

DUREE

NIVEAU

PUBLIC

ACTIVITES

30 - 45 min

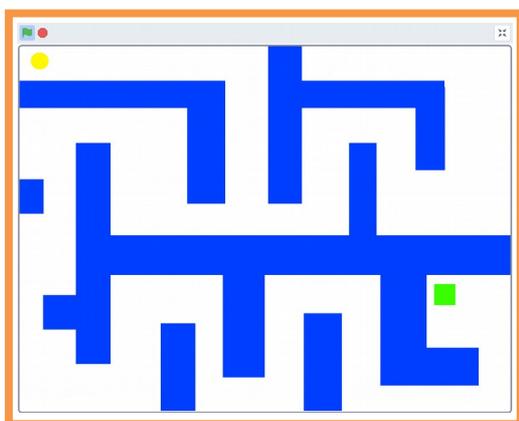
Débutant

enfant – adolescent –  
adulte

Programmation  
d'un jeu

## "Labyrinthe simple"

### FICHE D'ACTIVITE SCRATCH



#### ➤ Objectifs :

Le but de ce jeu est de créer de A à Z un labyrinthe que vous aurez dessiné et de jouer à atteindre la sortie avec un aventurier.

Pour programmer ce jeu, nous allons utiliser la plateforme **Scratch** qui a l'avantage d'utiliser du code visuel : pas besoin d'apprendre un langage de programmation pour construire ses propres jeux vidéo !

#### ➤ Compétences travaillées :

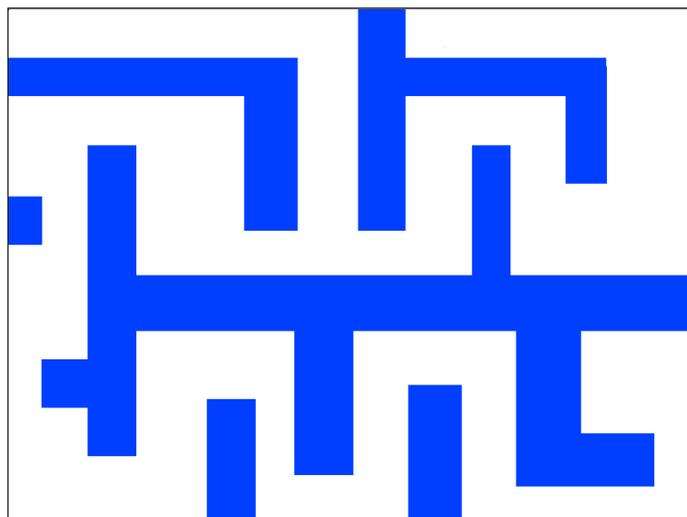
- Variables
- Aléatoire
- Boucles

## ➤ Étape 1 : Dessiner le labyrinthe

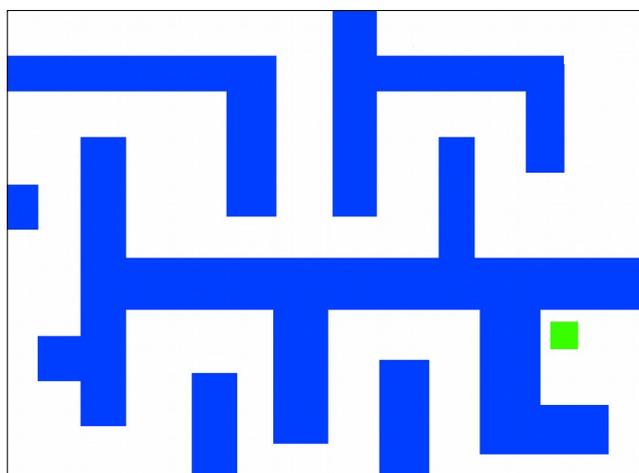
1. Nous allons dessiner notre labyrinthe. Dans la partie scène, cliquez sur l'icône de pinceau pour dessiner un nouvel arrière-plan.



Cette partie est libre, vous pouvez choisir la couleur de votre choix et dessiner votre parcours mais vous devez garder la même couleur. Le plus simple pour y arriver est de créer des rectangles remplis de la couleur.

