



Présentation de scratch

7 & 8



I. Scratch c'est quoi ?

Scratch c'est un **logiciel de programmation visuelle** développé par l'Institut de Technologie du Massachussetts aux Etats- Unis en 2006. C'est un logiciel gratuit.

II. Comment utiliser scratch ?

A. Ouvrir Scratch 3

Il y a moyen d'utiliser scratch : soit en ligne sur <https://scratch.mit.edu/> soit avec le logiciel hors ligne.



Scratch Desktop

Nous nous allons utiliser le logiciel **Scratch Desktop**.

Il est déjà installé sur ton ordinateur. Cherche le logo sur le bureau et double clique

A. L'interface de Scratch 3

The screenshot shows the Scratch Desktop interface with the following labeled components:

- Pour changer la langue**: Points to the language selection icon in the top left.
- Barre de menu**: Points to the top menu bar containing 'Fichier', 'Modifier', 'Tutoriels', and 'Projet Scratch'.
- Barre d'outils**: Points to the toolbar on the right side of the top menu bar.
- La scène**: Points to the main stage area where the cat sprite is running.
- Zone des scripts**: Points to the 'Scripts' block palette on the left.
- Palette des blocs**: Points to the 'Mouvement' block palette on the left.
- Les arrière-plans**: Points to the 'Backgrounds' panel on the right.
- Zone des lutins**: Points to the 'Sprites' panel on the right.

A. Le langage Scratch

Les lutins : dans Scratch, un lutin est un objet que tu vas animer à l'aide du code. Il peut s'agir d'un personnage mais aussi d'un objet ou d'un élément de décor.



Les arrière-plans : ce sont des décors que tu peux ajouter à tes histoires ou tes jeux.




Les blocs : Scratch permet de construire un programme en assemblant des instructions comme des pièces de puzzle.

Chaque instruction s'appelle un bloc. Il y a :


- **Les blocs déclencheurs** : de couleurs marron, ils indiquent un évènement et démarrent un bout de code.



- **Les blocs d'instructions** : ils sont classés par catégories (mouvement, sons, apparence, ...) et ont une encoche en haut et en bas pour s'accrocher les uns aux autres.

Le drapeau vert  : il donne le signal du départ. En cliquant dessus, tu ordonne à ton lutin d'exécuter les instructions que tu as codées.

Pour que ce bouton fonctionne, on place toujours en premier dans le bloc « quand le drapeau vert est cliqué ».

Le bouton rouge  : il arrête l'exécution du code.

La zone des lutins et des arrière-plans : ici, tu peux voir l'ensemble des lutins que tu as créés. Tu peux les modifier en cliquant sur leur icône.

La scène : elle te permet de voir le résultat de l'instruction que tu as donnée et donc de tester immédiatement ton code ! Pour que ton lutin joue ses instructions, clique sur le drapeau vert.

La zone des scripts : c'est dans cette zone que tu vas glisser les blocs d'instructions pour construire ton programme.

La palette des blocs : chaque bloc correspond à une commande qui a été programmée en langage informatique. Dans Scratch, elles sont classées par catégories.





Programmer un jeu avec scratch

8



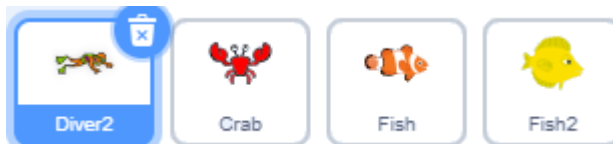
I. Travail demandé ?

On veut créer un jeu qui permet de déplacer un plongeur (Diver) dans un fond marin (Underwater) et d'attraper des poissons tout en évitant d'être attaqué par le Crabe.

II. Démarche de réalisation du jeu

A. Insertion d'un arrière-plan (Underwater)

B. Insertion des Lutins suivants :



C. Placer les lutins dans la scène selon la disposition suivante :



D. Ajouter le code pour le plongeur (Diver), pour les poissons et pour le Crab





	<p>C'est un bloc de mouvement qui modifie le style de rotation de l'image-objet dans le projet.</p>
	<p>Ce bloc sert à savoir si une certaine touche est pressée.</p>
	<p>Ce bloc sert à déterminer si un lutin touche le lutin choisi dans ce bloc.</p>
	<p>Les blocs contenus dans ce bloc seront dans une boucle sauf que la boucle ne se termine jamais (sauf si le bouton rouge est cliqué ou que le bloc stop tout est exécuté).</p>
	<p>Les blocs contenus à l'intérieur de ce bloc ne seront exécuté sauf si la condition est correcte</p>



Programmer un jeu avec scratch (Variable Score)



I. Travail demandé ?

8

On veut créer un jeu qui permet de déplacer une fusée (Rocketship) dans l'espace (Space City 2) et d'attquer des lunes (Beachball) tout en évitant d'être frappée par le rocher (Rocks).

