



Variables et listes

Avez-vous utilisé des blocs Variable ou Liste dans un projet Scratch ? Ils peuvent constituer un outil puisant lorsque vous créez des programmes personnalisés. Peut-être avez-vous utilisé une variable pour stocker un score de jeu*, mais savez-vous qu'une variable peut contenir des nombres ou du texte (également appelé « chaîne ») ? Et si vous stockez beaucoup d'informations personnalisées, l'utilisation d'une liste peut s'avérer plus efficace que la création de plusieurs variables. Explorons ce que sont les variables et les listes, et voyons plusieurs façons de les utiliser.

*Consultez nos didacticiels intégrés à l'éditeur ou nos fiches de codage pour obtenir des instructions sur la configuration d'une partition de base.

Dans ce guide, vous trouverez :

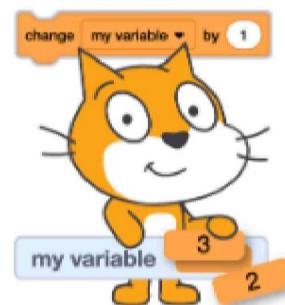
Qu'est-ce qu'une variable et une liste ?
• Blocs Scratch Reporter • Créer votre propre variable • Utilisation d'e variables • Variables de curseur

Afficher ou masquer des variables ou des listes

Qu'est-ce qu'une variable et une liste ?

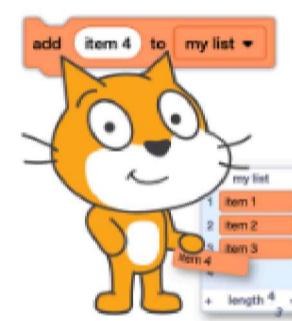
Vous pouvez considérer une variable comme un conteneur qui stocke une valeur/données/informations assignée. Les variables ne peuvent contenir qu'une seule information à la fois, donc à mesure que de nouvelles informations sont placées à l'intérieur de la variable, les anciennes informations sont remplacées.

Une liste, en revanche, peut stocker plusieurs informations, toutes classées comme des éléments distincts. Une liste est également appelée « tableau » dans d'autres langages de programmation.



L'avantage de stocker des informations dans une variable ou une liste est qu'elles peuvent être rappelées ultérieurement dans votre programme. Par exemple, vous pouvez :

demander au programme de vérifier le score actuel du jeu et de l'utiliser pour déterminer si un joueur a gagné ou perdu. • enregistrer une série de réponses fournies aux questions et reformuler les informations recueillies plus tard sous la forme d'une phrase.



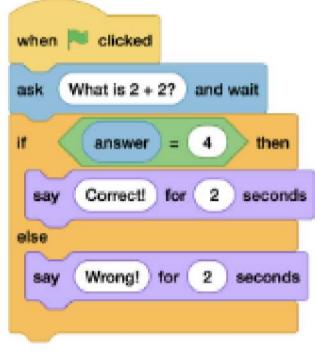
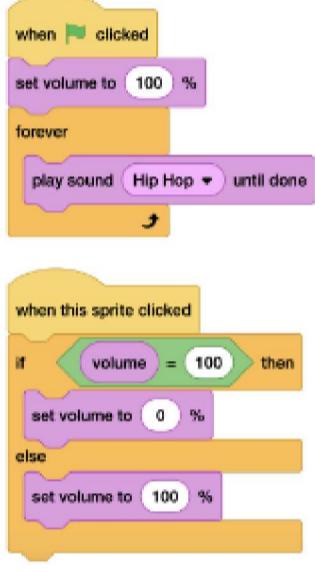
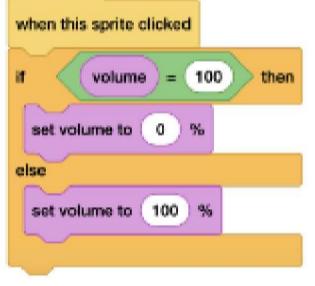
Les informations stockées peuvent changer la direction d'un jeu ou d'une histoire, activer et désactiver des actions, signaler des messages spéciaux, et bien plus encore !



Créé par la Scratch Foundation (scratchfoundation.org). Partagé sous la licence publique internationale Creative Commons Attribution-ShareAlike 4.0 (CCbySA 4.0).

