

# Introduction au logiciel Scratch

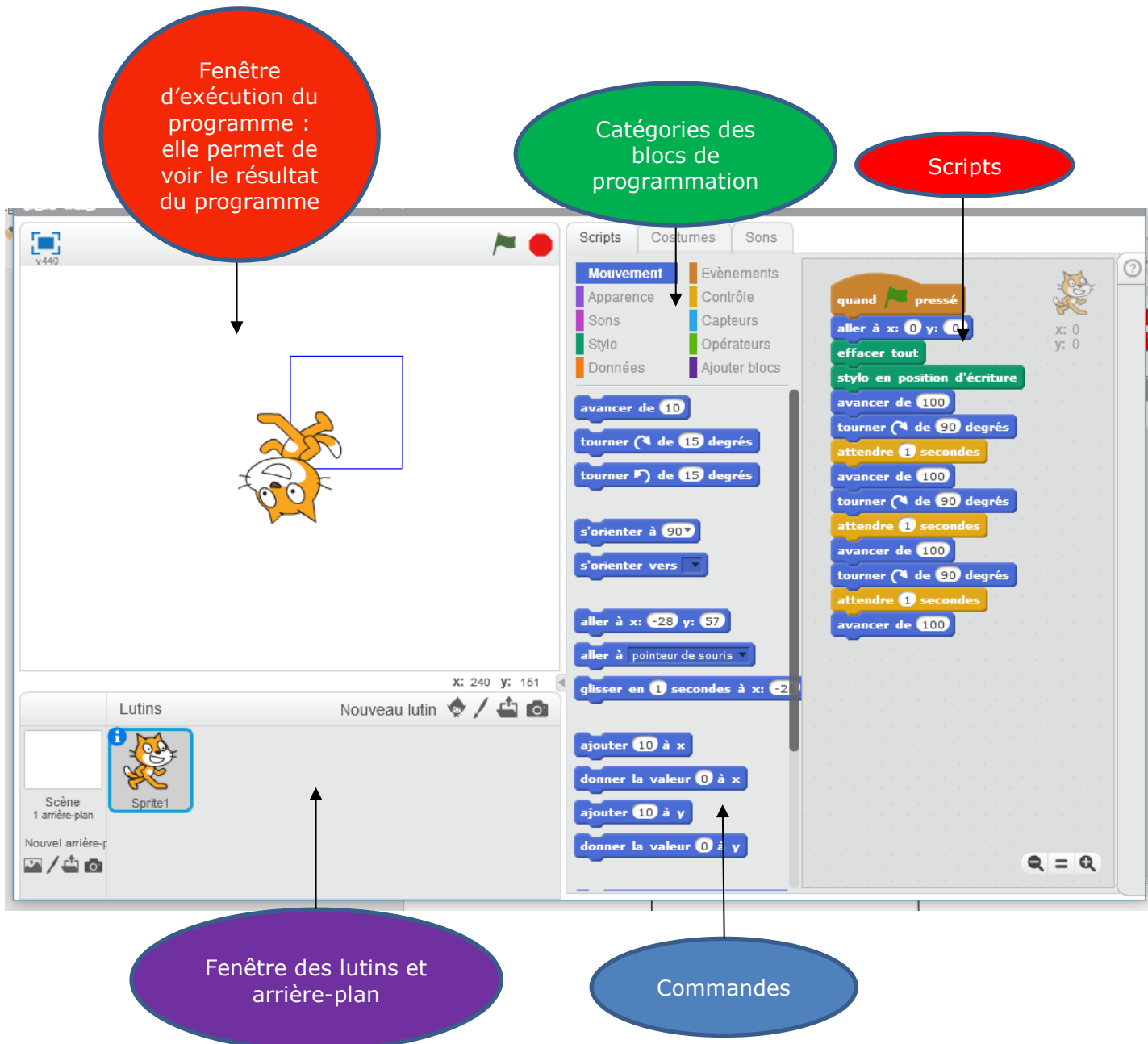
## I) Qu'est-ce que Scratch ?

