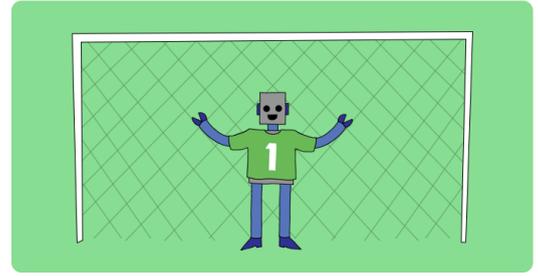


Battre le gardien de but

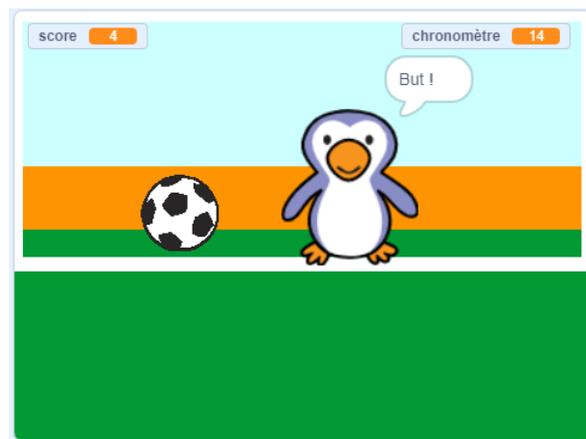
Combien de buts peux-tu marquer en 30 secondes ?



Étape 1 Introduction

Dans ce projet, tu apprendras à créer un jeu de football à 2 joueurs dans lequel tu devras marquer autant de buts que possible en 30 secondes.

Ce que tu feras



Ce que tu vas apprendre

- Se rappeler comment utiliser du code pour répondre aux pressions sur les touches
- Utiliser des blocs de détection pour détecter lorsque les sprites se touchent
- Utiliser des blocs de diffusion pour communiquer entre les sprites



Ce que tu auras besoin

Matériel informatique

- Un ordinateur capable d'exécuter Scratch 3

Logiciel

- Scratch 3 (soit **en ligne** (<https://rpf.io/scratchon>) soit **hors-ligne** (<https://rpf.io/scratchoff>))

Téléchargements

Le projet de démarrage peut être trouvé **ici** (<https://rpf.io/p/fr-FR/beat-the-goalie-go>).



Informations supplémentaires pour les éducateurs

Vous pouvez trouver le **projet terminé** **ici** (<https://rpf.io/p/fr-FR/beat-the-goalie-get>).

Étape 2 Faire bouger la balle de foot

Nous allons programmer la balle pour qu'elle se déplace en bas de la scène.

Ouvre le projet Scratch de démarrage « Battre le gardien de but ».



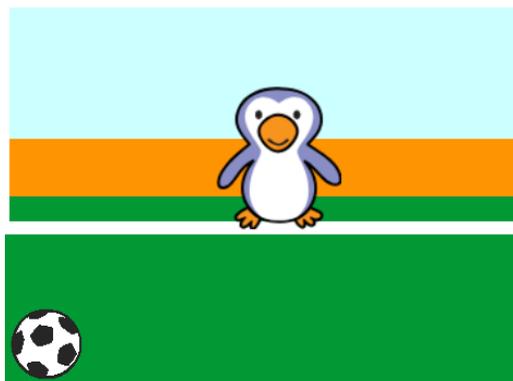
En ligne : ouvre le projet de démarrage à scratch.mit.edu/projects/406207989 (<https://scratch.mit.edu/projects/406207989>).

Si tu as un compte Scratch, tu peux en créer une copie en cliquant sur **Remix**.

Hors-ligne : ouvre le **projet de démarrage** (<https://rpf.io/p/fr-FR/beat-the-goalie-go>) dans l'éditeur hors-ligne.

Si tu dois télécharger et installer l'éditeur hors-ligne Scratch, tu peux le trouver à rpf.io/scratchoff (<https://rpf.io/scratchoff>).