Blocs Scratch Reporter

Les variables et les listes contiennent des informations que vous pouvez utiliser dans votre programme, mais Scratch est livré avec des blocs de rapport intégrés qui stockent également des informations. Contrairement à un bloc de pile, qui peut être placé directement au-dessus ou en dessous d'un autre bloc, les blocs rapporteurs vont à l'intérieur d'un autre bloc pour servir d'entrée, d'où leur forme ovale. Voici quelques exemples :

<p>Poser une question et répondre à la réponse Vous pouvez utiliser le bloc « demander » pour poser une question à un utilisateur. Lorsque l'utilisateur saisit une réponse dans la boîte de dialogue, elle est stockée dans un bloc de rapport appelé « réponse ». Vous pouvez ensuite utiliser une instruction conditionnelle « si alors » pour afficher une réponse si la réponse est correcte et une réponse différente si la réponse est incorrecte.</p>	
<p>Ajustez le volume. Un autre bloc de rapporteur est « volume ». Il stocke le numéro représentant le volume actuel du sprite, du clone ou de la scène. Vous pourriez écrire un programme vérifiant le volume d'un sprite lorsque l'utilisateur clique sur le sprite. Ensuite, utilisez une instruction conditionnelle « si alors » pour que le script règle le volume à 0 % ou 100 % (activant ou désactivant le son) en fonction du nombre actuel.</p>	
<p>Selon vous, quelle est la différence entre le bloc rapporteur « volume » dans la catégorie Son et le bloc rapporteur « loudness » dans la catégorie Détection ? « Loudness » indique le niveau de puissance de l'entrée du microphone. Vous pouvez essayer d'utiliser le « volume » pour contrôler l'esprit. Par exemple, à mesure que le bruit dans la pièce devient plus fort, votre projet pourrait demander au sprite de changer de costume ou vous pourriez programmer votre sprite pour qu'il se déplace à différentes distances en fonction du niveau de volume.</p>	

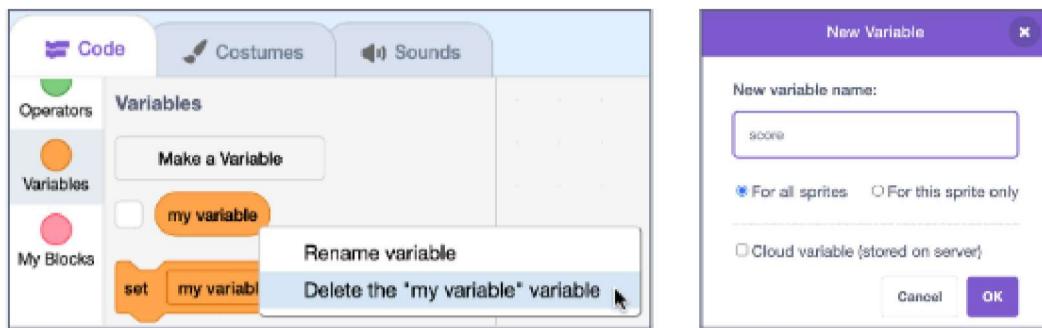


Créé par la Scratch Foundation (scratchfoundation.org). Partagé sous la licence publique internationale Creative Commons Attribution-ShareAlike 4.0 (CCbySA 4.0).

Créez votre propre variable

Que se passe-t-il s'il n'existe pas de bloc de rapporteur intégré pour stocker les informations dont vous avez besoin ? Dans ce cas, vous pouvez créer votre propre variable ! Sélectionnez la catégorie « Variable » dans la palette de blocs. Une variable générique appelée « ma variable » est déjà fournie.

Vous pouvez le renommer ou le supprimer en cliquant dessus avec le bouton droit. Ou cliquez sur la case « Créer une variable » pour en créer une nouvelle. Lorsque la zone de texte « Nouvelle variable » apparaît, tapez le nom des informations que vous souhaitez stocker dans la zone de texte pour le nom.

