

Initiation à la programmation en utilisant le logiciel SCRATCH

Le logiciel peut être téléchargé par exemple depuis le site :

<https://scratch.mit.edu/download>

A la première exécution du logiciel, commencer par choisir la langue « Français » dans le menu associé à la mappemonde en haut à gauche.

Comme nous aurons besoin de dessiner, ajouter l'extension STYLO en cliquant

en bas à gauche sur



INITIATION A LA PROGRAMMATION EN UTILISANT LE LOGICIEL SCRATCH..	1
A. PREMIERE SEANCE	2
1. SE FAMILIARISER AVEC L'INTERFACE	2
2. DESSINER UN CARRE.....	2
3. DESSINER PLUSIEURS CARRES DE TAILLES DIFFERENTES	3
4. FAIRE UN PROGRAMME QUI DESSINE AUTOMATIQUEMENT 10 CARRES EMBOITES.....	4
5. DEFINIR UN NOUVEAU BLOC APPELE « CARRE »	6
6. ECRIRE NOTRE PREMIER PROGRAMME COMPLET	8
7. TOURNER, OUI MAIS DE QUEL ANGLE ?	10
8. DESSINER UN HEXAGONE, PUIS PLUSIEURS HEXAGONES	11
9. DESSINER DES CERCLES	13
B. DEUXIEME SEANCE : ANIMER UNE HISTOIRE A PLUSIEURS PERSONNAGES.....	13

1.	FAIRE MARCHER LE CHAT	14
2.	FAIRE MARCHER UN DEUXIEME PERSONNAGE.....	15
3.	CHANGER DE DECOR.....	17
C.	TROISIEME SEANCE : PROGRAMMER UN PETIT JEU	20

A. Première séance

1. Se familiariser avec l'interface

Le chat (dont on peut réduire la dimension). Le « **Code** ».

Se concentrer sur le « **Stylo** » qui permet de dessiner des figures.
En **position d'écriture**, essayer « **Mouvement** » : **avancer, tourner** ...

Pour effacer et placer le chat au centre : **effacer tout, aller à x : 0 y : 0**,

Pour changer l'orientation du chat : **s'orienter à ...**,

Pour cacher, montrer le chat, dans « **Apparence** » : **cacher, montrer**.

Le reste de cette séance permet d'aborder des notions de géométrie : polygones, angles, ainsi que des algorithmes simples : répétition, variables, compteurs.

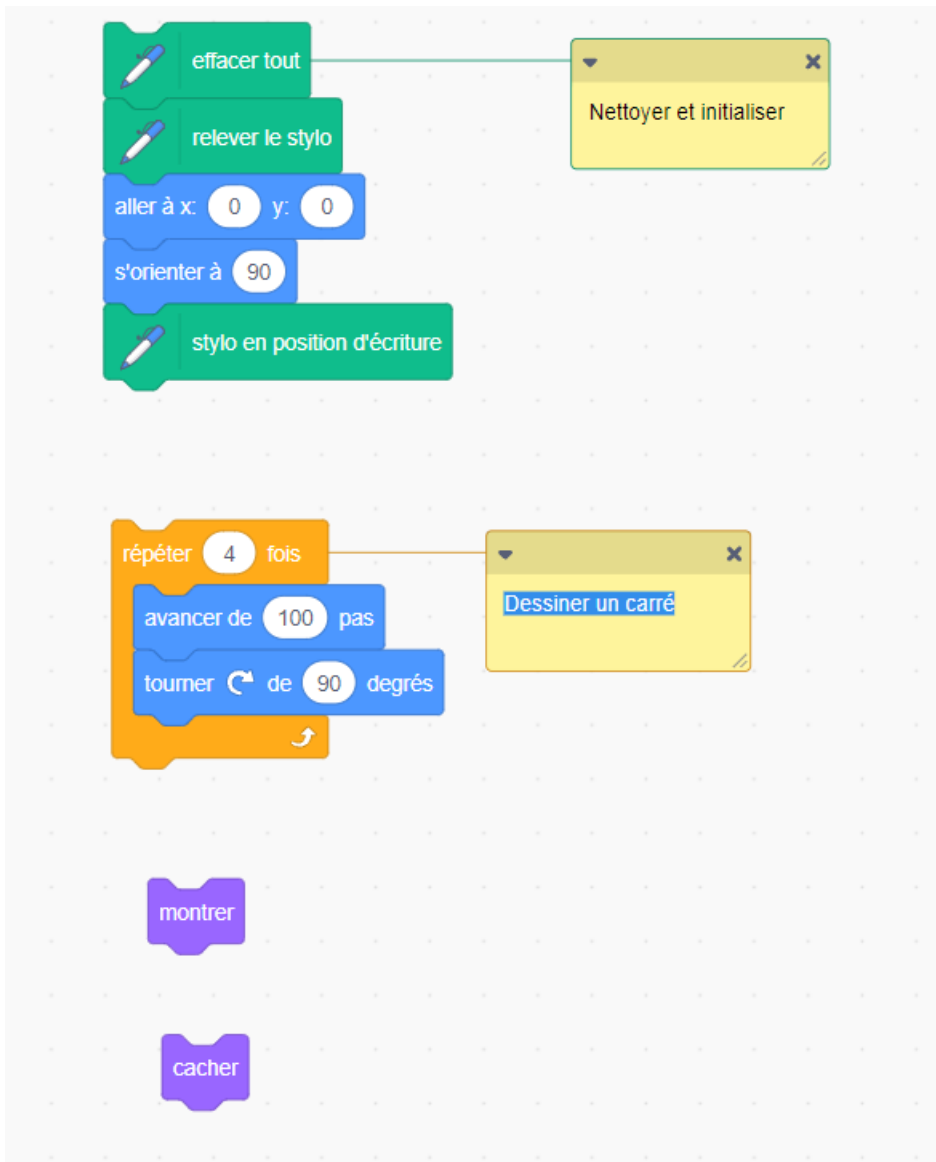
Introduit aussi la notion de programme avec paramètres.

2. Dessiner un carré

Laisser les enfants trouver une solution. Puis leur montrer le menu « **Contrôle** ».

Comment peut-on utiliser l'instruction : **répéter** ... ?

Les blocs de commandes s'agrègent pour former de petits programmes.



Remarque :

- Dans la section « **Apparence** » les 2 instructions de montrer ou cacher le chat



permettent

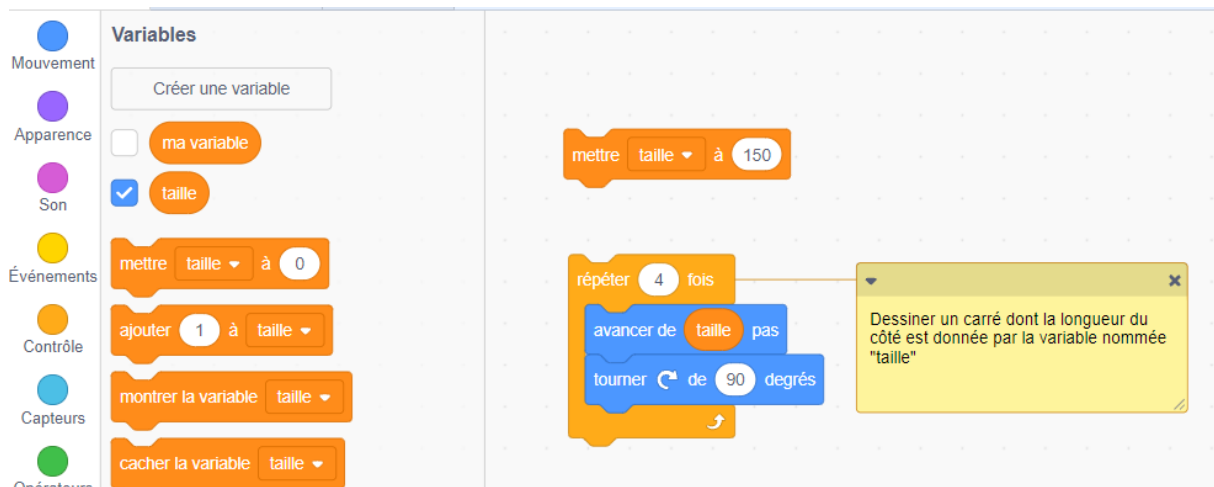
3. Dessiner plusieurs carrés de tailles différentes

Laisser les enfants se débrouiller quelques minutes puis introduire la notion de *variable*.

