

Un petit jeu vidéo



Programme un jeu tout simple.... Euh, pas si simple.

Respecte bien les consignes dans un premier temps.

Quand tu auras mis tout cela en place, tu pourras l'améliorer et le modifier à ta guise.

Tu peux essayer ce programme :

<https://scratch.mit.edu/projects/889898724>

Ce programme est inspiré par Inspiré par

<https://resources.scratch.mit.edu/www/cards/fr/scratch-cards-all.pdf>. Tu y trouveras d'autres jeux de cartes pour apprendre.

Au verso, on explique comment se servir de ces cartes.

Comment se servir de ces cartes ?

Chaque carte a un recto et un verso.

Au recto, tu trouveras ce symbole :

C'est la face Défi



Cette face te présente :

- un défi à réaliser,
- si nécessaire, des blocs qui pourront t'être utiles,
- si nécessaire, quelques conseils.

Si tu peux réaliser le défi sans aucune aide, si tu t'en sens capable, fais-le. Compare ensuite ta solution avec celle proposée au verso et passe à la carte suivante.

Si tu n'es pas trop sûr, fais-toi aider et retourne la carte

Au verso, un autre symbole qui représente « une solution ».

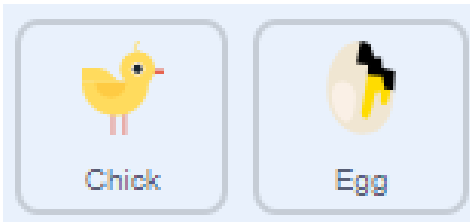
C'est la face Solution



Cette face présente :

- un exemple de programme « solution » ,
- des trucs et astuces pour relever le défi,
- et quelques explications.

Charge les sprites et l'arrière-plan



Charge les sprites **Chick** et **Egg** ainsi que l'arrière-plan **Blue sky**.



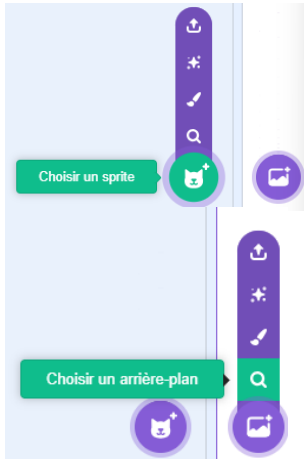
Supprime le **Sprite1**



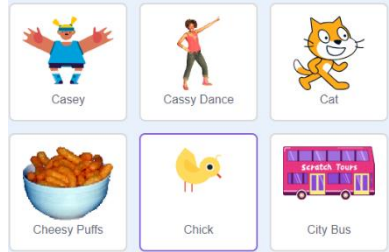
Initialise ton code en plaçant Chick au milieu de la scène sur le sol.

Tu auras peut-être envie de choisir d'autres sprite et un autre décor ? C'est très bien ! Mais je te conseille de commencer en suivant rigoureusement les instructions. Tu pourras modifier tout ce que tu voudras une fois le projet finalisé.

Charge les sprites et l'arrière-plan



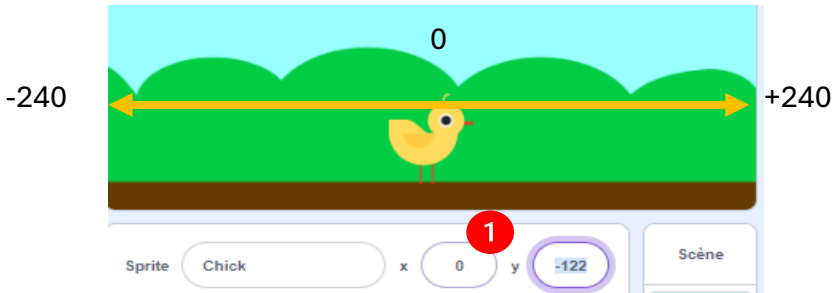
Clique sur choisir un sprite, sélectionne **Chick** et clique dessus pour l'ajouter.



Même procédure pour charger l'arrière-plan **Blue sky**.

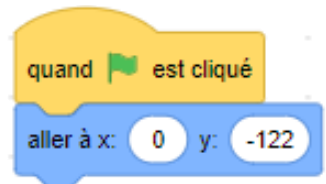
Place le poussin au milieu de la scène.