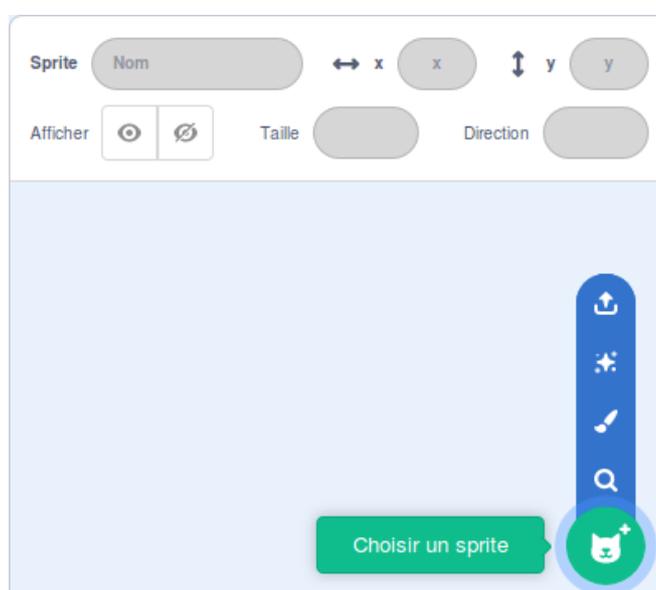


Une fois votre niveau terminé, rajoutez un carré d'une autre couleur pour localiser où se trouve la fin du labyrinthe. Il s'agit de la porte de sortie. Nous avons mis ici un carré vert.



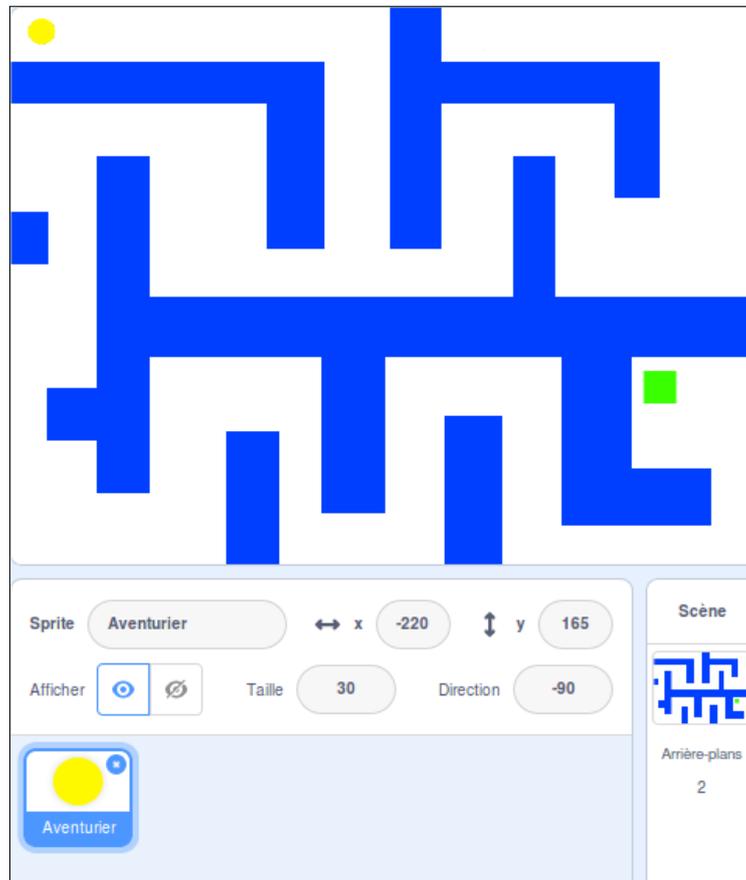
## ➤ Étape 2 : Parcourir le labyrinthe avec le sprite

1. Maintenant, nous allons créer un petit sprite qui aura la forme que vous voulez et qui va parcourir le labyrinthe.  
Pour cela, cliquez sur l'icône de loupe en dessous de la scène pour le choisir dans la bibliothèque Scratch, ou bien sur le petit pinceau pour le dessiner.



