

# OUTIL - GUIDE DE PRÉSENTATION DE SCRATCH

CRÉATION DE CONTENU > 3.4 PROGRAMMATION

CONVIENT POUR	AGE	NIVEAU DE COMPÉTENCE	FORMAT	DROITS D'AUTEUR	LANGUE(S)
Animateurs	N/A	Niveau 2	Fiche de préparation	Creative Commons (BY-SA)	Français

Cette fiche est faite pour présenter Scratch et son interface de création. Elle détaille les outils de l'interface, les blocs, les interfaces dessins et sons.

**Objectif général** Connaissances

**Temps de préparation pour l'animateur** moins d'une 1 heure

**Domaine de compétence** 3 - Création de contenu

**Nom de l'auteur** Samantha Giordano

**Ressource originellement créée** Français

## DÉROULÉ

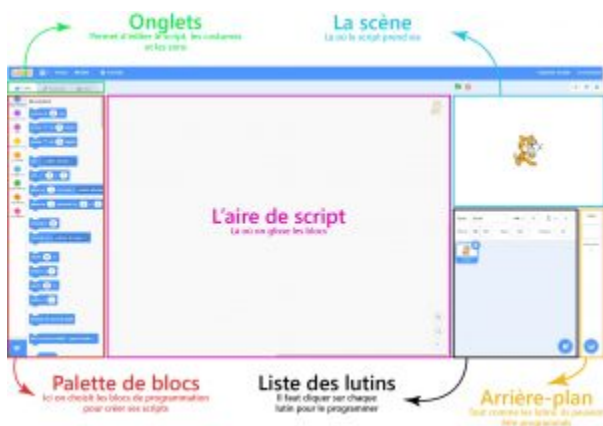
### 1 Un peu d'histoire pour le contexte

En 2006, le MIT développe un nouveau langage informatique : **Scratch**. Le but étant de rendre la programmation accessible dès 8 ans. Le logiciel est gratuit et est traduit en une vingtaine de langues. Les créations réalisées et partagées sur le site sont en « creative commons », ce qui permet de modifier les créations des autres utilisateur.ice.s.

Le langage se présente sous forme de **blocs** qu'il faut assembler pour former des suites de lignes de code. Il s'agit d'une version visuelle de Smalltalk, un langage de programmation orienté objet, qui a beaucoup influencé le développement d'autres langages tels que Java ou Ruby. Scratch est donc dynamique : il permet de modifier le code et ce, directement en cours d'exécution. Son utilisation est ludique, il suffit de manipuler les briques et d'observer immédiatement le résultat.

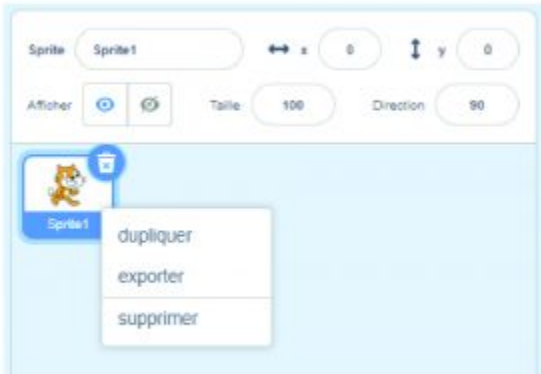
Plus concrètement, Scratch est un véritable langage car aborde les concepts de base de la **programmation** tels que les boucles, les conditions, les variables, etc. Il est idéal pour expliquer l'algorithme aux petit.e.s et grand.e.s et ce, de façon très accessible. C'est un outil efficace pour l'initiation de chacun.e. S'il est possible de créer des jeux ou des animations extrêmement simples, il est aussi possible d'aller plus loin et de créer des jeux complexes

