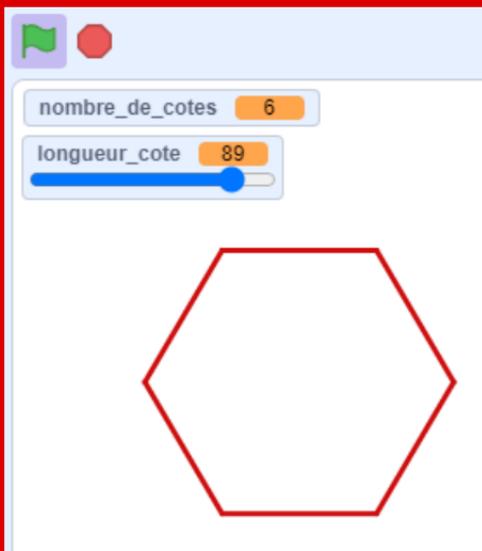


4 Polygones sur commande



Nous avons déjà abordé les polygones dans la première série. Faisons un pas de plus à l'aide de variables. L'utilisateur pourra choisir le nombre et la taille des côtés du polygone. Et en bonus, le nom du polygone affiché s'affichera



▼ Vidéo

<https://jeunesingenieurs.be/vid-s2-4>



▼ Proposition de solution

<https://scratch.mit.edu/projects/943847479>

▼ Tu te lances en mode Défi ?

Visionne la vidéo et essaie d'atteindre le résultat proposé.

Travaille étape par étape :

- Dessiner un carré (ou un autre polygone)
- Trouver une méthode pour calculer en degrés la rotation à effectuer aux sommets (=l'angle externe) en fonction du nombre de côtés du polygone.
- Ajouter les variables **nombre** et **longueur** des côtés et le bloc **demander ... et attendre** pour que l'utilisateur puisse choisir le nombre de côtés.
- Choisis le mode **barre de défilement** pour la variable **longueur** et ajoute une boucle **répéter indéfiniment** afin de pouvoir modifier sa valeur.
- Utilise un bloc personnalisé et supprime de rafraîchissement d'écran pour obtenir une modification instantanée de la taille du polygone.
- A l'aide d'une **liste** et du code adéquat, ajoute l'affichage du nom du polygone.

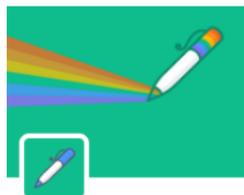
4.1 DESSINER UN POLYGONE

🎯 Objectifs

Prépare un premier script qui trace par exemple un hexagone avec un sprite quelconque et l'extension Stylo.

Analyse le script afin de répondre à cette question :
« Quelles sont les valeurs à modifier pour dessiner un triangle à la place d'un hexagone ? »

▾ Charge l'extension Stylo et initialise ton programme



Stylo

Ajoute ce code et prends le temps d'identifier le rôle de chaque bloc.

Il ne se passe rien, c'est normal 😊, il s'agit juste d'une initialisation.

DESSINER UN POLYGONE

Complète le script pour dessiner un hexagone

Ajoute la boucle répéter 6 fois pour dessiner un hexagone et teste ton programme.

Analyse le script

Comment modifier le code pour dessiner un carré, un triangle, un pentagone ?

Triangle :

répéter 3x, tourner de 120°

Carré :

répéter 4x, tourner de 90°

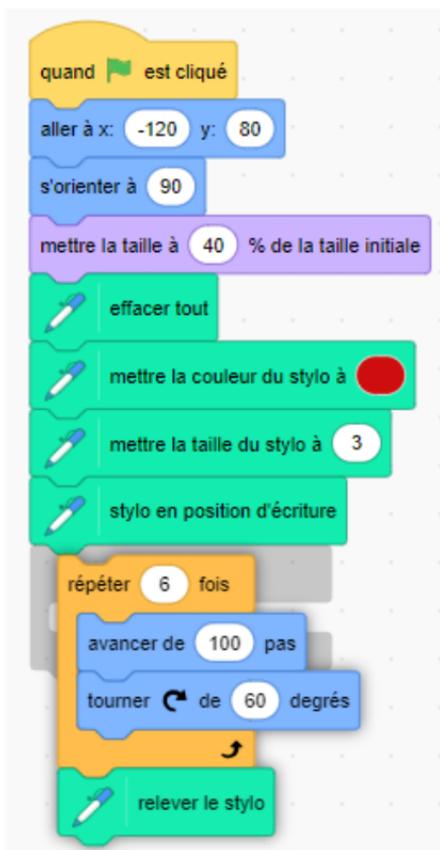
Pentagone :