Assurez-vous qu'il soit assez petit pour naviguer dans le labyrinthe !  
J'ai choisi une petite boule jaune pour mon exemple.

Une fois le sprite sélectionné, changez son nom, en "Aventurier" par exemple.



2. Mince ! Notre sprite se place sur un mur, mais nous voudrions qu'il se place en haut à gauche du niveau (ou ailleurs pour vous !). Nous allons donc dire au sprite de se placer à un endroit précis. Positionnez le lutin sur son lieu de départ en le faisant glisser avec la souris. Puis, utilisez le bloc « Aller à » avec les valeurs en x et y proposées (elles correspondent à l'emplacement actuel de votre lutin). Écrivez ensuite le code suivant (pour moi les valeurs sont -220 et 165) :

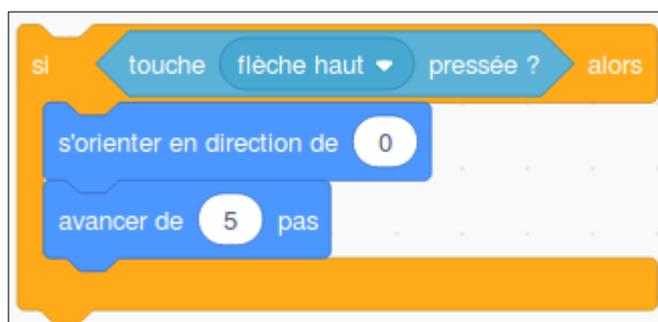


Quand le drapeau vert est cliqué, la balle va se placer sur la scène à abscisse (horizontalement) X = - 220 et à ordonnée (horizontalement) Y = 165.

## ➤ Étape 3 : On monte, on descend, on va à gauche, puis à droite

1. Désormais, il est temps d'animer notre sprite. Rendez-vous dans l'onglet Code une fois que vous avez sélectionné le sprite.

Le principe est simple : dès que l'on appuie sur l'une des touches fléchée, le lutin part dans cette direction. Voici comment faire :



Si la touche du haut est pressée, alors le sprite oriente sa direction vers le haut (0°) et avance de 5.

Quand on lance l'application, on place le sprite au début du labyrinthe, puis on attend qu'une des touches soit pressée et on change la direction.

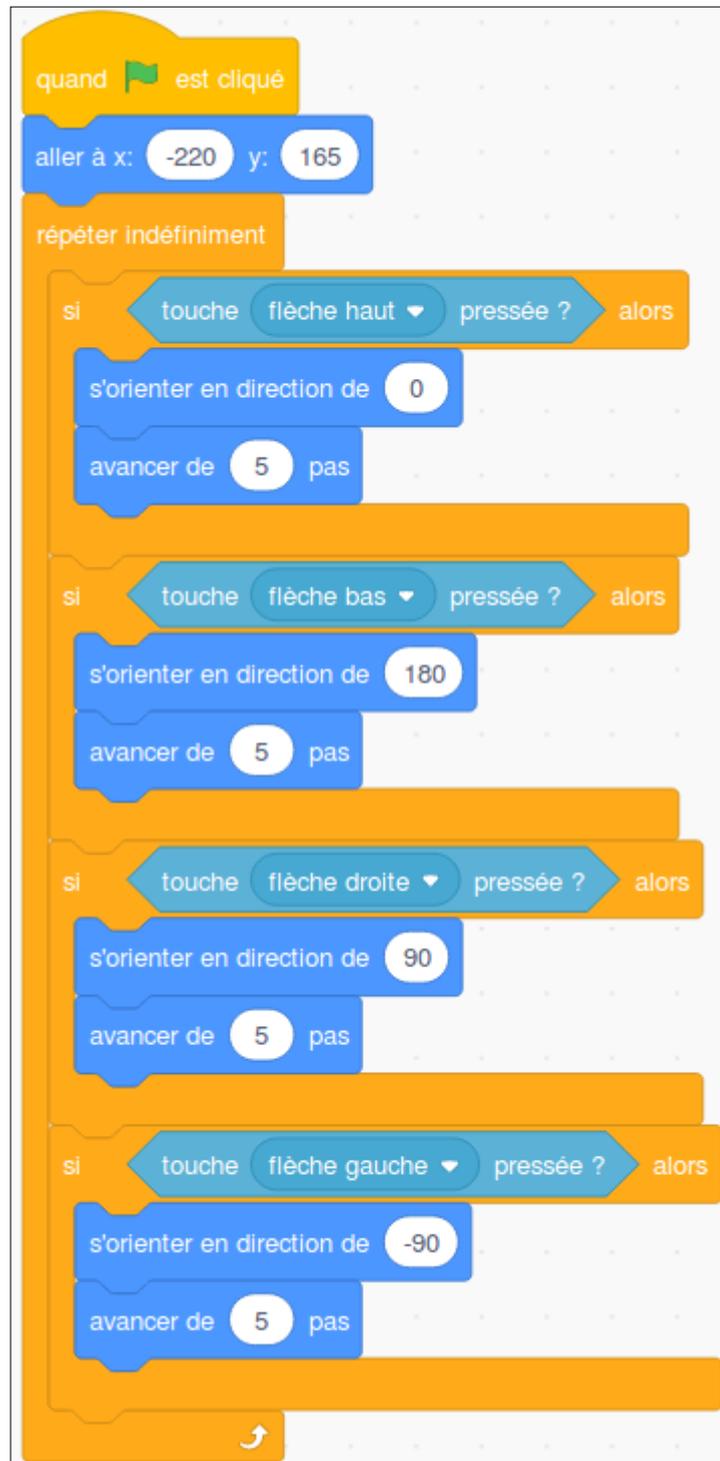
Il faut faire la même chose pour chacune des directions, voici ce que cela va donner :