II. Démarche de réalisation du jeu

A. Insertion de l'arrière-plan (Space City 2)

B. Insertion des lutins:



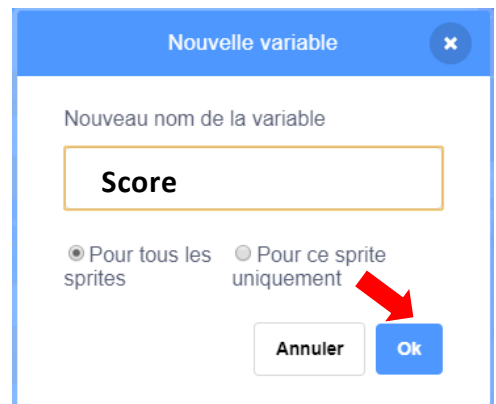
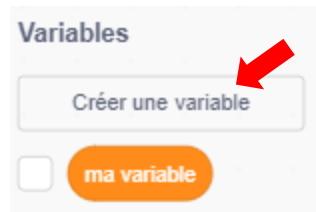
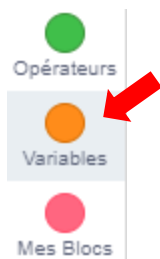
C. Placer les lutins dans la scène selon la disposition suivante :



D. Ajouter le code pour déplacer la fusée (Rocketship) avec les flèches du clavier.

E. Ajouter le code pour augmenter le score chaque fois que la fusée touche une lune (Beachball)

a. Créer une variable **Score**



b. On obtient la variable dans la liste des variables et aussi un affichage de la valeur du score sur la scène



c. On ajoute le code sur la scène pour mettre la valeur du score à Zéro (0) à chaque fois on démarre le jeu.

d. Modifier le code des lunes pour augmenter la valeur du score à chaque fois qu'ils touchent la fusée (Rocketship).

F. Ecrire le script (code) pour que les roches suivent la fusée, et pour que le jeu se termine quand ces derniers se heurtent (se touchent).

G. On crée un nouveau sprite avec l'option

Peindre , on écrit le texte (Gagné) avec l'outil **Texte**

et on ajoute le code pour que ce texte apparait (**montrer**) quand on obtient le score maximal du jeu.



Programmer un jeu (2) avec scratch (variables & opérateurs)



I. Travail demandé ?

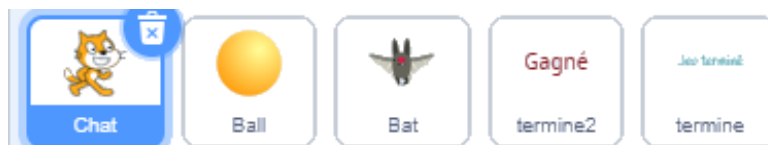
8

On veut créer un jeu qui permet de déplacer un chat (sprite1) dans la scène en montant et en descendant un escalier et de lancer des balles sur la chauve-souris (Bat).

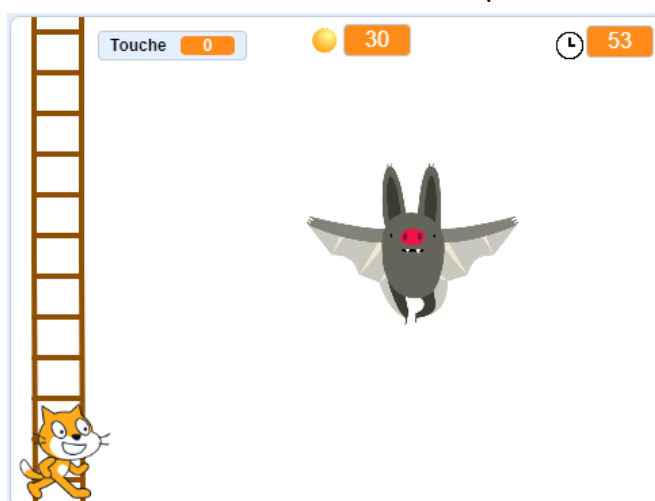
II. Démarche de réalisation du jeu

A. Dessiner l'escalier dans l'arrière-plan principal avec les outils de dessins.

B. Insertion des lutins:



C. Placer les lutins dans la scène selon la disposition suivante :



Variables

Créer une variable

- Balles
- pos_y_chat
- Temps
- Touche

D. Créer les variables **Temps**, **Touche**, **Balles** et **pos_y_chat**

- a. Le nombre de balles commence par 20.
- b. Le temps du jeu est de 60 secondes.
- c. Le nombre de touche commence à zéro 0.