Observe les **coordonnées X et Y** et **1** remplace la valeur de X par 0 et la valeur de Y par -122.



Ajoute le code d'initialisation

Ce premier script sert à positionner le poussin au départ de l'action . En modifiant la valeur de X, tu modifieras la position horizontale du sprite.



Le poussin saute, les œufs défilent



Faire sauter le poussin

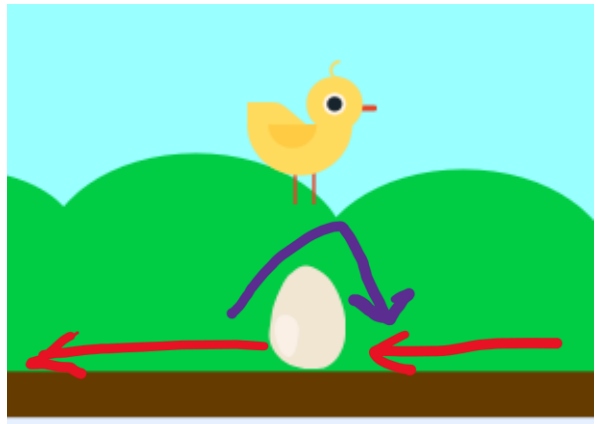
Faire sauter le poussin quand on appuie sur la touche flèche-haut du clavier .

Il devra sauter au-dessus de l'œuf. Tu dois donc prévoir un bond suffisamment important.

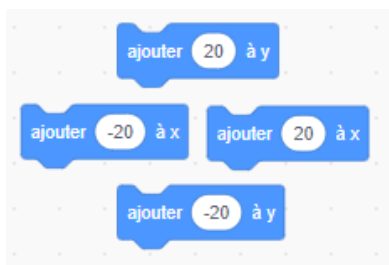


Faire défiler l'œuf

Faire en sorte que l'œuf défile de droite à gauche et puis recommence, indéfiniment .



Essaie les blocs avant de les utiliser !



Tu vas devoir utiliser les blocs **ajoute ... à x** et **ajoute ... à y** pour programmer les sauts du poussin et la glissade de l'œuf de gauche à droite.

Place ces blocs comme indiqué dans la fenêtre des scripts du poussin et essaie-les en cliquant dessus. Modifie les valeurs et essaie à nouveau. Remarque les signes négatifs... Ajouter -20 à x pourrait aussi se lire Retirer 20 à x...

Le poussin saute, les œufs défilent



Ajoute ce code au poussin

Ajoute ce second script qui va provoquer le saut du poussin quand tu appuieras sur la touche flèche-haut de ton clavier.

```
quand la touche flèche haut est pressée
répéter 12 fois
  ajouter 10 à y
répéter 12 fois
  ajouter -10 à y
```

Fais quelques essais en modifiant les valeurs de la boucle *répéter...fois* ou du bloc *ajouter ... à Y*.

Ajoute ce code pour *Egg*

L'œuf va traverser la scène de droite à gauche, revenir à son point de départ (x : 240 ; y :-130) et puis recommencer.

```
quand est cliqué
1 aller à x: 240 y: -131
2 répéter indéfiniment
  ajouter -6 à x
3 si absclisse x < -240 alors
  aller à x: 240 y: -131
```

Quelques explications

- 1 On place l'œuf au départ à droite de la scène avec *aller à x...y...*
- 2 La boucle *répéter indéfiniment* et le bloc *ajouter -6 à x* ont pour effet de faire avancer l'œuf vers la gauche.
- 3 On teste chaque fois si la position X (abscisse) n'est pas inférieure à -240. Si le test renvoie la valeur *True* (Vrai), alors l'œuf retourne à sa position de départ.



Le poussin casse l'œuf

Objectif : stopper le jeu si le poussin touche l'œuf.

A ce stade, quand tu démarres ton programme, les œufs défilent et tu peux faire sauter ton poussin.

Il faudrait que le jeu s'arrête quand un œuf est touché par le poussin.



