



Coder un 2048 en JavaScript

Niveau Débutant • 13-16 ans • 1h30 • version 2

1. Présentation du jeu

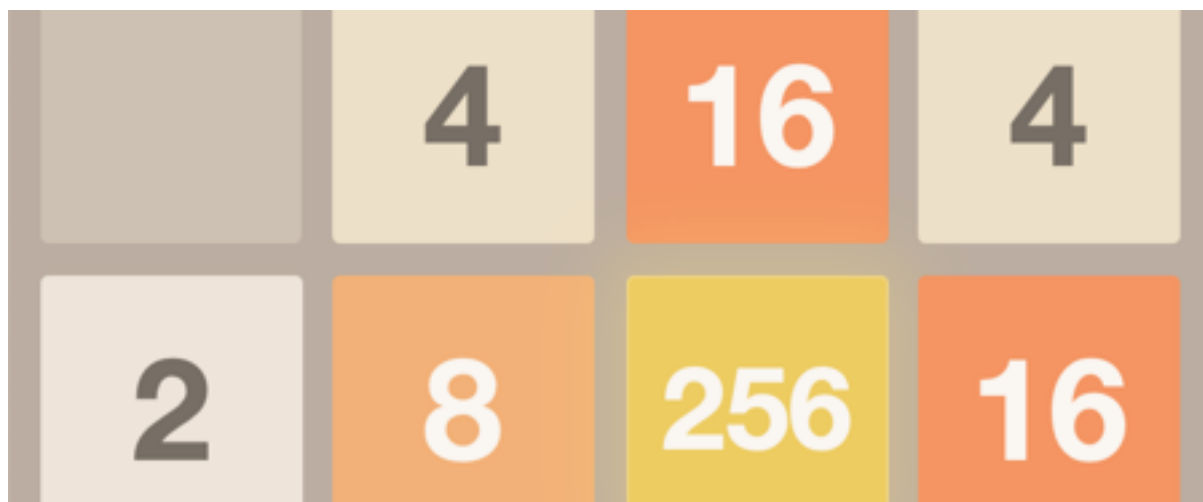


2048

2048 est un jeu vidéo de type Puzzle datant de 2014. Le but du jeu est de faire glisser des tuiles sur une grille, pour combiner les tuiles de mêmes valeurs et créer ainsi une tuile portant le nombre 2048.

Certains éléments du code de **2048** ont été perdus... À vous de compléter le code du jeu pour qu'il fonctionne de nouveau !

Dans ce sujet, vous apprendrez à programmer des boucles, des conditions en **JavaScript** et le fonctionnement global d'un jeu web.



2. Outils & Ressources

2.1 Ressources

Pour commencer, rendez-vous sur <https://repl.it/@EmmaEpitech/2048JS>.

L'écran est découpé en plusieurs parties : la partie fichier, à gauche : c'est ici que vous coderez le jeu. Et la partie rendu, à droite : c'est ici que vous verrez votre progression et pourrez tester votre jeu.

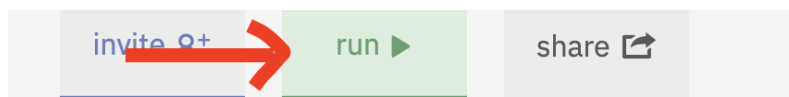
2.2 Prise en main de la plateforme

Sur la partie gauche de l'écran, tu peux voir une liste de fichiers.

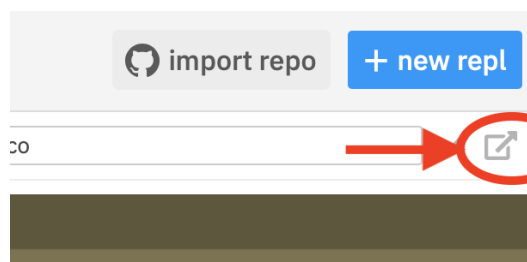
Les fichiers HTML, les images, et les fichiers **JavaScript**.

Aujourd'hui, tu n'auras besoin de modifier que le fichier "**game.js**". Tu peux toutefois aller voir ce qu'il se passe dans les autres fichiers si tu le souhaites.