E. Ajouter le code pour déplacer le Chat avec les flèches **↑** et **↓** du clavier.

F. Ajouter le code pour glisser la chauve-souris à des positions aléatoires.

G. Ajouter le code qui permet de lancer les balles avec la touche espace.

- a. Le nombre de balles diminue de 1 à chaque lancée.
- b. Si la balle touche la chauve-souris, on ajoute 1 à la variable touche et on diminue la taille du lutin de 20%
- c. Si on touche 3 fois le Bat, on affiche « Gagné » et le jeu s'arrête.

H. On ne peut plus lancer des balles si le nombre de balles devient égal à 0 zéro.

I. Le jeu s'arrête si le temps s'écoule (Temps devient égal à 0 zéro).

II. Codes des lutins (Solutions)

```

quand [drapeau] est cliqué
mettre Temps à 60
mettre Balles à 30
mettre Touche à 0
répéter indéfiniment
    attendre 1 secondes
    ajouter -1 à Temps
    si Temps = 0 alors
        envoyer à tous jeu_termine
        attendre 1.5 secondes
        stop tout
    
```

```

quand [drapeau] est cliqué
aller à x: -200 y: -140
répéter indéfiniment
    mettre pos_y_chat à ordonnée y
    si touche flèche haut pressée ? alors
        ajouter 10 à y
    si touche flèche bas pressée ? alors
        ajouter -10 à y
    
```

Jeu terminé

```

quand [drapeau] est cliqué
    cacher

quand je reçois jeu_termine
    répéter 4 fois
        montrer
        attendre 0.3 secondes
        cacher
    
```

Gagné

```

quand [drapeau] est cliqué
    cacher

quand je reçois gagne
    répéter 4 fois
        cacher
        attendre 0.3 secondes
        montrer
    
```




```

    quand le drapeau est cliqué
    cacher
    répéter indéfiniment
    si la touche espace est pressée ? alors
    si Balles > 0 alors
    ajouter -1 à Balles
    aller à Chat
    montrer
    glisser en 1 secondes à x: 240 y: pos_y_chat
    cacher
  
```



```

    quand le drapeau est cliqué
    mettre la taille à 100 % de la taille initiale
    montrer
    répéter indéfiniment
    glisser en 2 secondes à position aléatoire
    attendre 1 secondes
    si touche le Ball ? alors
    ajouter 1 à Touche
    ajouter -20 à la taille
    si Touche = 3 alors
    cacher
    envoyer à tous gagne
  
```



Programmer un jeu (3) avec scratch (Labyrinthe)



I. Travail demandé ?

8

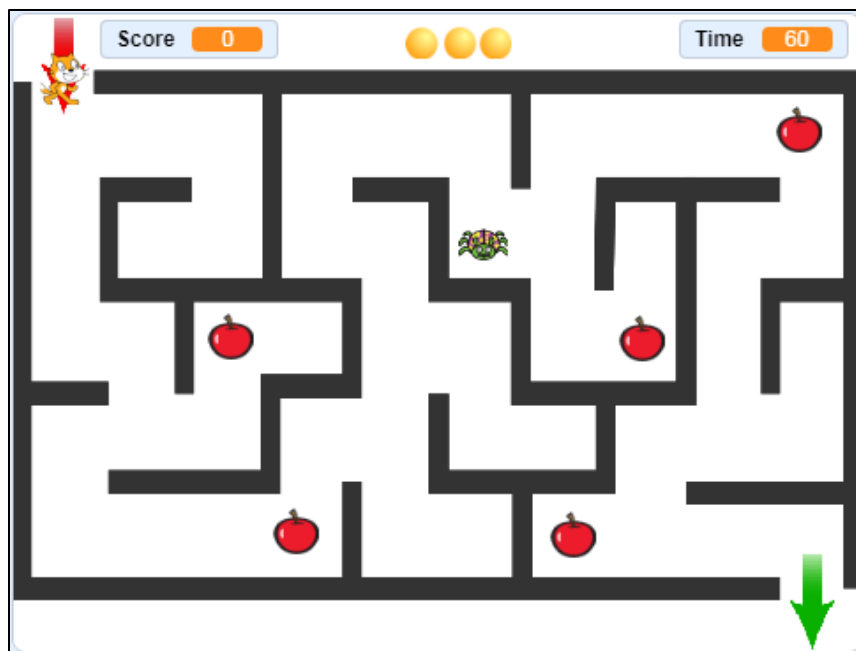
Créer un jeu du labyrinthe avec un joueur qui se déplace en mangeant des fruits (exemple : pommes) et en évitant des monstres. L'objectif est de sortir du labyrinthe en ayant mangé le maximum de fruits.

