
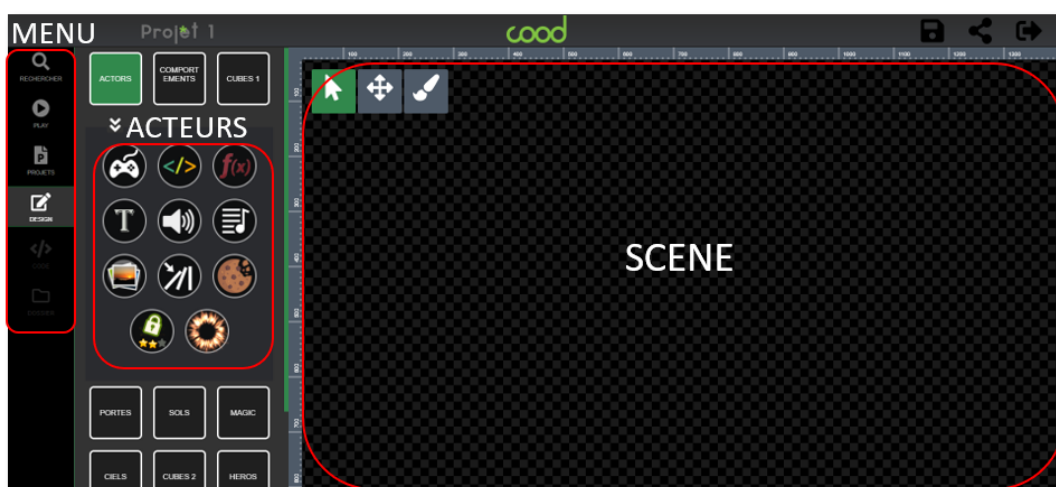



Animer un ennemi en continu

Maintenant qu'il est possible de contrôler son personnage, il est temps d'animer aussi les autres éléments dynamiques du jeu, comme les ennemis ou le décor (nuages, oiseaux, etc...).

Mais cette fois-ci, ils bougeront tout seuls !

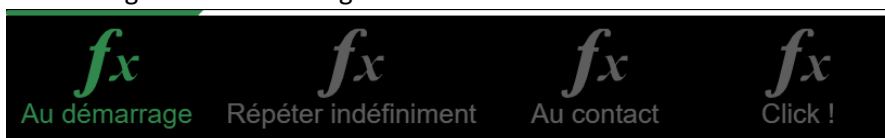
Pour commencer, aller dans le menu Design, et cliquer sur le logo  puis n'importe où sur la scène.



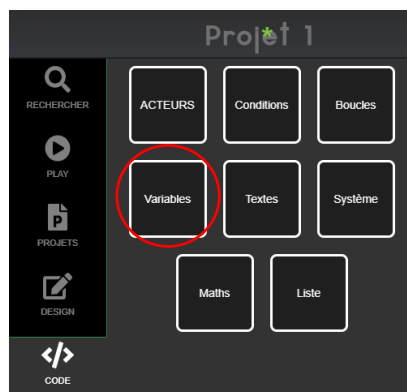
Cliquer sur le menu  du sprite et aller dans « Paramètres » pour lui choisir une apparence et le renommer. Pour ce tutoriel, appelons-le « Ennemi ».


Double-cliquer ensuite sur le sprite pour aller dans le menu Code.

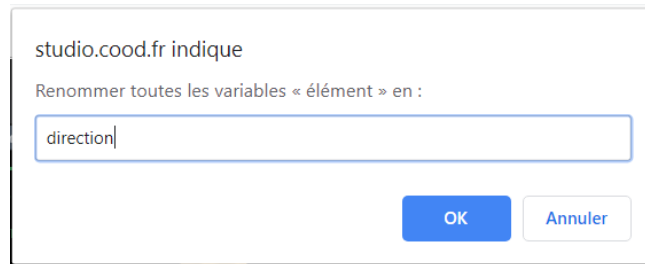
- Cliquer sur l'onglet « Au démarrage » en haut de l'écran :



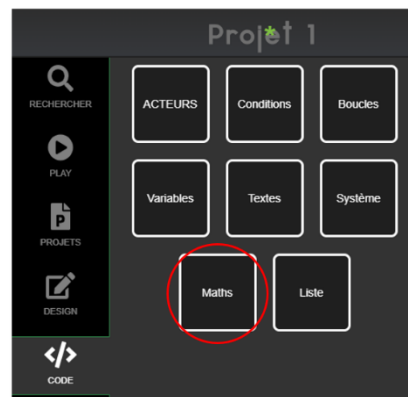
- Sélectionner le dossier « Variables » :




- Glisser le bloc  sur la scène.
Dans son menu déroulant, cliquer sur « Renommer la variable ».
 - Une nouvelle fenêtre va s'afficher pour nommer la nouvelle variable : appelons-la « direction ».



- Sélectionner ensuite le dossier « Maths » :



- Glisser le bloc  sur la scène et lui donner une valeur de 2.



Le sprite possède désormais une variable qui lui permet de se déplacer sur 2 mètres.

