

CASSE-BRIQUES

Scratch



CASSE-BRIQUES – CREATION MUR

PRINCIPE

Le mur est construit par le clonage d'une brique originale.

Nous utiliserons 2 boucles imbriquées :

1 boucle création ligne

1 boucle de répétition création ligne.

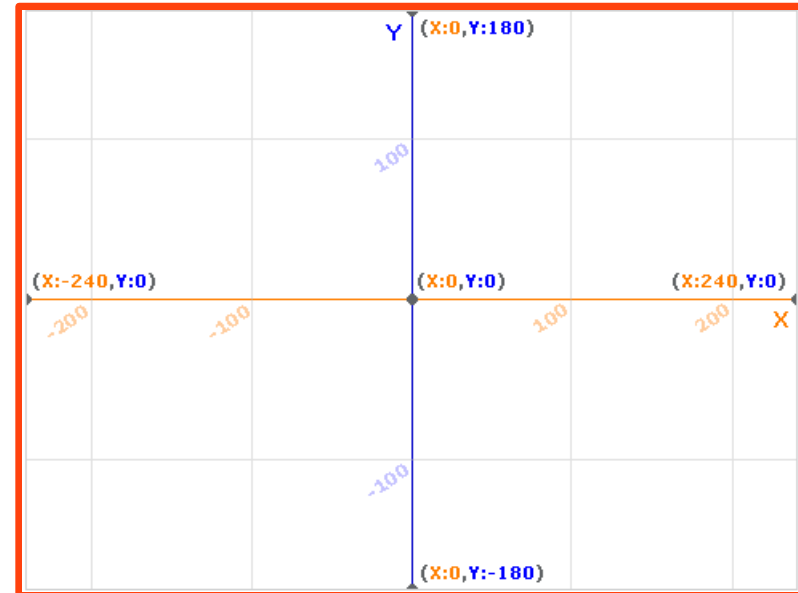
Les briques sont espacées de **25 horizontalement** et **25 verticalement**.

Le mur est rempli ligne par ligne.

Position début ligne : **X = -226** et **Y = 166**

Attention, si vous changez la taille de la brique, il sera nécessaire de recalculer les écarts et la position de départ.

SYSTEME DE COORDONNEES



ALGORITHMME

Définir position départ X

Définir position départ Y

Répéter 8 fois

Répéter 19 fois

Créer 1 clone

Déplacer 25 X droite

Fin répéter

Positionnement début ligne X

Déplacer 25 Y bas

Fin répéter

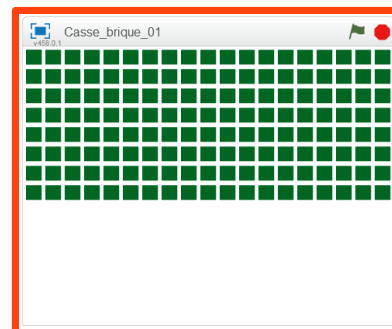
ELEMENTS DU PROGRAMME

Lutin (brique)

Contrôle (Boucle répéter, clone)

Variable (X_départ, Y_départ, chargement)

Mouvement (aller à...)



CODE SOURCE PROGRAMME

```
quand est cliqué
mettre Chargement à 0
mettre X_position à -226
mettre Y_position à 166
répéter 8 fois
  répéter 19 fois
    créer un clone de moi-même
    mettre X_position à -226
    ajouter à Y_position -25
  mettre Chargement à 1
quand je commence comme un clone
aller à x: X_position y: Y_position
ajouter à X_position 25
```

CASSE-BRIQUES – DEPLACEMENT RAQUETTE

PRINCIPE

Nous allons utiliser les touches du clavier pour déplacer la raquette. Lorsque l'utilisateur appuie sur la flèche de gauche, la raquette se déplace vers la gauche. Même chose pour le déplacement à droite avec la flèche droite. Pour ce faire, nous allons tester de façon permanente si ces touches sont enfoncées. Si c'est le cas, nous déplaçons la raquette.