Pour aller plus loin : Nous allons ajouter un chronomètre et un compteur de score qui ajoute des points lorsqu'un fruit est mangé et arrêter la partie lorsqu'un monstre est touché 3 fois puis afficher les résultats à la fin de la partie.

II. Démarche de réalisation du jeu

A. Rechercher l'image du labyrinthe sur Internet et l'insérer dans la scène.

B. Insertion des lutins dans la scène :



C. Créer les variables **Score**, **Vies**, **Times**

Le **Score** commence par 0, le nombre de **vies** commence par 3 et le temps (**Time**) du jeu est de 60 secondes.

D. Ajouter le code pour déplacer le **Chat** avec les flèches de directions : $\uparrow \downarrow \rightarrow \leftarrow$.

- Le Chat ne doit pas traverser les **lignes noires** du Labyrinthe

- Utiliser le bloc **couleur touchée** pour reculer le lutin.

couleur  touchée ?

E. Ajouter le code pour déplacer le monstre (**Ladybug2**) en continue dans la scène.

F. Si le **Chat** touche la couleur verte on affiche « Gagné » et le jeu s'arrête.

G. Le jeu s'arrête si le temps s'écoule (Temps devient égal à 0 zéro).

II. Codes des lutins (Solutions)

```

quand [drapeau] est cliqué
  aller à x: -211 y: 144
  répéter indéfiniment
    si [touche flèche haut ▼ pressée ?] alors
      ajouter 3 à y
    si [couleur [noir] touchée ?] alors
      ajouter -3 à y
  
```




```

quand [drapeau] est cliqué
  montrer
  répéter indéfiniment
    si [touche le Chat ▼ ?] alors
      cacher
      ajouter 1 à Score
  
```

```

quand [drapeau] est cliqué
  répéter indéfiniment
    si [touche le Ladybug2 ▼ ?] alors
      ajouter -1 à Vies
      aller à x: -211 y: 144
  
```

2-Créer un lutin (**vie**) pour la gestion des vies dans le jeu et créer les 4 costumes suivants

		
B3	B2	B1

```

quand [drapeau] est cliqué
  répéter indéfiniment
    aller à x: 23 y: 120
    glisser en 1 sec à x: 26 y: 50
    glisser en 1 sec à x: 69 y: 50
    glisser en 1 sec à x: 73 y: 120
    glisser en 1 sec à x: 68 y: 45
    glisser en 1 sec à x: 26 y: 48
    glisser en 1 sec à x: 25 y: 122
  
```

```

quand [drapeau] est cliqué
  répéter indéfiniment
    si [Vies = 3] alors
      basculer sur le costume B3
    si [Vies = 2] alors
      basculer sur le costume B2
    si [Vies = 1] alors
      basculer sur le costume B1
    si [Vies = 0] alors
      basculer sur le costume B4
  
```



Utiliser les sons avec scratch

8



I. Introduction

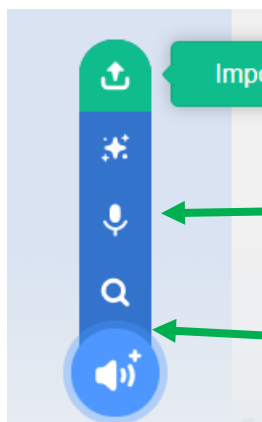
Sur Scratch, chaque lutin peut jouer des sons et de la musique.


Scratch vous offre le choix entre deux formats de fichiers courants, et différentes qualités (MP3 & WAV).


II. Importer une musique ou un son sur Scratch


Il est possible d'importer sa propre musique sur Scratch, afin de choisir quelle musique nous voulons mettre dans notre projet. Pour cela, il faut suivre cette démarche :

- Passez dans le menu sons:



 Permet d'importer un son de votre ordinateur

 Enregistrer un son avec son microphone

 Chercher et ajouter un son de la bibliothèque pour le lutin

- Cliquez sur *importer un son* et choisissez votre morceau
- Votre musique est insérée!