Ouvrir le menu « **Variables** » et créer une variable nommée **taille**.

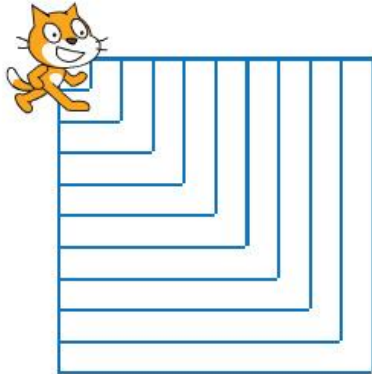
Repérer ce qu'on peut faire avec cette variable : lui donner une valeur, lui ajouter une valeur.

Reprendre le « programme » carré et utiliser cette donnée pour tracer les côtés. Pour dessiner des carrés de tailles différentes, il suffit alors de faire varier la valeur de taille et de relancer le « programme » carré.



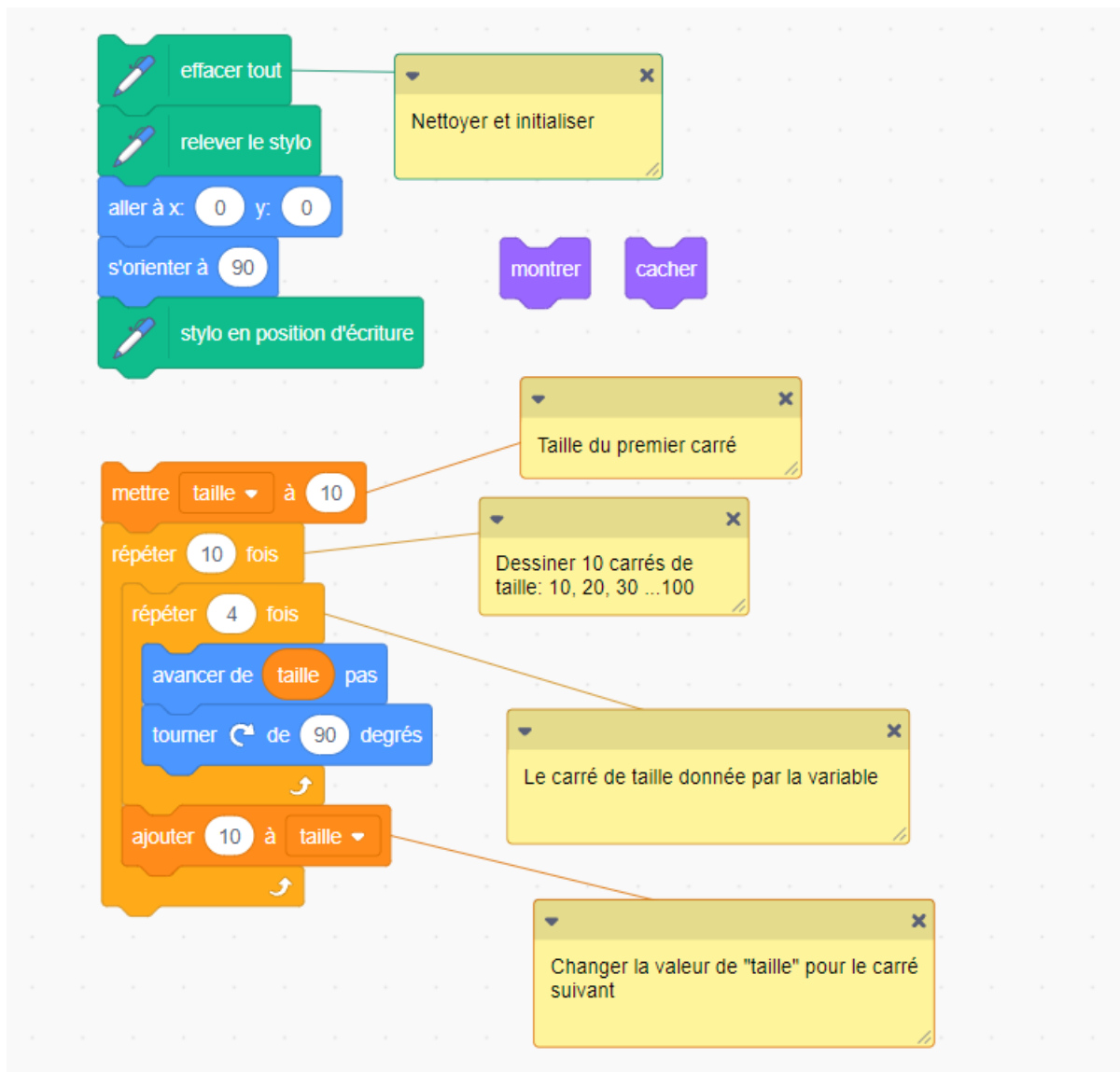
4. Faire un programme qui dessine automatiquement 10 carrés emboîtés

Voici la figure que l'on veut obtenir :

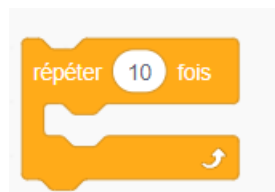


Plutôt que de faire varier à la main la variable **taille**, laissons un programme le faire pour nous. Dessiner 10 carrés de tailles : 10, 20, 30 ... peut se faire aussi grâce à l'instruction **répéter ...**

Il suffit d'entourer le « programme » carré pour qu'il s'exécute 10 fois. Mais attention à chaque tour il ne faut pas oublier de changer la valeur du côté du carré. Laisser les enfants essayer de trouver la solution. La voici ci-dessous :



Remarque :

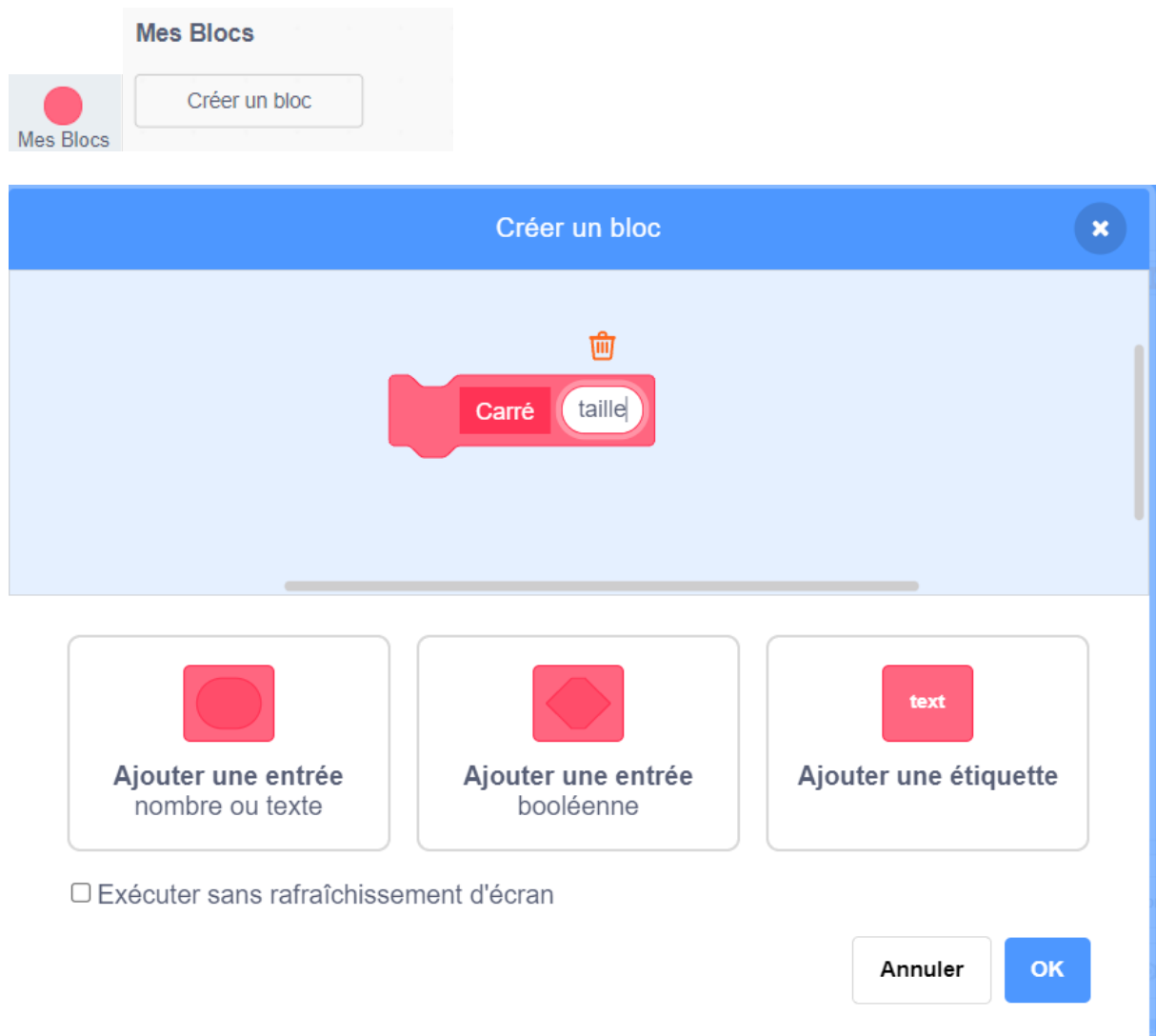


- L'instruction encadre l'instruction « répéter » qui dessine le carré suivie de l'instruction « ajouter » qui calcule la taille du prochain carré. En informatique on appelle souvent « boucle » cette instruction « répéter ».