ELEMENTS DU PROGRAMME

Lutin (brique)

Contrôle (Boucle répéter, condition SI)

Capteur (Touche pressée)

Mouvement (Ajouter à, Aller à)

ALGORITHMME

Positionner la raquette

Répéter à l'infini

Si flèche gauche alors

 Déplacer la raquette à gauche

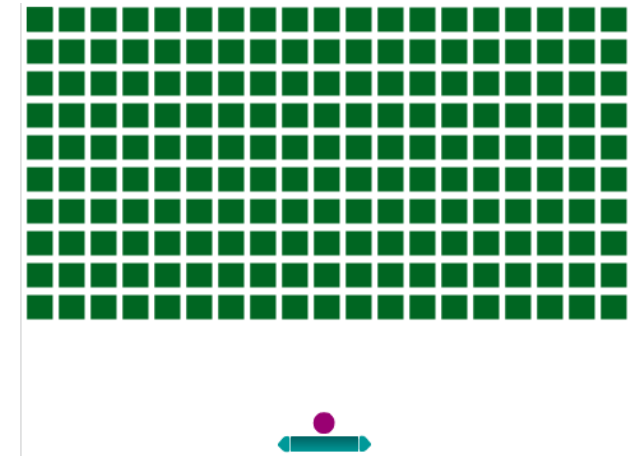
Fin si

Si flèche droite alors

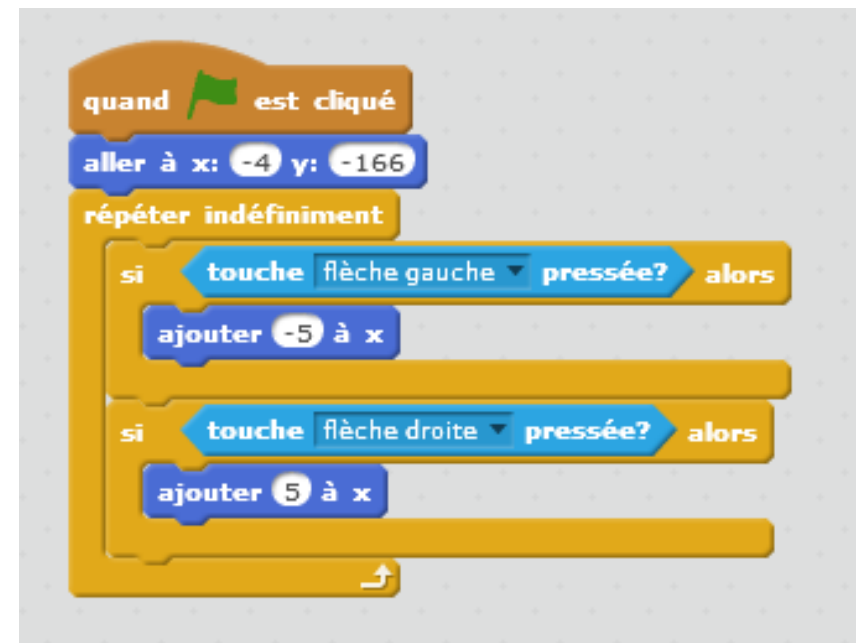
 Déplacer la raquette à droite

Fin si

Fin répéter



CODE SOURCE PROGRAMME



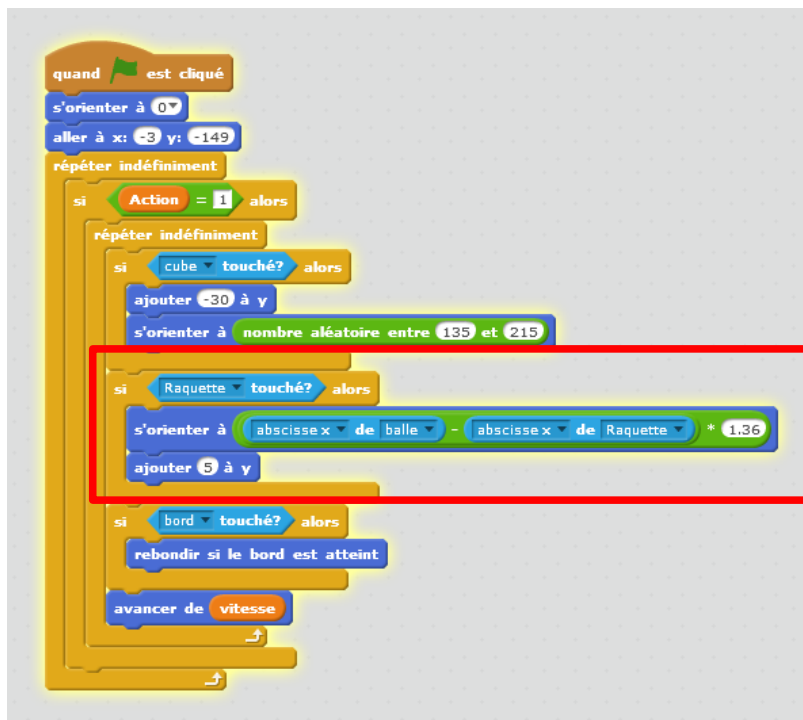
CASSE-BRIQUES – GESTION DE LA BALLE 1

PRINCIPE

Nous allons ici gérer le comportement de la balle. La balle touche plusieurs éléments : Les briques, la raquette et les parois du jeu. Nous allons donc tester en permanence le contact avec ces éléments.

- Si la balle touche les bords alors elle rebondie
- Si la balle touche la raquette alors elle est renvoyée avec un angle qui peut varier en fonction du point de contact.
- Si la balle touche une brique alors la brique est détruite et la balle rebondie.
- La balle avance en permanence (vitesse).

CODE SOURCE PROGRAMME (BALLE)



REBOND BALLE SUR RAQUETTE