- Scratch est un logiciel de programmation visuelle qui permet de faire exécuter, par un ordinateur, certains algorithmes.
- Coder avec Scratch est beaucoup plus simple que coder avec un vrai langage de programmation traditionnel, puisque la création de scripts est réalisée à partir d'instructions simple se présentant sous la forme d'un assemblage de blocs.

Scratch est une implémentation visuelle et dynamique du langage de programmation Smalltalk basé sur Squeak. Scratch est un projet du groupe Lifelong Kindergarten au Media Lab du MIT. Il est disponible gratuitement.

## II) Description du logiciel

### 1) Description générale





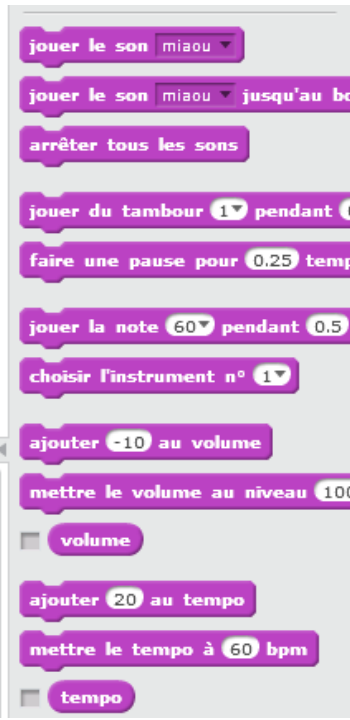
Le drapeau vert permet de démarrer le programme

Le signe stop arrête tous les scripts






## 2) Description des principaux menus

Mouvement	Apparence	Stylo
<p>Dans ce menu, on trouve toutes les manières de déplacer le lutin. Il se déplace dans un repère allant de -240 à 240 en abscisse et de -180 à 180 en ordonnée.</p>	<p>Ce menu permet de modifier l'apparence du lutin, sa taille, écrire un message ou modifier l'arrière-plan</p>	<p>Dans ce menu se trouvent les instructions pour voir la trace d'une partie ou de tous les déplacements du lutin</p>


Evènements	Contrôle	Sons
<p>Dans ce menu, les instructions servent à démarrer les actions du programme. On peut commencer une action à l'aide du clavier ou de la souris</p>	<p>Ce menu permet de répéter une action un certain nombre de fois, ou indéfiniment. (Si on veut effectuer des répétitions sans jamais s'arrêter ou jusqu'à ce qu'une condition soit vérifiée...)</p> <p>On utilise les blocs du type « Si ...alors.... » lorsque le problème impose de dissocier différents cas dans sa résolution.</p> <p>Ces types de programme se verront en 4°</p>	<p>Dans ce menu, on trouve tous les types de sons que l'on peut intégrer dans notre programme</p>
		

### III) Programmer avec Scratch

#### 1) Le drapeau vert

Le drapeau vert permet de lancer de nombreux scripts en même temps. Si on ajoute la commande, au début du script  alors Il suffit de cliquer sur  en haut de la zone « exécutions des scripts » pour exécuter les scripts et sur  pour l'arrêter.

#### 2) Changer le lutin

Cliquez sur  pour changer le lutin qui se trouve dans la fenêtre des lutins.

Ne pas oublier de cliquer sur OK pour valider votre choix.

Supprimer ensuite le lutin « sprite 1 » à l'aide du clic droit.