### 2 L'interface globale

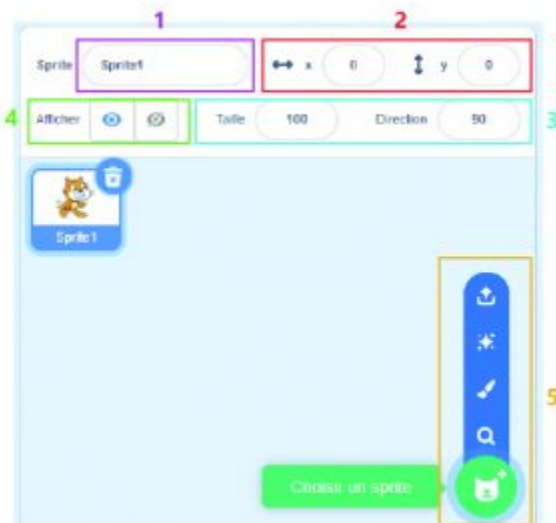


## 3 Les lutins

Les lutins (appelés également « Sprite ») sont les éléments que vous allez pouvoir programmer **individuellement** en glissant les blocs de programmation dans l'aire de script.  
Attention ! Avant d'écrire votre code, assurez vous d'avoir sélectionné le bon lutin.



En effectuant un clic droit sur l'icône représentant le lutin, un menu déroulant s'ouvre. Vous pourrez alors dupliquer, exporter ou supprimer votre lutin.



Voici les différents paramètres accessibles depuis l'encadré destiné à la gestion des lutins :

1 : modifier le nom

2 : modifier l'emplacement du lutin sur la scène en modifiant les abscisses et les ordonnées (vous pouvez également le faire en déplacement le lutin directement sur la scène à l'aide de votre souris).

3 : changer la taille et la direction du lutin. En cliquant sur la direction, vous pourrez accéder à des paramètres supplémentaires qui vous permettront de choisir entre 3 style de rotations (de gauche à droite) : le lutin pivote à 360° lors de ses mouvements, le lutin ne pivote que vers la gauche ou vers la droite, le lutin ne pivote pas.

4 : choisir de rendre visible ou non le lutin sur la scène

5 : ajouter un lutin. Il existe différentes façons de créer un lutin : on peut le choisir dans la bibliothèque de Scratch, le dessiner ou l'importer.

Il est possible de personnaliser chaque lutin grâce à l'onglet « costume », disposé en haut à gauche de l'écran. Les costumes permettent de définir l'apparence d'un lutin, et il est possible pour un lutin d'avoir plusieurs costumes. Cela permet de créer des animations et des effets visuels en passant d'un costume à l'autre (à réaliser grâce aux blocs de programmation).

Voici les divers espaces qui sont à votre disposition :



1 : l'onglet costume

2 : l'espace de personnalisation de votre lutin. Vous pouvez changer sa forme, sa couleur, ajouter des éléments...

3 : créer ou ajouter un nouveau costume en allant chercher dans la bibliothèque, en le dessinant, ou en l'important depuis son ordinateur.

4 : visualiser les différents costumes associés à un lutin. En faisant un clic droit sur un costume vous pourrez le dupliquer, l'exporter ou le supprimer

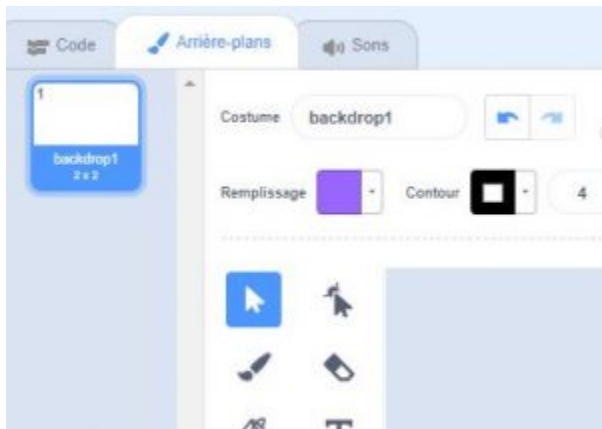
## 4

## Les arrières plans

Scratch permet d'ajouter des arrière plans à vos scènes et de les programmer de la même manière que les lutins. Pour cela, il suffit de sélectionner votre arrière plan, et de glisser les blocs de programmation de votre choix dans l'aire de script.

