**Support de cours - Snake JavaScript**

Nima Zarrabi

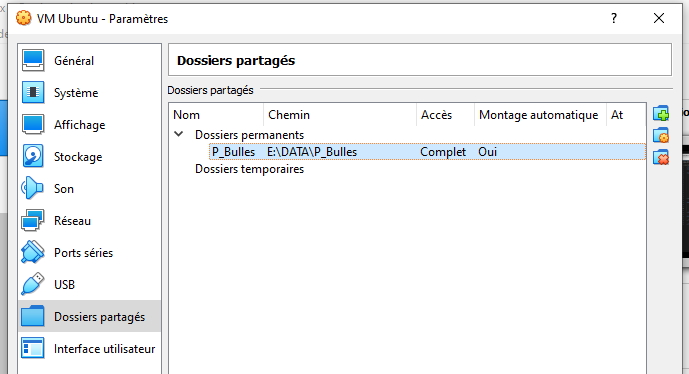
Importation de la VM Ubuntu depuis une clef USB (VM déjà un peu configuré)

Lancer la VM..

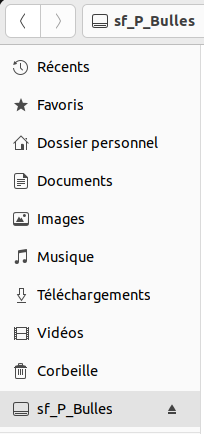
# Création d’un dossier partager :

Ouvrir l’interface Oracle VirtualBox

Bouton  sur le côté



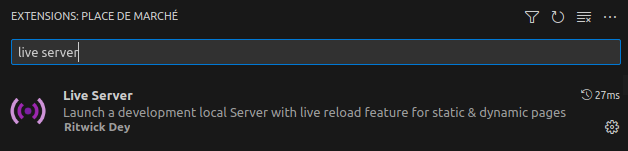
Dossier visible dans l’explorateur de fichier Ubuntu, en tant que Disque