1. Si la touche du haut est pressée, alors le sprite oriente sa direction vers le haut (0°) et avance de 5.
2. Si la touche du bas est pressée, alors le sprite oriente sa direction vers le bas (180°) et avance de 5.
3. Si la touche de droite est pressée, alors le sprite oriente sa direction vers la droite (90°) et avance de 5.
4. Si la touche de gauche est pressée, alors le sprite oriente sa direction vers la gauche (-90°) et avance de 5.

Remarquez que la seule différence entre les 4 scripts est le bloc « s'orienter » qui change de direction.

2. Pour que notre sprite bouge correctement dans les quatre directions, il reste à assembler ces quatre blocs. Assemblez les blocs ainsi, avec le premier bloc que vous aviez réalisé, plus une boucle « répéter indéfiniment » :



Quand le drapeau vert est cliqué, le sprite va se placer sur la scène, et les quatre conditions réalisées précédemment sont vérifiées en permanence, pour permettre au programme de toujours détecter si une des touches est pressée ou non.

## ➔ Étape 4 : Et les murs alors ?

1. Le sprite est capable de se déplacer, mais quand on passe sur un mur il passe à travers ! Il ne devrait pas.  
Nous allons donc ajouter une condition qui va dire que si le mur est touché, on revient à notre position initiale ([-220,165] pour moi / à vous de mettre vos valeurs).



Si le sprite touche la couleur bleu des murs, alors il retourne à sa position initiale de X = - 220 et Y = 165. Pour mettre la bonne couleur dans le bloc « couleur touchée » : cliquez sur le rond de couleur, puis cliquez sur la pipette, puis cliquez sur un des murs du labyrinthe.

2. Ajoutez un dernier bloc pour que votre sprite crie « victoire » quand vous atteignez la sortie du labyrinthe.



Si le sprite touche la couleur verte du point d'arrivée, alors le lutin dit « Gagné !! » pendant 2 secondes, puis le jeu se termine.

Ajoutez ces deux blocs à la suite de votre script :



Quand le drapeau vert est cliqué, le sprite va se placer sur la scène, et les six conditions réalisées précédemment sont vérifiées en permanence, pour permettre au programme de toujours détecter si oui ou non une des touches est pressée, ou si une couleur est touchée.

## ➔ Bonus

Nous avons à présent un jeu fonctionnel, mais pas forcément passionnant. Pour l'améliorer vous pouvez :

- Ajouter un ennemi
- Ajouter une porte qui s'ouvre avec une clé
- Ajouter un autre niveau