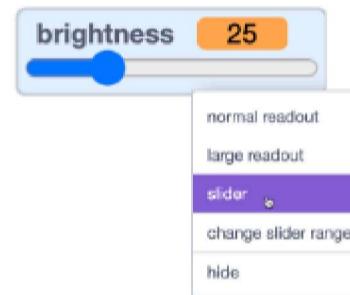


Vous pouvez alors choisir si tous les sprites auront accès à la variable, ou seulement celui que vous avez actuellement sélectionné. Plusieurs fois, vous choisirez probablement de le laisser par défaut « pour tous les sprites », afin que n'importe quel sprite de votre programme puisse modifier la variable, réinitialiser la variable ou vérifier les données que la variable contient actuellement (par exemple, un score global de jeu). Cependant, il peut arriver qu'une variable ne s'applique qu'à un seul sprite. Par exemple, lorsque différents sprites peuvent avoir leur propre jauge de santé dans un jeu. Si vous choisissez d'utiliser les variables cloud, elles sont stockées « dans le cloud » ou sur les serveurs de Scratch afin que les valeurs soient conservées lorsque les projets sont rechargés.

Notez les différents types de blocs disponibles dans la catégorie Variables. Il existe des blocs pour définir ou modifier la variable, par exemple, et même un bloc rapporteur de variable (ce bloc ovale) qui peut être utilisé à l'intérieur d'un autre bloc.

Utiliser des variables

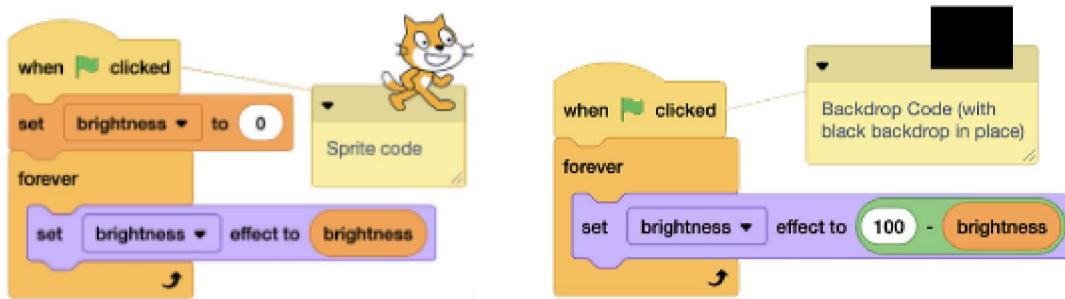
Créons un projet dans lequel la variable contrôle la luminosité du sprite. Vous verrez qu'il n'y a pas de bloc rapporteur pour la luminosité disponible dans la palette de blocs. Créez une variable appelée « luminosité » et cochez la case à côté de votre variable « luminosité » personnalisée dans la palette de blocs pour qu'elle soit affichée sur la scène. Maintenant, cliquez avec le bouton droit sur son moniteur de scène et vous verrez que vous avez quelques options : « lecture normale », « lecture large » et « curseur ». Faisons-en un curseur, afin que l'utilisateur puisse contrôler la luminosité du sprite.



Créé par la Scratch Foundation (scratchfoundation.org). Partagé sous la licence publique internationale Creative Commons Attribution-ShareAlike 4.0 (CCbySA 4.0).

Vous pouvez créer un court script qui dit : Lorsque vous cliquez sur le drapeau vert, la variable « luminosité » doit être définie sur 0. Ensuite, le programme doit définir pour toujours l'effet de luminosité sur le nombre dans la variable « luminosité ». Cliquez sur le drapeau vert et ajustez le curseur pour tester !

Et si vous souhaitez que la luminosité de la toile de fond soit à l'opposé de celle du sprite ? Vous pouvez créer une boîte noire comme costume de toile de fond, puis créer un script indiquant que la toile de fond doit toujours définir l'effet de luminosité à 100 moins ce qui est dans la variable « luminosité ». Testez à nouveau pour voir comment cela fonctionne. La luminosité de la toile de fond et du sprite doivent être opposées. Que pourriez-vous essayer d'autre ?