III. Modifier un son

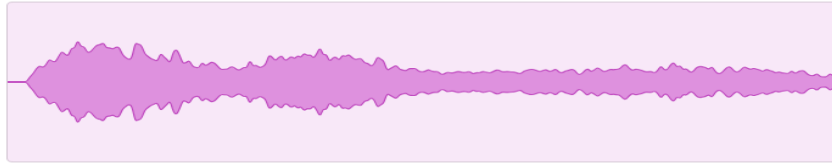
Il existe 13 boutons permettant de modifier un son dans l'éditeur scratch :

A. Boutons en haut



1. **Copy / Copier**: Il vous permet de copier un bout de votre morceau
2. **Paste / Coller**: Quand vous avez copié quelque chose, il vous permet de le coller
3. **Copy to New / Copier à un nouveau son**: Vous permet de coller un bout de votre son comme un nouveau son.
4. **Delete / Supprimer**: Quand vous avez sélectionné quelque chose, vous permet de le supprimer

B. Piste du son



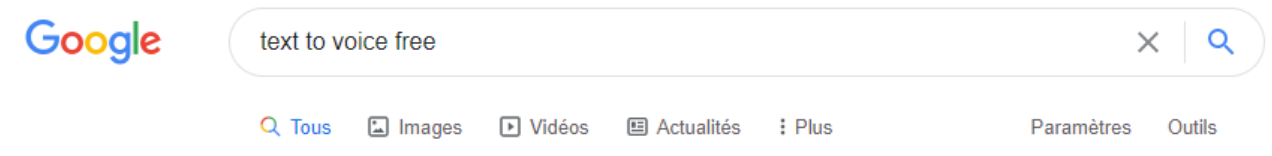
C. Boutons en bas



5. **plus vite**: accélère le son
6. **plus doucement**: ralentit le son
7. **plus fort**: accentue le son
8. **plus doux**: baisse le son
9. **mute**: supprime le son
10. **fondre en ouverture**: donne un effet crescendo (le son augmente peu à peu)
11. **fondre en fermeture**: donne un effet decrescendo (le son diminue peu à peu)
12. **inverser**: inverse le son
13. **robot**: prend une intonation de robot

IV. Text to voice (Lire un texte sur Internet)

1. Dans **Google**, chercher un site pour convertir du texte en voix (**Text to Voice**)



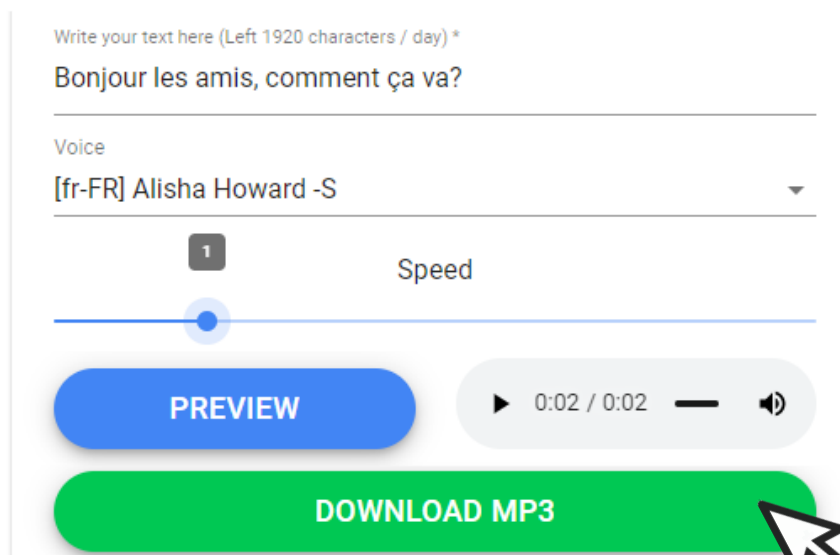
Environ 1 670 000 000 résultats (0,58 secondes)

<https://wideo.co> > text-to-speech ▾ Traduire cette page

[Free Text to Speech Software \(TTS\) - by Wideo](#)