5. Définir un nouveau bloc appelé « Carré »

Maintenant que nous savons comment dessiner un carré de taille quelconque enregistrons le une fois pour toute pour pouvoir s'en servir quand on en aura besoin sans à chaque fois écrire son code.

On peut alors créer un nouveau bloc dans la rubrique « **Mes Blocs** » qu'on appellera donc « carré »



On ajoute au bloc « Carré » un paramètre, nommé taille, qui définit donc la longueur des côtés du carré. On clique sur **ok**.

Il ne reste plus qu'à écrire le contenu du bloc Carré, comme on l'a appris précédemment :

Initiation à la programmation en utilisant le logiciel SCRATCH

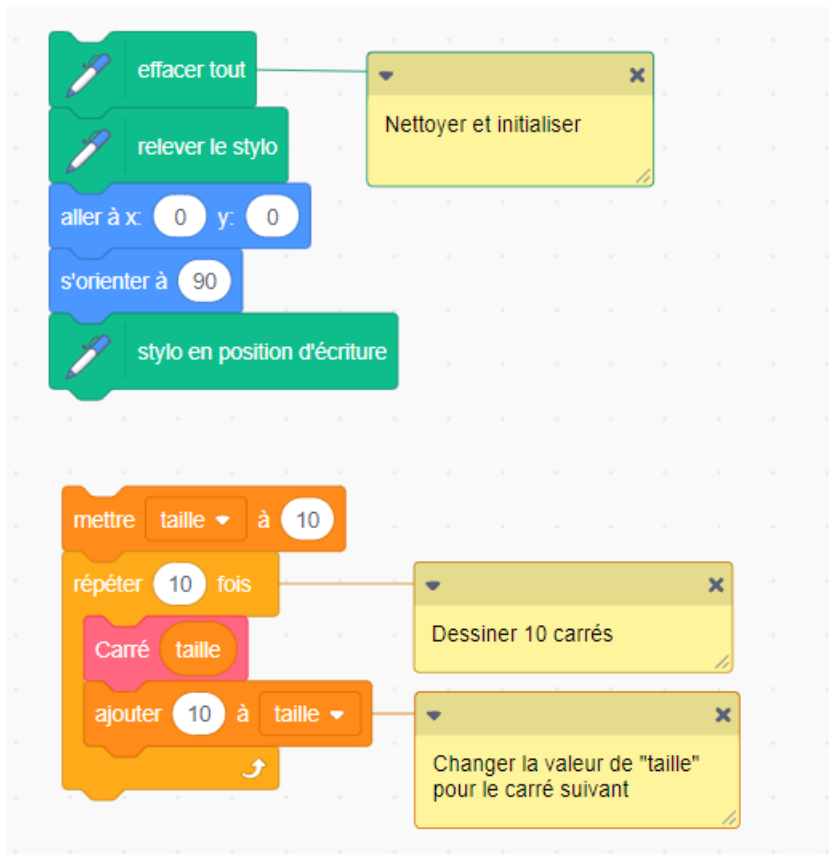


Attention : le « taille » de  est le paramètre du bloc « Carré », donc il faut le tirer (avec la souris) de l'entête  pour le copier à cette place.

On possède alors maintenant un nouveau bloc qu'on peut utiliser comme les autres instructions de Scratch, par exemple :



On peut ainsi réécrire plus simplement le programme des 10 carrés emboîtés :



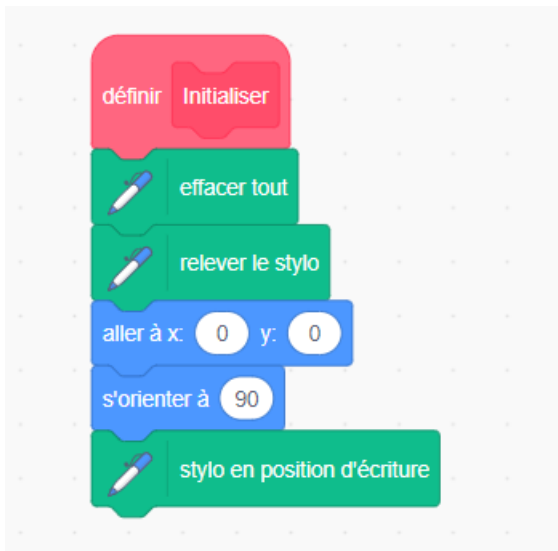
6. Ecrire notre premier programme complet

Le programme que nous voulons écrire est :

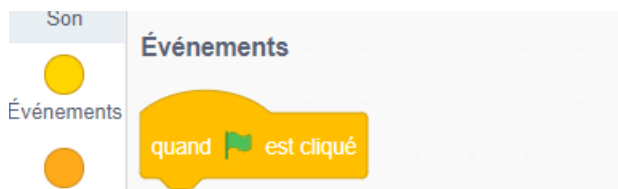
Initialiser (pour tout effacer et se placer au centre de l'écran)

Dessiner 10 carrés emboîtés.

Créons alors un nouveau bloc qui efface tout, appelons le **Initialiser**



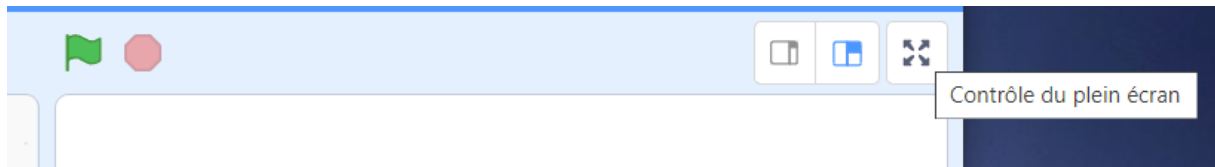
Pour indiquer où commence un programme, on marque le début avec un drapeau vert qu'on trouve dans la rubrique « **Évènements** »



