

1

Tester une égalité

Écrire un programme dans lequel un lutin demande le résultat de 9×6 . Si la réponse est correcte, le lutin répond « Bravo ! », dans le cas contraire, le lutin répond « Mauvaise réponse ».

Demander "Combien font 9 fois 6 ?"

Si **réponse** = 54 alors :

Dire "Bravo !"

Sinon

Dire "Mauvaise réponse"

Fin du test

Algorithme

Image 1

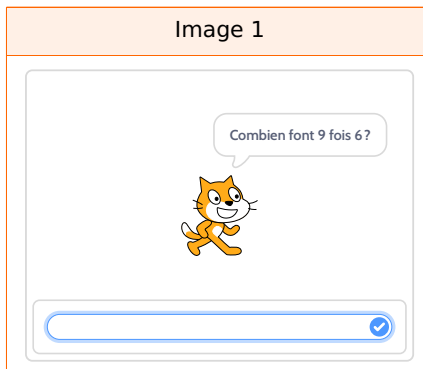
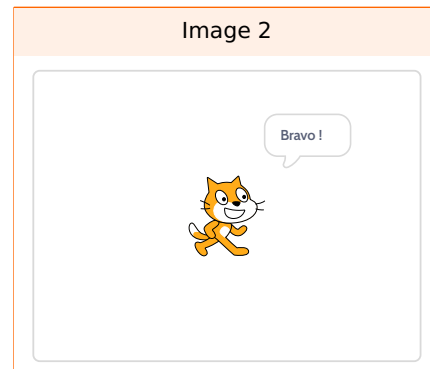
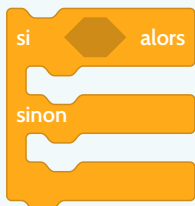


Image 2



Les blocs suivants s'avéreront utiles dans la réalisation du programme demandé :

Aide



Ce bloc permet de tester une condition (écrite entre « si » et « alors ») et exécute les instructions situées après « alors » et avant « sinon » lorsque la condition est réalisée ou bien les instructions situées après « sinon » quand la condition est fausse.

Aide



Ce bloc permet de tester une égalité et renvoie *vrai* lorsque les deux valeurs sont égales ou *faux* sinon.

2

La table des 9

Il s'agit de modifier le programme précédent pour tester, au hasard, un produit de la table des 9.

Il faut donc créer une variable qui contiendra un nombre entier, compris entre 1 et 10, choisi au hasard par l'ordinateur.

Image 3



Initialiser la variable **facteur** à nombre aléatoire entre 1 et 10

Demander "Combien font 9 fois " **facteur**

Si **réponse** = $9 \times$ **facteur** alors :

Dire "Bravo !"

Sinon

Dire "Mauvaise réponse"

Fin du test

Algorithme

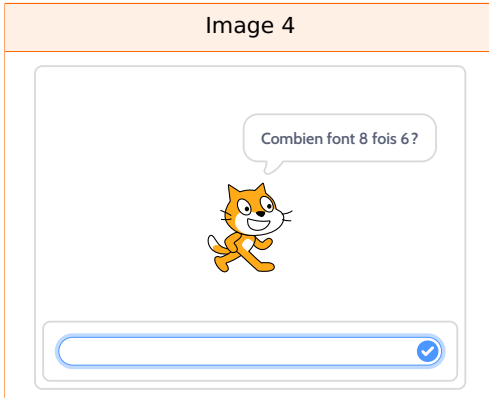
nombre aléatoire entre 1 et 10

Ce bloc choisit aléatoirement un nombre entier compris entre les deux valeurs spécifiées (incluses).

3

Généralisation

Image 4



Modifier le programme précédent de telle sorte que chacun des deux facteurs soit un nombre entier, compris entre 1 et 10, choisi au hasard par l'ordinateur.

Pour ce faire, il suffit de créer une seconde variable aléatoire afin d'y stocker la valeur de l'autre facteur.

4

Améliorations possibles

Quelques améliorations sont encore possibles, par exemple :

- reprendre le programme précédent en répétant 10 fois les questions sur les tables ;
- ajouter un compteur (en utilisant une troisième variable) pour établir le score de réussite à l'issue de la série des 10 calculs ;
- ajouter un autre compteur pour numérotter les questions ;
- choisir un arrière plan pour la scène ;
- modifier le costume du lutin en fonction de la phase de jeu.

Image 5



On pourra, par exemple, choisir le lutin *Tera* et opter pour les costumes suivants :



Question



Bonne réponse



Mauvaise réponse

basculer sur le costume costume1

Ce bloc permet de modifier le costume du lutin : on peut fournir le nom ou le numéro du costume (ce qui permet d'utiliser une variable).

À retenir

- Dans Scratch, les structures de **tests** sont définies à l'aide des blocs