Dans le projet de démarrage, tu devrais voir un arrière-plan de but, un ballon de foot et un sprite de gardien.



Clique sur ton sprite de ballon de foot. Ajoute ce code, de sorte que la balle de foot se déplace en bas de l'écran jusqu'à ce que la barre d'espace soit enfoncée.



quand  est cliqué

répéter jusqu'à ce que touche pressée ?

avancer de pas

rebondir si le bord est atteint

Clique sur le drapeau vert pour tester ton projet. Ta balle de foot devrait rebondir en bas de l'écran jusqu'à ce que la barre d'espace soit enfoncée.



Ajoute ce code à ton sprite de ballon de foot, afin que la balle de foot se dirige vers le but après que la barre d'espace ait été pressée.



```
quand  est cliqué
  répéter jusqu'à ce que  space  pressée ?
    avancer de 10 pas
    rebondir si le bord est atteint
  répéter 15 fois
    ajouter 10 à y
```

Clique sur le drapeau vert pour tester ton code. Cette fois-ci, appuie sur la barre d'espace et ta balle de foot devrait aller vers le but.



Clique sur le drapeau vert pour tester ton code. Que se passe-t-il si tu cliques sur le drapeau une deuxième fois ? Peux-tu résoudre ce problème ?



? J'ai besoin d'un indice

Ton code devrait ressembler à ceci :



```
quand drapeau vert est cliqué
  aller à x: -200 y: -140
  répéter jusqu'à ce que touche espace pressée ?
    avancer de 10 pas
    rebondir si le bord est atteint
  répéter 15 fois
    ajouter 10 à y
```

Étape 3 C'est but ?

Une fois que le ballon a atteint le but, il y a une décision à prendre. **Si** le ballon touche le gardien de but alors il a été sauvé, **sinon** c'est but.

Ajoute ce code à la fin de ton code de sprite de ballon de foot, afin que tu puisses vérifier si la balle touche le gardien. 



```
répéter 15 fois
  ajouter 10 à y
si touche le Gardien ? alors
sinon
```

Démarre le son « rattle » si le gardien a sauvé la balle. 



```
si touche le Gardien ? alors
  jouer le son rattle
sinon
```

Tu peux également diffuser un message au gardien, afin qu'ils puissent te dire que la balle a été sauvée.



Diffuser un message « sauvé » lorsque la balle a été sauvée.



```
si touche le Gardien ? alors
  jouer le son rattle
  envoyer à tous sauvé
sinon
```

Tu peux maintenant coder ton gardien de but pour dire « Sauvé ! » quand il reçoit le message.



```
quand je reçois sauvé
  dire Sauvé ! pendant 1 secondes
```

Teste ton code en essayant de marquer un but. Si ton gardien a sauvé son but, il devrait dire « Sauvé ! ».



Étape 4 But !

Peux-tu jouer un son et coder ton gardien pour dire « But ! » quand un but a été marqué ?



Rappelle-toi qu'un but a été marqué si le ballon ne touche pas le gardien.



? J'ai besoin d'un indice

Ton code devrait ressembler à ceci :



```
si touche le Gardien ? alors
  jouer le son rattle
  envoyer à tous sauvé
sinon
  jouer le son cheer
  envoyer à tous but
```



quand je reçois but ▼

dire But ! pendant 1 secondes

Étape 5 Ajouter un chronomètre

Ajoutons un chronomètre, pour que le joueur doive marquer autant de buts que possible en 30 secondes.

Créer une nouvelle variable appelée chronomètre.