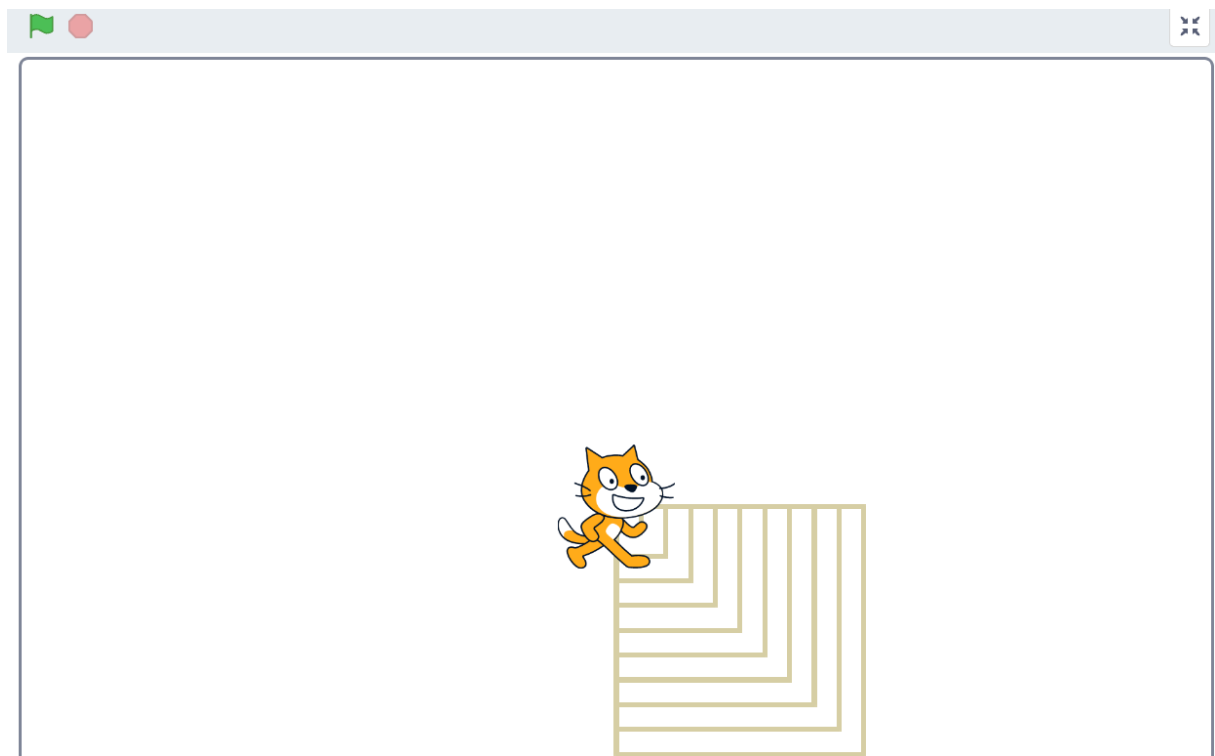
On peut alors écrire le programme complet :



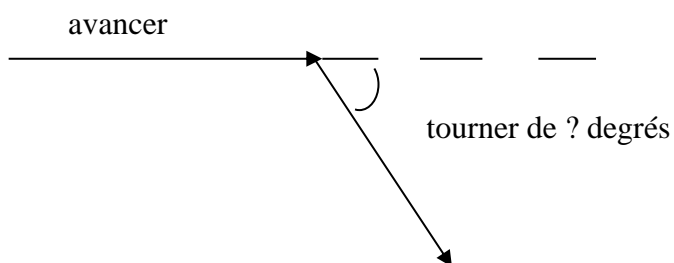
Pour exécuter le programme, passons en plein écran en cliquant en haut à droite sur la croix :



Pour démarrer le programme il suffit de cliquer sur le drapeau vert.



7. Tourner, oui mais de quel angle ?



Pour dessiner le carré on tourne 4 fois avec un angle de 90 degrés à chaque fois. Au total on aura tourné de $4 * 90 = 360$ degrés, ce qui nous ramène au point de départ.

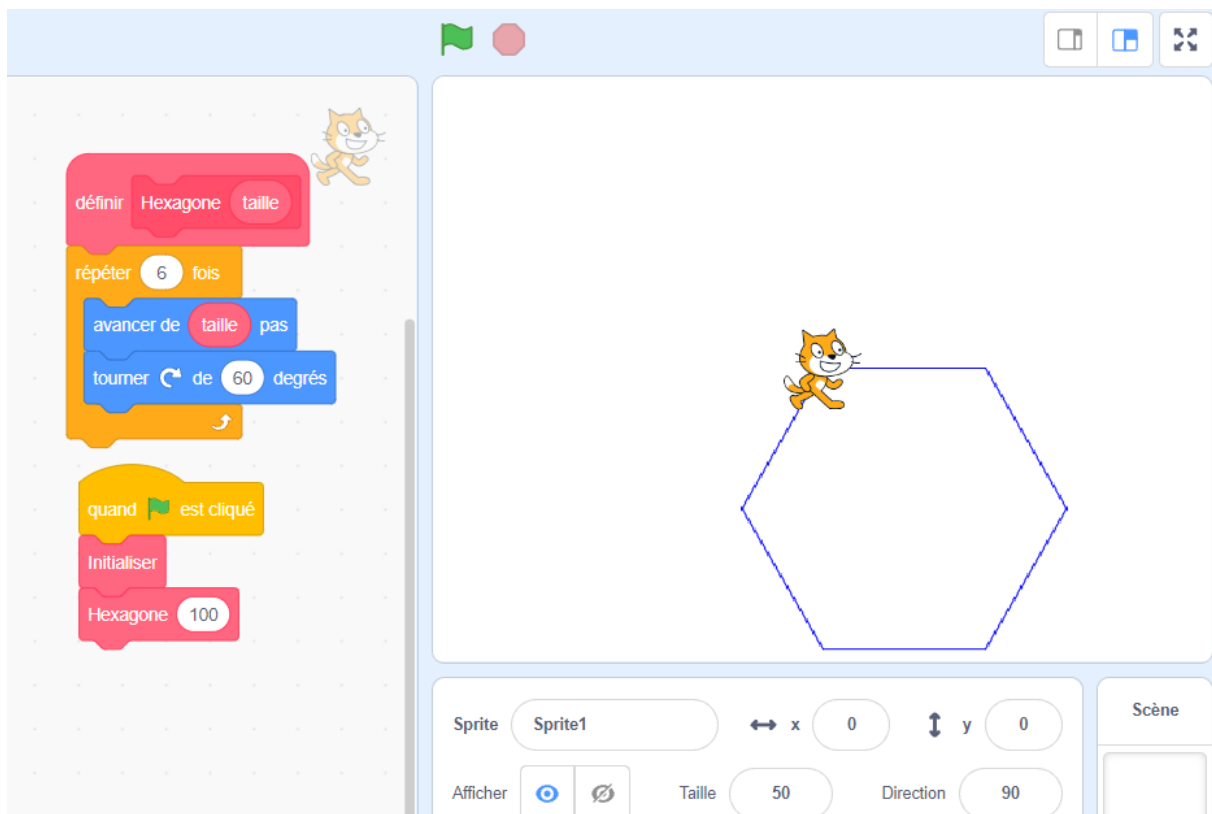
On peut alors dessiner d'autres polygones réguliers en tournant d'un angle différent. Mais il faudra qu'au total on tourne de 360 degrés pour fermer la figure.

8. Dessiner un hexagone, puis plusieurs hexagones ...

Le programme est similaire à celui du carré, sauf le nombre de côtés qui vaut donc ... ?

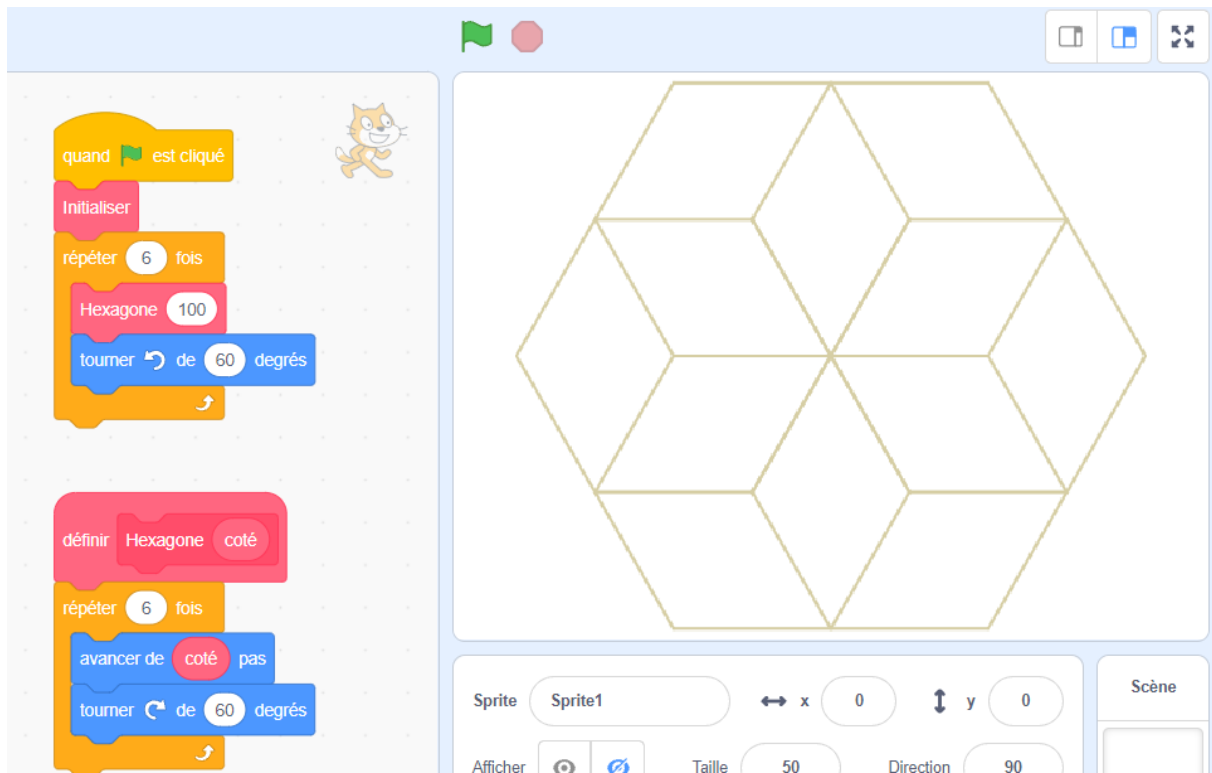
L'angle de rotation n'est pas 90° , mais ... ?