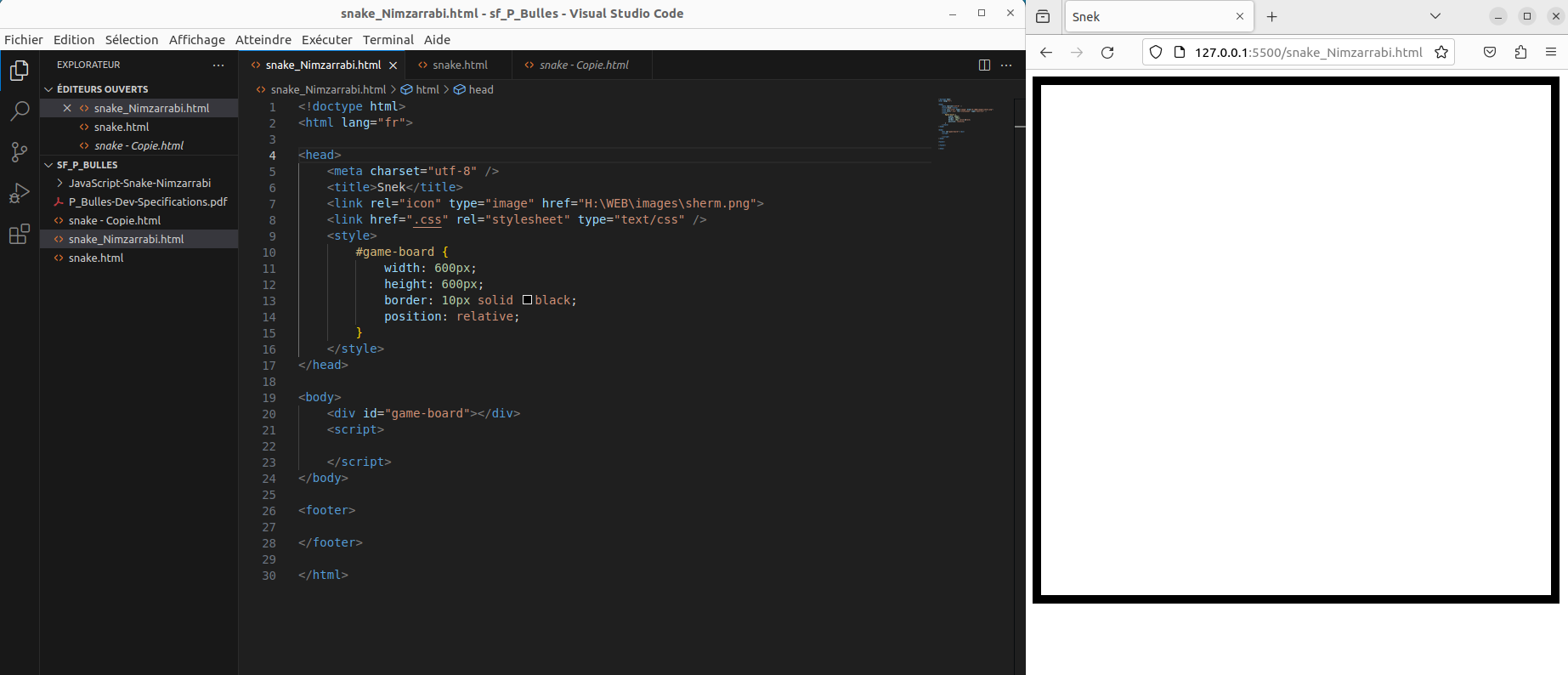
# Code Snake

Visual Studio Code est déjà présent sur la VM.

Installer l’Extension « Live Server » (pas nécessaire mais pratique)



La page se met à jour toute seul à chaque ctr+S sur le programme.



Le carrée vert apparait finalement

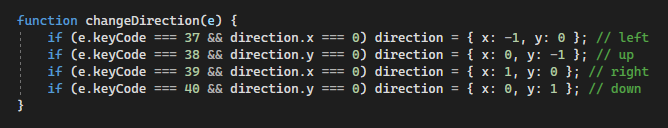


Le snake apparait au millieu du plateau

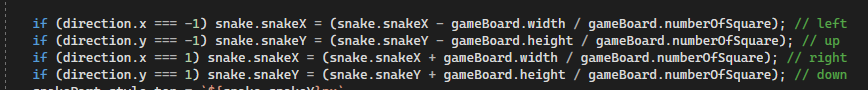


La pomme aussi

Changement de direction



Dépanadent de la position, le snake avance vers le haut, bas, droite ou gauche

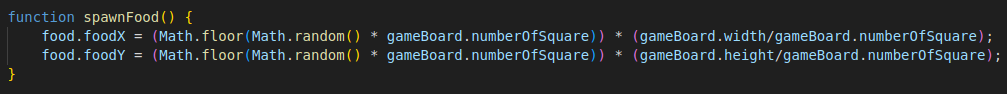


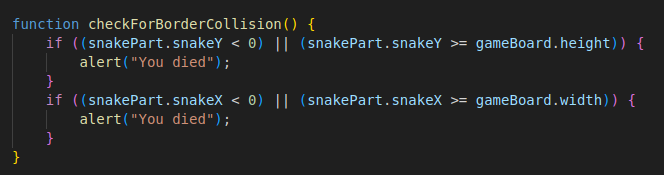
Ajout de la function « spawnFood »

La position de la pomme est random entre 0 et le nombre de carrées dans le plateau (20)

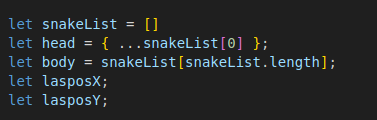
Puis, ce chiffre est multiplié par le nombre de pixels par carrée du plateau (longueur / nombre de carrées, et hauteur / nombre de carrées)

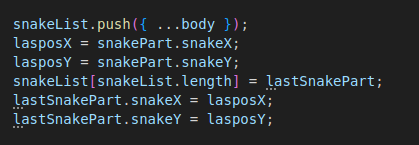
Cela donne une position x et y en pixels, cette positon ne peut qu’être des multiples de la taille d’un carrée (30px dans ce cas, car le plateau fait 600px en X et Y, et il comporte 20 cases. 600/20 = 30)



