

# Création d'un support d'apprentissage Boomwhackers sur Scratch

Valentin Leroux

Interlocuteur Académique au Numérique (IAN) en Éducation Musicale



Scratch est un logiciel libre disponible en ligne ou sous forme de logiciel (Windows/Mac) : <https://scratch.mit.edu>

Cet environnement de programmation se destine aux enfants ou aux débutants en programmation. Il est donc parfaitement adapté pour l'enseignement au collège et certaines épreuves du DNB de mathématiques s'appuient sur cette interface de programmation visuelle « par blocs ».

## Mise en route

Si vous utilisez la version logicielle (hors-ligne), cliquer sur l'icône  en bas à gauche de l'écran pour activer l'extension « Musique ». Cette extension est active par défaut dans la version en ligne.

Le planisphère  vous permet de modifier la langue du programme.

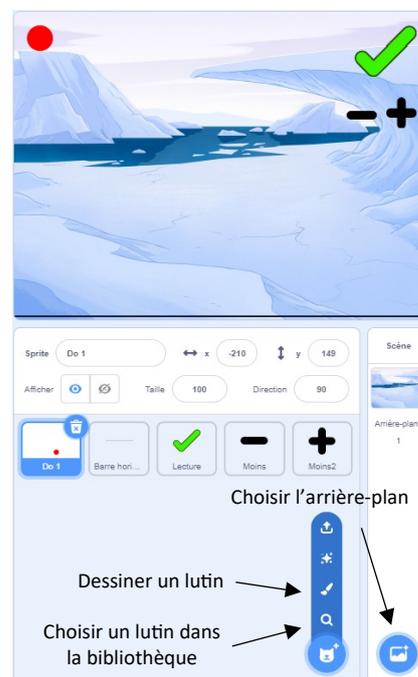
## Étape 1 : création des éléments graphiques

Depuis l'interface principale de Scratch, effacez le chat nommé « Sprite 1 » (ou « lutin » selon les versions) , puis importez ou créez les éléments graphiques suivants :

- Un petit cercle de la couleur de la note voulue.
- Un bouton qui servira à déclencher la lecture.
- Un bouton [+] et un bouton [-]
- *Facultatif* : une barre horizontale servant de repère en bas de l'écran.
- *Facultatif et beaucoup plus joli* : un arrière-plan

Placez ces éléments graphiques sur l'écran.

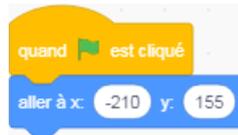
*Pour mieux vous y retrouver ensuite, il est possible de renommer les lutins. Ceci est particulièrement utile pour les notes de musique !*



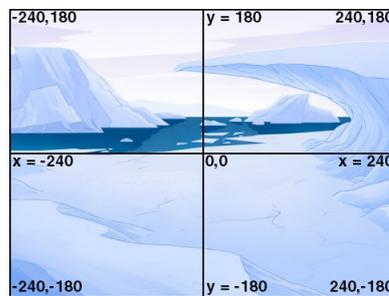
## Étape 2 : Animer une note

Avant de créer l'ensemble de la gamme, nous allons tout d'abord programmer notre première note.

- Dans la liste des lutins, sélectionnez le cercle correspondant à notre première note. Selon l'onglet en haut à gauche, nous pouvons en modifier le Code, les Costumes (l'apparence) ou des Sons (effets sonores, que nous n'utiliserons pas ici pour la musique). Pour le moment, seul le Code va nous intéresser.
- Le cercle devra toujours se trouver au même endroit de l'écran. Pour que celui-ci apparaisse en haut à gauche, nous pouvons lui donner les coordonnées  $x = -215$  et  $y = 155$ . Pour cela, naviguez dans les catégories Événements et Mouvement, puis faites glisser et assemblez dans la fenêtre de code les blocs suivants :



En cliquant sur le drapeau vert en haut à droite de Scratch :  , le cercle se place automatiquement au bon endroit. Déplacez le cercle à la main et cliquez à nouveau sur le drapeau vert : le cercle reviendra toujours à sa place.

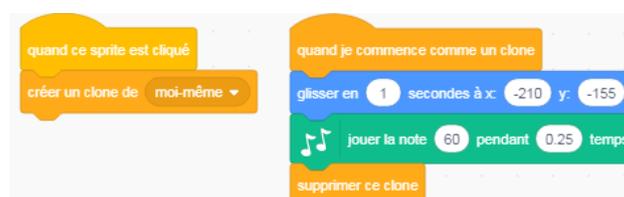


Le système de coordonnées de Scratch : un espace de 480x360 pixels.