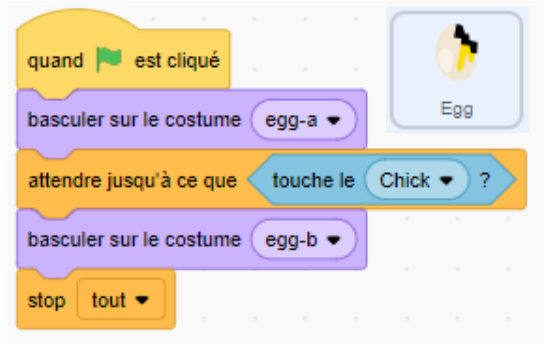
Il y a plusieurs manières d'obtenir ce résultat.

À toi d'en trouver une qui fonctionne.

Le poussin casse l'œuf



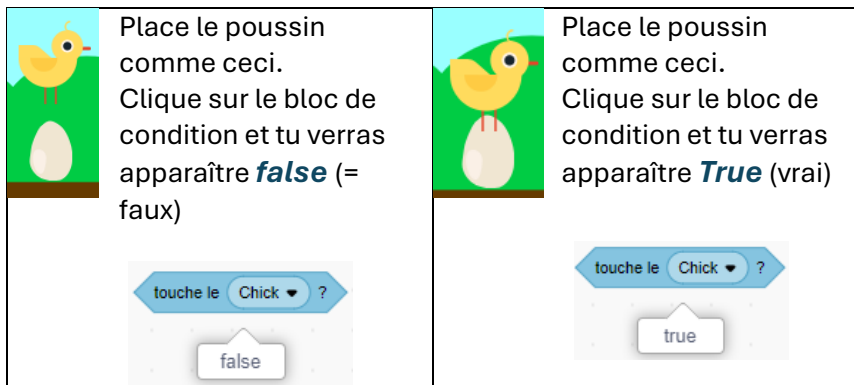
Le bloc **attendre jusqu'à ce que** est une boucle qui teste si l'œuf est touché par le poussin. Ce bloc passe la main au suivant dès que la condition **touche le chick** renvoie la valeur **Vrai** (True). Lorsque cela se produit, on




bascule sur le costume egg-b (œuf cassé) et ensuite, le jeu s'arrête avec le bloc **stop tout**.

Teste la condition « **touche le chick** »

Duplique le bloc **touche le chick** et place-le seul hors de tes scripts.



Ces blocs en forme de  ne peuvent contenir que deux valeurs : Vrai ou Faux. Le bloc ci-dessous se lit donc : attendre jusqu'à ce que la condition **touche le Chick** soit Vraie.



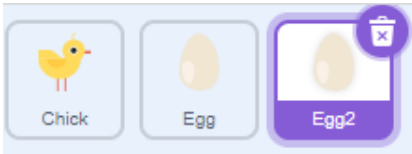
Ajoute un deuxième œuf



Objectif :

Le deuxième œuf doit défiler avec un peu de retard (+/-une seconde) et se comporter de la même manière que le premier.

Cette opération peut se réaliser simplement en dupliquant le sprite Egg...



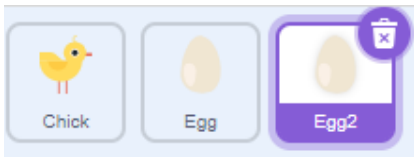
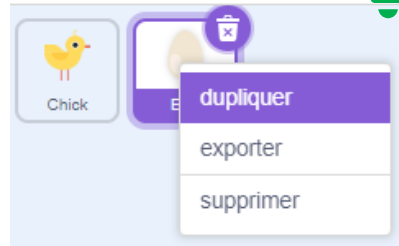
Rappel important !

Quand on duplique un sprite, on duplique aussi son code et tous ses costumes.

Ajoute un deuxième œuf



Clique droit sur le sprite *Egg* et ensuite sur *dupliquer*. En dupliquant, tu crées une réplique du sprite mais tu reproduis aussi ses scripts (le code).



Retarde le départ du deuxième œuf d'une seconde en ajoutant un bloc *attendre 1 seconde* (1) au démarrage.



Teste ton code !

Ajoute un message « perdu »



Lorsque le poussin touche l'œuf, l'œuf change de costume, le jeu s'arrête et il faudrait en plus qu'un message apparaisse : « Perdu ».

