

**Atelier kids-lab.io**  
**Créé par Yann Caron**



**kids-lab.io**

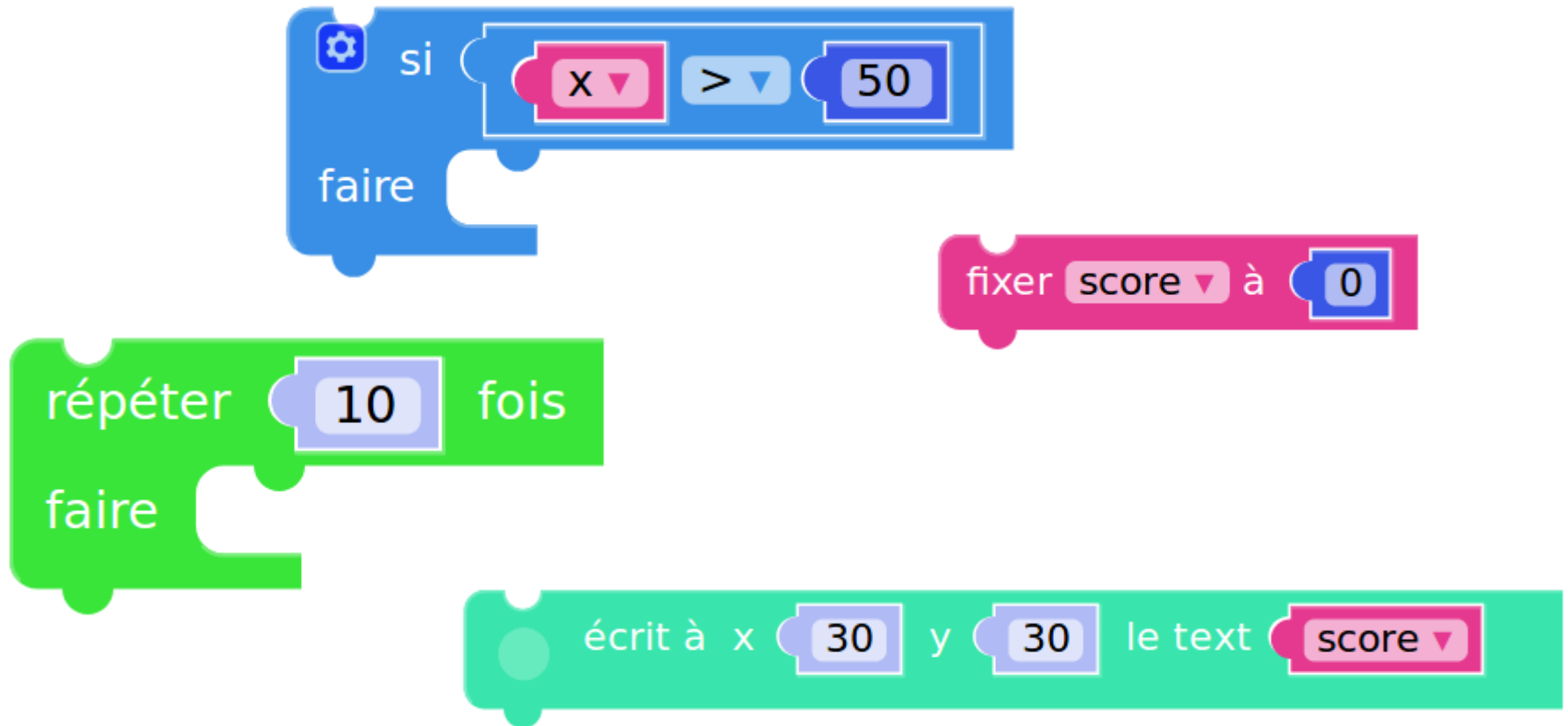
# Rappel :: Un programme

- Suite d'instructions :: ce que le programme doit faire
- Structure :: comment le programme est organisé