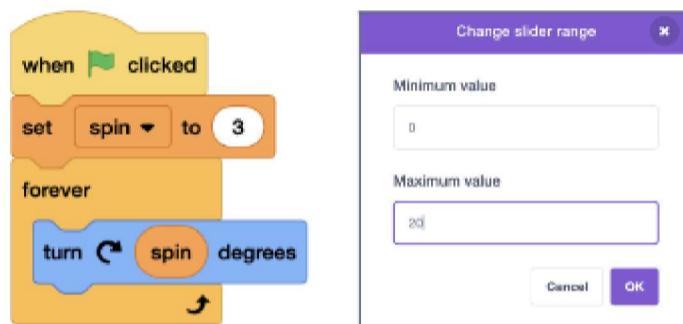
Variables du curseur

Les curseurs peuvent également être utiles pour d'autres choses, comme la vitesse ou le volume.

Si vous souhaitez définir des limites sur la plage de nombres disponibles dans le curseur, vous pouvez cliquer avec le bouton droit sur le moniteur de scène de la variable pour définir une « plage de curseur » minimum et maximum.

Par exemple, vous pouvez créer un projet avec un code comme celui ci-dessous, dans lequel la variable du curseur contrôle la vitesse de rotation en définissant la variable « spin » comme nombre de degrés à tourner.

S'il tourne trop vite, cela vous donne peut-être le vertige, vous souhaiterez donc peut-être définir un nombre maximum inférieur pour le curseur.

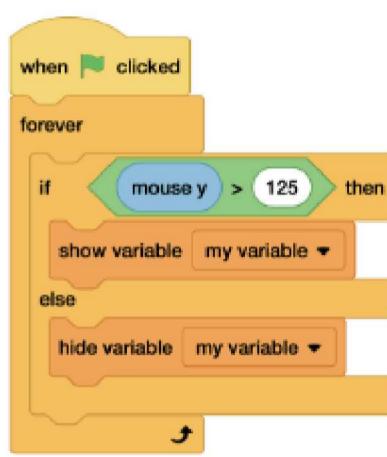


Créé par la Scratch Foundation (scratchfoundation.org). Partagé sous la licence publique internationale Creative Commons Attribution-ShareAlike 4.0 (CCbySA 4.0).

Afficher ou masquer des variables ou des listes

Utilisez les blocs « afficher la variable » et « masquer la variable » dans votre programme si vous souhaitez contrôler quand le moniteur de scène de la variable est visible. Par exemple, il pourrait démarrer de manière visible, puis se cacher lors que l'utilisateur atteint un certain point dans le programme, comme l'écran de fin. Vous pouvez utiliser des blocs similaires pour afficher et masquer des listes.

Vous pouvez également choisir d'afficher ou de masquer les variables sur la scène en fonction de quelque chose comme le placement de la souris, comme dans l'exemple ici. De cette façon, les moniteurs variables n'encombrent pas l'écran jusqu'à ce que l'utilisateur veuille apporter une modification.



Créez votre propre liste

Pour créer une liste, sélectionnez la catégorie « Variables » dans la palette de blocs et cliquez sur la case « Créez une liste ». Aucune liste par défaut n'est présente dans un nouveau projet, donc aucun bloc de liste ne sera visible jusqu'à ce que vous créeiez une liste.



Lorsque la zone de texte « Nouvelle liste » apparaît, tapez le nom des informations que vous souhaitez stocker dans la zone de texte du nom. Vous pouvez ensuite choisir si tous les sprites auront accès à la liste, ou seulement celui que vous avez sélectionné. Une fois qu'une liste est créée, un mélange de blocs de pile et de blocs rapporteurs apparaîtra sous la catégorie Variables pour que vous puissiez l'utiliser.

Une liste peut être utilisée pour stocker plusieurs éléments d'informations connexes, ou elle peut être utilisée à la place de la création de plusieurs variables.

Transmettre des données dans des variables et des listes

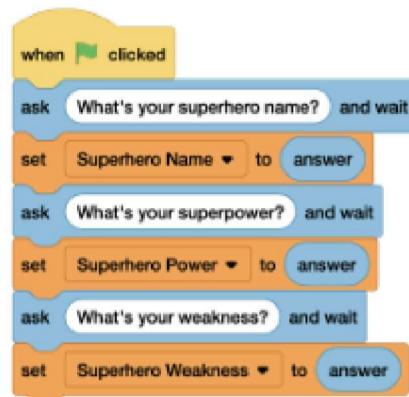
Vous pouvez transmettre des informations à une variable ou une liste en cliquant ou en déplaçant un sprite, en ajustant un curseur, via des blocs de code, et bien plus encore ! Vous pouvez également transmettre des informations d'un bloc, d'une variable ou d'une liste de rapporteur à un autre. Cela pourrait être utile car les variables et les blocs rapporteurs ne peuvent contenir qu'une seule information à la fois.