- Nous allons ensuite définir le son de l'instrument qui nous intéressera. Parmi les instruments proposés dans Scratch, le son des cordes en Pizzicato est celui qui se rapproche le plus des Boomwhackers. Complétez la programmation existante avec le bloc de choix d'instruments, dans la catégorie Musique :



- Finalement, nous rajoutons deux nouveaux blocs de programmation pour notre cercle afin de réaliser 6 actions successives :
  - Détecter le clic
  - Créer un clone du cercle  
*puis, à côté :*
  - Détecter la création d'un clone
  - Faire glisser le clone au travers de l'écran : la valeur [x] reste identique mais [y] passe de 155 (tout en haut) à -155 (tout en bas)
  - Jouer la note voulue (ici, un do dont la hauteur vaut 60)
  - Disparaître



Lorsque vous cliquez sur le cercle, celui-ci crée un clone qui traverse l'écran de haut en bas, produit la note voulue et disparaît.

### Étape 3 : Créer l'ensemble des notes

- Depuis la liste des lutins, faites un clic-droit sur le cercle du do puis cliquez sur Dupliquer autant de fois que vous souhaitez créer de nouvelles notes.
- En basculant sur l'onglet Costumes, modifiez chaque lutin pour que la note jouée corresponde à sa couleur.
- En revenant sur l'onglet Code, modifiez le numéro de chaque note jouée. Par exemple 60 pour do, 62 pour ré, 64 pour mi, 65 pour fa, 67 pour sol, etc.
- Toujours dans l'onglet code, changez également l'emplacement initial de chaque cercle sur l'axe des abscisses [x]. Par exemple, - 210 pour do, -170 pour ré, -130 pour mi, -90 pour fa, -50 pour sol etc.



### Étape 4 : Programmer une mélodie

Il est très simple de programmer une suite de notes, qui sera jouée automatiquement lors d'un clic sur le lutin Lecture (✓) créé au départ. Pour cela, sélectionnez le lutin dans la liste puis dans l'onglet code, créez des clones des notes voulues espacées par des pauses. L'ensemble se déclenche lorsque l'utilisateur cliquera sur le lutin.



*Il est possible de déclencher plusieurs notes simultanément en les plaçant immédiatement l'une après l'autre.*

### Étape 5 : Gérer le tempo

Pour le moment, le tempo est toujours le même (60 BPM par défaut). Cochez la case  tempo dans la catégorie musique pour faire apparaître celui-ci à l'écran.

Programmer les lutins « moins » et « plus » de telle sorte que l'utilisateur puisse modifier le tempo facilement :



Lutin « moins »

Lutin « plus »

## Étape 6 : Organiser la forme musicale

Lorsqu'une musique comporte plusieurs parties, il est judicieux d'optimiser les suites de notes programmées pour éviter d'avoir à se repérer dans de trop longues suites d'instructions.

Prenons l'exemple d'une comptine, dans laquelle une même séquence mélodique est entendue trois fois (structure A-A-B-A).

The image shows two staves of musical notation. The first staff is labeled 'PARTIE A' and contains the melody for the first line of the song: 'Au clair de la lune mon ami Pierrot Prête-moi ta plume pour écrire un mot'. The second staff is labeled 'PARTIE B' and contains the melody for the second line: 'Ma chandelle est morte je n'ai plus de feu Ouvre-moi ta porte pour l'amour de Dieu'. The melody for 'PARTIE A' is repeated in the second staff.

Il est possible d'optimiser le code de la façon suivante, en utilisant le système de messages (rubrique Événements) :

The image displays a Scratch script designed to play the song sequence. It starts with a 'when clicked' event block. The script is organized into three columns, each representing a different part of the song. The first column contains three 'send to all' blocks, each followed by an 'and wait' block, to trigger the three instances of Part A. The second and third columns contain the specific musical notes and rests for Part A and Part B, respectively, using 'create clone of' blocks followed by 'wait' blocks. The notes are: Do 1, Ré 1, Mi 1, Ré 1, Do 1, Mi 1, Ré 1, Do 1, Ré 1, Do 1 for Part A; and Ré 1, Ré 1, La 0, La 0, Ré 1, La 0, Si 0, La 0, Sol 0 for Part B. The rests are 0.25 and 0.5 seconds.

*Le bloc « envoyer à tous [nom de la partie] et attendre » permet de ne pas déclencher simultanément les différentes séquences. La deuxième séquence ne s'active que si la première est terminée.*