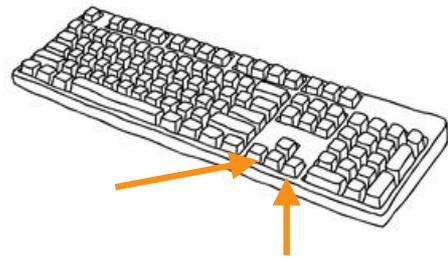




L'IDÉE DU JEU

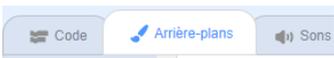


Il s'agit d'un jeu d'arcade, où le joueur dans un wagon de train de mine doit éviter les obstacles. Pour cela il faut orienter le wagon de droite à gauche à l'aide des flèches du clavier.

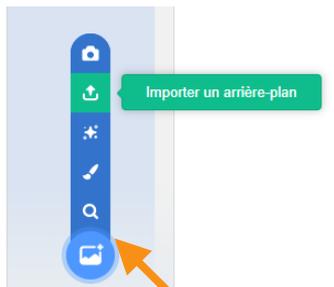


STRUCTURE GRAPHIQUE DU JEU

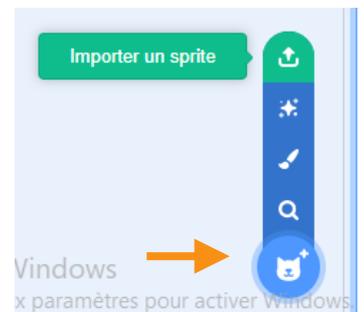
1 Ajouter en arrière plan l'image des rails dans un tunnel. Enregistrez le a l'adresse suivante <http://www.undergroundminers.com/ashland29.jpg> ou sur [le site de technologie](#)



Cliquer ensuite sur **arrière plan** en haut a gauche puis sur **importer un arriere plan**

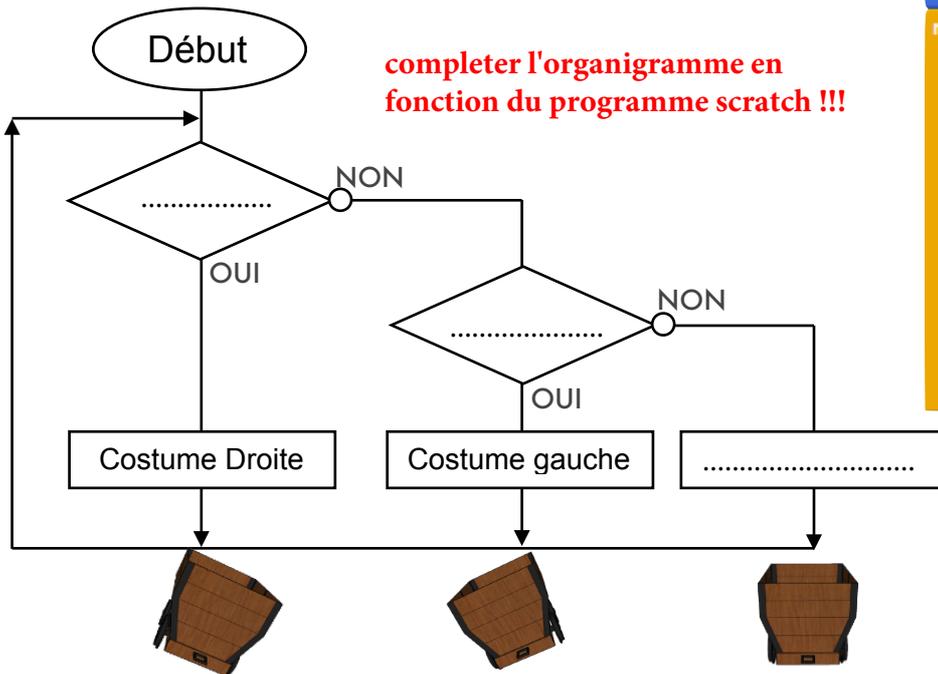


2 Ajouter le lutin correspondant au wagon . Telecharger le fichier (le lutin se trouve sur le site de technologie) , et dezipper le .
Cliquez ensuite en bas a droite sur **choisir un sprite** puis **importer** .