2. Taper le texte à convertir, choisir la **voix** et la **langue**, modifier la **vitesse** (Speed)

wideo





Coder les sons pour le jeu Labyrinthe

8



I. Ajouter un son de fond pour la scène ?

Pour ajouter un son qui se joue en continu dans notre jeu, on peut :



mettre le volume à 50 %

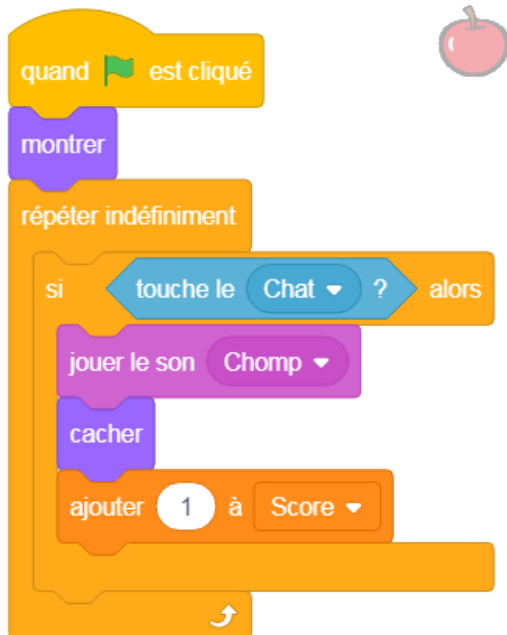
Ce bloc permet de modifier le volume en pourcentage.

Le niveau ne peut pas dépasser 100%.

jouer le son Video Game 1 jusqu'au bout

permet de jouer un des sons du lutin **en entier**

II. Coder le son pour les lutins (fruits et messages)



jouer le son pop

Ce bloc permet de jouer un des sons du lutin (si une autre action est effectuée, elle coupe le son).

arrêter tous les sons

Ce bloc permet de couper tous les sons joués au moment où il est activé pour tous les lutins.

volume

Ce bloc contient le volume actuel du son.



ajouter 10 à l'effet hauteur

Ajoute un effet sonore à un son

mettre l'effet hauteur à 100

Ce bloc met l'effet sonore d'un son à une certaine valeur.



Programmer un jeu AppleCatcher

8

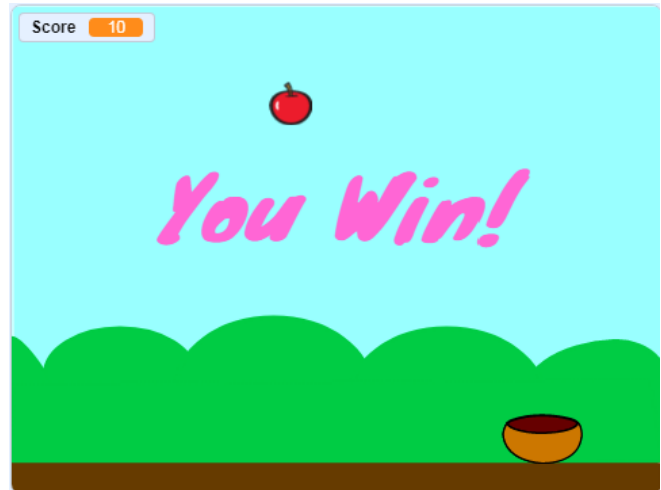


I. Travail demandé ?

Créer un jeu **AppleCatcher**, qui permet de déplacer un panier à droite et à gauche, en essayant d'attraper les pommes qui tombent du ciel.

Le jeu se termine lorsqu'on obtient un score bien défini (Ex : 10).

Pour aller plus loin : Nous allons ajouter les sons pour améliorer notre jeu : fond musicale, son pour les pommes et un son d'applaudissement lorsque le jeu se termine.



II. Démarche de réalisation du jeu

A. Choisir l'arrière-plan **Blue Sky**



B. Insertion des lutins **Bowl** et **Apple** dans la scène :



C. Créer un nouveau lutin (**Peindre**) contenant le texte « **You Win !** »

D. Ajouter une variable **Score** qui va recevoir le nombre de pommes attrapées.

E. Ajouter le code pour déplacer le panier (**Bowl**) à droite et à gauche.

```

quand [drapeau] est cliqué
  aller à x: -8 y: -135
  répéter indéfiniment
    si [touche flèche gauche ▼ pressée ?] alors
      ajouter -10 à x
    si [touche flèche droite ▼ pressée ?] alors
      ajouter 10 à x
  
```

F. Ajouter le code pour le lutin Apple

G. Ajouter le code sur le lutin Win pour afficher « You Win ! » et terminer le jeu

H. Ajouter le code dans la scène (sur l'arrière-plan) pour écouter un fond de musique qui se répète tant que le jeu est lancé.