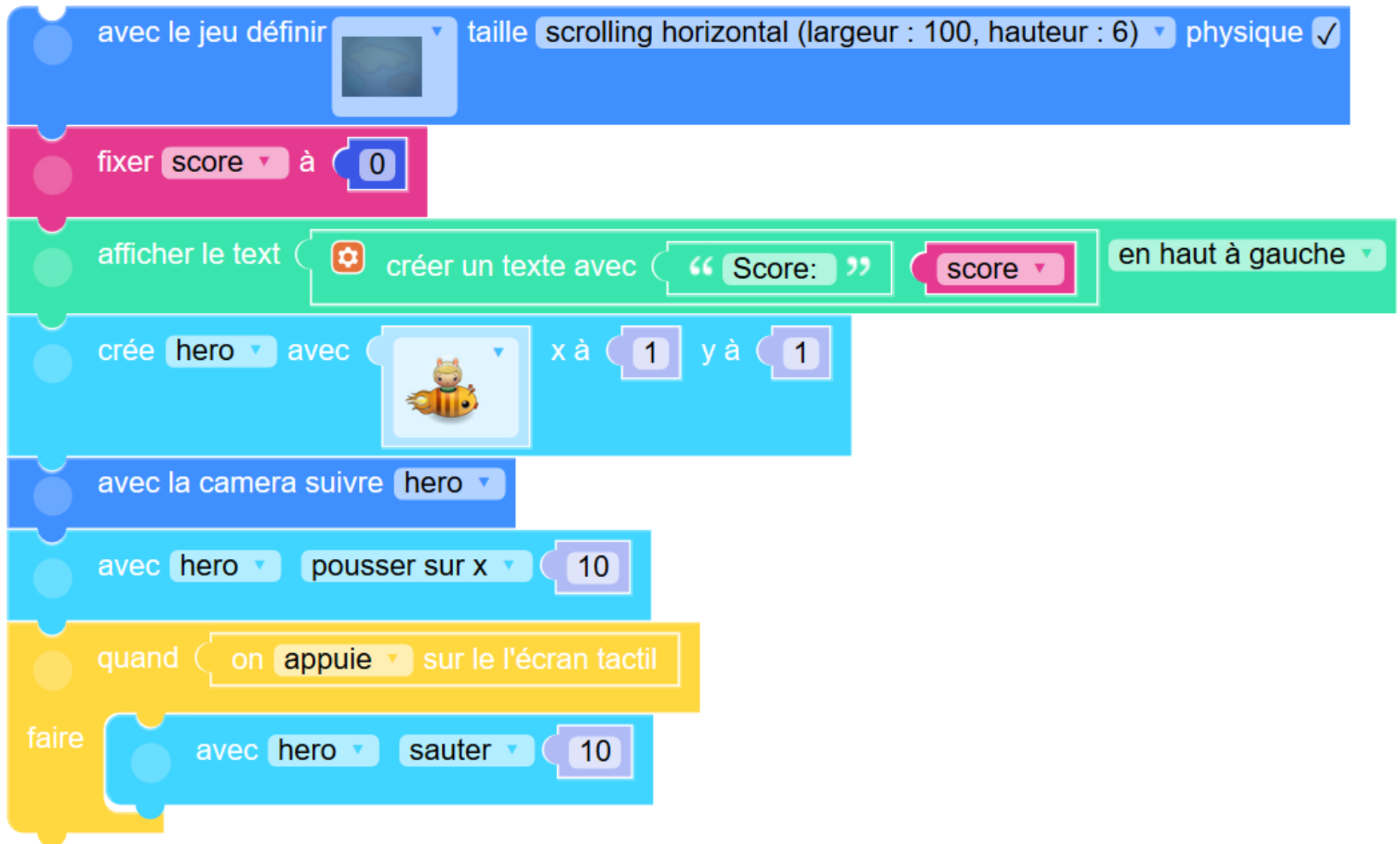


# Les blocs

- Un bloque :: une instruction :: un ordre

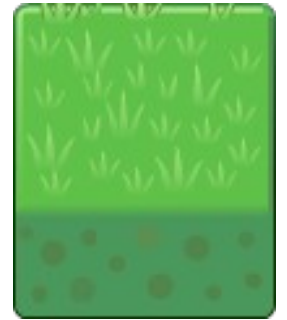


# Un programme avec des blocs

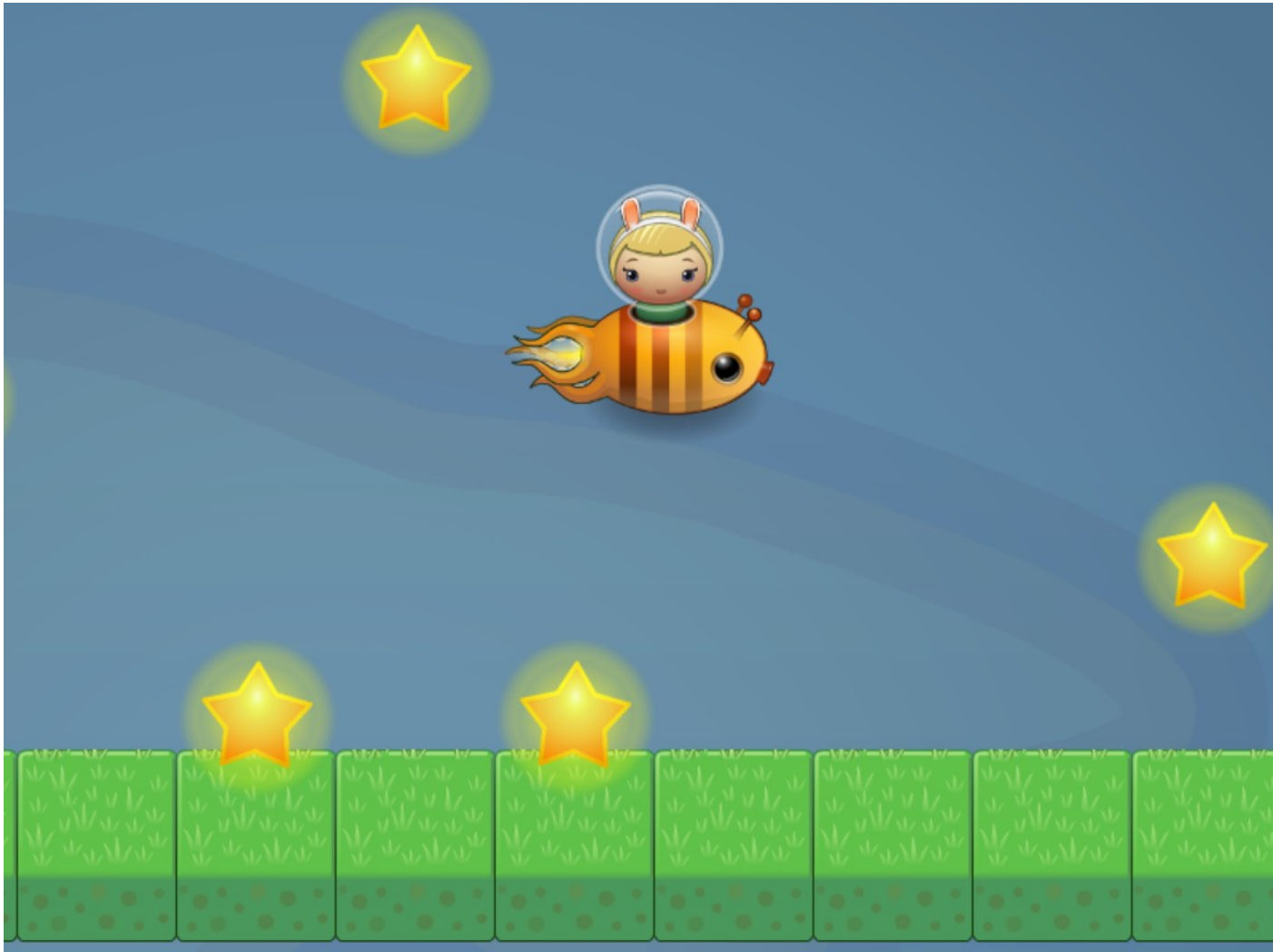


# Un moteur de jeu ?

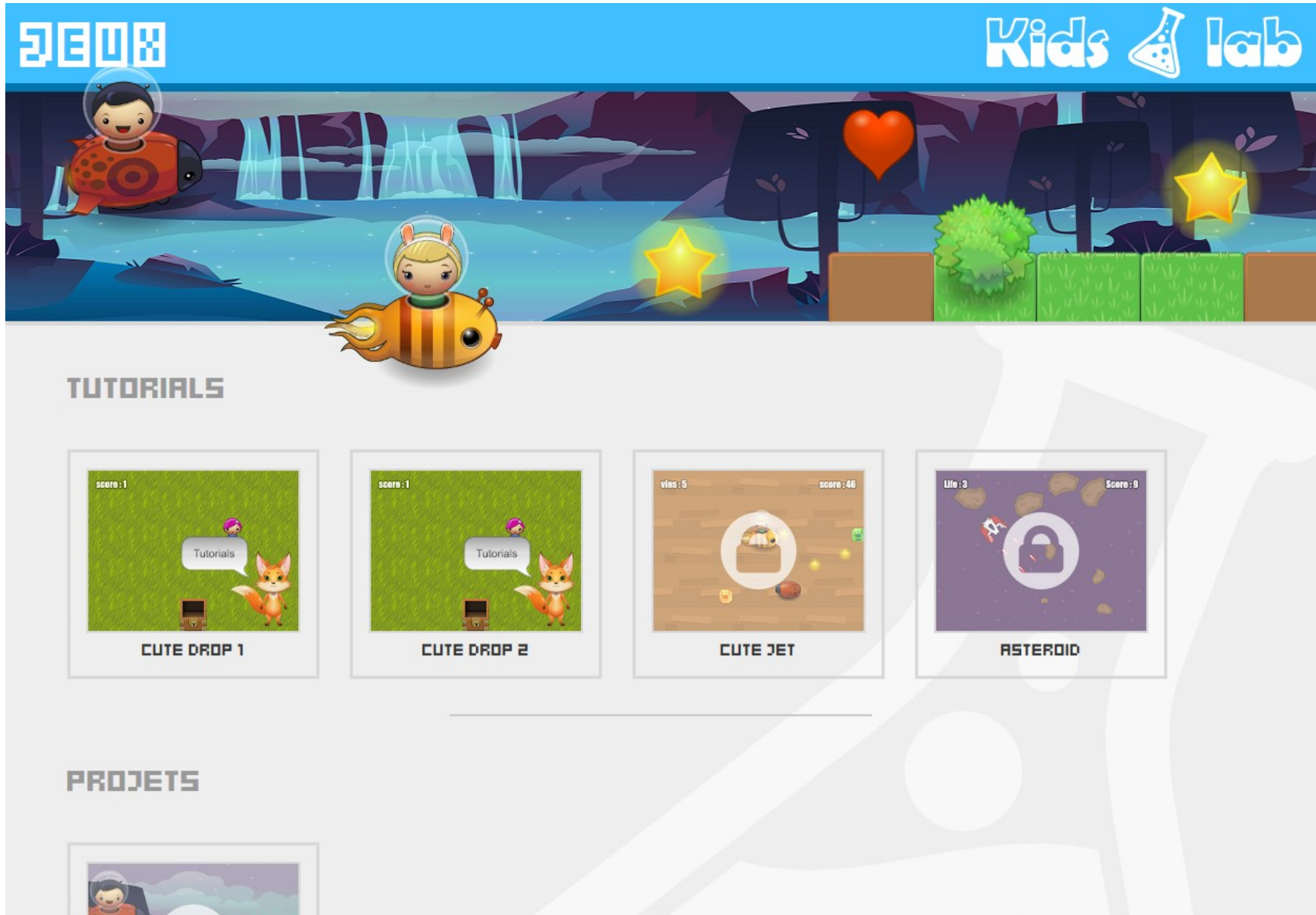
- Gère des acteurs ::
  - Propriétés (ex taille)
  - Méthodes (ex sauter, pousser)
  - Événements (ex quand touche)
- Gère la physique
- Gère les collisions