Le logiciel scratch choisit d'office un arrière plan blanc, mais il est possible de changer celui-ci en allant piocher dans la bibliothèque de Scratch, de le dessiner ou de l'importer depuis son ordinateur.

De la même manière que pour les lutins, il est possible d'avoir plusieurs arrière plans afin de créer des effets d'animation. Dans la vignette qui apparaît en bas à droite de l'écran vous ne verrez en revanche, que l'arrière plan qui est actuellement visible dans votre scène. Pour voir l'ensemble de votre arrière plan et en ajouter de nouveaux, il faut vous rendre dans l'onglet « arrière plan » (en haut à gauche de l'écran). Dans cet onglet, comme pour celui des « costumes » (voir les détails plus haut), vous pourrez modifier votre arrière plan et en ajouter des nouveaux.











## 5 Les différents types de blocs

Chaque bloc est classé dans différentes catégories très repérables par les couleurs.



En parcourant ces différentes sections, vous vous rendez compte que tous les blocs n'ont pas la même forme. Avant de rentrer dans les détails, définissons ces formes.

On peut distinguer 5 types de blocs :

	<p>Blocs instructions. Ils définissent les actions du lutin.</p>
	<p>Blocs de données. Ils renvoient à la variable de l'élément choisi. Par exemple, si on examine cette instruction :</p>  <p>Le code renvoie à la variable abscisse "x" qui est la position horizontale du lutin et retourne sa position.</p> 
	<p>Blocs chapeaux/entrées. Ils sont le point de départ du script. Ils sont les événements qui déclenchent le début d'une action.</p>
	<p>Blocs de structure de contrôle. Ces blocs influent sur le déroulement des instructions (ceux-ci sont identifiables par leur couleur).</p>
	<p>Blocs booléens. Ils indiquent quelle condition est testée dans les blocs structure de conditions (cel qui est ci-dessus). Le résultat renvoyé par ces blocs est soit vrai, soit faux. On construit le script ainsi :</p>  <p>Ainsi, lorsque le drapeau vert est cliqué, si le lutin touche un bord de la scène, la condition est vraie et il avance de 10. Si en revanche il ne touche pas le bord de la scène, la condition est fautive, il n'avance pas.</p>

## 6 Les blocs "mouvement"

	Déplace le lutin d'un certain nombre de pas. Le lutin se déplace en avant ou en arrière (ou vers le haut, vers le bas, de gauche à droite) suivant la direction où il est positionné.
	Fait tourner le lutin vers la droite d'un nombre de degrés à définir.
	Fait tourner le lutin vers la gauche d'un nombre de degrés à définir.
	Le lutin se déplace sur l'emplacement d'un autre lutin ou sur le curseur de la souris (choix multiples dans le menu déroulant).
	Le lutin se déplace sur une position à définir sur la scène.
	Le lutin se déplace jusqu'à un point à définir en un temps à définir.
	Le lutin se déplace jusqu'à un point à définir sur l'abscisse x et l'ordonnée y en un temps à définir.
	Définit la direction du lutin: 90° droite, -90° gauche, 0° haut, 180° bas.
	Orienté le lutin vers un autre lutin ou vers le pointeur de la souris (choix multiples dans le menu déroulant).
	Le lutin se déplace d'un nombre à définir sur l'abscisse x. Un nombre positif déplace le lutin vers la droite, un nombre négatif déplace le lutin vers la gauche.
	Le lutin se déplace sur un emplacement à définir sur l'abscisse x.
	Le lutin se déplace d'un nombre à définir sur l'ordonnée y. Un nombre positif déplace le lutin vers le haut, un nombre négatif déplace le lutin vers le bas.
	Le lutin se déplace sur un emplacement à définir sur l'ordonnée y.