L'angle de rebond varie en fonction de la position de contact sur la raquette. Pour chaque contact, nous calculons l'écart entre la position horizontale de la balle et celle du centre de la raquette. Nous multiplions ensuite cet écart avec un coefficient. Ainsi, plus l'écart est important, plus l'angle de renvoi va être important. Pour déterminer le coefficient, nous avons appliqué une règle de trois avec comme paramètre un renvoi à 45° ou -45° sur les extrémités de la raquette.

Remarquez également qu'à chaque contact avec la raquette, nous déplaçons la balle de 5 vers le haut. Il s'agit ici de ne pas laisser la balle sur la raquette après le premier contact.

ELEMENTS DU PROGRAMME

Lutin (raquette, balle, brique)

Contrôle (Boucle répéter, Si)

Variable (X_départ, Y_départ, action, vitesse)

Mouvement (aller à..., s'orienter, avancer)

Capteur (objet touché)

ALGORITHME GENERAL

Positionner la balle

Répéter à l'infini

SI Cube touché **alors**

Détruire cube

Rebondir

Fin SI

SI raquette touchée **alors**

Renvoyer avec angle

Fin SI

SI bord touché **alors**

Rebondir

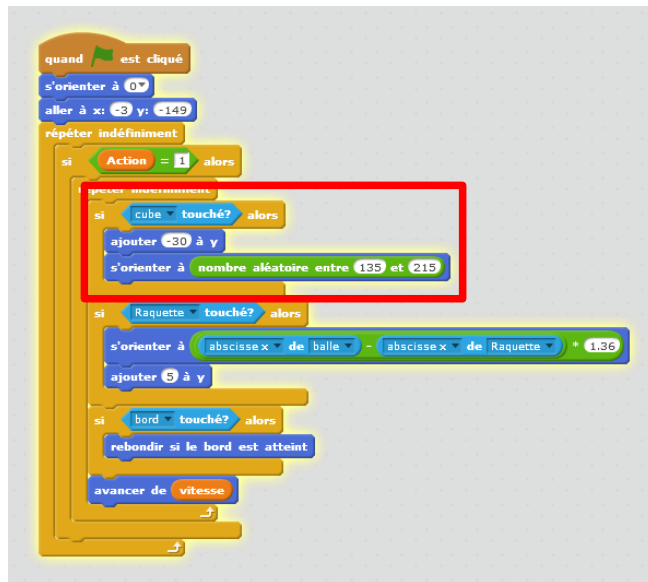
Fin SI

Avancer

Fin répéter

CASSE-BRIQUES – GESTION DE LA BALLE 2

CODE SOURCE PROGRAMME (BALLE)