Pour lancer ton code appuie sur la flèche verte "**RUN**" en haut de l'écran.



Si ton écran est trop petit pour afficher le jeu en entier, tu peux l'ouvrir dans un nouvel onglet en cliquant sur le bouton suivant :



2.3 Game.js

Plusieurs fonctions sont présentes dans le fichier **game.js**. Ton rôle va être de les compléter.

```
function prepareGame(game)
```

i La fonction **prepareGame** est appelée une seule fois au début du jeu, lors de l'initialisation.

```
function gameLoop(game, tile1, tile2, canMerge, position)
```

i La fonction **gameLoop** est appelée à chaque tour de jeu, lorsque le joueur appuie sur une des flèches du clavier. Il s'agit de la **boucle de jeu**. Dans cette fonction **tile1** correspond à une tuile à déplacer. **tile2** n'aura une valeur que lorsque les deux tuiles vont fusionner. Elle sera alors égale à la deuxième tuile en jeu dans la fusion.

3. Découverte du jeu

3.1 Préparer le jeu

2048

Si tu lances le jeu, tu peux t'apercevoir qu'il n'y aucune tuile sur la grille, et que tu ne peux pas la déplacer !

La première étape va donc être **d'ajouter les tuiles** au démarrage et **d'initialiser le jeu**.

Pour cela, nous allons ajouter les deux instructions suivantes au début de la fonction **prepareGame** :

```
game.startTiles = 2;  
game.setup();
```

i Teste ton code en appuyant sur **"RUN"** !

i Tu as sans doute remarqué que les tuiles ont toutes une valeur de "-1" et qu'elles ne peuvent pas bouger sur la grille. Nous allons coder cela dans les prochaines étapes.

3.2 Déplacer les tuiles



Maintenant que nous avons généré aléatoirement nos tuiles, nous allons faire en sorte qu'elles se déplacent lorsque l'on appuie sur l'une des flèches du clavier.

Dans la fonction **gameLoop**, ajoute la ligne suivante :

```
game.moveTile(tile1, position.farthest);
```



Lance le jeu pour tester ton code !

3.3 Combiner les tuiles

2048

La prochaine étape va être de **combiner les tuiles** lorsque deux tuiles ayant **une valeur égale** se trouvent côte à côte. Pour cela, nous allons devoir **comparer** leur valeur.

Au **début** de la fonction **gameLoop**, ajoute une nouvelle condition pour comparer la valeur des tuiles et vérifier qu'on peut les combiner :

```
if (canMerge && tile1.value == tile2.value) {  
  }  
else {  
  }  
}
```

Tu te souviens de la ligne qu'on a ajouté à l'étape **3.3 Déplacer les tuiles** ?

Celle-ci ne doit être appelée **que** si on ne combine pas les tuiles.

Déplace la ligne de code entre les accolades du **“else”** de la condition précédente :

```
else {  
  game.moveTile(tile1, position.farthest);  
}
```



À présent, si une combinaison est possible, la tuile ne se déplacera plus.

Retour dans la condition “if” :

Nous allons créer une nouvelle tuile, qui viendra remplacer les deux tuiles à combiner. La valeur de la nouvelle tuile doit être **2 fois plus grand** que celle des tuiles d’avant.

Ajoute les deux lignes suivantes à l’intérieur de la condition **if** pour créer une tuile :

```
var tile = new Tile(position.next, 2 * tile1.value);
tile.mergedFrom = [tile1, tile2];
```

Puis ajoute la ligne suivante pour l’insérer dans la grille afin qu’elle s’affiche :

```
game.grid.insertTile(tile);
```



Lance à nouveau le jeu pour voir ce qu’il se passe !

Tu peux voir que la nouvelle tuile s’ajoute dans la grille, mais que l’une des anciennes tuiles est toujours là.

Nous allons à présent voir comment **supprimer les tuiles inutiles**.

