

Frédéric Pain

Illustrations de Morgan Thomas

Cahier d'activités

SCRATCH

pour les

KIDS

Dès 8 ans



Apprends
à programmer
en t'amusant !

EYROLLES

Cahier d'activités SCRATCH pour les KIDS

Pars à l'aventure avec Scratch !

Grâce à ce cahier d'activités, apprends à programmer avec Scratch en visitant le parc américain de Yellowstone et ses environs. Suis Tom et Lila sur les traces des animaux sauvages, observe les geysers du parc, participe à la course du grand pow-wow annuel des Indiens Crows et découvre les terribles épreuves des pionniers en route vers l'Ouest. Découpé en cinq étapes de difficulté croissante, ce livre détaille pas à pas la réalisation des programmes et des animations, en s'appuyant sur les ressources numériques disponibles sur Internet. À conseiller pour tous, à partir de 8 ans.

Enseignant chercheur en physique à l'université Paris-Sud à Orsay, **Frédéric Pain** s'intéresse depuis plusieurs années à l'apprentissage de la programmation informatique pour les plus jeunes. Dans ce cadre, il anime des ateliers autour de Scratch et d'Arduino dans des classes de primaire et de collège.

À qui s'adresse ce livre ?

Aux enfants (dès 8 ans), parents et enseignants !

Sur www.editions-eyrolles.com/go/cahierscratch

Télécharge les ressources du livre (fichiers sources, jeux, bonus...).



Dans la même collection

www.editions-eyrolles.com

Studio Eyrolles © Éditions Eyrolles

Code éditeur : G14218
ISBN : 978-2-212-14218-1

Cahier d'activités

SCRATCH

pour les **KiDS**

Groupe Eyrolles
61, bd Saint-Germain
75240 Paris Cedex 05
www.editions-eyrolles.com

En application de la loi du 11 mars 1957, il est interdit de reproduire intégralement ou partiellement le présent ouvrage, sur quelque support que ce soit, sans autorisation de l'éditeur ou du Centre français d'exploitation du droit de copie, 20, rue des Grands-Augustins, 75006 Paris.

© Morgan Thomas pour les illustrations de l'ouvrage

© Frédéric Pain pour les photos de l'ouvrage, sauf p. 49 haut et centre gauche © Shutterstock

© Groupe Eyrolles, 2015

ISBN : 978-2-212-14218-1

Frédéric Pain

Cahier d'activités

SCRATCH pour les **KiDS**

Avec les illustrations de Morgan Thomas



Quelques mots avant ton départ en voyage !

Dans la vie de tous les jours, les programmes informatiques sont partout : dans les ordinateurs bien sûr, mais aussi dans les téléphones, les distributeurs de café, les voitures et même les ascenseurs. La plupart du temps, on les utilise sans s'en rendre compte, mais dans un avenir proche, chacun pourra programmer les objets qui l'entourent. Il est donc important de connaître les bases de la programmation en prévision du monde de demain, et Scratch peut t'y aider !

Le langage Scratch est développé depuis 2003 par le laboratoire Lifelong Kindergarten (qu'on peut traduire par « À la maternelle toute la vie ! ») du MIT (*Massachusetts Institute of Technology*) dans la ville de Boston, aux États-Unis. Il a pour but d'apprendre à programmer à tous, et aux enfants en particulier, en créant des animations et des jeux. Chacun progresse dans son apprentissage à partir de ses essais et de ses erreurs, mais aussi en discutant avec les autres Scratcheurs.

Scratch est un logiciel gratuit qui ne nécessite pas d'installation sur ton ordinateur : tu y accèdes directement par Internet à l'adresse <http://scratch.mit.edu>. Il est très simple à utiliser : tu peux écrire ton premier programme et voir comment il fonctionne en quelques minutes. Pour cela, il te faut assembler des blocs de couleur dans le bon ordre, mais tu vas voir, c'est facile ! Si tu te crées un compte sur le site de Scratch (il te faut juste un nom d'utilisateur et un mot de passe), tu peux stocker tes projets en ligne et les partager avec d'autres enfants à travers le monde. Il y a déjà plus de 8 millions de projets partagés qui t'aideront et te donneront certainement des idées pour créer tes propres jeux et animations. Mais tu peux aussi utiliser Scratch sans te créer un compte.

Avec ce livre sur Scratch, tu vas partir en voyage avec Lila et son frère Tom dans l'Ouest américain pour découvrir le parc naturel de Yellowstone, l'un des plus beaux des États-Unis. Au cours des 5 étapes de ce séjour, tu vas programmer des jeux très différents, où tu devras reconnaître des traces d'animaux, calculer la prochaine éruption d'un geyser, participer à une course de chevaux... À chaque étape, tu découvriras de nouveaux outils de programmation et tu gagneras un badge !