	Si le lutin atteint le bord de la scène, il rebondit.
	Définit le style de rotation du lutin. Il est possible de le fixer dans les infos du lutin, mais il est aussi possible de le modifier au cours de l'animation.
<input type="checkbox"/>	Fournit la valeur de la position x du lutin.
<input type="checkbox"/>	Fournit la valeur de la position y du lutin.
<input type="checkbox"/>	Fournit la valeur de la direction actuelle du lutin.

## 7 Les blocs "apparence"

	Le lutin affiche le message choisi dans une bulle de pensée pendant un temps à définir avant de disparaître
	Le lutin affiche le message choisi dans une bulle de pensée
	Le lutin affiche le message choisi dans une bulle de pensée pendant un temps à définir
	Le lutin affiche le message choisi dans une bulle de pensée
	Bouton sur un attribut choisi (choix multiples dans le menu déroulant)
	Bouton sur la couleur choisie dans la liste des couleurs. Arrivé au dernier couleur, la liste recommence au début.
	Bouton vers l'arrière jusqu'à un point dans la liste des arrière plans (choix multiples dans le menu déroulant)
	Bouton vers l'avant jusqu'à un point dans la liste des arrière plans
	Ajoute le lutin à une liste de lutins spécifiés. Un nombre négatif le fera retirer.
	Permet d'agrandir ou de réduire le lutin. 100% étant sa taille initiale.
	Permet d'ajouter un effet au lutin. Le déroulant permet plusieurs choix couleur, taille position, bouger, glisser, pousser, tourner, faire tourner. Plus le chiffre associé est élevé, plus l'effet est marqué. (choix multiples dans le menu déroulant)
	Lait l'effet graphique du lutin à une valeur spécifiée. Les effets sont les mêmes que ceux évoqués précédemment (choix multiples dans le menu déroulant)

## 8 Les blocs "son"









	Joue le son choisi jusqu'au bout avant de continuer la suite de l'animation. (choix multiples dans le menu déroulant)
	Joue le son choisi sans arrêter la suite de l'animation (choix multiples dans le menu déroulant)
	Arrête tous les sons en cours.
	Joue le son choisi en modifiant la hauteur du son : un nombre négatif jouera un son plus grave, un son positif jouera un son plus aiguë (choix multiples dans le menu déroulant)
	Permet de choisir la hauteur, plus grave ou plus aiguë, du son choisi (choix multiples dans le menu déroulant)
	Annule tous les effets sonores de l'animation
	Permet de modifier le volume du lutin. Le volume est de 100 par défaut. Un nombre négatif réduit le volume, un nombre positif l'augmente.
	Définit le volume sonore du lutin. 100 % étant le volume par défaut.
	Fournit la valeur du volume actuel du lutin



9

## Les blocs "événements"

Les blocs événements définissent l'action qui déclenchera l'animation des lutins. Cette action peut être différente pour chaque lutin.

	Exécute le script du lutin lorsque le drapeau vert au dessus de la scène est cliqué
	Exécute le script du lutin lorsque la touche "espace" est cliquée. (choix multiples dans le menu déroulant)
	Exécute le script du lutin lorsque celui-ci est cliqué.
	Exécute le script lorsque l'arrière-plan change pour un autre défini.
	Exécute le script lorsque le "volume sonore", le "chronomètre" ou le "mouvement vidéo" (ou choix) est supérieur à une valeur à définir. (choix multiples dans le menu déroulant)
	Exécute le script lorsque le lutin reçoit un message à définir. (choix multiples dans le menu déroulant)
	Envoie à tous les autres lutins un message à spécifier. (choix multiples dans le menu déroulant)
	Le lutin envoie à tous les autres lutins un message à spécifier et attend qu'ils aient fini les animations liées à cet événement avant de continuer sa propre animation. (choix multiples dans le menu déroulant)

Pour le bloc « envoyer à tous et attendre » : définir un message permet de synchroniser les éléments (lutins et/ou arrière-plans) entre eux. Par exemple, un premier lutin peut, au cours de son animation, envoyer un message à un second lutin qui ne commencera son animation que lorsqu'il aura reçu ce message. Si on commande au premier lutin d'« envoyer à tous message1 », il enverra son message tout en continuant son animation jusqu'au bout. En revanche, si on commande au premier lutin d'« envoyer à tous message1 et attendre », il enverra son message et attendra que le second lutin finisse l'animation liée au message avant de continuer la sienne.

