



W1 - JavaScript

W-JSC-501

Bootstrap

Puissance 4

Bootstrap

repository name: puissance4
language: JS, HTML, CSS



- The totality of your source files, except all useless files (binary, temp files, obj files,...), must be included in your delivery.

OBJECTIF

Vous devez développer un Puissance 4 sous la forme d'un plugin javascript.

Dans ce Bootstrap, nous allons nous intéresser au développement du plateau de jeu lui-même.

Rappel: Votre plateau de jeu doit faire 'x' cases de large et 'y' cases de hauteur.

LA PAGE HTML

CHOIX INITIAUX

La première étape va donc être de créer la page HTML qui contiendra notre jeu. Comme nous souhaitons que notre jeu soit complètement sous forme de plugin, nous allons créer un simple container, par exemple un `<div>` ou un `<canvas>` que notre plugin viendra modifier. Cela permettra en théorie à n'importe quel site d'utiliser notre plugin dès lors qu'un container y aura été placé.

DIV OU CANVAS ?

Le `<div>` est une division classique de page HTML dans laquelle on peut insérer d'autres balises, alors que le `<canvas>` est une sorte de "toile" sur laquelle on peut "peindre" avec du javascript. Pour des jeux vidéos, on utilise souvent le canvas ; mais dans notre cas le `<div>` et les tableaux HTML se prêtent plutôt bien à un

Puissance 4. Nous allons donc utiliser le **<div>**.



Tout votre Puissance 4 doit se situer dans le **<div>**. Votre plugin peuplera le **<div>** avec les autres éléments dont vous aurez besoin pour afficher les joueurs, etc.



Vous aurez besoin d'une feuille de style CSS pour votre jeu. Pensez à séparer le CSS de votre jeu et le CSS du site, même si ce dernier est minimal !

LE PLUGIN

Tout votre projet doit tenir dans un plugin, c'est à dire un objet javascript (voir <https://developer.mozilla.org/fr/docs/Web/JavaScript>). Comme le jeu s'affichera dans le **<div>**, le plus simple est d'appeler votre plugin sur le **<div>**.

LE PLATEAU

AFFICHER UN TABLEAU

La principale chose que doit faire votre plugin, c'est d'afficher le plateau de jeu du Puissance 4. Pour modéliser une grille de Puissance 4, le plus simple est d'utiliser un tableau stylisé, c'est à dire un élément **<table>**. Comme ses dimensions peuvent être variables selon les options, vous devrez le générer dynamiquement dans votre fonction.



Vous pouvez créer dynamiquement des éléments en utilisant la syntaxe suivante :

```
monparagraphe = document.createElement('p');
```

STYLE

Une fois que votre tableau est créé, vous pouvez le styliser dans votre feuille CSS pour obtenir l'apparence désirée. Traditionnellement, la grille de Puissance 4 est bleue et se remplit de jetons ronds, par défaut jaunes et rouges (mais la couleur doit aussi être une option).



Vous pouvez créer des éléments carrés au background bleu, incluant des éléments ronds au background blanc, rouge ou jaune. Les éléments ronds peuvent être faits avec la propriété CSS "border-radius".

ET ENSUITE ?

Maintenant que vous avez votre plateau de jeu et les options de changement de couleur, vous pouvez commencer à coder les mécaniques de jeu (par exemple le changement d'une couleur à une autre quand un joueur clique sur une colonne). Mais pour pouvoir coder les mécaniques, il faut que vous stockiez quelque part les "états" du système : "cette case est rouge", "cette case est jaune", etc.



Vous trouverez une liste d'informations importantes dans le document de présentation du Projet.

Pour faire ça bien, vous aurez besoin de la POO (Programmation Orientée Objet). Commencez par vous renseigner sur la POO, et particulièrement la POO en JS. Imaginez par exemple que vos cases sont des objets qui ont des attributs (couleur, position, ...) et des méthodes (colorer, ...). Il en va de même pour vos colonnes, pour le plateau de jeu, ...

Ce n'est pas grave si vous n'y arrivez pas du premier coup : un développeur professionnel expérimenté saura planifier en avance la structure du système de données, les objets et leurs attributs en fonction des besoins qu'il prévoit. Les premières fois, vos systèmes et structures de données seront sans doute peu cohérents ou optimaux ; vous devrez peut-être les refaire complètement en cours de route quand vous réaliserez qu'ils ne sont pas adaptés. Ce n'est pas grave, c'est un processus nécessaire à l'apprentissage. Lancez-vous et commencez à coder ! Vous améliorerez votre code en cours de route.

Bon courage et amusez-vous bien !