Eventuellement, penser à renommer les lutins si besoin en fonction du programme , ce qui facilitera l'écriture et la relecture du programme.

GESTION DU DÉPLACEMENT DU WAGON



```
quand flag est cliqué
aller à x: 0 y: -95
répéter indéfiniment
  si touche flèche droite pressée? alors
    basculer sur le costume Droite
  sinon
    si touche flèche gauche pressée? alors
      basculer sur le costume Gauche
    sinon
      basculer sur le costume Centre
```



```
aller à x: 0 y: -95
```

Cette fonction permet de placer le Wagon au centre de l'écran (x=0) et en bas (y=-95).

Sachant que l'écran de la scène fait 480 (-240 < x < 240)
par 360 pixels (-180 < y < 180)

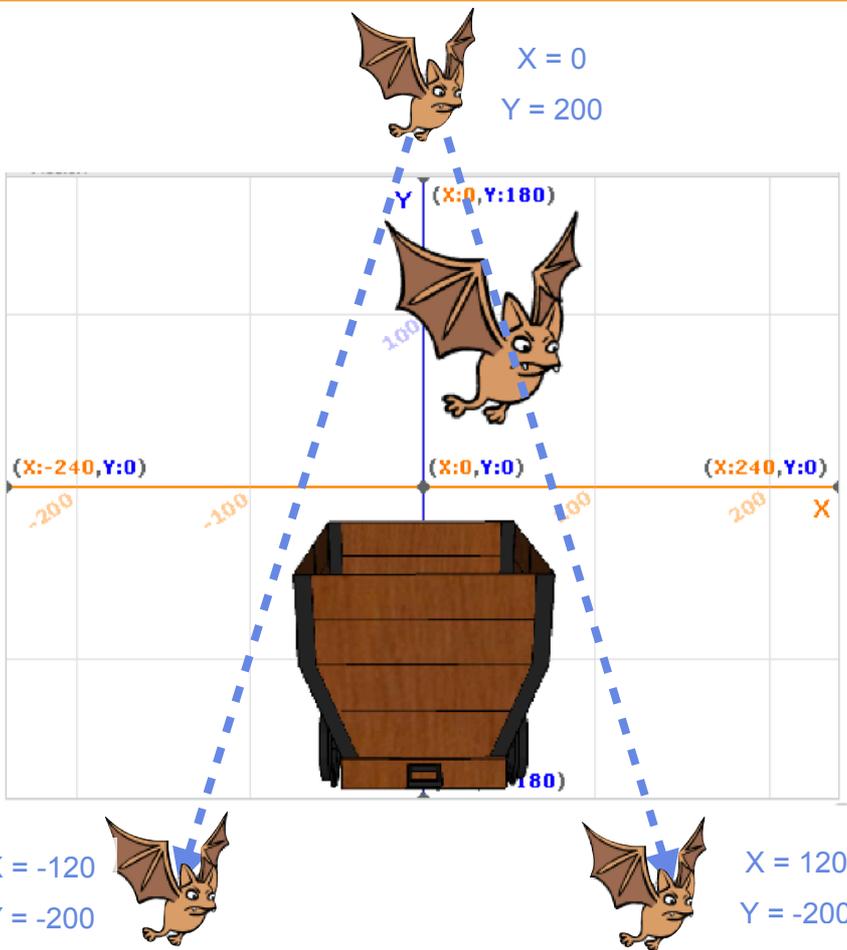
GESTION DES OBSTACLES



Bat

- 1 Ajouter un lutin « obstacle », ici une chauve souris nommée Bat directement disponible dans la bibliothèque Scratch.

ATTENTION : dans les programmes vous devrez mettre bat et non bat 1 !!!



La chauve souris est placée au centre en haut en dehors de la scène : en $x = 0$ et $y = 200$.