répéter 5x, tourner de 72°

$$3 \times 120^\circ = 360^\circ,$$

$$4 \times 90^\circ = 360, \quad 5 \times 72^\circ = 360^\circ, \dots$$

Pour calculer l'angle de rotation (= angle externe) à effectuer à chaque sommet du polygone, il suffira de faire $360^\circ / \text{nombre de côtés}$.



4.2 CHOISIR LE NOMBRE DE CÔTÉS

🎯 Objectifs

Le programme doit

- interroger l'utilisateur : « Combien de côtés doit avoir le polygone ? » et mémoriser cette valeur,
- calculer l'angle externe , c'est-à-dire la rotation à effectuer en degrés à chaque sommet,
- exécuter le dessin

➤ Crée deux variables et affiche-les

Afin de pouvoir modifier le type de polygone et sa taille, crée deux variables :

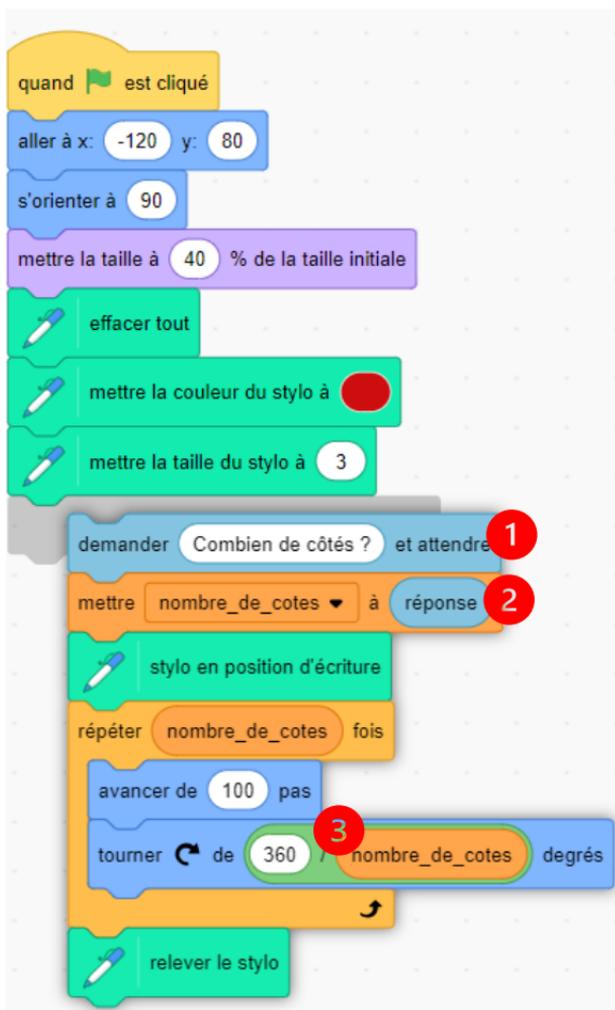
- ***longueur_cotes***
- ***nombre_de_cotes***



Les programmeurs recommandent de composer des noms de variables à l'aide de caractères simples, en minuscule et le tiret underscore (tiret bas).

Assure-toi que les deux variables apparaissent bien sur la scène à l'aide de la case à cocher qui les précèdent. Tu peux les repositionner si tu le souhaites.

CHOISIR LE NOMBRE DE CÔTÉS



(1) Avec le bloc **demander ... et attendre**, on interroge l'utilisateur et la réponse est stockée dans le bloc de valeur **réponse**.

(2) Le contenu de **réponse** est stocké dans la variable **nombre_de_cotes**.