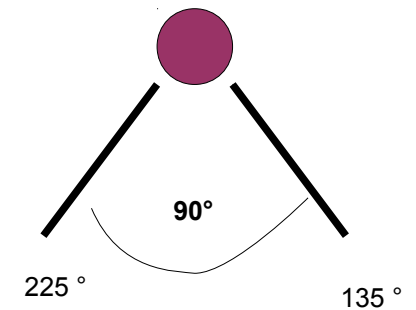
CODE SOURCE PROGRAMME (BRIQUE)



Les contact de la balle sont gérés au niveau du code de 2 lutins :
LA BRIQUE ET LA BALLE

REBOND BALLE SUR BRIQUE (BALLE)

Lorsque la balle touche une brique elle est renvoyée vers le bas. Il suffit donc de changer sa direction. Grâce à la fonction aléatoire, nous renvoyons la balle selon un angle compris entre **135 et 225** degrés. Le rebond va ainsi changer pour chaque brique. L'angle de renvoi sera donc pris dans un angle de **90° vers le bas**



REBOND BALLE SUR BRIQUE (BRIQUE)

La suppression de la brique touchée est géré au niveau du lutin brique. Pour rappel, les briques du mur sont des clones de la brique originale. pour chaque création d'un clone de brique, nous allons tester en permanence si la balle est touchée grace à une **boucle infinie**. Si c'est le cas, nous détruisons la brique. Ainsi le contact avec la balle est géré de façon indépendante pour chaque brique.

REBOND BALLE SUR BORD (BALLE)

Dans le jeu, la balle doit rebondir lorsqu'elle touche les bords. Scratch propose un capteur **bord touché**. Lorsque le bord est touché, on utilise alors la fonction «**Rebondir si bord atteint**» qui va renvoyer la balle avec un angle aléatoire.

Nous pourrions améliorer le jeu et augmenter la difficulté en supprimant le rebond sur le bord du bas. Si la balle touche ce bord alors le jeu se termine ou une faute est comptabilisée.

CASSE-BRIQUES – Pour aller plus loin...

SORTIE DE LA BALLE

Si la balle touche le bord inférieur alors le joueur perd une vie. La partie reprend jusqu'à ce que le joueur n'ai plus de vie.

Indices

Vous devrez utiliser une variable vie pour gérer le niveau de vie du joueur. Lorsque la balle sort, j'enlève 1 à cette variable. Au début de la partie, je l'initialise avec le nombre de vie que je compte donner à mon joueur.

GESTION DU SCORE

Dans un jeu il y a toujours un score. Le principe est simple; pour chaque brique détruite, le joueur marque un point. Lorsque le joueur atteint la fin du jeu, vous devez afficher son score.

Indices

Créer une variable score. pour chaque brique détruite, on ajoute 1 point au score. N'oubliez pas de remettre le score à 0 en début de partie.

SONORISER VOTRE JEU

Il n'y a pas de bon jeu vidéo sans une bonne bande son. pour rendre votre jeu plus attractif, rajouter des bruitages sonores sur les actions clefs (balle qui touche les bords, la raquette, destruction d'une brique, sortie de la balle, musique d'ambiance.

Indices

Importer de nouveaux sons dans votre programme. jouez les sons grâce aux fonctions de son de Scratch.

GESTION DES ECRANS

Un jeu vidéo comporte toujours plusieurs écrans. Le joueur débute par un écran d'accueil qui lui explique les règles ou lui permet de choisir le niveau de difficulté. Lorsque la partie est terminée, l'écran de fin apparaît alors pour annoncer le score et faire le bilan de la partie. Alors à vous de jouer. Ajoutez votre écran d'accueil et votre écran de fin. N'oubliez pas les boutons pour démarrer la partie ou changer d'écran.

Indices

Scratch n'est pas une application qui gère les écrans. Pour palier à ce problème, affichez ou cachez les objets de votre programme en fonction des écrans théoriques que vous aurez créés. Utilisez les fonctions de messages de Scratch pour faire communiquer les objets entre eux.