Fonction pour faire le snake manger des pommes

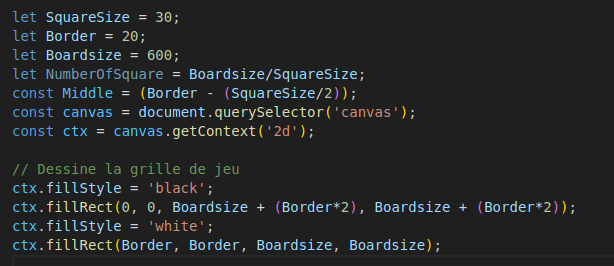
J’ai essayé de faire le snake s’agrandir, mais je n’ai pas encore réussi



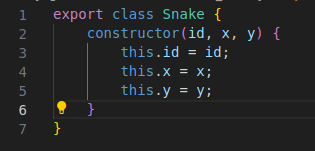


J’ai décidé de tout changer afin d’implémenter des classes et d’utiliser le « ctx » mis à disposition, j’ai tout de même garder certaines fonctions.

Création de la grille de jeu

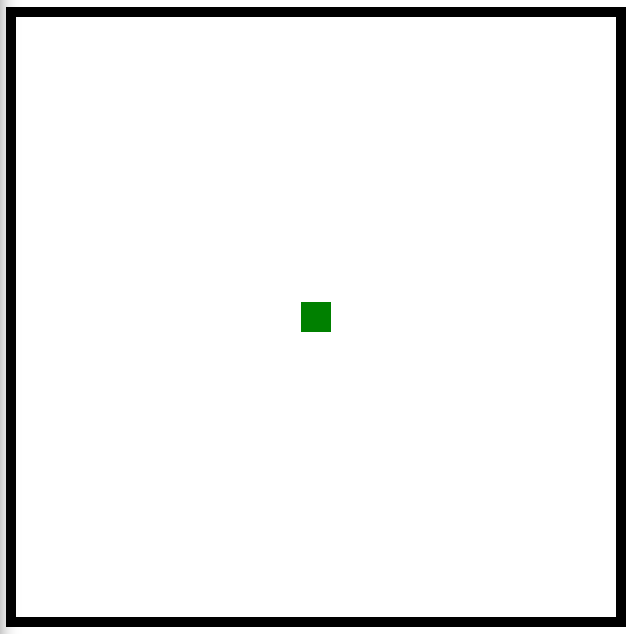


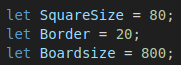
Class « Snake »