(3) L'angle externe est calculé à l'aide du bloc **opérateur de division**



4.3 MODIFIER LA TAILLE

🎯 Objectifs

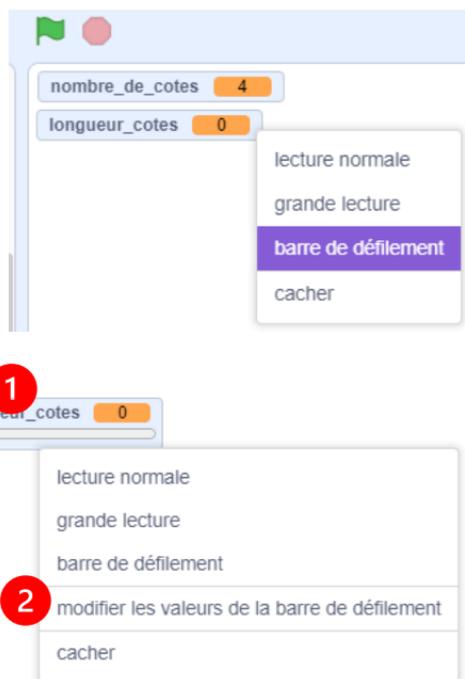
À l'aide de la variable *longueur_cotes* convertie en curseur, modifier la taille du polygone une fois qu'il est dessiné et en continu.

▼ Modifie l'apparence d'une variable

Clique-droit sur la variable *longueur_cotes* et choisis *barre de défilement*.

La variable se présente comme un curseur (1).
Clique-droit une deuxième fois et choisis cette fois *modifier les valeurs de la barre de défilement*. (2)

À l'invite, encode les valeurs 50 minimum et 100 maximum.



MODIFIER LA TAILLE



Modifie le script

Ajoute :

(1) une valeur de départ pour la longueur des côtés.

(2) une boucle **répéter indéfiniment** pour adapter la longueur des côtés en continu, chaque fois que la valeur de la variable est modifiée grâce au curseur.

(3) le bloc variable **longueur_cote**

Teste ce script

Cela marche ? Mais tu es peut-être un peu déçu par la vitesse de rafraîchissement de l'écran et le clignotement ...

On va arranger cela.

4.4 À TOUTE VITESSE

🎯 Objectifs

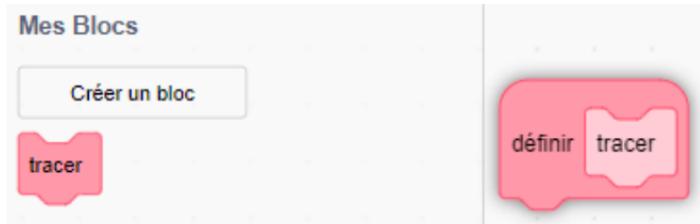
À l'aide d'un bloc personnalisé et en supprimant le rafraîchissement d'écran, débarrasse-nous de ce clignotement...

📌 Crée un bloc personnalisé

Dans la catégorie Mes blocs, clique sur **Créer un bloc**.