Pour t'aider dans la découverte de Scratch, télécharge les ressources du livre qui sont à l'adresse www.editions-eyrolles.com/go/cahierscratch. Il y a 7 dossiers :

- les 5 dossiers Etape1 à Etape5 qui contiennent les fichiers nécessaires pour programmer les jeux de chaque étape ;
- le dossier Jeux_complets qui contient les solutions des jeux à programmer ;
- le dossier Bonus qui contient des exercices supplémentaires avec leurs solutions.

Avant de partir, vérifie que tu n'as rien oublié. Pour ce voyage avec Scratch, il te faut :

- de la curiosité, de l'imagination et de la bonne humeur ;
- un goûter pour faire des pauses ;
- des copains, des copines, des frères, des sœurs ou qui tu veux, pour écrire les programmes avec toi et jouer ;
- un adulte pour t'aider si besoin... et surtout pour admirer tes réalisations ;
- un ordinateur avec Internet si possible (mais pas obligatoire !).



Comment jouer sans accès à Internet ?

Si tu n'as pas accès à Internet, demande à un adulte d'installer Scratch sur ton ordinateur en suivant les instructions fournies à l'adresse <http://scratch.mit.edu/scratch2download/>.
N'oublie pas aussi de télécharger les ressources du livre !

Allez, assez parlé, il est temps maintenant de se mettre en route pour le grand départ, avec Tom, Lila... et Scratch !

Sommaire



Étape 1

Décollage immédiat ! p. 8



Étape 2

Sur les traces
des animaux sauvages p. 18



Étape 3

Le secret du geyser p. 28



Étape 4

La course de chevaux
de Crow Agency p. 40



Étape 5

Sur la piste de l'Oregon p. 50



SCRATCH'AIR

S1

SA714



Étape 1

Décollage immédiat !

Réveil très tôt ce matin : Tom et Lila partent pour l'aéroport, décollage prévu à 9 h 37, vol SA714 pour les États-Unis, destination finale le parc de Yellowstone. Tom emporte des livres sur les animaux et les plantes, Lila, un appareil photo.

Découvre l'univers de Scratch, avec ses lutins et ses costumes, et apprends à les créer et à les animer. Change de décor et organise l'animation en plusieurs séquences.



Attention, lis bien ceci d'abord !

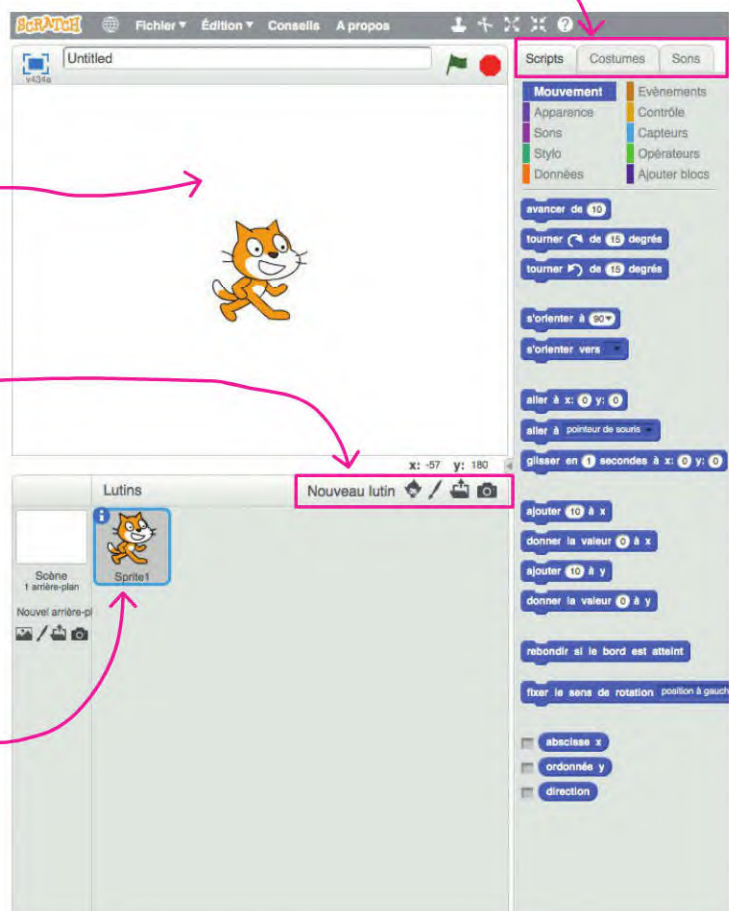
Commence par te familiariser avec l'univers de Scratch : va sur son site Internet (<http://scratch.mit.edu>), ou lance-le directement si tu l'as installé sur ton ordinateur (voir page 6). En cliquant en haut sur Créer, tu obtiens la fenêtre ci-dessous. Repère bien les différentes zones : la scène, la liste des lutins et la zone des scripts avec ses trois onglets (scripts, costumes, sons).

