

PROJET - PARTIE JAVASCRIPT

OBJECTIF

Vous devez réaliser en binôme le jeu du morpion.

Le contenu des fichiers HTML et CSS vous sont fournis. Ils permettent de créer l'interface du jeu. Votre objectif est de compléter le fichier JavaScript afin de rendre votre jeu fonctionnel.

Lorsque les parties ci-dessous seront terminées, vous êtes libre de personnaliser votre jeu, c'est-à-dire que vous pouvez :

- Modifier l'apparence
- Ajouter des éléments non demandés
 - Le score des joueurs
 - Saisir et afficher le nom des joueurs
 - ...

Le projet sera à rendre dans un dépôt Moodle dans la partie web dynamique, avant le dimanche 10 mars. Vous devrez fournir dans un dossier .zip, tous les fichiers nécessaires au fonctionnement du jeu. Le dossier .zip devra être nommé comme suit : NOM1_Prénom1_NOM2_Prénom2_Morpion.zip (remplacer les noms et prénoms par ceux de votre binôme).

PAGE HTML

Dans le fichier **morpion.html**, modifier les points suivants :

- Dans la balise `<div id= "regle">` ajouter :
 - Le titre « Le jeu du morpion »
 - Et les règles du jeu
- Dans la partie `<head>`, ajouter les lignes permettant de relier cette page html aux pages css et js.

PAGE CSS

Le fichier **style.css** est déjà fourni, mais pas définitif. Libre à vous de personnaliser votre projet.

PROJET - PARTIE JAVASCRIPT

PAGE JAVASCRIPT

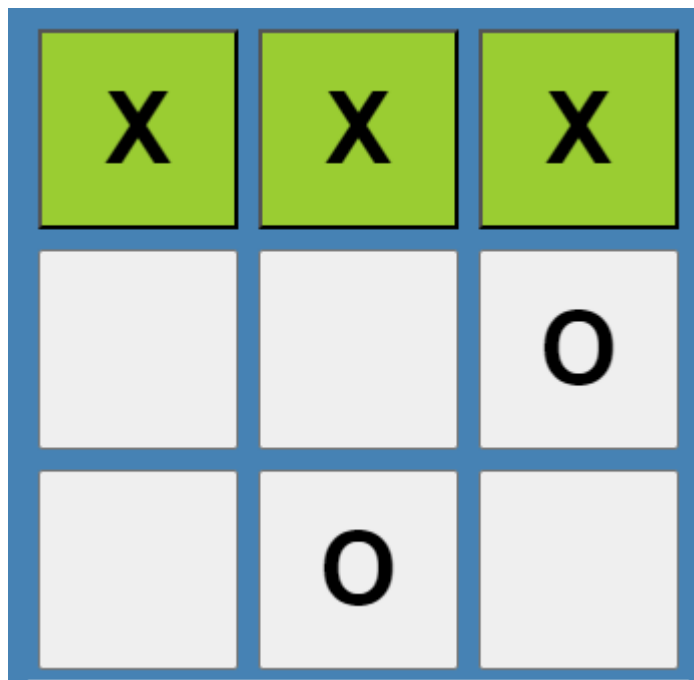
Modifier le fichier **script.js**, en complétant les 5 fonctions ci-dessous.

estValide(bouton) : prend en paramètre un bouton (case de la grille) et retourne vrai si le bouton est vide (case jouable) ou faux si le bouton est plein (case déjà joué).

setSymbol(bouton, symbole) : prend en paramètre un bouton (case) et un symbole (X ou O) et ajoute le symbole sur le bouton (=> modifie le texte du bouton).

rechercheVainqueur(boutons, joueurs, tour) : prend en paramètre la liste des boutons, la liste des joueurs et le tour. Cette fonction va tester tous les cas possibles où un joueur peut gagner. C'est-à-dire vérifier si le joueur correspondant au tour a aligné son symbole sur trois cases consécutives (horizontales, diagonales ou verticales). Si vous trouvez trois symboles alignés, alors changer la couleur de fond de ces trois cases (#9ACD32 couleur verte) et retourner vrai (faire un return permet de stopper le programme lorsque vous avez trouvé une victoire).