10

## Les blocs "contrôle"

	Le lutin attend 1 seconde avant de poursuivre le script.
	Le script compris dans la boucle est répété un nombre défini de fois avant de continuer.
	Le script compris dans la boucle est répété indéfiniment.
	Si la condition déterminée est vérifiée, le lutin exécute le script compris dans la boucle.
	Si la condition déterminée est remplie, le lutin exécute le script compris dans la partie supérieure de la condition. Si la condition n'est pas vérifiée ("sinon"), le lutin exécute le script compris dans la partie inférieure.
	Le lutin attend jusqu'à ce qu'une condition (à définir) soit vraie avant de continuer son script.
	Le script compris dans la boucle s'exécute tant qu'une condition (à définir) n'est pas vraie.
	Stop l'exécution soit pour tous les lutins, soit pour le script en cours, soit pour un autre script du lutin.
	Indique aux clones du lutin ce qu'ils doivent faire une fois créés.
	Le lutin crée un clone de lui-même ou d'un autre lutin (sous multiples dans la menu déroulant)
	Supprimer le clone.

## 11 Les blocs "capteurs"

	Si un élément spécifique (bouton, pointeur souris ou bord) est touché, le résultat sera vrai, sinon il sera faux (choix multiples dans le menu déroulant)
	Si une couleur spécifique est touchée, le résultat sera vrai, sinon il sera faux.
	Si une couleur spécifique touche une autre couleur spécifique, alors le résultat sera vrai, sinon il sera faux.
	Fournit la valeur de la distance entre le lutin et un autre lutin ou le pointeur de la souris (choix multiples dans le menu déroulant)
	Le lutin pose une question et ouvre une boîte de dialogue dans laquelle il est possible de répondre
	Fournit la valeur de la réponse donnée dans la boîte de dialogue
	Si la touche définie est pressée, alors le résultat sera vrai, sinon il sera faux (choix multiples dans le menu déroulant)
	Si la souris est pressée, alors le résultat sera vrai, sinon il sera faux.
	Fournit la valeur de la position de la souris sur l'axe des x (horizontale)

	Fournit la valeur de la position de la souris sur l'ordonnée y (verticale)
	Fournit la valeur du volume sonore capté par les micros.
	Fournit la valeur du chronomètre
	Réinitialise le chronomètre
	Fournit la valeur d'un paramètre du lutin ou de la scène. (choix multiples dans le menu déroulant)
	Fournit la valeur de l'heure ou la date actuelle. (choix multiples dans le menu déroulant)
	Fournit la valeur du nombre de jours depuis l'an 2000
	Fournit la valeur du nom d'utilisateur.

- Le bloc « volume sonore » : l'utilisation de ce bloc nécessite un ordinateur équipé d'un microphone. Il permet de détecter le son capté par celui-ci. En cela, il est différent du bloc « volume » qui fournit la valeur du son du lutin
- Le bloc « vidéo mouvement sur ce lien » : Il est nécessaire d'activer la webcam au préalable, à l'aide du bloc « activer la vidéo ». Le bloc permet à la webcam de vérifier le nombre de mouvements d'un lutin ou de l'ensemble de la scène, ou bien la direction du mouvement d'un lutin ou de l'ensemble de la scène.
- Le bloc « chronomètre » : Le chronomètre est toujours en cours d'exécution. Si on veut qu'au début de l'action (ou même au cours de l'action) le chronomètre parte de zéro, il faut bien penser à mettre le bloc « réinitialiser le chronomètre ». Le chronomètre ne s'arrêtera jamais, mais il repartira alors de zéro.