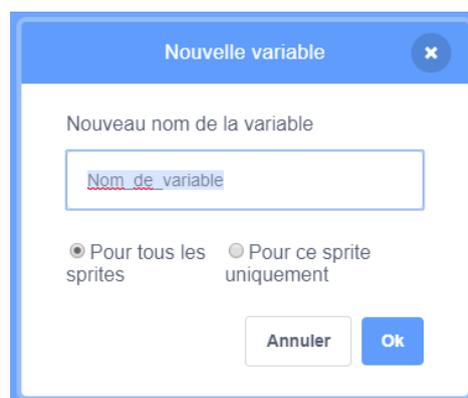


Ajoute une variable dans Scratch

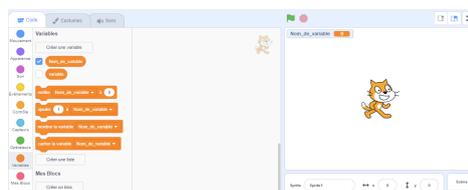
- Clique sur **Variables** dans l'onglet Scripts puis clique sur **Créer une variable**.



- Écris le nom de ta variable. Tu peux choisir si ta variable est disponible pour tous les lutins ou seulement pour celui ci. Appuie sur **OK**.



- Une fois que tu as créé la variable, elle s'affiche sur la scène, ou tu peux la décocher dans l'onglet Scripts pour la masquer.



Clique sur ta **scène**, et ajoute ce code pour définir le **chronomètre** à 30 au début du jeu.



```
quand [drapeau] est cliqué
mettre chronomètre à 30
```

Ensuite, tu devras ajouter un bloc **répéter jusqu'à**, afin que le chronomètre puisse s'exécuter jusqu'à ce qu'il atteigne 0.



```
quand [drapeau] est cliqué
mettre chronomètre à 30
répéter jusqu'à ce que chronomètre = 0
```

Réduis ton chronomètre de 1 seconde jusqu'à ce qu'il atteigne 0.



```
graph TD
    Start[quand drapeau est cliqué] --> Set[mettre chronomètre à 30]
    Set --> Loop[répéter jusqu'à ce que chronomètre = 0]
    Loop --> Wait[attendre 1 secondes]
    Wait --> Dec[ajouter -1 à chronomètre]
    Dec --> Loop
```

Une fois que le chronomètre a atteint 0, tu devras jouer le son « whistle » puis arrêter le jeu.



```
quand drapeau est cliqué
mettre chronomètre à 30
répéter jusqu'à ce que chronomètre = 0
  attendre 1 secondes
  ajouter -1 à chronomètre
jouer le son whistle jusqu'au bout
stop all
```

Clique sur le drapeau vert pour tester ton code. Ton chronomètre devrait commencer à 30 et se terminer à 0.



Tu peux changer ton chronomètre pour commencer à 10 si tu ne veux pas attendre 30 secondes !

Tu as seulement la chance de marquer 1 but ! Pour avoir plus de 1 chance, ajoute un bloc **répéter indéfiniment** autour de ton code **balle de foot**. Tu peux également ajouter un bloc **attendre** entre les tentatives.



```
quand [drapeau] est cliqué
  répéter indéfiniment
    aller à x: -200 y: -140
    répéter jusqu'à ce que [touche space ▼] pressée ?
      avancer de 10 pas
      rebondir si le bord est atteint
    attendre 1 secondes
  répéter 15 fois
    ajouter 10 à y
  si [touche le Gardien ▼ ?] alors
    jouer le son rattle ▼
    envoyer à tous sauvé ▼
  sinon
    jouer le son cheer ▼
    envoyer à tous but ▼
```

Défi : ajouter un score

Peux-tu ajouter une variable `score` à ton jeu ?

- Tu devras définir le score à 0 au début de la partie ;
- Tu devras ajouter 1 à ton score chaque fois qu'un but est marqué.

Tu auras besoin de ces blocs :



Étape 6 Contrôler le gardien de but

C'est beaucoup trop facile de marquer un but ! Laissons un deuxième joueur essayer de sauver des buts.

Clique sur ton sprite **Gardien** et ajoute ce code pour changer la position x du gardien lorsque tu appuies sur la flèche de gauche.