Puis elle se déplacera en bas (en dehors de la scène en $y = -200$) soit à droite en $x = 120$ soit à gauche en $x = -120$.

```

quand est cliqué
mettre à 70 % de la taille initiale
répéter indéfiniment
  aller à x: 0 y: 200
  glisser en 2 secondes à x: 120 y: -200

```



mettre à 70 % de la taille initiale

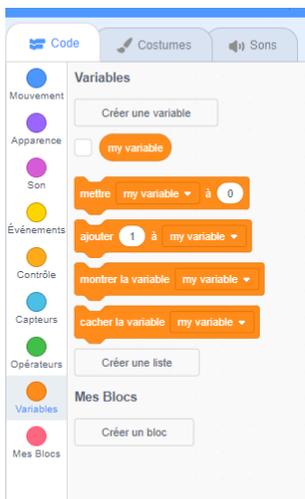
Permet de diminuer la taille de la chauve souris sans utiliser l'interface graphique.

2

Il va falloir maintenant gérer aléatoirement l'arrivée des chauves souris à droite ou à gauche.

Pour cela, utiliser une variable **Aleatoire-Droite_Gauche** qui sera égale soit à 1 soit à 2.
 Si la variable est égale à 1 : direction vers la droite

Si la variable est égale à 2 : direction vers la gauche



```

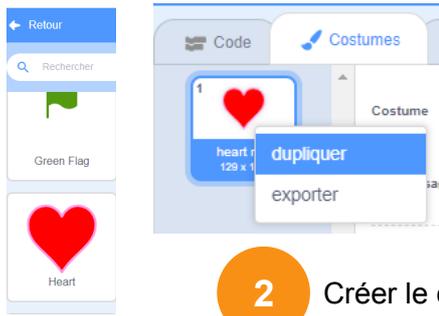
quand est cliqué
mettre à 70 % de la taille initiale
répéter indéfiniment
  aller à x: 0 y: 200
  mettre Aleatoire-Droite_Gauche à nombre aléatoire entre 1 et 2
  si Aleatoire-Droite_Gauche = 1 alors
    glisser en 2 secondes à x: 120 y: -200
  sinon
    glisser en 2 secondes à x: -120 y: -200

```

GESTION DES VIES

1

Ajouter un lutin coeur . Puis dans le menu costume cliquer sur dupliquer et creer **3 costumes** supplementaires (2 coeurs pour 2 vies , 3 coeurs pour 3 vies et game over)

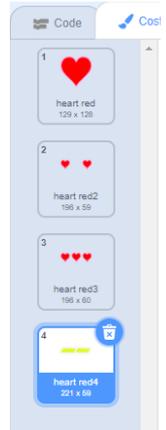
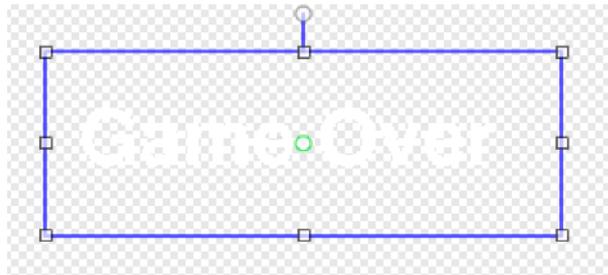
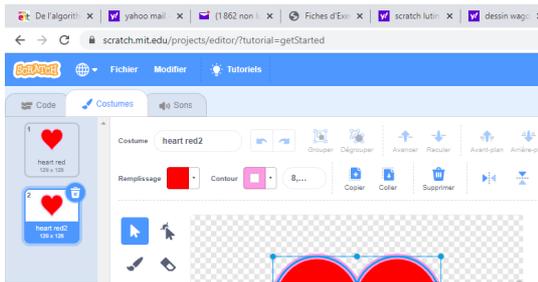


A chaque fois qu'une chauve souris touche le wagon, le joueur perd une vie.