## 12 Les blocs "opérateurs"

Les opérateurs servent au calcul. Il est à noter qu'un chiffre pouvant être négatif ou positif, il est important de prendre en compte l'ordre des nombres dans les opérations. On trouve dans les opérateurs soit des nombres, soit des chaînes de caractères (qui sont des mots ou des suites de mots, des espaces et des caractères spéciaux).

	Additionne deux nombres
	Soustrait le second nombre du premier.
	Multiplie deux nombres.
	Divise le premier nombre par le second.
	Fournit la valeur d'un nombre aléatoire compris entre deux valeurs.
	Si le premier nombre est supérieur au second, le résultat est vrai, sinon il est faux.
	Si le premier nombre est inférieur au second, le résultat est vrai, sinon il est faux.
	Si le premier nombre est égal au second, le résultat est vrai, sinon il est faux.
	Si les deux conditions sont remplies, la condition est vraie.
	Si l'une des deux conditions est remplie, la condition est vraie.

	Si la condition n'est pas remplie, le résultat est vrai.
	Permet de regrouper deux chaînes de caractères. Par exemple, un mot et un élément d'une liste (à l'aide d'un bloc "renvoi")
	Fournit la valeur de la lettre désignée dans une chaîne spécifiée
	Fournit la valeur de la longueur de la chaîne spécifiée, c'est-à-dire le nombre de caractères de la chaîne
	*Fournit la valeur du reste de la division entière du premier nombre par le second nombre.
	Fournit la valeur du nombre entier le plus proche du nombre spécifié (l'arrondi de 7,89 est 8).

## 13 Les blocs "variables"

### Créer une variable

Une variable est un espace de mémoire dans lequel l'ordinateur peut stocker des données. Par exemple, le score d'un jeu est une variable: on stocke une valeur de base et elle change au cours de la partie

selon le calcul des bonus et malus. Pour créer une variable, il suffit de cliquer sur l'onglet "créer une variable". Cette fenêtre s'ouvre alors :

Nous appellerons, pour l'exemple, notre variable "score". Il faut ensuite décider si cette variable ne s'appliquera qu'à ce seul lutin ou bien également à tous les autres. Si elle ne s'applique qu'au seul lutin sélectionné, alors elle fonctionne indépendamment des autres lutins créés. Si elle s'applique à tous les lutins, alors elle peut varier en fonction des actions de chacun des lutins créés. Toutefois, si vous décidez en cours de route de modifier cette variable pour un seul lutin, cela ne fonctionnera pas.

	Fournit la valeur de la variable "score"
	Porte la variable à une valeur à définir. (choix multiples dans le menu déroulant)
	Ajoute à la variable une valeur à définir. (choix multiples dans le menu déroulant)
	Montre sur la scène la variable "score" ainsi que sa valeur. (choix multiples dans le menu déroulant)
	Cache sur la scène la variable "score" et sa valeur. (choix multiples dans le menu déroulant)

## Créer une liste

Une liste est une structure de données permettant de regrouper, d'organiser et de traiter des informations plus simplement. Par exemple, il est possible de créer une liste de mots que mon lutin affichera de manière aléatoire. Pour créer une nouvelle liste, il suffit de cliquer sur "créer une liste". On peut ensuite paramétrer la liste "pour tous les lutins", ce qui signifie qu'elle s'appliquera à tous les lutins créés. Ou bien on peut paramétrer la liste "pour ce lutin uniquement", ce qui signifie qu'elle s'appliquera seulement au lutin sélectionné. Pour l'exemple, nous créerons une liste de course.



## 14 Pour aller plus loin

N'hésitez pas à vous rendre sur le site de Scratch, vous y trouverez de nombreuses ressources :

- [Des projets publiés](#)
- [Des tutoriels](#)
- [Des cartes à imprimer pour vous guider dans vos créations](#)
- [Des idées pour animer des ateliers \(en anglais\)](#)

Et bien sur, les fiches d'activités scratch disponibles sur notre site !