Le texte « Perdu » peut être un nouveau sprite ou ajouté sur un nouvel arrière-plan.

Ajouter un message « perdu »

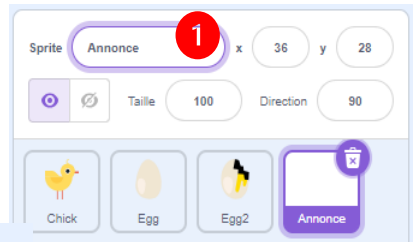


Crée un **nouveau sprite** par **peindre** et dessine nouveau un costume.

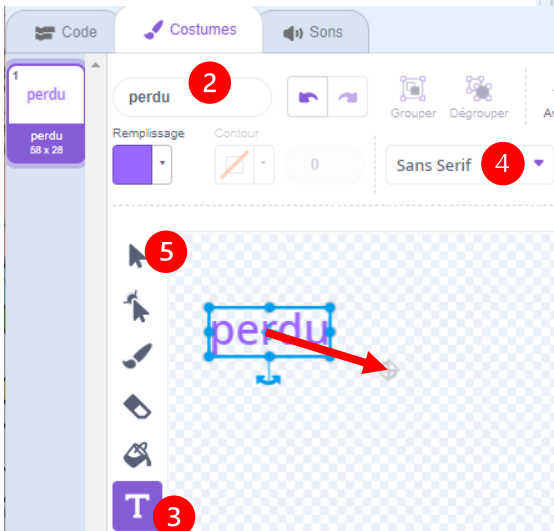


Renomme le nouveau sprite **1** **Annonce** et le costume en

2 **perdu**. Clique sur le bouton



3 **Texte** et ensuite dans la fenêtre d'édition. Tape le texte **perdu**.



Modifie la couleur de **Remplissage**, positionne le texte au milieu de la fenêtre et choisis la **4** police que tu préfères.

Faire apparaître « perdu »



Positionne **perdu** au milieu de la scène et cache-le au démarrage.

« Perdu » apparaît quand un œuf casse...

Ajoute ce code au sprite **Annonce** et crée un nouveau message ❶ **œuf cassé**.

Envoyer le message **œuf cassé**

Retourne dans le code de ton premier sprite **Egg** et ajoute un bloc ❷ **envoyer à tous**.

Supprime le bloc **stop tout**.

Fais de même pour le deuxième œuf.

Ajouter un score



Compte le nombre d'œufs qui arrivent indemnes à gauche de la scène.



Pour cela, il faudra créer une variable score, l'initialiser (= la mettre à zéro au début du jeu) et ensuite y ajouter 1 chaque fois qu'un œuf arrive à gauche de la scène.

Crée une variable « score »



Crée une **1** *variable* et nomme-là *score*. La case qui précède le bloc doit être **2** cochée pour que le score apparaisse sur la scène.



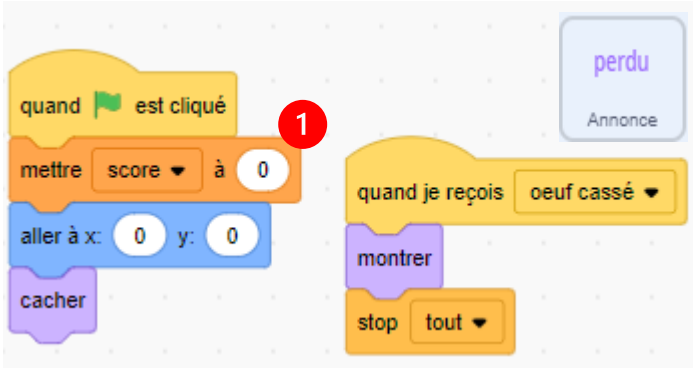


Ajouter un score

Initialise la variable score

Quand une partie commence, le score est remis à zéro.

Ajoute le bloc **mettre score à 0** dans le code du sprite **Annonce**.



Ajoute un score

Ajoute un point quand l'œuf a terminé une traversée.

Dans le premier script de **Egg**, ajoute le bloc **ajouter 1 à score** et fait de même pour le deuxième œuf. En plaçant ce bloc à cet endroit, un point sera ajouté à **score** chaque fois qu'un œuf arrivera à l'extrême gauche de la scène.