Rappel pour comparer plusieurs éléments dans une même condition, vous pouvez utiliser "ET" et "OU". En python on les écrit "and" et "or", en Javascript "ET" = "&&" et "OU" = "||" (alt + 6).



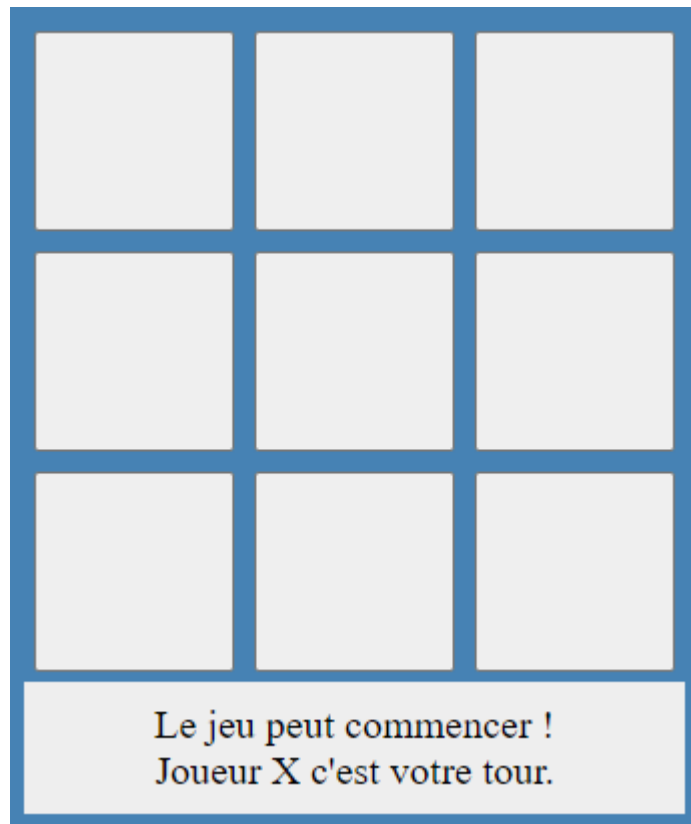
matchNull(boutons) : prend en paramètre la liste des boutons et vérifie chaque bouton pour savoir si la partie est nulle ou non. Si un des boutons est vide (taille = 0) retourner faux (= partie pas finie). Retourner vrai si tous les boutons ont été testés et qu'aucun n'est vide (toutes les cases sont pleines et donc partie nulle).

PROJET - PARTIE JAVASCRIPT

main() : fonction qui permet de dérouler le jeu.

- Initialiser les variables suivantes :
 - o boutons : liste des boutons
 - o joueurs : liste des joueurs représenté par leur symbole (X et O)
 - o tour : nombre de tour commence par 0
 - o jeuFini : faux de base (passe a vrai quand la partie est fini)
 - o afficheur est un objet qui permet de modifier le texte de la balise <div id= "StatutJeu"> par le texte qu'on lui demande (la création de afficheur est donné, vous avez juste à l'utiliser).
- Modifier le statut du jeu pour que s'affiche le nom du joueur a qui c'est le tour (voici comment utiliser afficheur).
`afficheur.sendMessage("Le jeu peut commencer !
 Joueur " + joueurs[tour] + " c'est votre tour.")`

Au début du jeu vous devez avoir cette grille et ce message.