Le programme est calqué sur celui des carrés. Seuls le nombre de côtés et les angles sont différents évidemment.

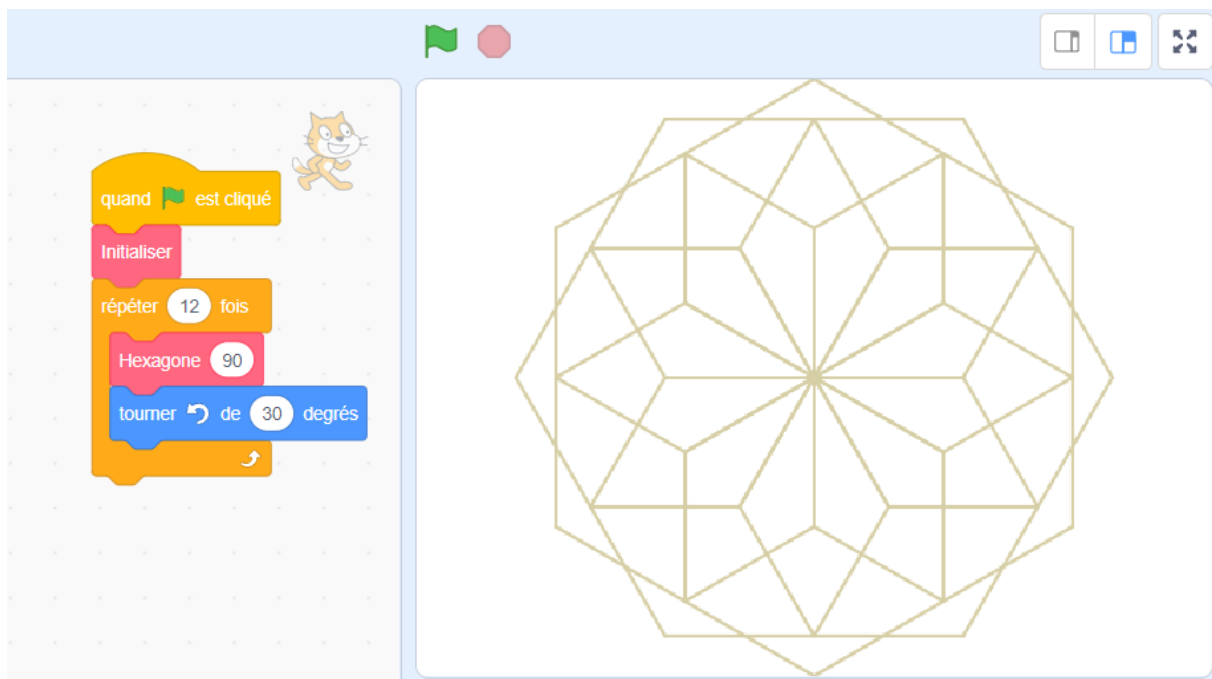


On peut alors afficher plusieurs hexagones pour composer des dessins plus complexes mais réguliers :

- Faisons tourner un hexagone 6 fois pour dessiner une jolie figure :

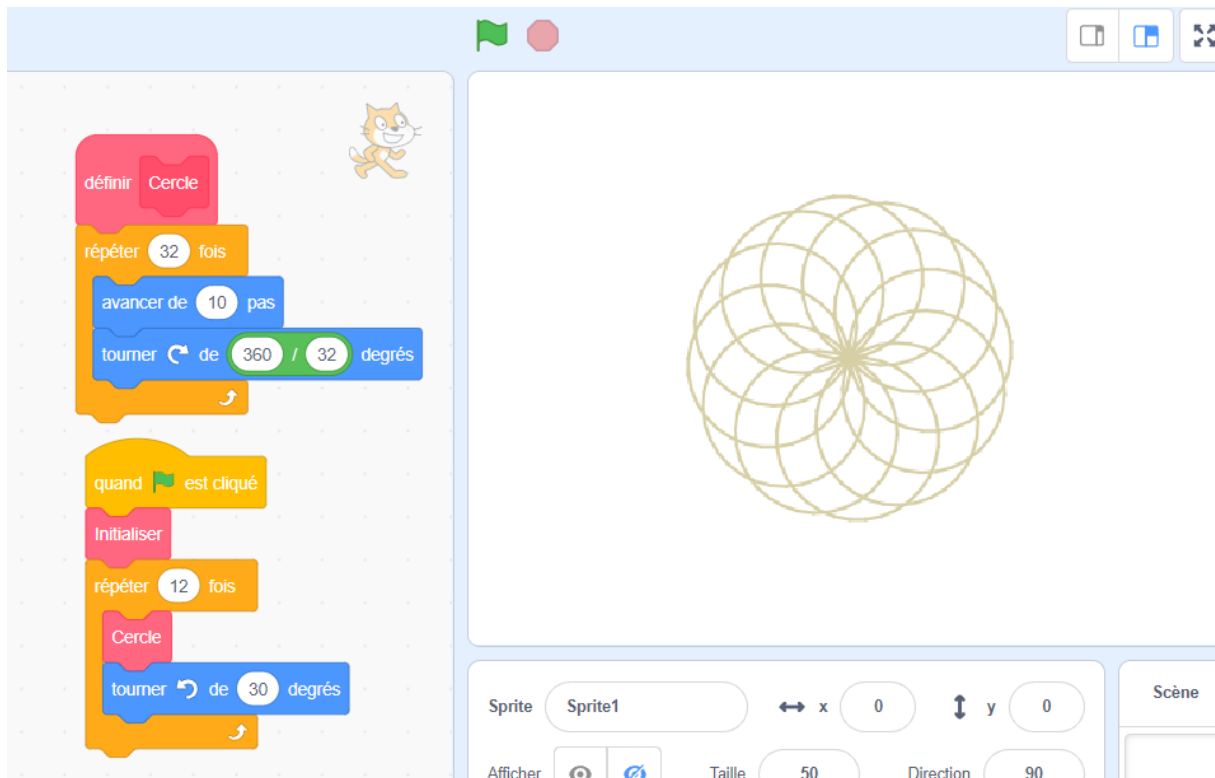


Et si on le faisait tourner 12 fois ...



9. Dessiner des cercles

Bizarrement Scratch n'offre pas d'instruction pour dessiner un cercle. On peut toutefois dessiner un cercle par approximation en dessinant un polygone ayant un grand nombre de côtés de petite taille. Par exemple :



B. Deuxième séance : animer une histoire à plusieurs personnages

Raconter une histoire à plusieurs personnages.

On montre l'exemple, on explique le principe du codage. Les élèves s'inspirent de cet exemple pour créer leur propre scénario.