# Notre jeu :: cute-jet



# Notre logiciel :: kids-lab.io



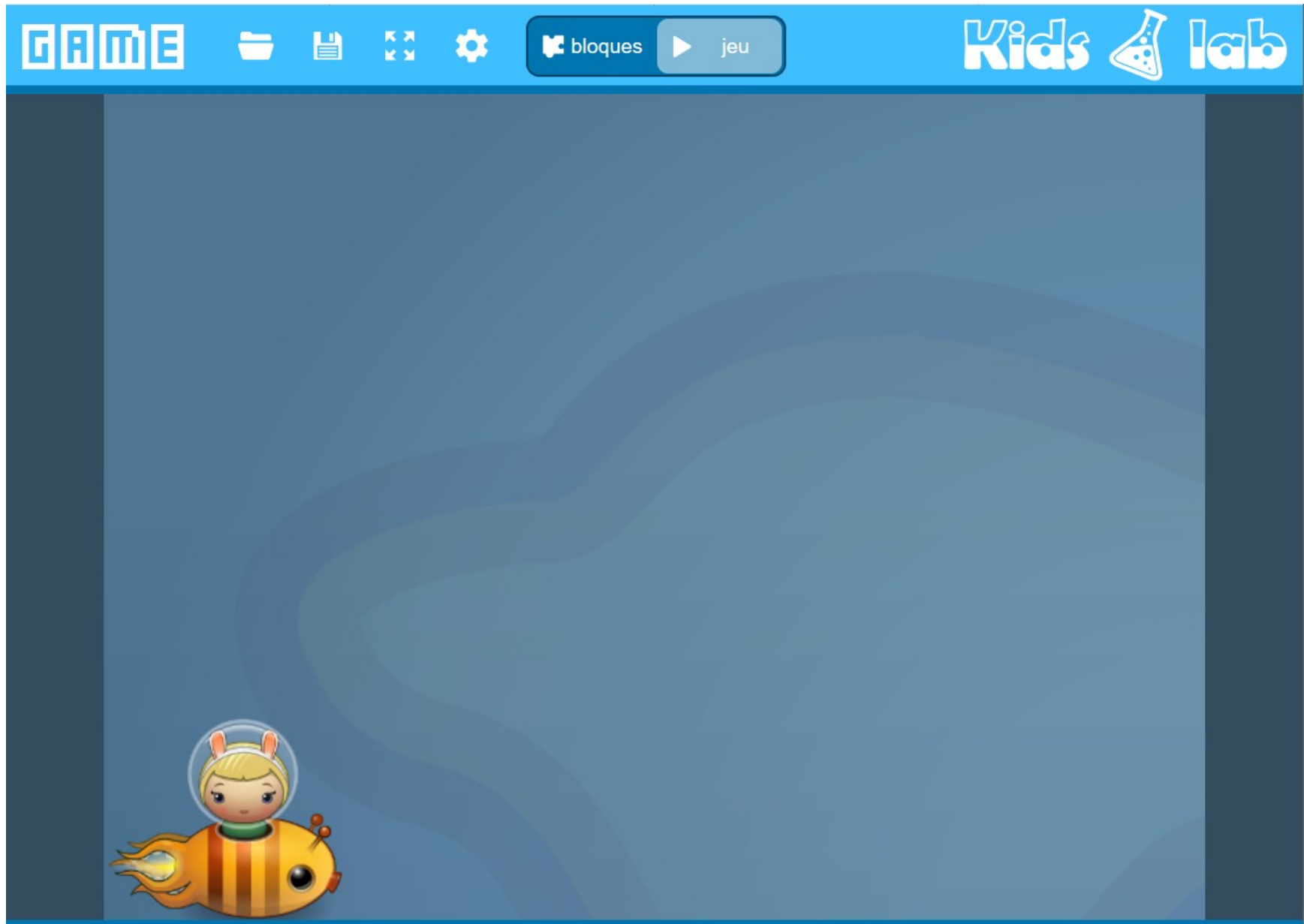
# Notre logiciel :: projet cute-jet

The screenshot displays the Kids Lab software interface. At the top, a blue header bar contains the 'GAME' logo, a folder icon, a document icon, a window icon, a gear icon, and buttons for 'bloques' and 'jeu'. The 'Kids lab' logo is on the right. A vertical sidebar on the left lists categories: Jeu, Acteur, Évènement, Animation, Sons, Images, Logique, Boucles, Math, Texte, and Variables. The main workspace shows a sequence of three code blocks: 1. 'avec le jeu définir' (with the game define) block for 'taille' (size) set to 'scrolling horizontal (largeur : 100, hauteur : 6)' and 'physique' (physics) checked. 2. 'crée' (create) block for 'hero' with a cat icon, 'x à 1' (x at 1), and 'y à 1' (y at 1). 3. 'avec la camera suivre' (with the camera follow) block for 'hero'. The background features a faint pattern of beakers. On the right side of the workspace, there are zoom controls (+, -, and a target icon) and a trash can icon.



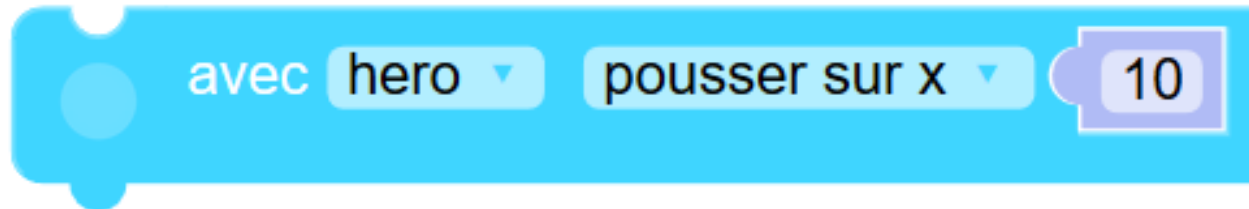
**A nous de jouer !  
Le game play :-)**

# Lancer le jeu

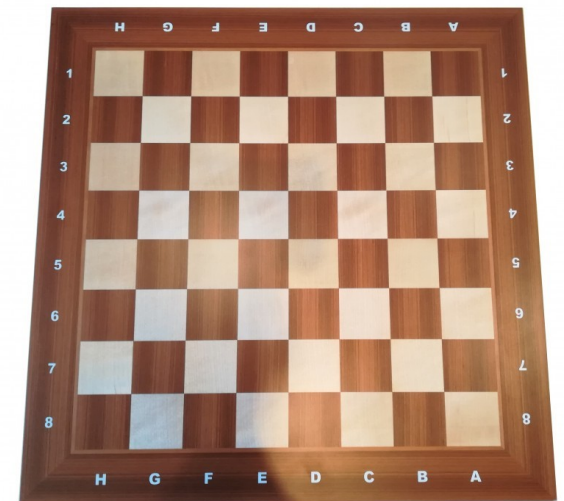
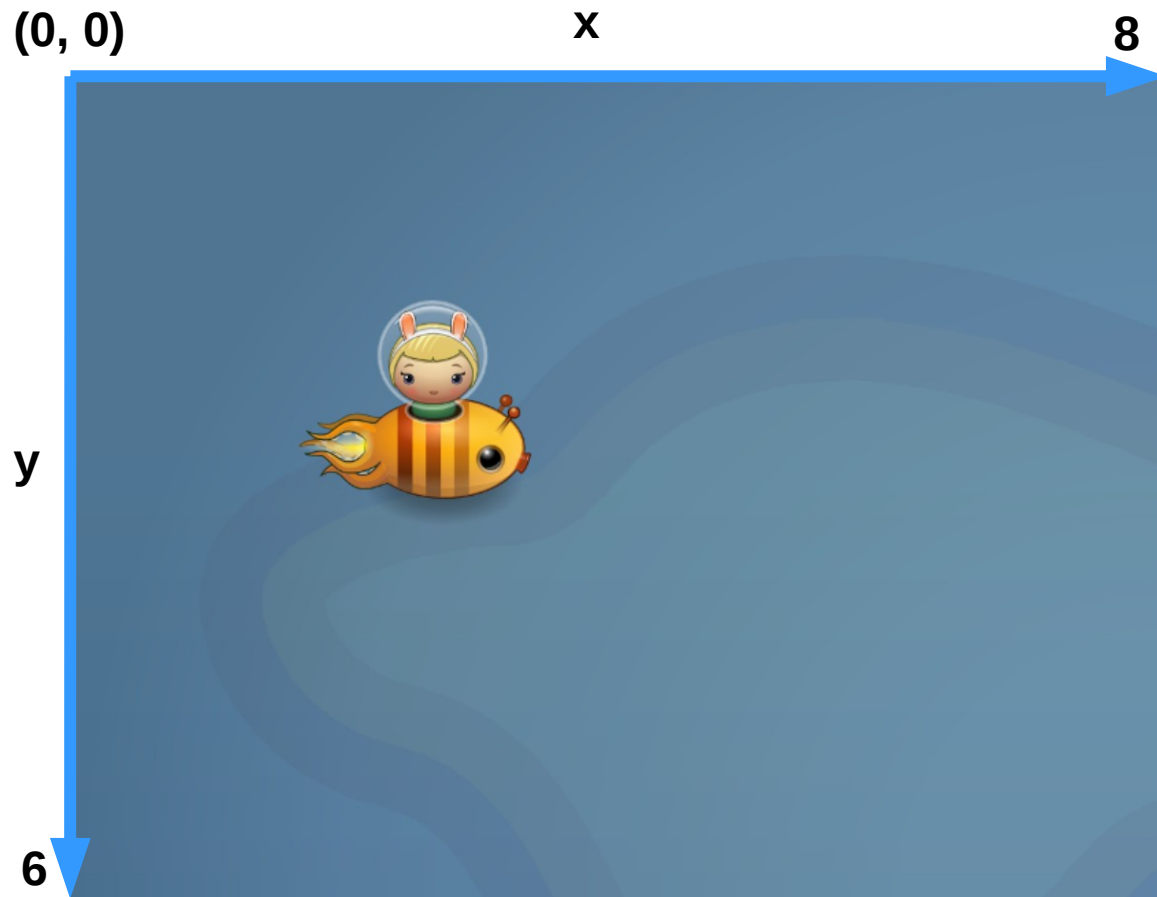


# Gameplay

- Il ne se passe rien !
- Il faut faire avancer le héros !