Par exemple, supposons que vous souhaitez créer un projet qui collecte les réponses de l'utilisateur à plusieurs questions. Chaque réponse peut être transmise et stockée dans des variables individuelles en utilisant le bloc « set[variable] to » pour transmettre la « réponse » dans une variable après que chaque question connexe soit posée. Cela nécessite la configuration de plusieurs variables.

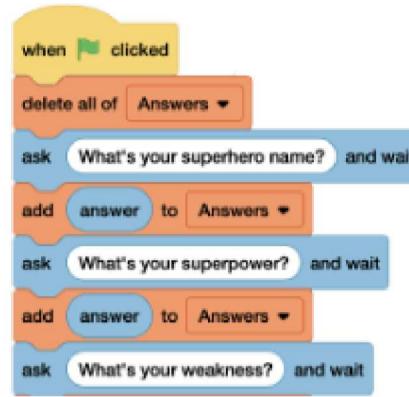


Créé par la Scratch Foundation (scratchfoundation.org). Partagé sous la licence publique internationale Creative Commons Attribution-ShareAlike 4.0 (CCbySA 4.0).

Ou bien, chaque réponse peut être ajoutée à une liste en utilisant le bloc « ajouter (chose) à (liste) », où « chose » est la « réponse ». Voir des exemples de transmission de données à des variables par rapport à la transmission à une liste dans les scripts ci-dessous. Une liste peut remplacer plusieurs variables. Avec un e liste, les nouvelles informations sont simplement ajoutées à la liste, plutôt que les données précédentes soient écrasées. Et la liste s'allonge de plus en plus jusqu'à ce que vous lui disiez de supprimer tout ou partie des informations à l'aide des blocs de code « supprimer ». Si vous collectez beaucoup de données, le stocker dans une liste peut vous faire gagner du temps.



Cette pile de codes montre la transmission des données du rapporteur « réponse » vers différentes variables après que chaque question ait été posée.

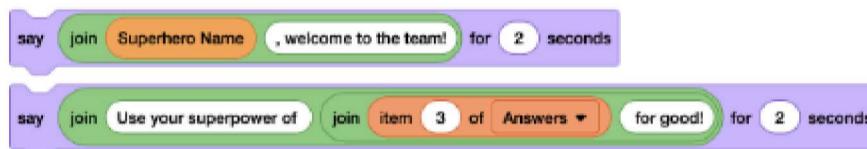


Cette pile de codes montre la transmission des données du rapporteur « réponse » dans une seule liste après que chaque question soit posée.

demande-t-il.

Données de sortie à partir de variables et de listes

Les informations stockées dans une variable ou une liste peuvent être rappelées et utilisées ultérieurement. Une liste est constituée de nombres associés à des éléments, de sorte que les informations des listes peuvent être demandées en fonction de leur emplacement dans la liste à l'aide du bloc de rapport « élément de (liste) ». Par exemple, vous pouvez partager les données sous la forme d'une phrase à l'aide du bloc « dire » pour créer un message personnalisé pour chaque utilisateur. Ou essayez d'ajouter la variable et la liste d'entrées dans d'autres blocs pour voir les résultats !



Utiliser des listes

Vous pouvez cocher une case à côté de la liste dans la palette de blocs pour l'afficher sur scène via un moniteur de scène. Lorsqu'il est visible sur la scène, vous pouvez cliquer avec le bouton droit sur le moniteur de scène et choisir d'exporter les données de la liste vers un fichier sur votre ordinateur si vous souhaitez l'enregistrer. Vous pouvez également importer des données dans une liste (à partir d'un fichier .txt ou .csv), si vous devez ajouter rapidement de nombreuses informations à la fois. Vous pouvez également cliquer sur le signe plus (+) en bas



Créé par la Scratch Foundation (scratchfoundation.org). Partagé sous la licence publique internationale Creative Commons Attribution-ShareAlike 4.0 (CCbySA 4.0).

coin gauche pour ajouter un nouvel élément à une liste via le moniteur de scène, et cliquez sur les éléments pour les modifier en tapant dans les cases du moniteur de scène.