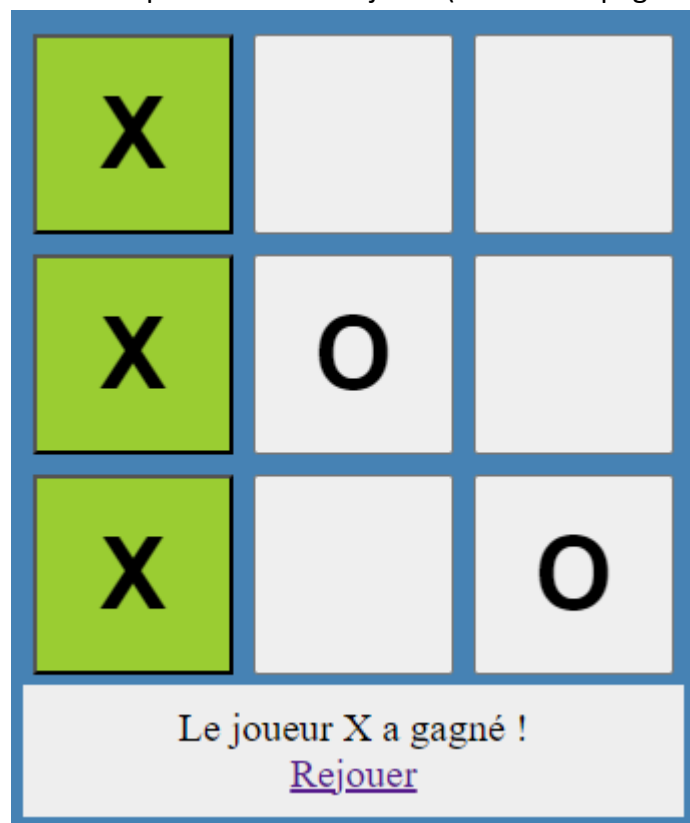


PROJET - PARTIE JAVASCRIPT

- Déroulé du jeu : pour chaque coup joué (9 coups maximum = 9 cases dans la grille) créer un événement au clic sur chaque pion.
 - A chaque fois vérifier si la case est libre, si elle est déjà occupée, modifier le statut du jeu pour prévenir le joueur que la case est déjà occupée et que c'est toujours à lui de jouer.

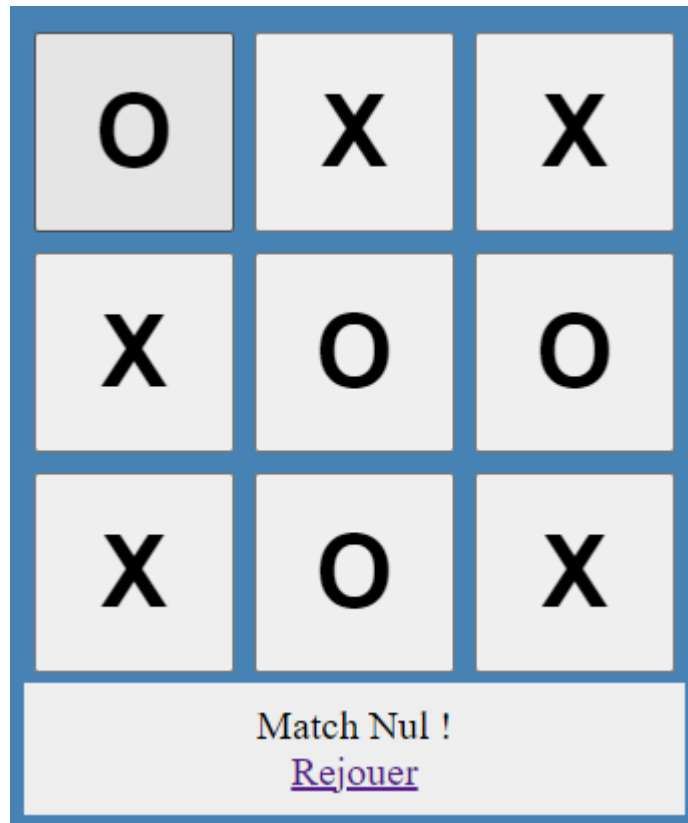
Case occupée !
Joueur X c'est toujours à vous !

- Sinon (case libre) écrire le symbole du joueur dans la case qu'il a choisi.
Vérifier si le jeu est fini ou non
Si la partie est fini, modifier le statut du jeu pour prévenir le joueur qui a gagné et ajouter un lien permettant de rejouer (lien vers la page du jeu).



Si la partie est nul, modifier le statu du jeu pour prévenir que la partie et nul et proposer de rejouer.

PROJET - PARTIE JAVASCRIPT



Incrémenter le tour.

Modifier la valeur du tour pour avoir uniquement les valeurs 0 ou 1. (la valeur 0 représente le tour du joueur X et la valeur 1 représente le tour du joueur O)

Modifier le statut de la partie pour savoir quel joueur doit jouer.