#### 3) Exemple 1 : Les déplacements

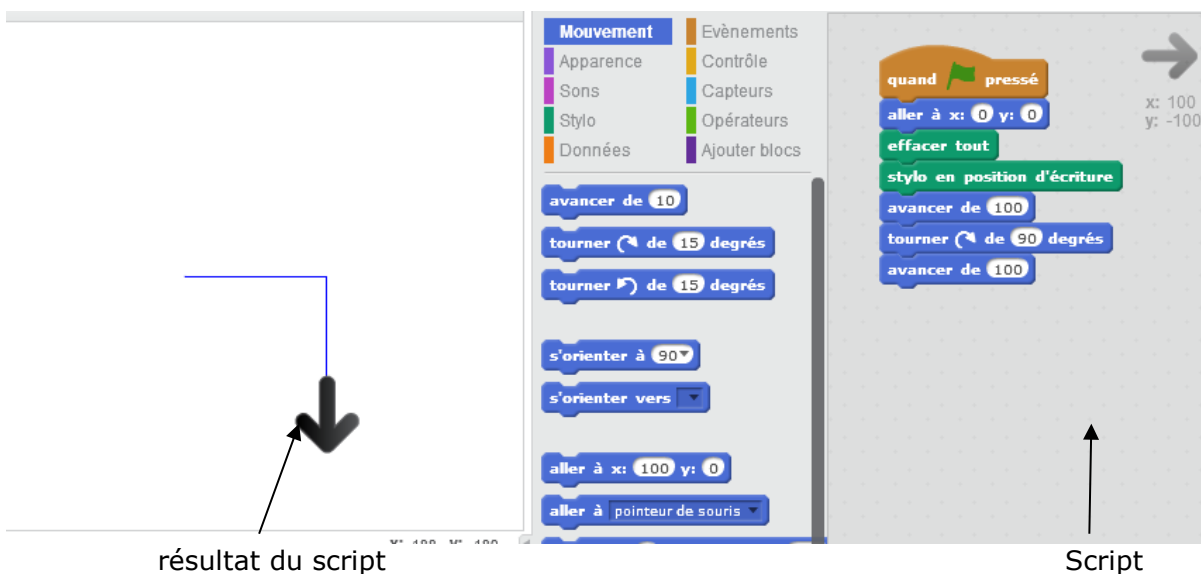
- On modifie le lutin en  « Arrow2 »

- On supprime le lutin chat « sprite 1 »

- On va reproduire l'algorithme suivant :

En partant de l'origine du repère (  $x = 0$  et  $y = 0$  )  
Avancer de 100  
Tourner dans le sens  $\curvearrowright$  de  $90^\circ$   
Avancer de 100

Si on ne veut voir apparaître que les segments sans la flèche, il suffit de cacher le lutin (exemple 2)



The screenshot shows the Scratch interface. On the left, a stage displays the result of a script: a black arrow pointing down, with a blue line indicating its path from the origin (0,0) to (0,100) and then to (100,100). An arrow points to this result with the label "résultat du script". In the center, the "Scripts" palette is visible, showing a script with the following blocks: "when green flag clicked", "go to x: 0 y: 0", "clear", "set pen style to drawing", "move 100", "turn 90 degrees clockwise", and "move 100". On the right, the "Sprite" palette shows the "Arrow2" sprite selected, with an arrow pointing to it and the label "Script".



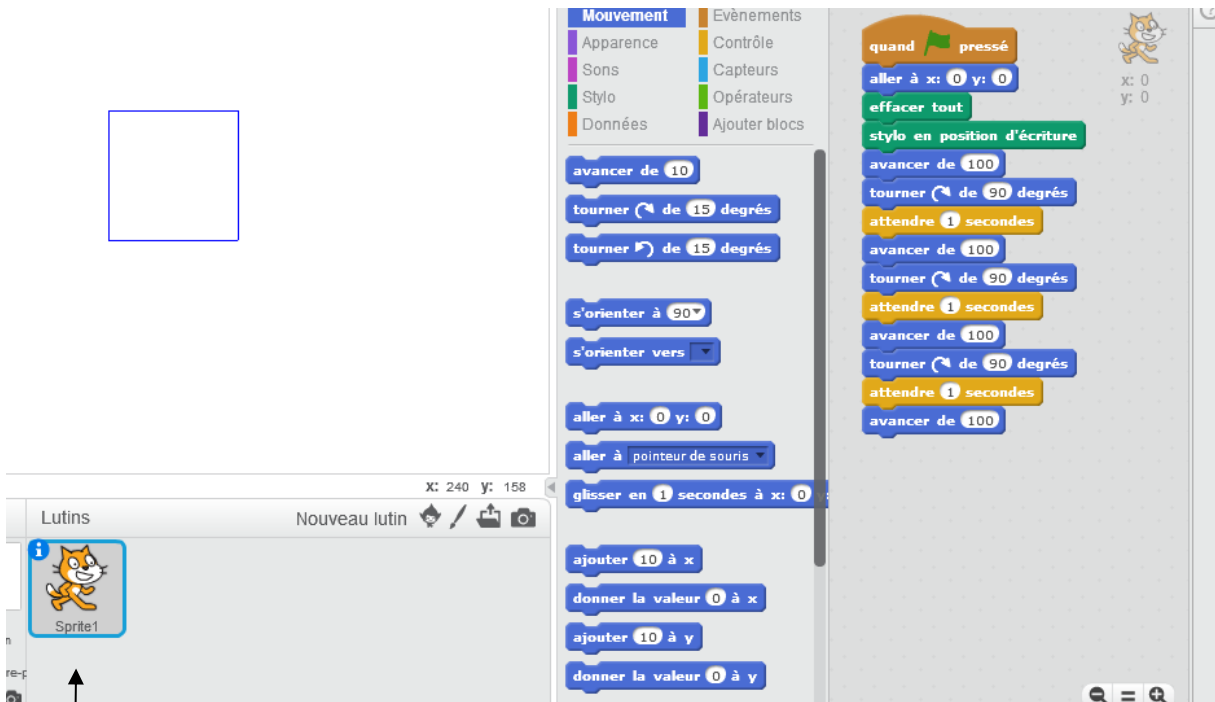
On se positionne sur l'origine du repère