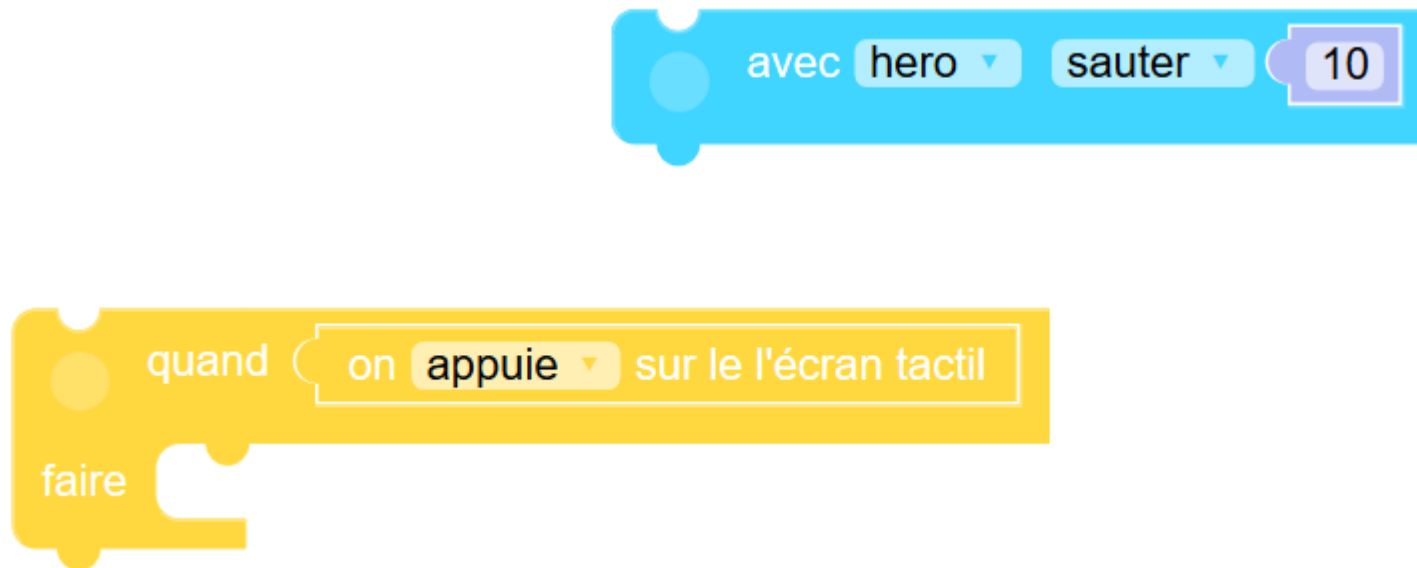


# Pourquoi X ?

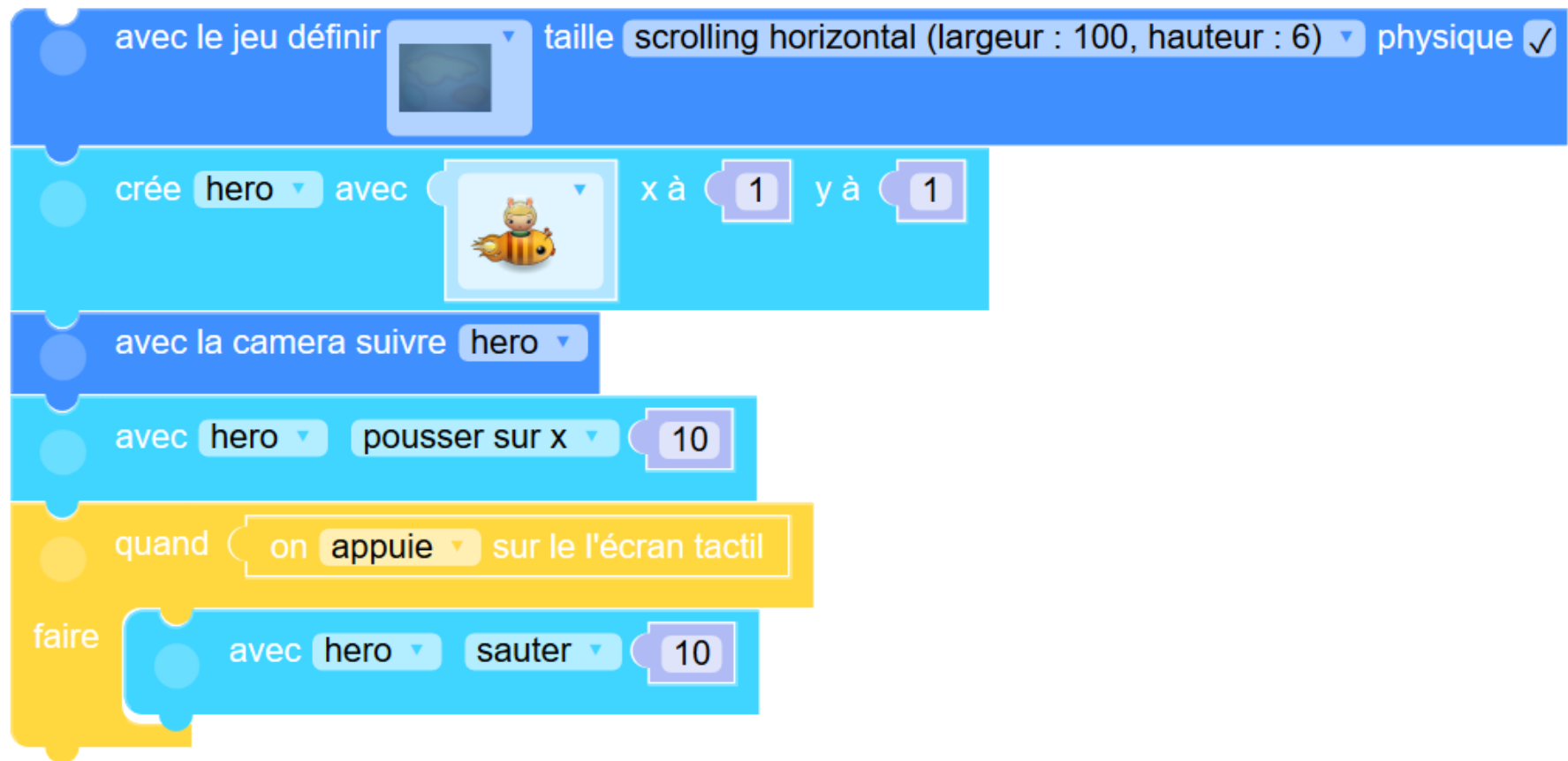


# Événement

- Quand on touche l'écran, le héros saute



# Résultat



**Bravo !**  
**Le décor :-)**

# But :: créer un décor

- Ajouter des blocs en bas du jeu





# Etape 1 :: créer l'acteur « bloque »

- Menu Acteur / Créer un nouvel acteur...

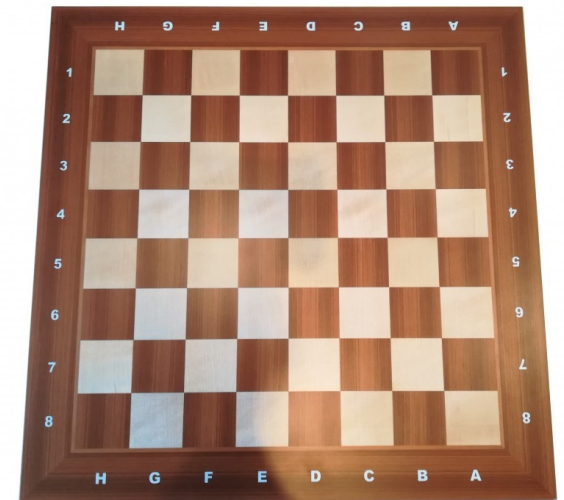


# Etape 2 :: ajouter l'acteur en bas

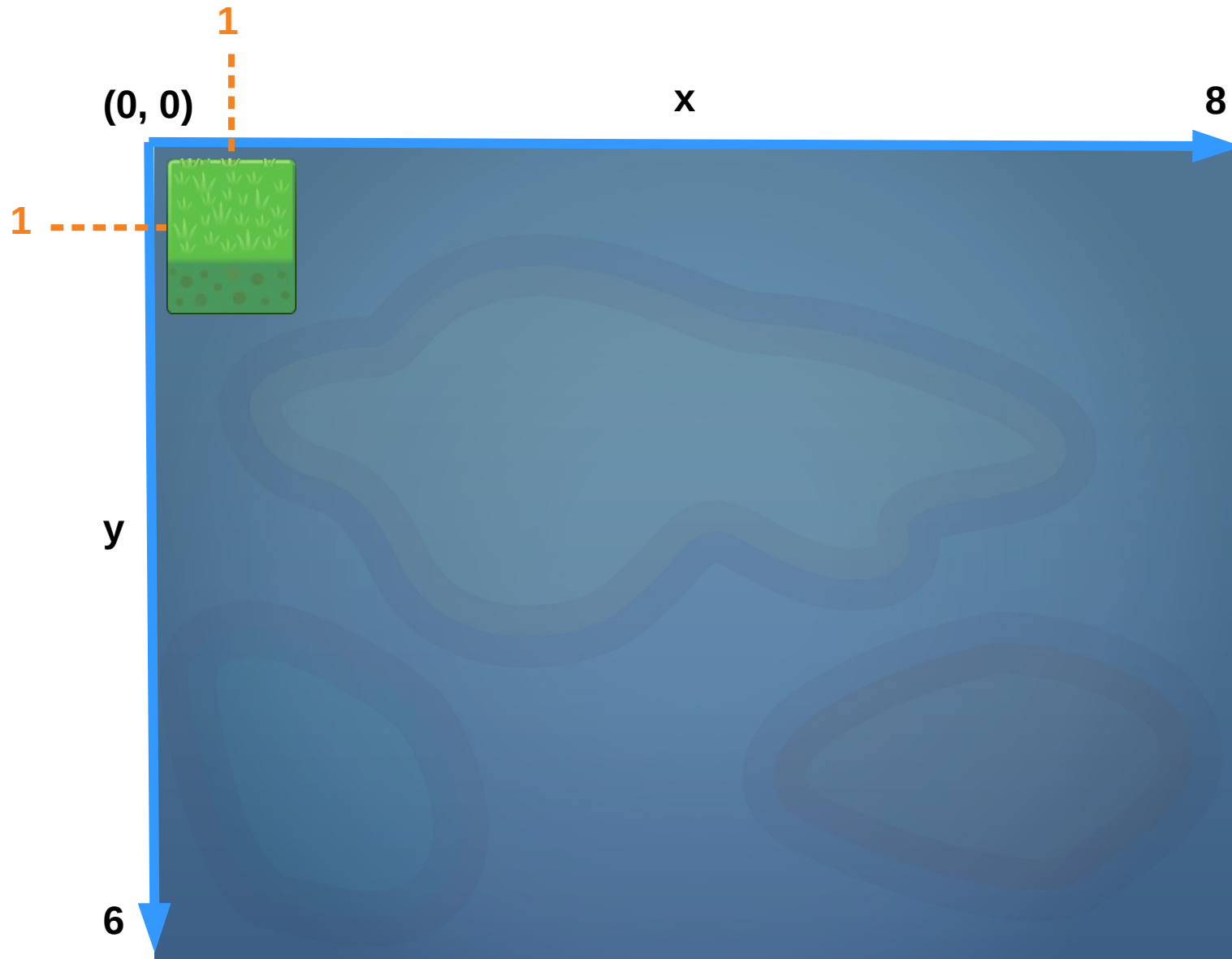
- Ajouter l'acteur dans le programme
- Question :: comment le placer en bas ?