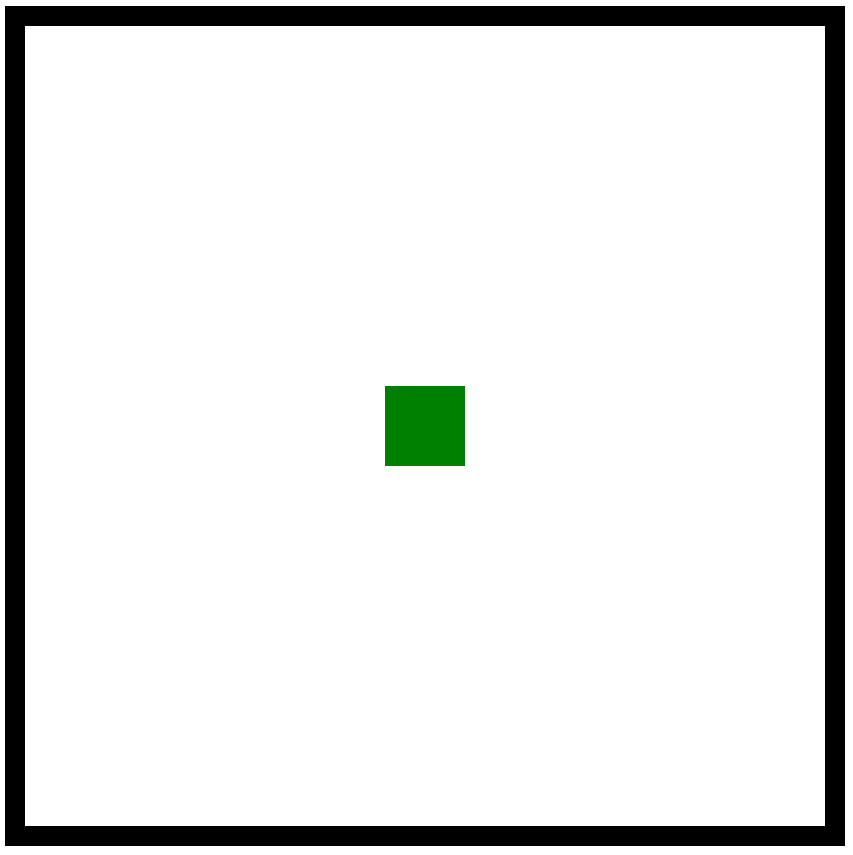


L’utilisation de variables « SquareSize », « Border », « BoardSize » et « NumberOfSquare » permet au tableau d’être complétement variable, si une des valeurs change, le reste devra pouvoir s’adapter, cela vas être utile pour crée de diffèrent modes de jeu.

Plateau avec ces valeurs : 

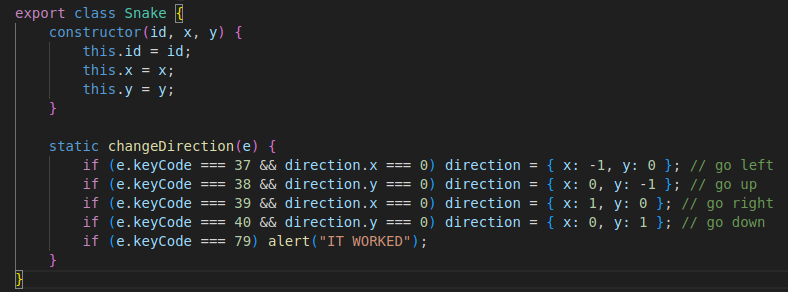
 600/30 = 20x20 Cases.

Plateau avec ces valeurs : 

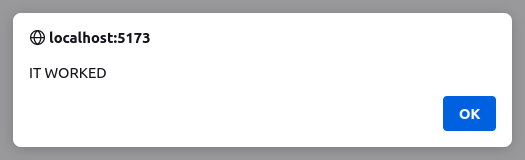
 800 / 80 = 10x10 Cases

Méthode dans la classe « Snake » qui permet de changer la direction du Snake avec les touches de flèches (le dernier « if » sert juste de test).

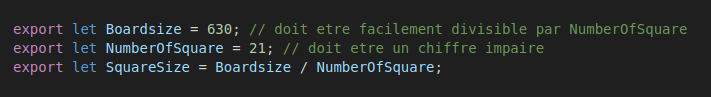




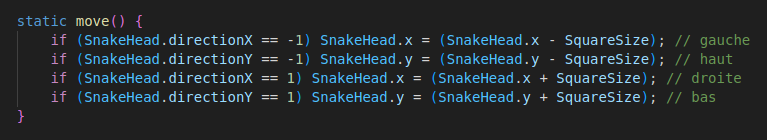
Message quand la touche 79 (« o ») est pressée.

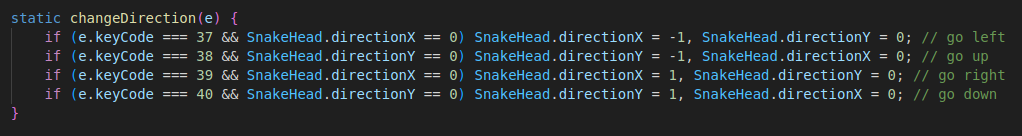


J’ai rajouter une classe « GameEngine ».

Changement du système du plateau, il est à présent encore plus pratiquement changable. 

Les variables sont exportées car elles sont utilisées dans « Snake », avec la méthode « Move »



Le déplacement fonctionne aussi 



Le Snake avance sans laisser de traînée.

Beaucoup d’export/import

Sources :

Documentation JavaScript donnée :

<https://fr.javascript.info/>

W3Schools : chapitre JavaScript (source principale d’information)

<https://www.w3schools.com/js/default.asp>

StackOverflow :

<https://stackoverflow.com/questions/66800735/creating-populating-and-appending-a-div>

<https://stackoverflow.com/questions/5597060/detecting-arrow-key-presses-in-javascript>

Aide de CHAT PGT :

Autre sites :

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Logical_OR>