A chaque fois que l'on va cliquer sur le drapeau vert, le lutin part de l'origine du repère et les anciens traits sont effacés

Pour tracer les segments

Script qui correspond à l'algorithme ci-dessus

#### 4) Exemple 2 : Dessiner un carré



Avec le clique droit, je cache le lutin afin de ne voir que la construction géométrique

#### 5) Création de variables

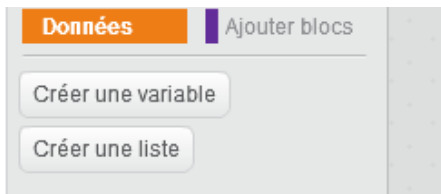
##### Exemple 3 : Faire une division Euclidienne

L'objectif de notre programme est de nous donner le reste de la division euclidienne de 120 par 36.

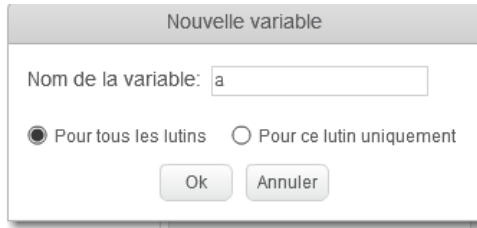
Pour cela nous devons créer 3 variables : « **a** » qui est le dividende donc **a = 120**

« **b** » qui est le diviseur donc **b = 36** et le reste **r** qui est le reste de la division de a par b

Pour créer une variable on utilise le bloc :



Dans ce bloc il n'y a que deux éléments : Créer une variable ou créer une liste. On clique dans « créer une variable »

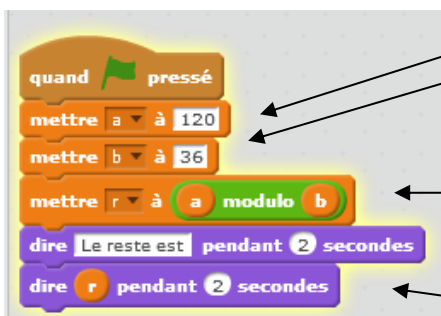


Il s'affiche « nom de la variable : » on y entre la 1<sup>ère</sup> variable **a** puis on clique sur OK. On fait de même avec b et r

Dans le bloc « données » il s'affiche :



Maintenant on peut écrire notre programme permettant d'effectuer cette division euclidienne:



a = 120  
b = 36

« **a modulo b** » donne directement le reste de la division euclidienne de a par b

Pendant 2 secondes le lutin dit « Le reste est » puis il affiche le reste pendant 2 secondes

pendant 2 secondes :



puis pendant 2 secondes :