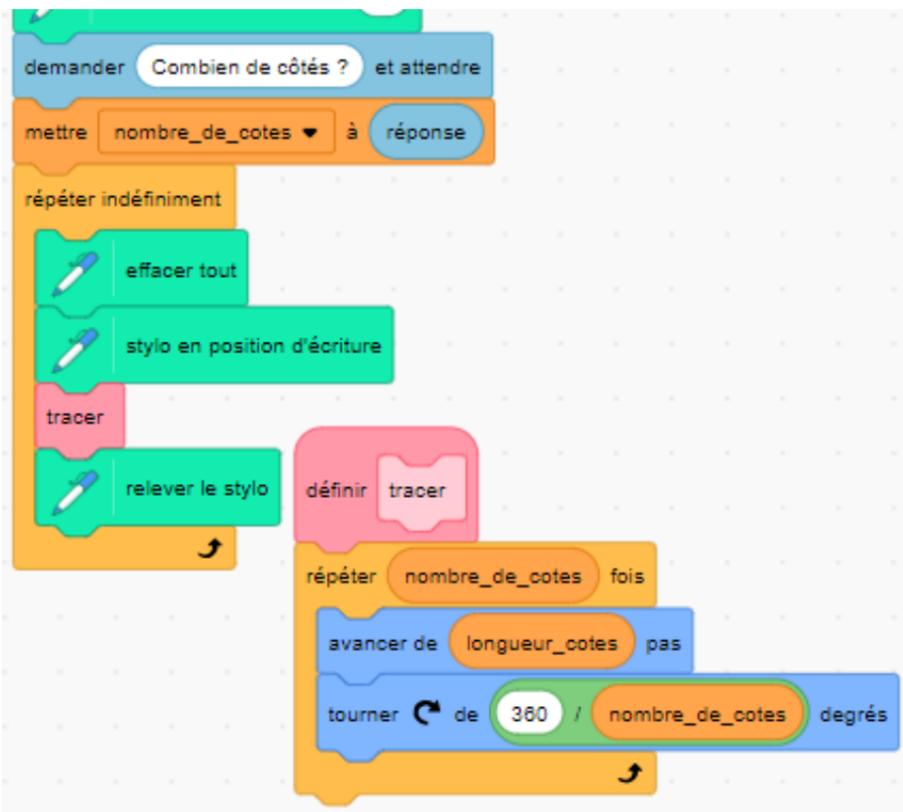
(1) Appelle-le **tracer** et supprime le rafraîchissement d'écran en cochant **Exécuter sans rafraîchissement** (2).

Un nouveau bloc **tracer** et un bloc de tête du même nom apparaissent.



À TOUTE VITESSE

Modifie ton script



La boucle répéter... fois placée sous le bloc de tête définir... tourne beaucoup plus vite quand on supprime le rafraîchissement d'écran et le clignotement disparaît...ou plutôt, on ne le voit plus.

4.5 LE NOM DU POLYGONE

Objectifs

Affiche le nom du polygone dessiné en fonction du nombre de côtés choisi par l'utilisateur. Utilise les listes pour mémoriser ces noms.

Créer une liste de noms

- (1) Dans la catégorie **Variables**.
- (2) Nomme-la **polygones**.



Une liste vide apparaît sur la scène.

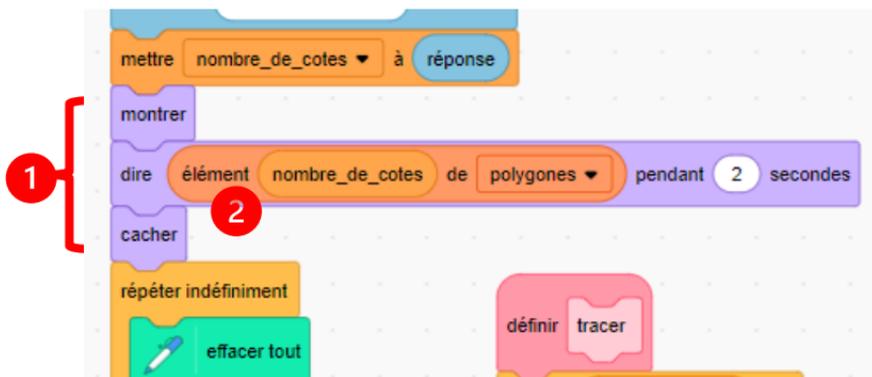
Clique sur le signe + (3) et complète-la. Commence par deux tirets , ainsi, le numéro de l'élément de liste va correspondre au nombre de côtés : 3 pour le triangle, 4 pour le carré, etc.



LE NOM DU POLYGONE

▼ Afficher le nom du polygone

Ajoute les blocs *montrer*, *dire...pendant ... secondes* et *cacher* (1) juste avant la boucle *répéter indéfiniment*.



Pour comprendre son fonctionnement, teste le bloc de valeur *élément de* (2) hors du script. Modifie la valeur de l'index (le numéro : 3, 4, 5...) de l'élément. Clique sur le bloc de valeur et le nom correspondant à l'index de la liste s'affiche.

