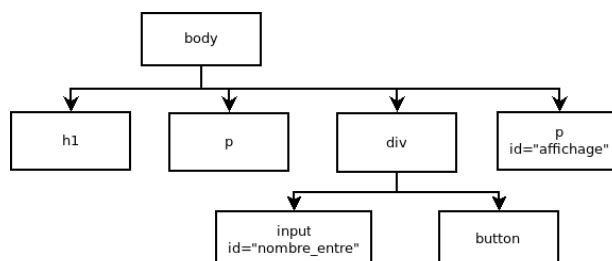
2 HTML

Les sites n'utilisent plus de pop-up pour l'interaction depuis environ 25 ans. Construisez une page web pour communiquer avec l'utilisateur.

2.1 Construction de la page

Réalisez une page ayant cette structure

The screenshot shows a web page layout with the following elements: a title "Le jeu le plus nul du monde !", a subtitle "Ici se trouve le jeu le moins intéressant jamais créé, en ce monde.", an input field with the placeholder "Jouer !", and a button labeled "Jouer !".



2.2 Adapter le jeu à la programmation réactive

Avant, nous avions une "boucle principale" pour continuer le jeu jusqu'à avoir trouvé. A présent le navigateur nous en fournit une. Il faut légèrement changer le code de la fonction `tour` précédente.

On peut faire en sorte que la validation du bouton lance un tour du jeu. Pour cela, nous allons faire une fonction `game` qui ne prend pas de paramètres. Nous allons dire au bouton de lancer la fonction lorsque l'on clique dessus :

```
<button onclick="game()" type="button">
```

Structure de `game` :

```
function game() {
    nombre_demande = get_input_number();
    tour(nombre_demande); // tour est la fonction précédente
}
```

Code supplémentaire fourni :

```
function get_input_number(){
    // fonction permettant de récupérer le nombre dans le champs de texte
    return parseInt(document.getElementById('nombre_entre').value);
}
```

2.3 Améliorer l'interaction

Un utilisateur n'a pas à regarder la console de son navigateur pour savoir s'il a le bon nombre ou pas...

Utilisez le code suivant pour remplacer le `console.log` et mettre à jour des éléments de la page web.

Code supplémentaire fourni :

```
function update_display(text){
    // fonction permettant de mettre à jour le paragraphe de résultat.
    document.getElementById('affichage').style.visibility = "visible";
    document.getElementById('affichage').textContent = text;
}

function update_background_color(color){
    // fonction permettant de mettre à jour la couleur de fond de la page.
    document.body.style.backgroundColor = color;
}
```

3 CSS

Cette page est moche. Améliorons là.

3.1 Centrer les textes

Utiliser cette déclaration : `text-align: center;` pour centrer les textes sur une ligne.

Où doit-on la placer pour tout centrer ?

3.2 Couleurs

Nous pouvons changer les couleurs du titre et du contenu.

Que pensez-vous de : `color: rgb(15, 15, 15);` ou : `color: rgb(55, 55, 55);`

Nous pouvons également changer la couleur de fond de la page pour indiquer à l'utilisateur si c'est plus ou moins.

Utiliser `update_background_color` défini précédemment pour le mettre à jour.

4 Bonus

4.1 Si l'utilisateur ne rentre pas un nombre...

Comment détecter si un utilisateur n'a pas entré un nombre ? Aller regarder la documentation des NaN pour voir comment les repérer.

4.2 Centrer verticalement

Le centrage vertical dans une page est (était...) un vrai problème en CSS. Regardez la documentation de flexbox pour avoir une solution actuelle.