Le costume démarre en début de jeu sur 3 vies pour finir sur 1 vie puis Game Over

2

Créer le costume Game Over à la suite des costumes des vies



Utiliser les outils pour copier , diminuer la taille du coeur pour faire les costumes....



L'image d'arrière plan du jeu est sombre, penser à écrire le texte en blanc pour qu'il soit plus lisible

3

Créer une variable **Vie**



4

Programmer la gestion des vies

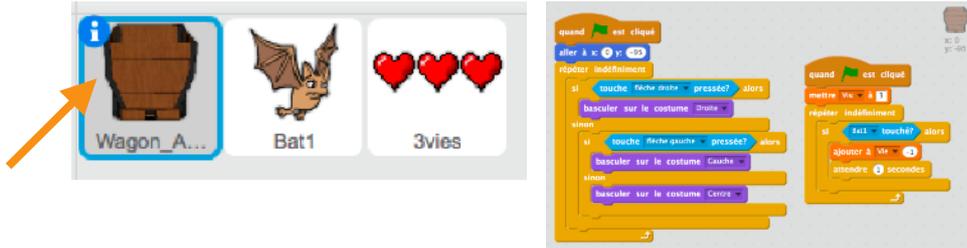
- Réduire les images →
- Positionner les images en haut à droite de la scène →
- Afficher le costume 3 vies dès le début du jeu →
- Si la variable vie = 2 alors afficher le costume 2vies →
- Si la variable vie = 1 alors afficher le costume 1vie →
- Si la variable vie = 0 alors afficher le costume GameOver →
- Arrêter tous les programmes ! →



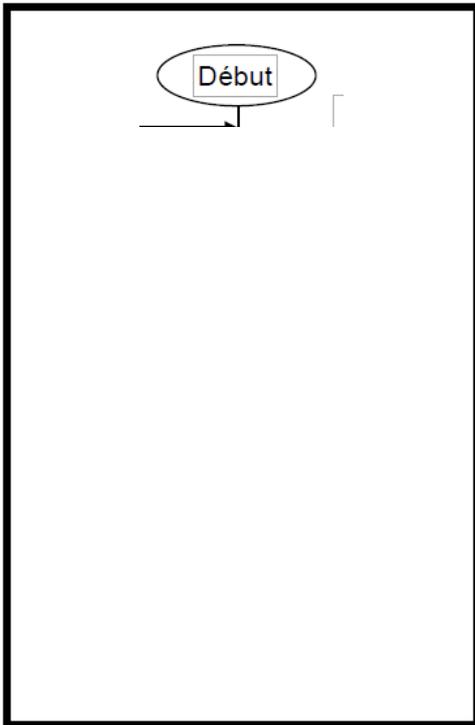
5

Créer le programme qui permet d'enlever une vie à chaque fois que la chauve souris (nommée Bat1) touche le wagon.

Pour cela revenir dans le lutin du Wagon et ajouter le programme en question.



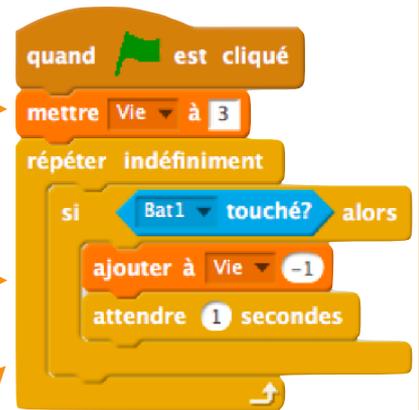
Créer l'organigramme correspondant au programme si contre !!!



Initialisation de la variable Vie à 3

Si la Bat1 touche le Wagon alors enlever 1 à Vie

Attendre 1 seconde, permet de ne pas perdre les 3 vies immédiatement !



GESTION DU SCORE

1 Créer une variable **Score**



2 Dans un des lutins créer le programme qui gère le score : C'est à dire, toutes les secondes le joueur gagne 1 points

Initialisation de la variable Score à 0

Toutes les secondes Ajouter 1 à la variable Score



La suite du jeu dépend de votre créativité ...