*Permet de comprendre comment les personnages communiquent par **messages** (voir menu **Evènements**).*

*Utiliser les instructions de **Contrôle : répéter... et si ... alors***

Utiliser les instructions d'**Apparence** pour changer d'image de **costume** pour simuler une marche avec attente d' $1/10^{\text{ème}}$ de seconde entre 2 images.
Dialoguer avec **dire** et **penser**

1. Faire marcher le chat

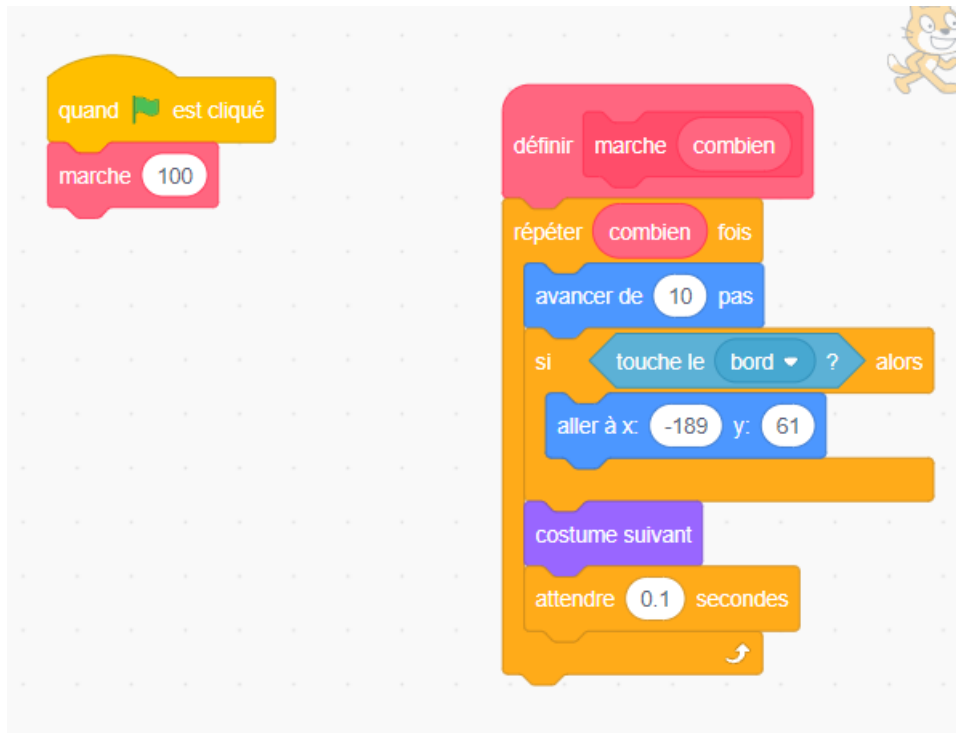
Remarquer que le chat a 2 costumes, si on fait apparaître les 2 images alternativement on simule une marche, à condition bien sûr de le faire avancer.

```
Répéter ...[  
  Avancer 10  
  Costume suivant  
]
```

2 problèmes : c'est trop rapide et quand on arrive au bord le chat est bloqué.
- Ralentir en attendant 0.1 seconde entre 2 images.
- Tester après chaque avancée si on a atteint le bord grâce aux **Capteurs : touche le bord ?** (remarquer que l'instruction touché ? a plusieurs options : pointeur de souris, et bord : choisir « bord »).

Que faire quand on atteint le bord ? Replacer le chat à l'autre extrémité de l'écran, grâce à l'instruction de **Mouvement : aller à x : y :** (remarquer que les coordonnées actuelles d'un personnage sont toujours indiquées en bas à droite de l'écran sous l'icône du personnage. On peut ainsi en plaçant le personnage à gauche de l'écran choisir les **x et y** pour l'instruction **aller à .**)

Pour le scénario qu'on envisage on aimerait faire varier le nombre de pas d'une marche : c'est ce nombre qu'on utilise pour l'instruction **répéter ... fois** .
On définit donc un sous-programme, grâce à **Ajouter block**, qu'on appelle marche et qui a 1 paramètre : combien .



On peut alors tester le programme : **marche 100** par exemple

2. Faire marcher un deuxième personnage.

Créer un nouveau lutin qui a des costumes de marche, par exemple : « Avery wakking » ou « Bear-walking » (*on apprend l'anglais au passage !*)

Définir pour ce nouveau personnage le même sous-programme :

marche (combien)

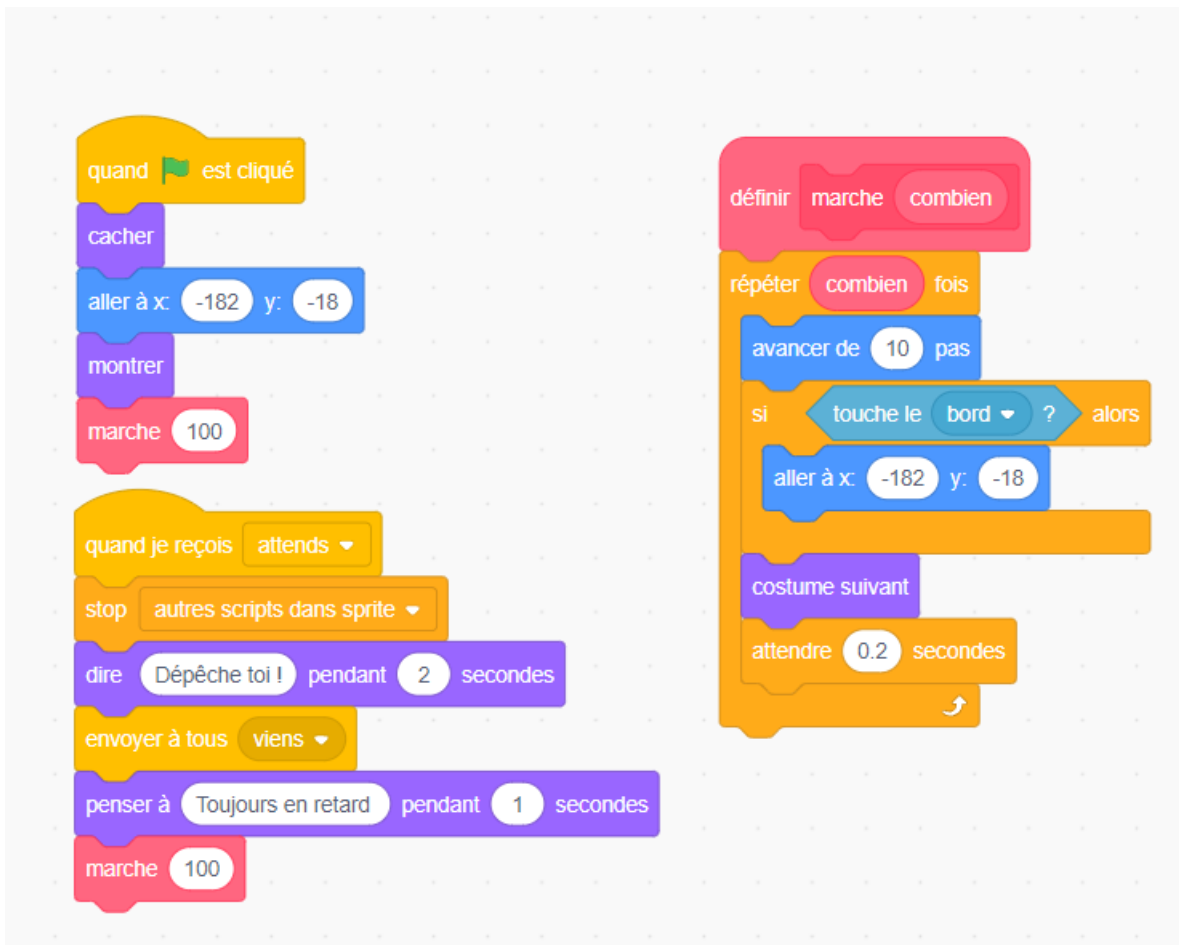
Adapter les coordonnées x et y pour que les 2 personnages partent d'endroits différents.

Commencer l'histoire : quand **le drapeau vert** est cliqué, Avery se met en marche, alors que le chat apparaît au bout d'une seconde et **dit** à Avery de l'attendre. Il lui envoie le **message : attends** .