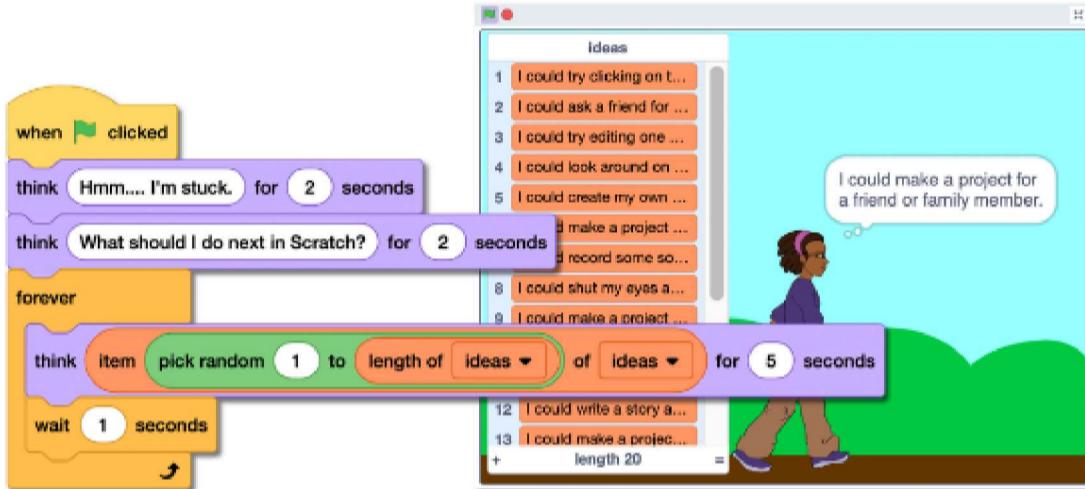
Lorsque vous créez un projet Scratch et remplissez une liste, les informations de la liste seront conservées jusqu'à ce que vous demandiez au programme de supprimer des éléments de la liste ou de supprimer la liste entière. L'état de votre projet est conservé lorsque vous l'enregistrez, ce qui signifie que s'il y a des informations dans votre liste, vous n'avez pas besoin de dire au programme d'ajouter ces mêmes éléments à votre liste à chaque fois qu'il est exécuté, sauf si vous les avez supprimés de la liste et qu'ils doivent être ajoutés.

Lorsqu'une liste a été remplie, cliquez sur les blocs du rapporteur de liste pour voir la valeur que chacun contient.

Il existe une variété d'utilisations simples et complexes des listes, telles que :

utiliser une liste prédéfinie pour déterminer une animation (comme un projet de mélodie) des lettres ou afficher des mots/sous-titres sur la scène (comme dans un projet Mad Lib) • programmer un générateur d'idées aléatoires (en utilisant plusieurs listes pleines d'adjectifs ou de catégories, lorsque vous avez besoin d'idées sur les types de projets à créer) ou une version simple où le programme sélectionne un élément aléatoire dans une liste d'idées

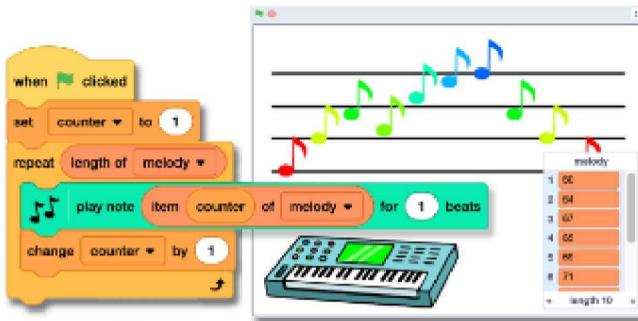
Ce ne sont là que quelques-unes des possibilités. Parcourez tous les blocs disponibles pour les listes. À votre avis, que fait chacun ? Comment pouvez-vous utiliser certains blocs d'un programme pour ajouter, modifier ou réinitialiser les données d'une liste ? Expérience ! Comment allez-vous utiliser les variables et les listes pour créer votre propre programme personnalisé ?