- Cliquer ensuite sur l'onglet « Répéter indéfiniment » en haut de l'écran :





- Sélectionner le dossier « Conditions » :

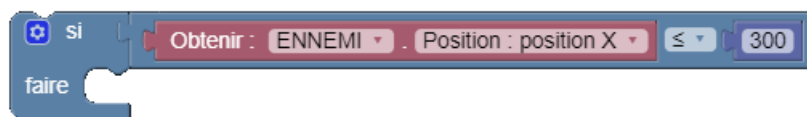




- Glisser les blocs  et  sur la scène, et encastrez-les ensemble. Sélectionner « ≤ » dans le menu déroulant du second bloc.

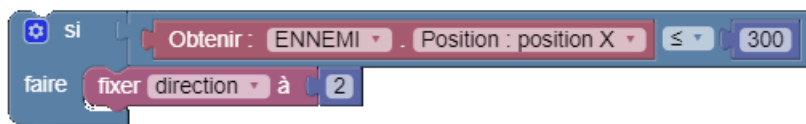
- Puis, sélectionner l'onglet « Acteurs » :



- Glisser le bloc  sur la scène. Dans ses menus déroulants, choisir « Ennemi » puis « Position : position X ».
- Depuis l'onglet « Maths », placer un bloc  sur la scène et lui donner une valeur de 300 (par exemple).
- Encastrer les blocs comme ici :



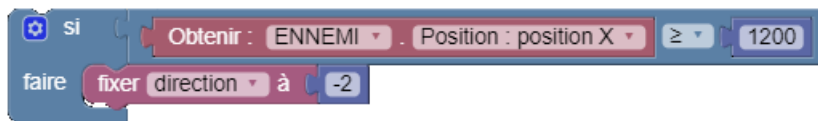
- Aller ensuite dans l'onglet « Variables » et glisser le bloc  sur la scène. Dans son menu déroulant, choisir la variable « direction ».
 - Si la variable « direction » n'apparaît pas, cliquer sur « Renommer la variable » puis l'appeler à nouveau « direction ».
- Aller dans l'onglet « Maths » et glisser le bloc  sur la scène puis lui donner la valeur 2.
- On obtient ce code :



Si l'ennemi se trouve dans la zone supérieure ou égale à 300 mètres, la valeur de sa direction augmentera de 2 mètres en continu.

Maintenant que le personnage va dans un sens, il faut qu'il se retourne pour effectuer sa ronde.

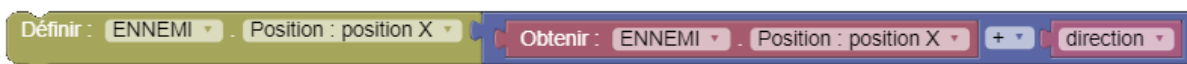
- Recommencer l'exercice en modifiant la première valeur en « 1200 » et la seconde valeur en « -2 ».
- On obtient alors ce code :



Si l'ennemi se trouve dans la zone inférieure ou égale à 1200 mètres, la valeur de sa direction diminuera de 2 mètres en continu.

Maintenant que les valeurs de déplacement ont été affectées à la variable « direction », on peut demander au sprite de se déplacer en fonction de sa position.

- Aller dans l'onglet « Acteurs » et glisser le bloc sur la scène. Dans ses menus déroulants, choisir « Ennemi » puis « Position : position X ».
- Placer ensuite le bloc et choisir « Ennemi » puis « Position : position X » dans ses menus déroulants.
- Aller dans le menu « Maths », choisir le bloc puis imbriquer tous les blocs comme ici :



L'ennemi se déplace désormais de droite à gauche en continu !

Le sprite se déplace enfin mais il ne regarde pas dans la bonne direction.

- Pour cela, aller dans le menu « Acteurs » et placer le bloc . Dans ses menus déroulants, choisir « Ennemi » puis « Position : regarder vers ».
- Glisser ensuite le bloc et les imbriquer. Ajouter ces blocs au second paragraphe, destiné à faire reculer le sprite.
- Répéter cette opération pour qu'il regarde à droite cette fois, et les imbriquer au premier paragraphe, destiné à faire avancer le sprite.

- Le code final ressemble à ceci :

```

graph TD
    subgraph " "
        direction TB
        S1[si] --- F1[faire]
        F1 --- F1_1[fixer direction à 2]
        F1 --- F1_2[Exécuter : ENNEMI . Position : Regarde vers Droite]
        S2[si] --- F2[faire]
        F2 --- F2_1[fixer direction à -2]
        F2 --- F2_2[Exécuter : ENNEMI . Position : Regarde vers Gauche]
        D[ Définir : ENNEMI . Position : position X ] --- A[ Obtenir : ENNEMI . Position : position X + direction ]
    end

```

The image shows a sequence of Scratch-style code blocks:

- A **si** (if) block with the condition: **Obtenir : ENNEMI . Position : position X ≤ 300**.
- Inside the first **si** block, a **faire** (do) loop containing:
 - fixer direction à 2**
 - Exécuter : ENNEMI . Position : Regarde vers Droite**
- A second **si** block with the condition: **Obtenir : ENNEMI . Position : position X ≥ 1200**.
- Inside the second **si** block, a **faire** (do) loop containing:
 - fixer direction à -2**
 - Exécuter : ENNEMI . Position : Regarde vers Gauche**
- A final **Définir : ENNEMI . Position : position X** block followed by an **Obtenir : ENNEMI . Position : position X + direction** block.