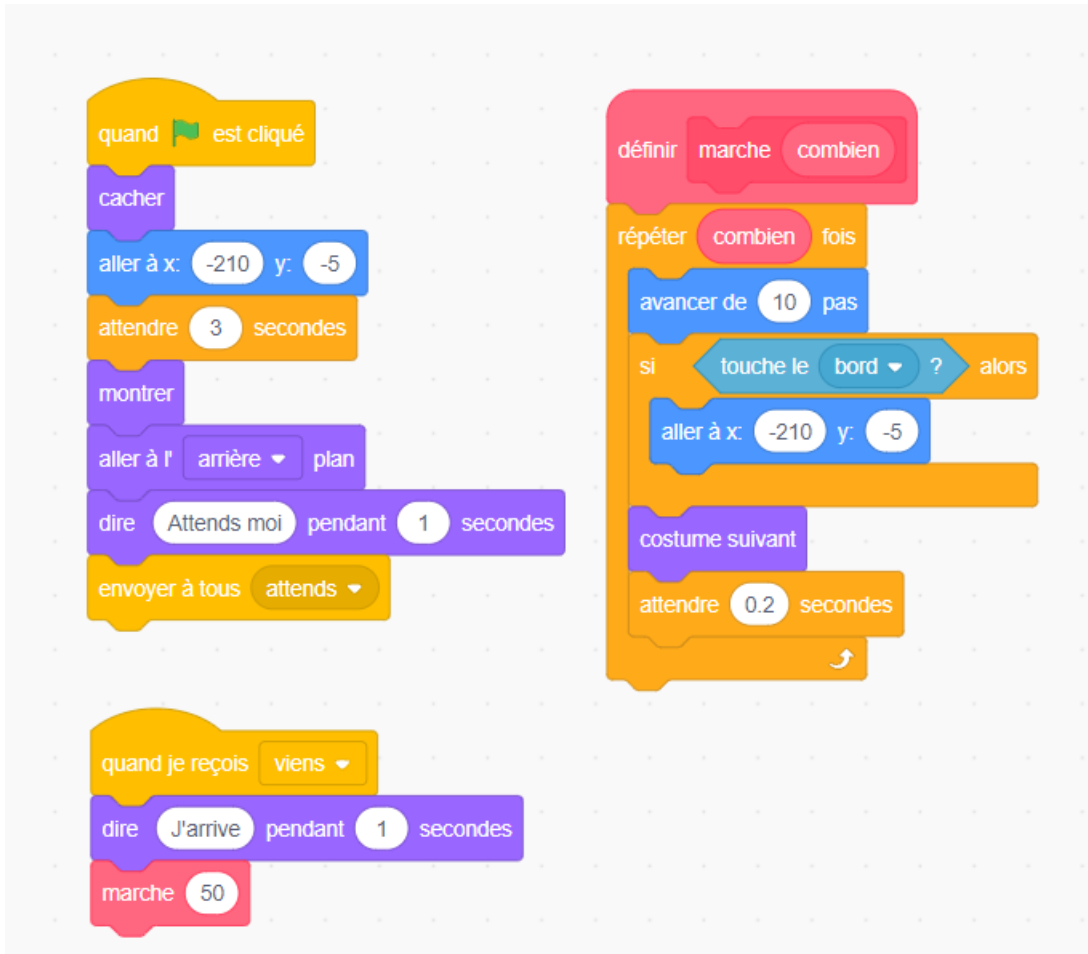
Quand Avery reçoit le message « **attends** », elle s'arrête de marcher (**stop autres scripts du lutin**) lui **dit : dépêche toi** , lui envoie le **message : viens** et **pense** que le chat est décidément toujours en retard.

Quand le chat reçoit le message « **viens** », il dit : « **j'arrive** » et il commence à marcher.

Avery :



Le Chat :



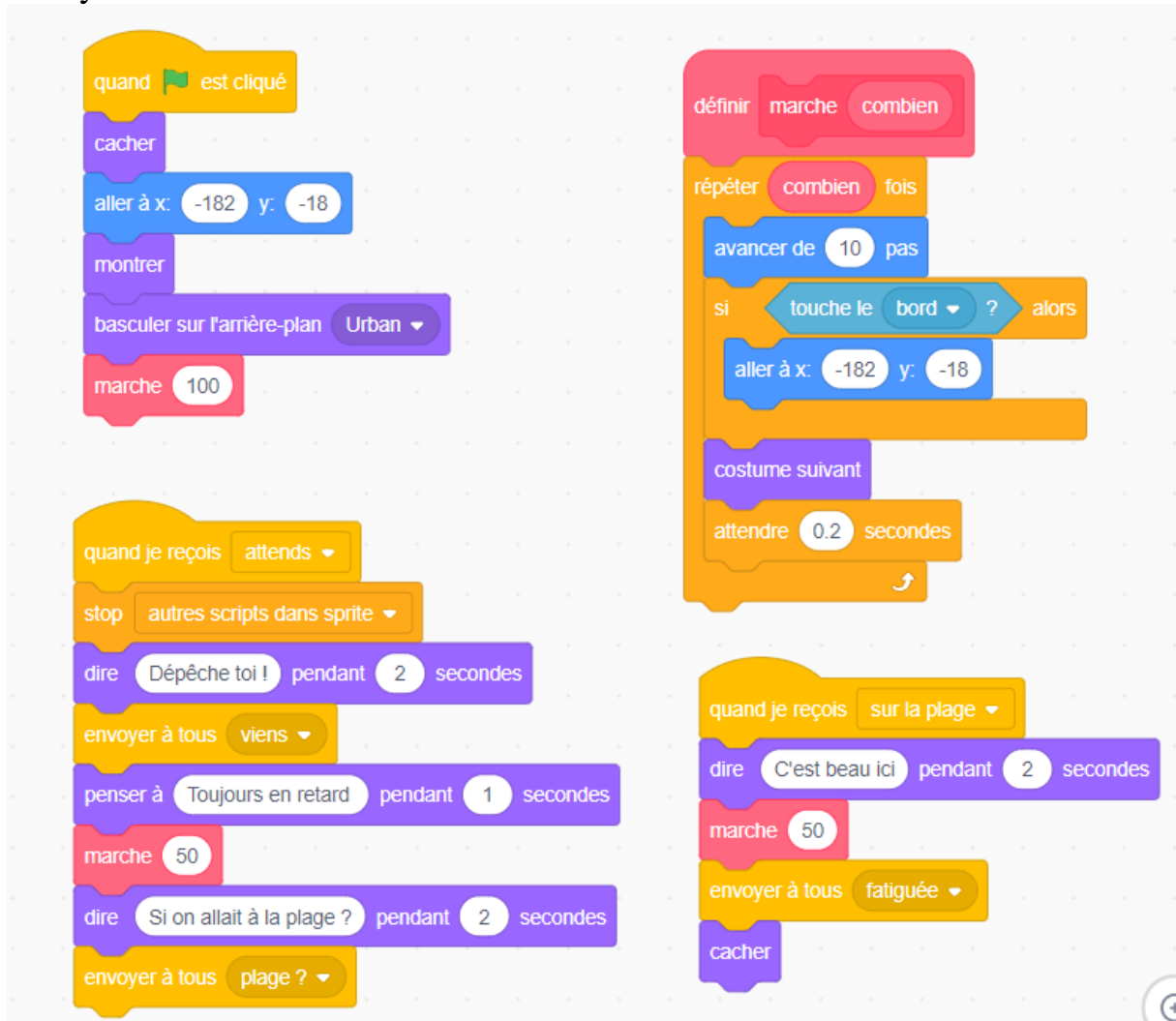
3. Changer de décor

Si vous ne l'avez pas encore fait, choisir un décor pour la scène. Mon histoire commence dans la rue. Puis Avery s'arrête et demande au chat : « si on allait à la plage ? ». Le chat reçoit la **question** répond « bonne idée » et c'est lui qui choisit le décor de la plage grâce à l'instruction : **basculer sur l'arrière plan ...** Le chat envoie le message « sur la plage », qui est reçu par Avery qui reprend alors sa marche.