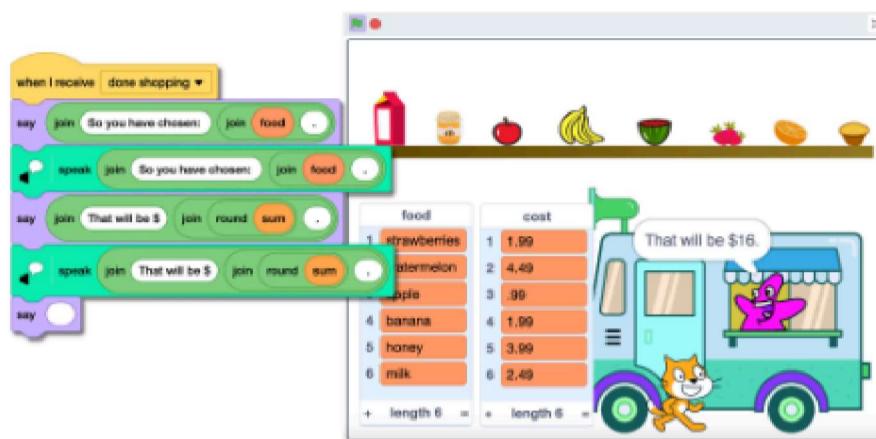
Exemple de projet où le programme sélectionne un élément aléatoire dans une liste d'idées (<https://scratch.mit.edu/projects/962298830>).



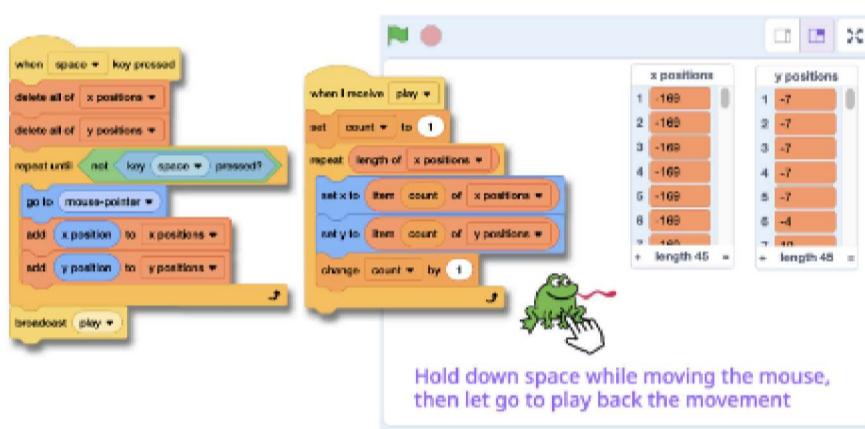
Créé par la Scratch Foundation (scratchfoundation.org). Partagé sous la licence publique internationale Creative Commons Attribution-ShareAlike 4.0 (CCbySA 4.0).



Exemple de projet où une liste prédéfinie détermine l'animation (<https://scratch.mit.edu/projects/962308051>).



Exemple de projet où le programme effectue des calculs sur des nombres dans une liste (<https://scratch.mit.edu/projects/286059732>).



Exemple de projet où le programme enregistre les positions d'une souris (<https://scratch.mit.edu/projects/962767941>).

Consultez nos cartes de codage associées : Cartes de codage de variables et de listes

Regardez nos vidéos de ressources complémentaires ici pour en savoir plus :

Que sont les variables et les listes dans Scratch ? (Partie 1) | [Tutoriel](#)

Utiliser des variables et des listes dans Scratch (Partie 2) | [Tutoriel](#)



Créé par la Scratch Foundation (scratchfoundation.org). Partagé sous la licence publique internationale Creative Commons Attribution-ShareAlike 4.0 (CCbySA 4.0).

Astuce : Si vous souhaitez traduire ce guide, cliquez ici pour faire une copie de ce document Google.



Créé par la Scratch Foundation (scratchfoundation.org). Partagé sous le
Licence publique internationale Creative Commons Attribution-ShareAlike 4.0 (CCbySA 4.0).