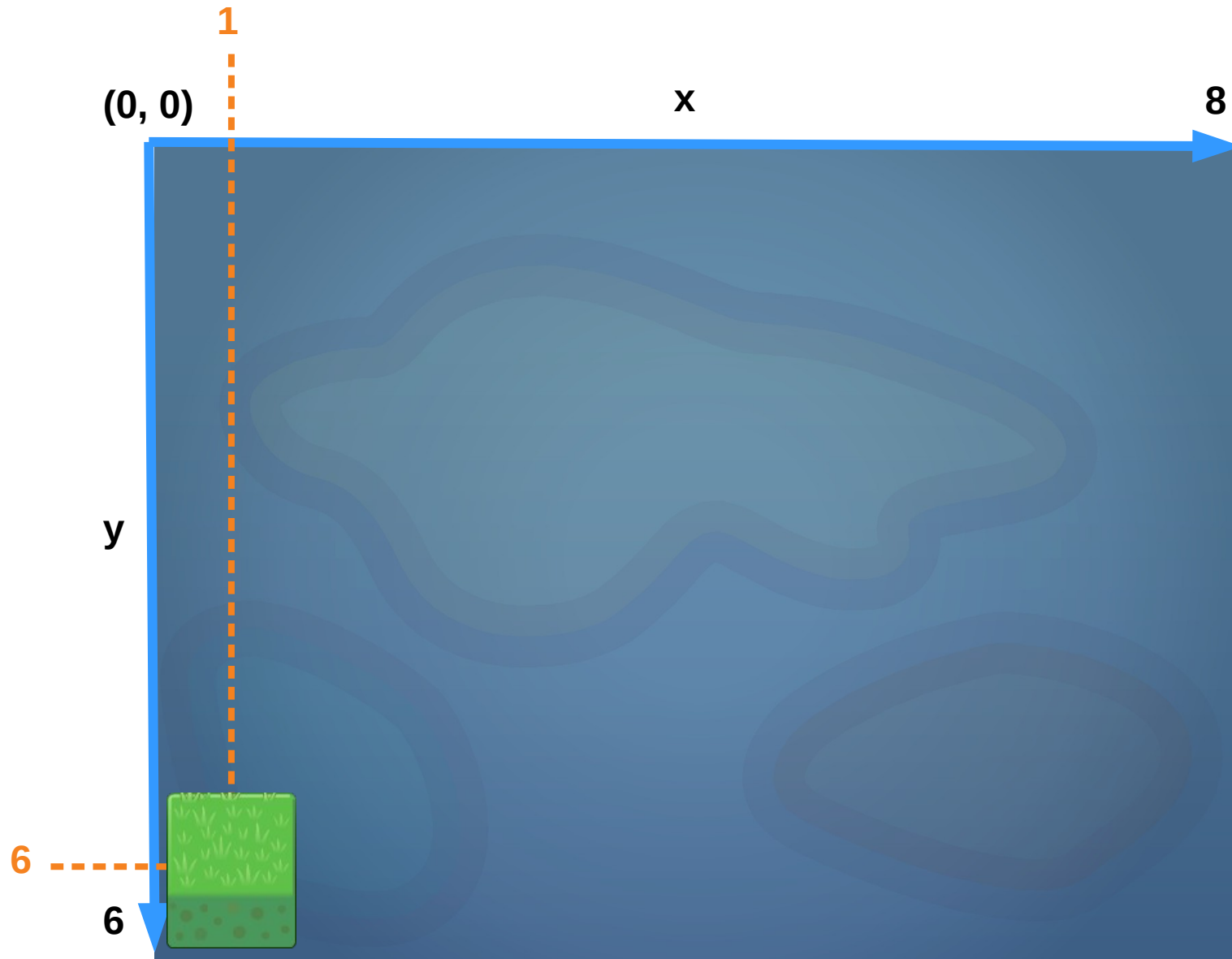
# En bas, Y ?



# Au départ (haut, gauche)



# En bas :: Y

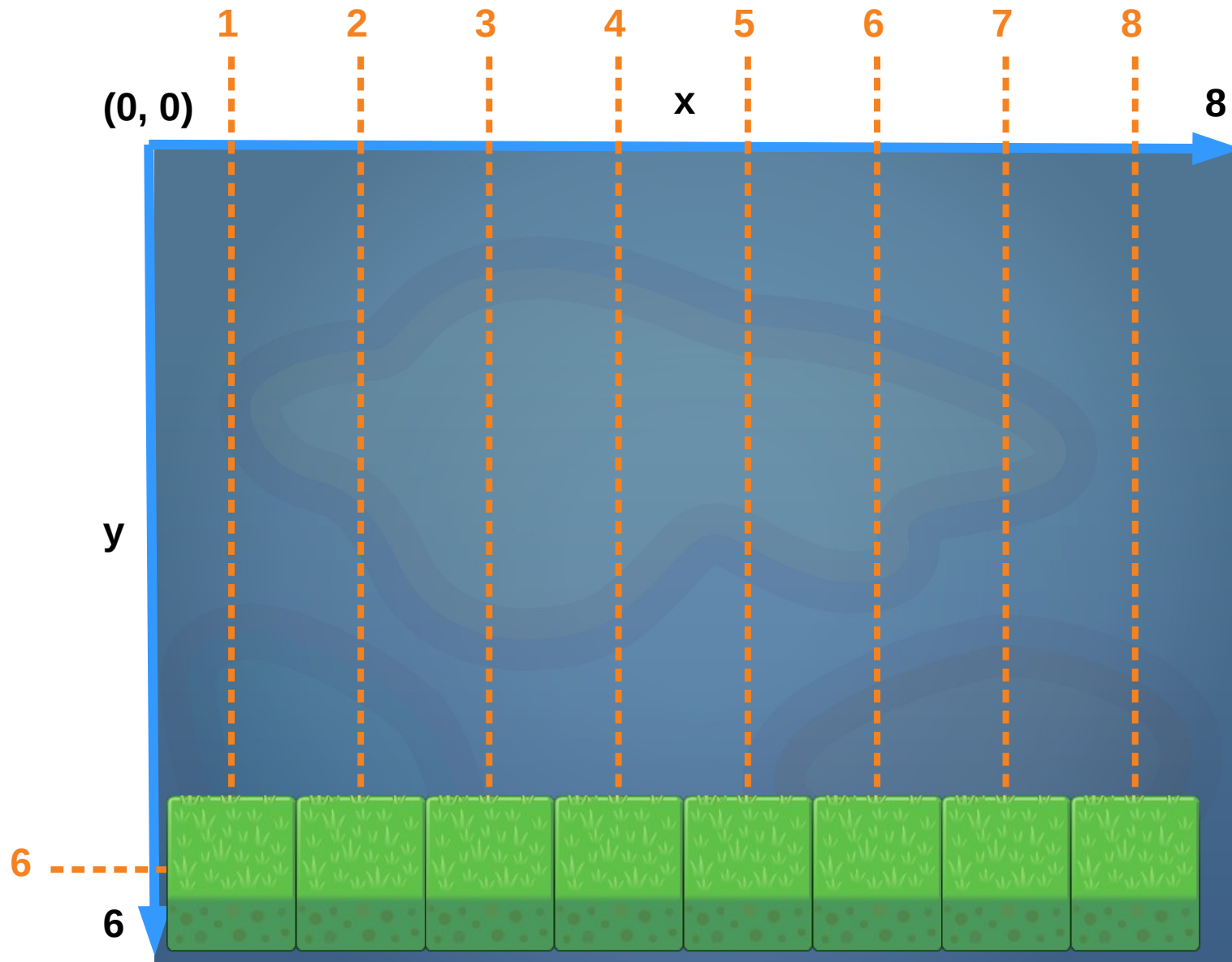


# Etape 3 :: créer une ligne



- Ajouter plusieurs acteurs
- Question :: comment les placer en ligne ?

