TP JavaScript Jeu du chaton

1. LE JEU DU CHATON

Pour commencer, copiez dans votre dossier personnel les 4 fichiers de l'application (html, css, jpg, et js). Notez dans le fichier HTML le lien avec le fichier JS :

<script type="text/javascript" src="trouveLeChat.js" > </script>

Ouvrez le fichier JS :

Le traitement est découpé en plusieurs « fonctions ».

Le début du code consiste à définir des constantes (const) et déclarer des variables (var).

La partie « programme principal » se résume à 1 seule ligne : demarrerJeu() ;

La fonction « demarrerJeu » enchaîne l'exécution dans l'ordre de :

setPlateau(); setSecret(); setChats(); masquer();

TRAVAIL : L'appui sur le bouton « Voir » appelle la fonction « voir » Quand on relâche le bouton, on appelle la fonction « masquer »

2. SETPLATEAU : GENERER DU HTML AVEC DU JAVASCRIPT

On veut créer une grille de ce type :

<div> ligne1</div>	<div> cellul</div>	<div> cellul</div>	<div> cellul</div>			

La structure HTML est un ensemble de <div> ...

Chaque ligne est une <div> qui contient plusieurs <div> (que j'appelle cellules). Dans cet exemple il faut donc créer 6 <div> lignes, et 48 <div> cellules !!! On va utiliser JavaScript pour le faire :

2 boucles imbriquées :

Boucle principale : crée les lignes Boucle secondaire : crée les cellules sur la ligne

C'est la fonction « setPlateau() » qui réalise cette opération.

TRAVAIL : Analysez le code pour déterminer où ajouter la gestion de l'évènement suivant : Quand on clique sur une cellule, on appelle la fonction « demasquer »

3. DEMASQUER : FAIRE APPARAITRE L'IMAGE DU CHAT

La fonction « demasquer() » teste si la case contient une image. Si c'est le cas, elle la fait apparaitre (display = block).

Cette fonction contrôle également les bonnes et les mauvaises réponses.

Une des techniques utilisées est de rechercher un élément fils (childNode) d'une balise :

```
celluleClicked.childNodes[0].style.display = 'block';
```

« celluleClicked » représente la « target » de l'événement, donc la cellule sur laquelle on a cliqué.
Si la cellule n'est pas vide, on peut supposer qu'elle contient une .
On y accède utilisant childNode[0] (le premier élément enfant).

TRAVAIL : Quand on clique sur une mauvaise case, on veut qu'elle passe en rouge et qu'apparaisse le nombre d'essais restant.

4. FONCTION « VOIR »

La fonction « voir » permet de repérer la position des chats. Actuellement, on peut l'utiliser sans limite.

TRAVAIL : Mettre en place le code qui empêche de « VOIR » plus de 2 fois.

5. DES MESSAGES D'ERREUR ...

En affichant la console, vous verrez des cas non prévu initialement qui génèrent des messages d'erreur :

- Si on reclique sur une image qui est apparu
- Si on reclique sur une case rouge qui contient le nombre d'essais. Elle n'est plus vide donc le test :
 if (celluleClicked.innerHTML == "") {

renvoi vrai alors que ce n'est pas une ...

TRAVAIL : Trouvez la solution pour qu'il n'y ait pas d'erreur...