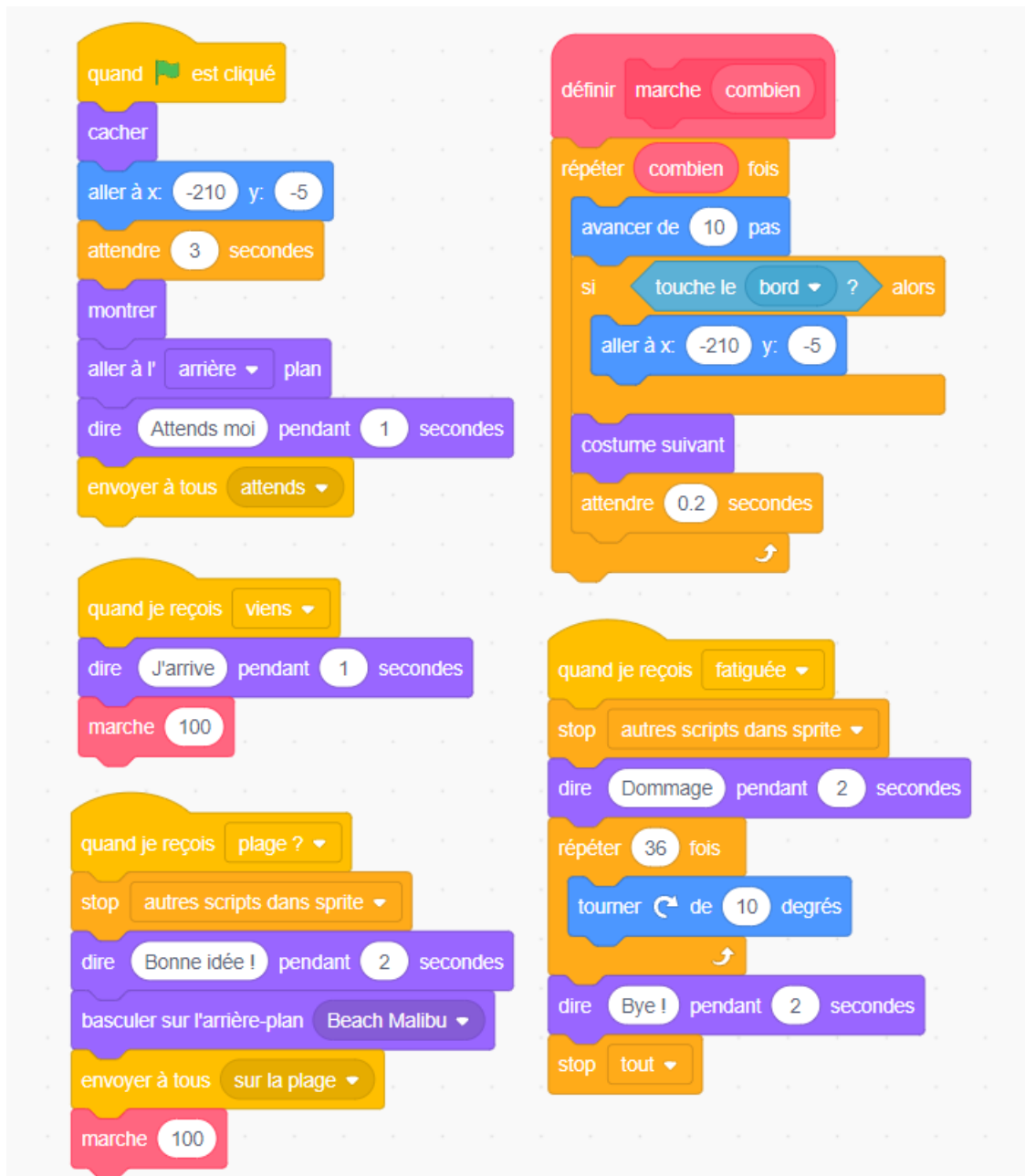
Au bout d'un certain temps Avery s'arrête dit qu'elle est fatiguée et envoie le message « arrêt ».

A la réception de ce message, le chat dit « dommage » et fait une acrobatie (dessine un cercle comme à la première séance).

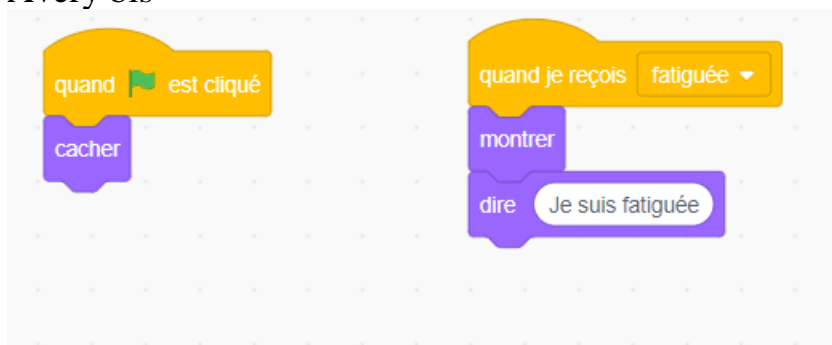
Avery :



Le Chat :



Avery bis

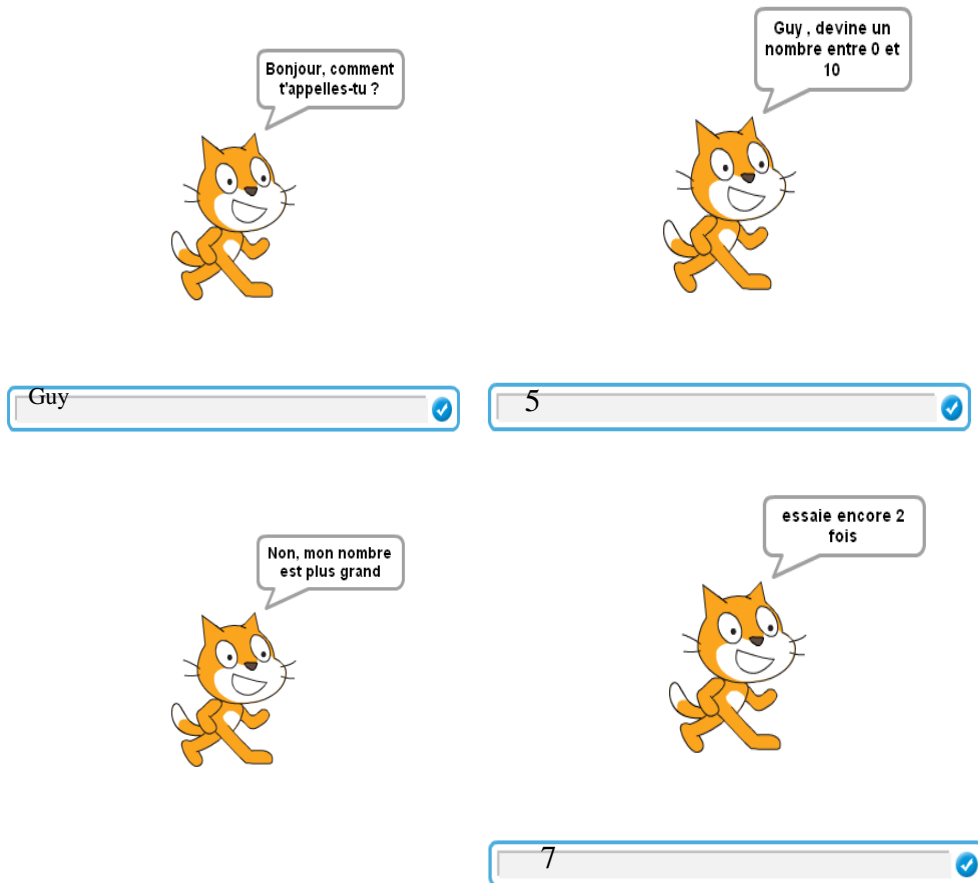


C. Troisième séance : Programmer un petit jeu

Il s'agit de programmer un petit jeu interactif. Le chat choisit un nombre entre 0 et 10 et l'utilisateur doit trouver ce nombre en maximum 3 essais.

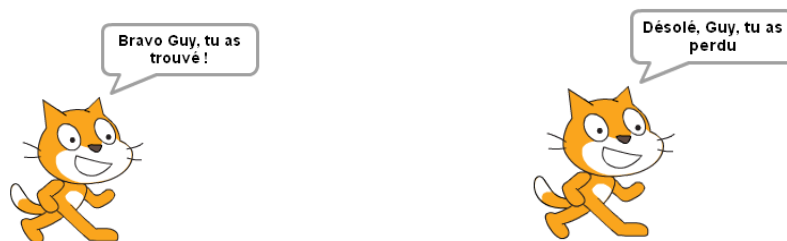
(Notion de logique : comparaison de nombres, instruction conditionnelle).

Voici la séquence qu'il faudra programmer :



Et ça finit par : ...

ou bien après 3 essais infructueux :



Le programme ci-dessous ne doit pas être réalisé tel quel en une seule fois, mais bien progressivement, pour des raisons pédagogiques.

- (1) commencer par : "bonjour comment tu t'appelles ?" et la saisie du nom, mettre le nom dans une variable, pour pouvoir s'en rappeler plus tard. Annoncer le but du jeu: "Guy, tu dois deviner un nombre entre 0 et 10")
- (2) tirer au hasard le nombre entre 0 et 10 et mettre ce nombre dans une variable (utiliser la fonction « nombre aléatoire »)
- (3) un seul essai d'abord : trouvé ou pas (instruction de test SI ...), puis affiner "plus grand" , "plus petit"
- (4) boucle (répéter 3 fois) pour permettre 3 essais