# En ligne



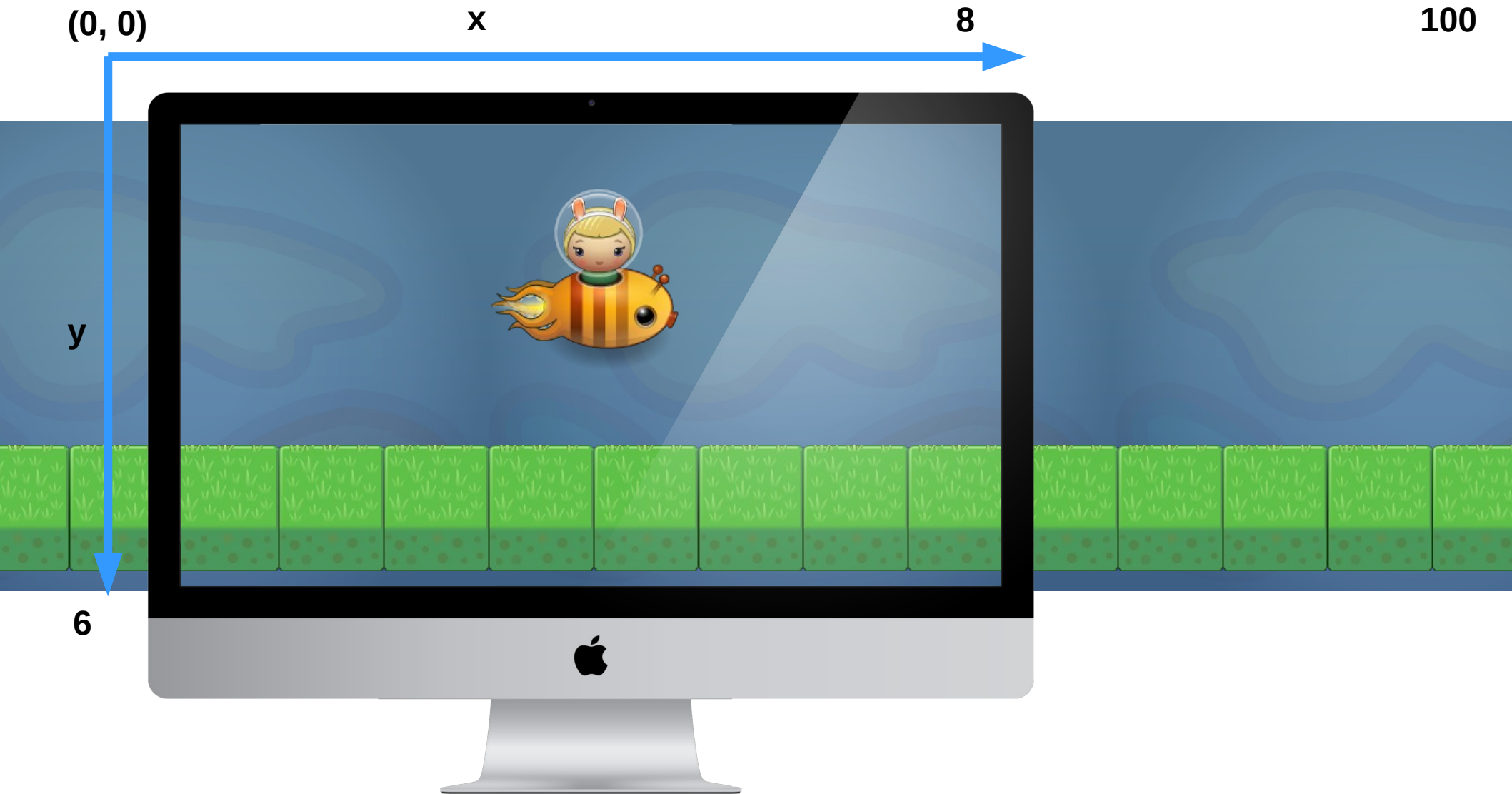
# Solution 1 :: distribuer

- $X = 1$ ,  $X = 2$ ,  $X = 3$



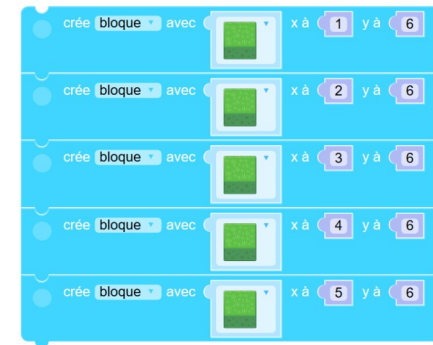
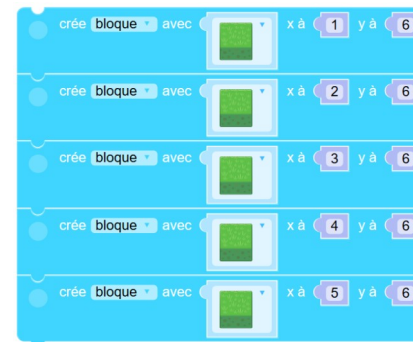
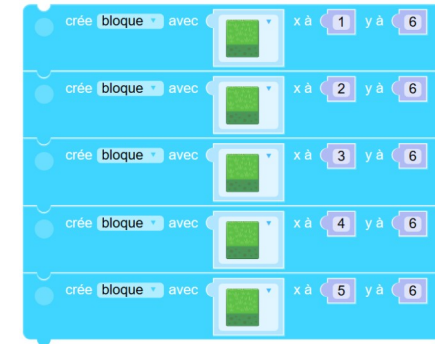
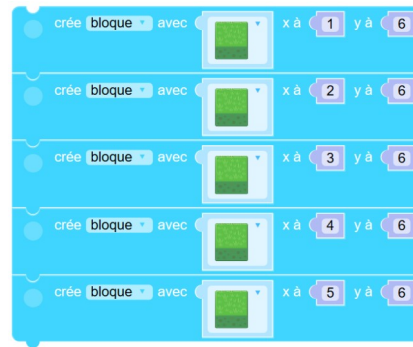


# Problème :: Scrolling



# Problème

- Il faut en créer 100 !



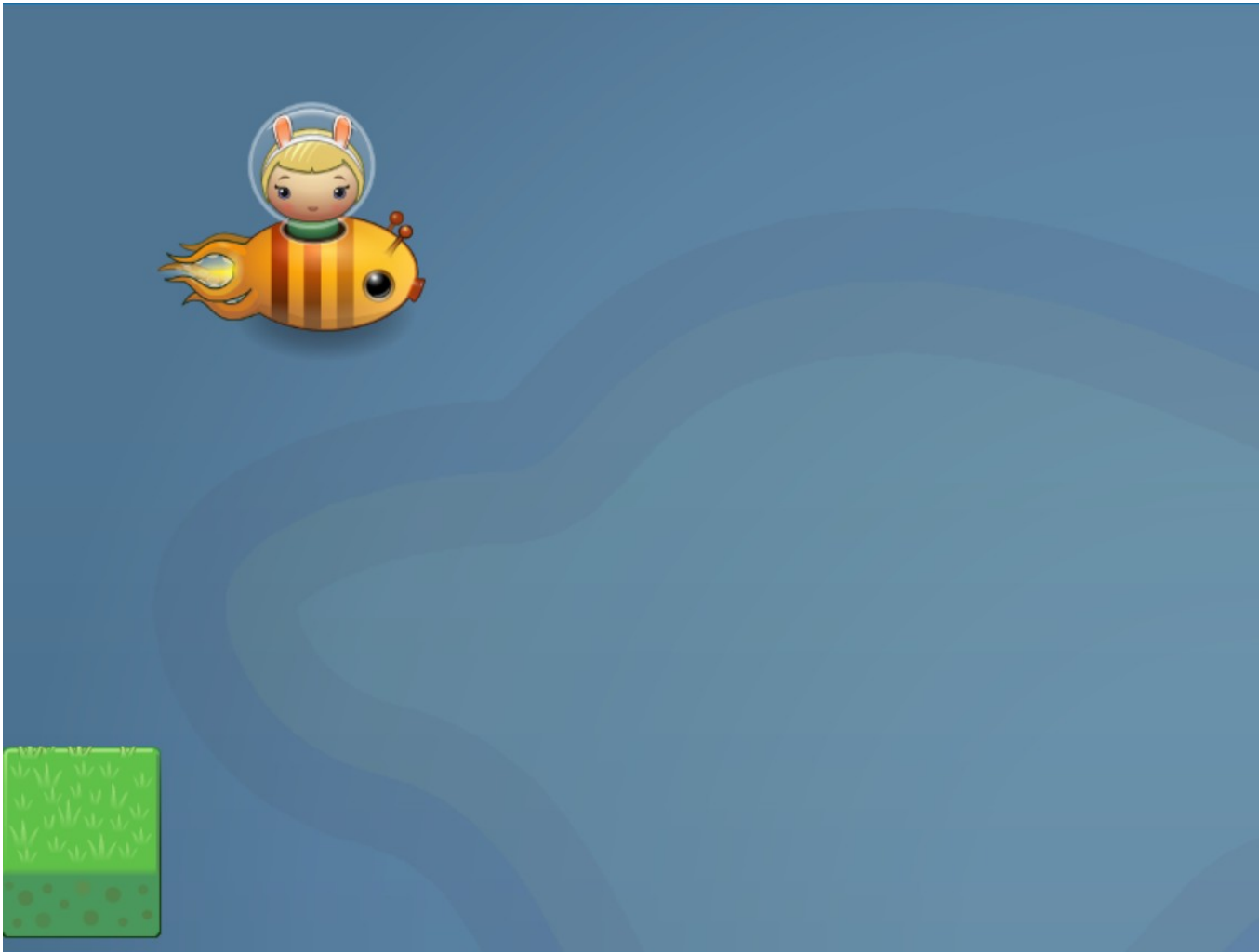
# Solution 2 :: répéter

- Répéter 100 fois !



# Problème

- Ils sont tous au même endroit !



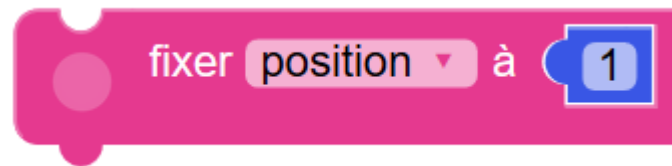
# Solution 2 et demi :: itéré

- $X + 1$  !
- Oui mais comment ?