The code consists of two blocks. The first is a green 'when key pressed' block with 'left arrow' selected in the dropdown menu. The second is a blue 'add -10 to x' block, which is nested under the 'when key pressed' block.

Appuie sur la flèche de gauche pour tester ton nouveau code. Ton gardien devrait se déplacer vers la gauche.

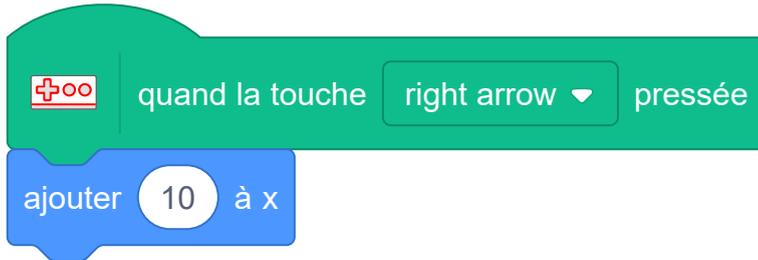


Utilise des blocs similaires à ceux ci-dessus pour que le **Gardien** se déplace vers la droite lorsque la touche fléchée droite est enfoncée.



J'ai besoin d'un indice

Ton code devrait ressembler à ceci :



Défi : contrôle manuel

Au lieu de déplacer la balle vers la gauche et la droite automatiquement, peux-tu permettre à ton joueur de contrôler la balle avec les touches **q** et **d** ?

Pour ce faire, tu devras supprimer le code permettant de déplacer la balle vers la gauche et la droite.

```

quand [drapeau] est cliqué
  répéter indéfiniment
    aller à x: -200 y: -140
    répéter jusqu'à ce que [touche space ▼] pressée ?
      avancer de 10 pas
      rebondir si le bord est atteint
    répéter 15 fois
      ajouter 10 à y
    si [touche le Gardien ▼ ?] alors
      jouer le son rattle ▼
      envoyer à tous sauvé ▼
    sinon
      jouer le son cheer ▼
      envoyer à tous but ▼
  
```

Tu peux alors ajouter du code pour déplacer la balle lorsque les touches sont enfoncées. Voici quelques blocs de code pour t'aider :

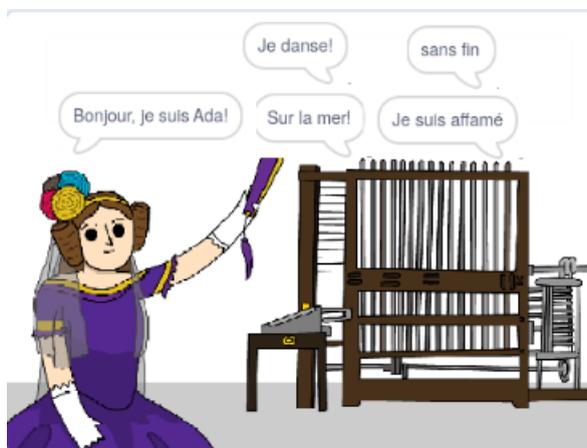
ajouter -5 à x

si alors

touche a ▼ pressée ?

Étape 7 Et ensuite ?

Jetez un coup d'œil au projet **Générateur de poésie** (<https://projects.raspberrypi.org/fr-FR/projects/poetry-generator>) d'Ada.



Ce projet a été traduit par des bénévoles:

Jonathan Vannieuwerkerke

Michel Arnols

Grâce aux bénévoles, nous pouvons donner aux gens du monde entier la chance d'apprendre dans leur propre langue. Vous pouvez nous aider à atteindre plus de personnes en vous portant volontaire pour la traduction - plus d'informations sur rpf.io/translate (<https://rpf.io/translate>).

Publié par Raspberry Pi Foundation (<https://www.raspberrypi.org>) Sous un Creative Commons license (<https://creativecommons.org/licenses/by-sa/4.0/>).

Voir le projet et la licence sur GitHub (<https://github.com/RaspberryPiLearning/beat-the-goalie>)