À la suite de ton code, ajoute les deux lignes suivantes :

```
game.grid.removeTile(tile1);
tile1.updatePosition(position.next);
```

3.4 Ajouter les points



Dans le jeu **2048**, lorsque 2 tuiles se combinent, tu gagnes un certain nombre de points, correspondant à la valeur de la nouvelle tuile créée.

Par exemple, si tu combines deux **tuiles 8** ensemble, tu gagnes alors **16** points !


Ajoute la ligne suivante à ton code :

```
game.score += tile.value;
```

Dernière étape : détecter lorsque la partie est gagnée, c'est-à-dire, lorsque l'on a obtenu une tuile de valeur **2048**.

Pour cela, il suffit d'ajouter la condition suivante à la suite de ton code :

```
if (tile.value == 2048) {  
  game.won = true;  
}
```

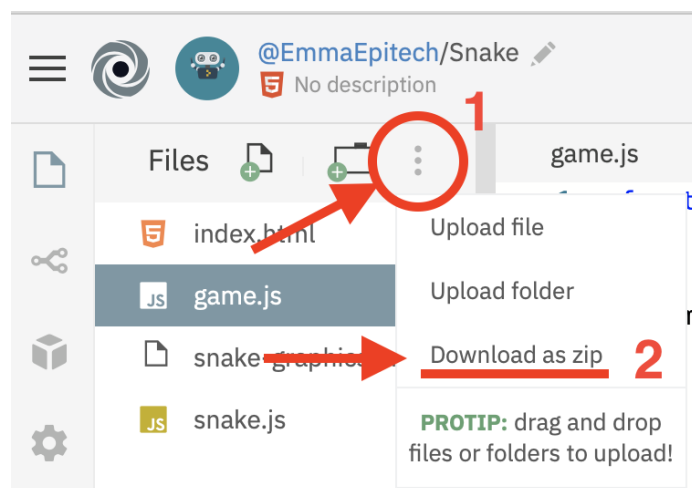
 *Appuie sur **RUN** et essaye de gagner un maximum de points !*

Bonne chance à toi !

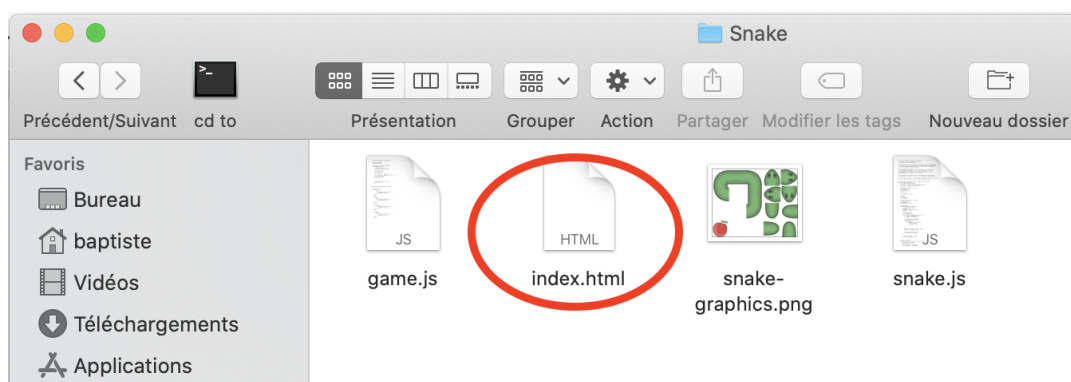
5. Sauvegarde ton jeu



Tu peux enregistrer ton jeu sur ton ordinateur pour le modifier ou y jouer plus tard. Pour cela, clique sur “**Download as zip**”.



Pour lancer le jeu sur ton ordinateur, ouvre l'archive zip, puis double-clique sur le fichier **index.html**.



6. Quelques liens utiles

Pour apprendre le JavaScript :

→ <https://www.w3schools.com/js>

Pour refaire l'exercice :

→ <https://repl.it/@EmmaEpitech/2048JS>

Pour voir nos autres exercices :

→ <https://repl.it/@EmmaEpitech>

Pour plus d'informations sur nos activités :

→ <https://www.e-mma.org>