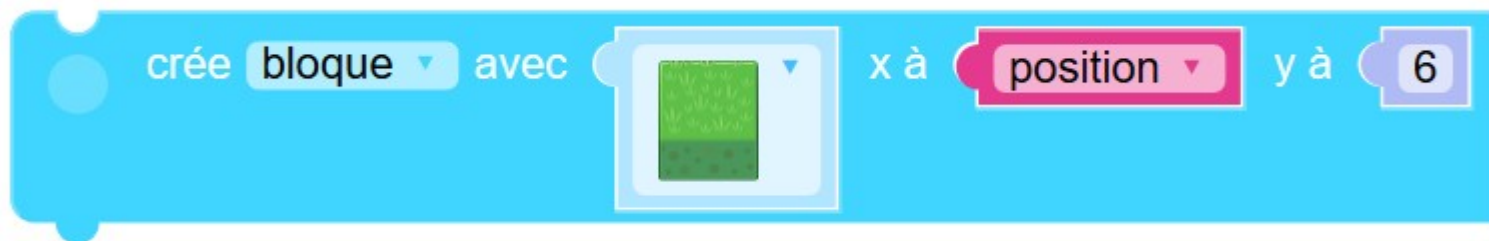


# Nouveau :: Les variables

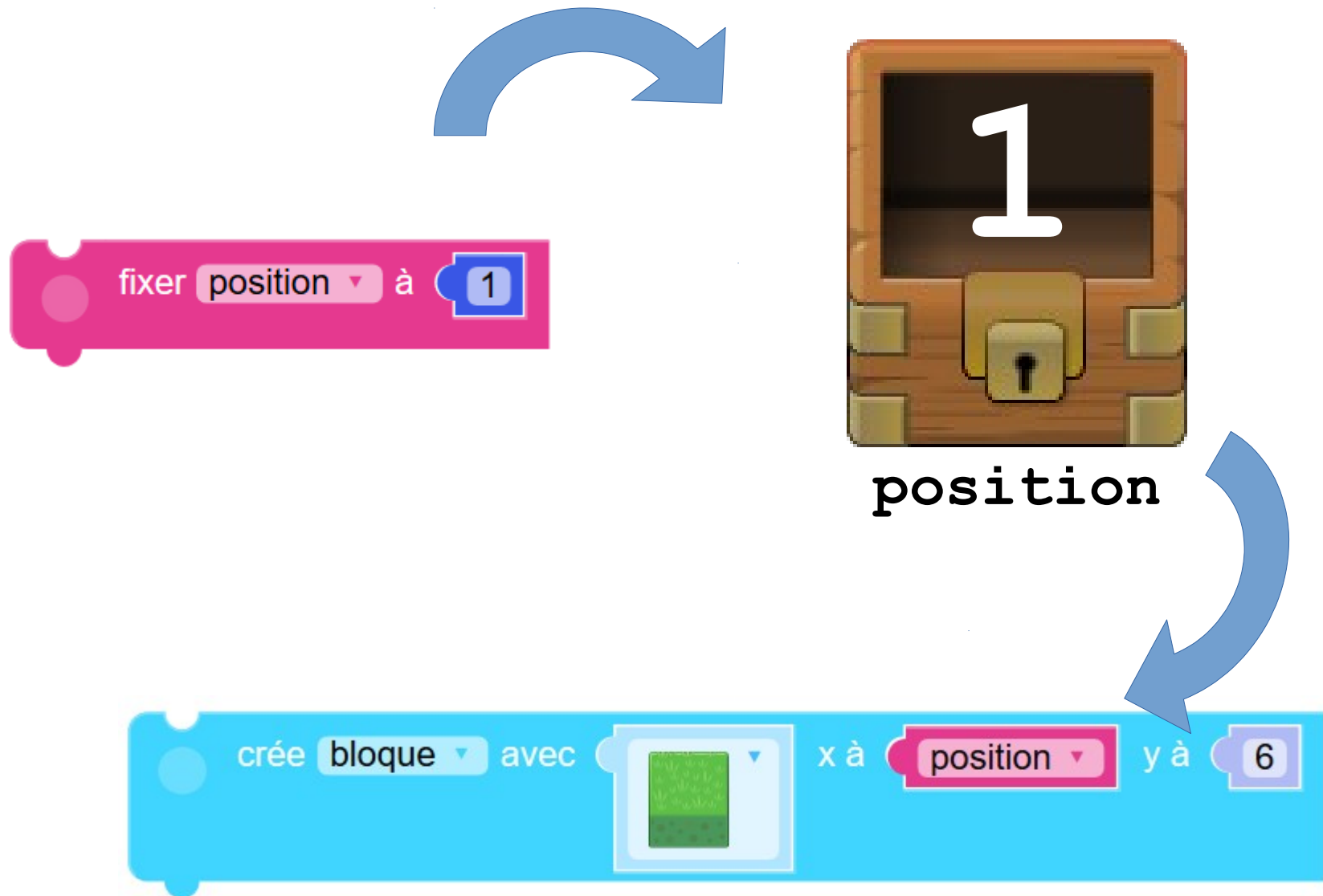
- Une boîte dans la mémoire de l'ordinateur qui a un nom
- On peut mettre un nombre dedans (écrire)



On peut utiliser le nombre (lire)



# Nouveau :: Les variables



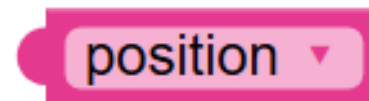
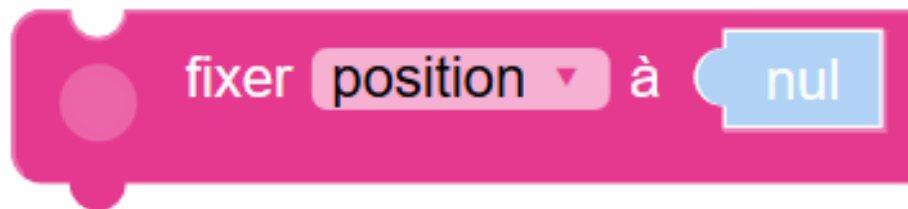
# Question

- Comment ajouter 1 à la position ?
- Chaque fois qu'un nouveau bloque est créé !

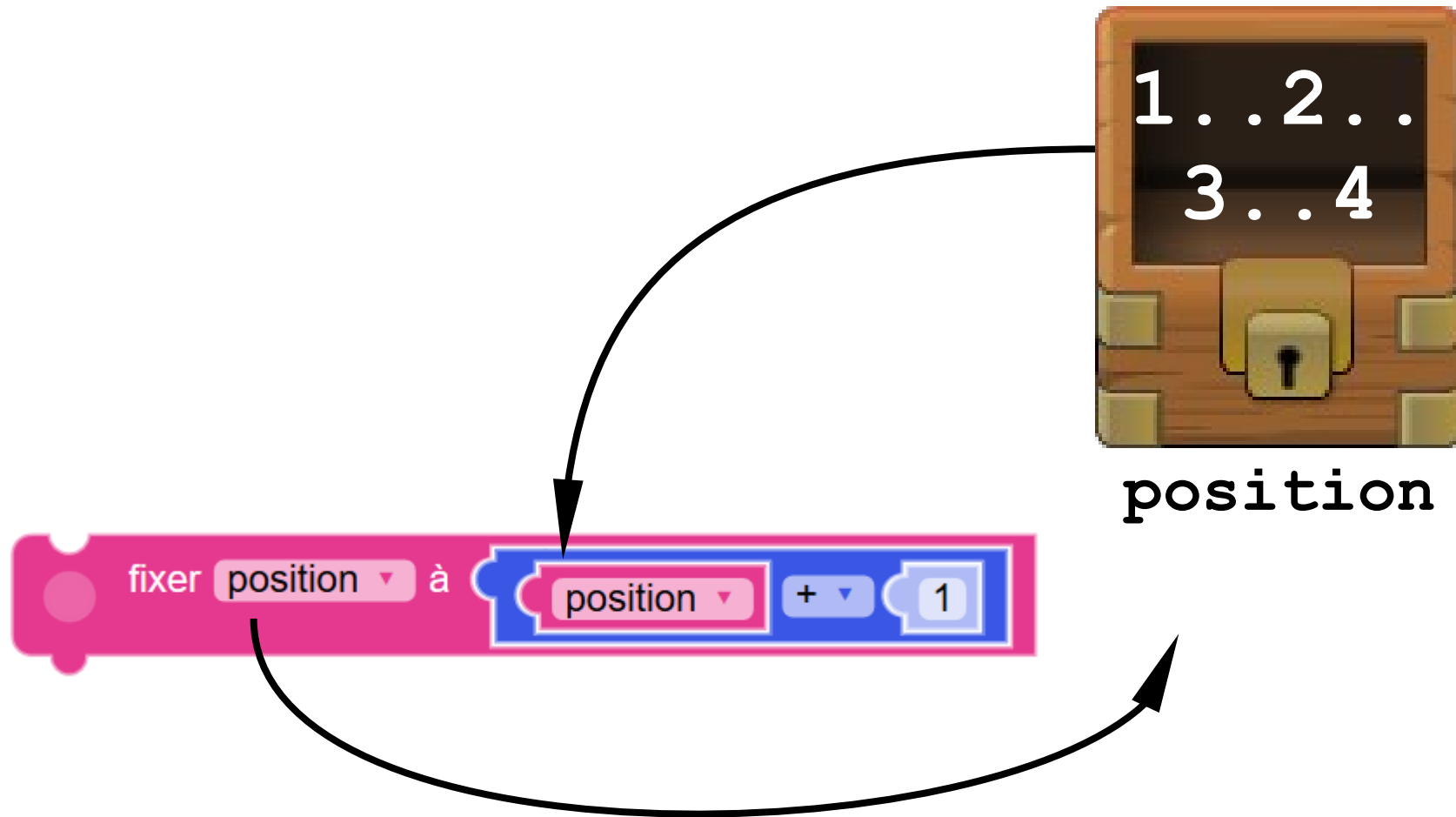




# Indice :: math



# Solution :: math



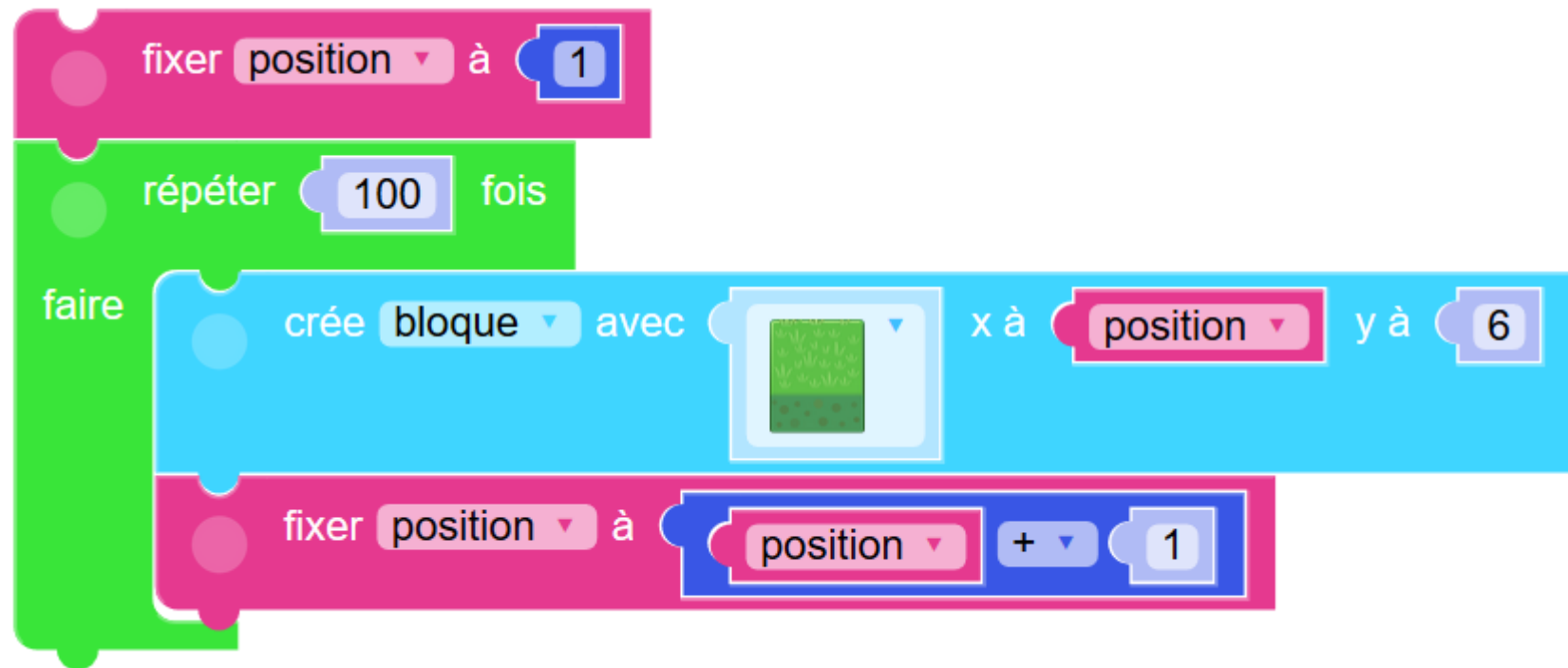
# Solution

- `position = position + 1`



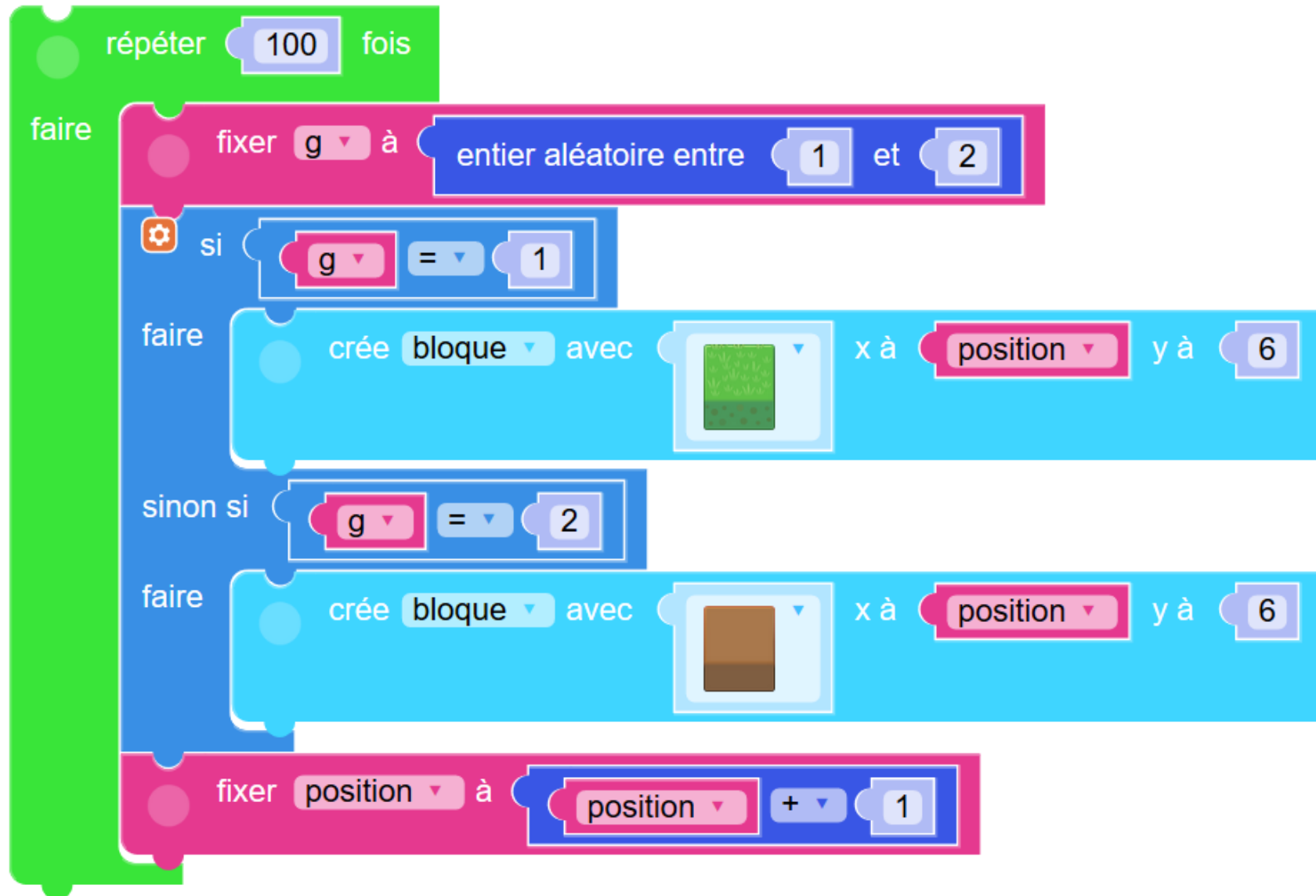
# Tout ensemble

- `position = position + 1`

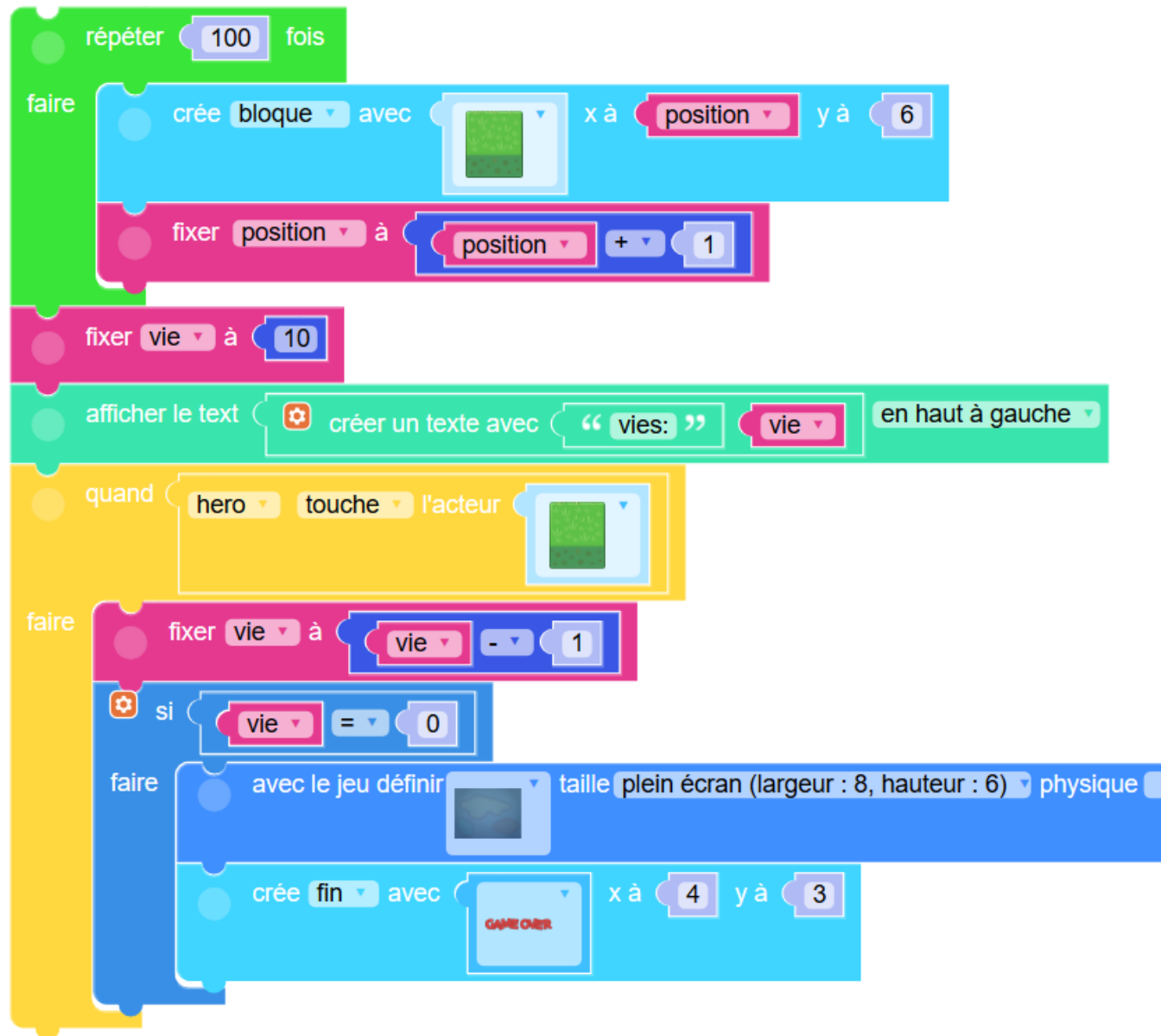


**Bravo !**  
**Et merci de ta participation :-)**

# Idée 1 :: blocques au hasard



# Idée 2 :: gestion des vies



# Idée 3 :: étoiles et score