La zone des scripts. Ces trois onglets décrivent les scripts, costumes et sons du lutin sélectionné.

La scène où le jeu se déroule.

Les 4 manières de créer un lutin : à partir de la bibliothèque, par l'outil de dessin, à partir d'un fichier, ou en prenant une photo avec une webcam.

La liste de tous les lutins du jeu. Pour sélectionner un lutin, clique dessus. Si tu cliques alors sur le **i**, les propriétés du lutin apparaissent dans une fenêtre.



Tout en haut de l'écran, tu vois le menu général de Scratch comme ci-dessous.



En cliquant tout à gauche sur le logo Scratch, tu retournes sur la page d'accueil où tu peux voir les projets Scratch partagés. Le menu Fichier te permet de créer un

nouveau projet, de sauvegarder ton projet sur ton ordinateur (Télécharger dans votre ordinateur), ou d'ouvrir un projet que tu as déjà enregistré (Importer depuis votre ordinateur). Si tu t'es connecté à ton compte Scratch, tu as aussi la possibilité d'enregistrer ton projet sur Internet (Sauvegarder maintenant) ou retrouver tes projets (Aller à mes projets).

Les icônes à droite dans le menu général sont très pratiques. Le tampon  permet de copier un objet ou un script. Les ciseaux  te servent à supprimer un objet. Clique dessus, puis sur le chat qui est dans la scène : il disparaît et la liste des lutins est maintenant vide. Enfin, avec les deux icônes , tu peux agrandir ou rétrécir un objet.



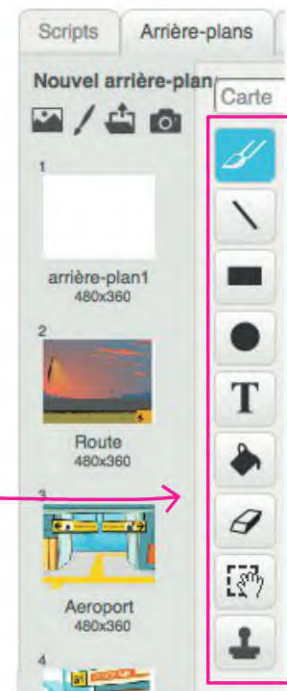
Petite astuce

Si tu ne sais pas à quoi sert une icône, passe ta souris dessus et alors tadam ! Une petite bulle apparaît pour te l'expliquer.

Allez, te voici prêt maintenant pour le grand départ ! Pour ta première animation avec Scratch, tu vas créer 4 séquences : d'abord, Tom et Lila prennent le bus pour l'aéroport, puis ils se déplacent dans l'aérogare. Ensuite, ils enregistrent leurs bagages, et enfin, ils prennent l'avion jusqu'à Denver.

Installe les décors

Dans la liste des lutins, vérifie que la scène est sélectionnée. Clique sur l'onglet Arrière-plans dans la zone des scripts, puis sur l'icône . Va alors chercher le dossier des ressources de l'Étape 1 (voir page 6) et ouvre, une à une et dans cet ordre, les images Route.png, Aeroport.png, Enregistrement.png et Carte.png. Chacune de ces images va être le décor d'une séquence de ton animation. Tu vois alors la fenêtre ci-contre.



Ces outils te permettent de modifier l'image sélectionnée.

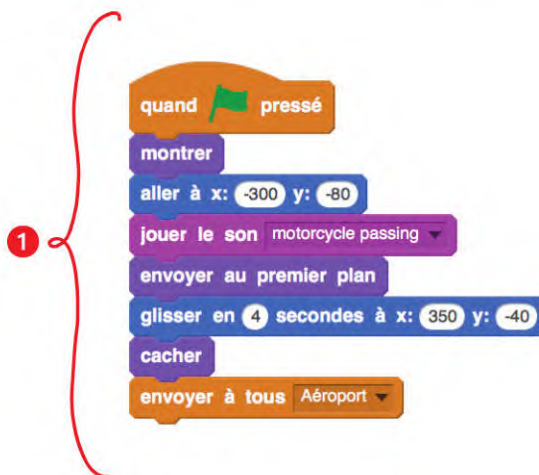
Crée ton premier lutin et ton premier script

Dans la première séquence, le bus de Tom et Lila roule vers l'aéroport. À côté de la liste des lutins, tu as l'icône  que tu as déjà vue. Clique dessus et ouvre le fichier Bus.png des ressources de l'Étape 1 : tu as créé ton premier lutin ! Sur la scène, tu le vois maintenant qui vole sur l'arrière-plan Route. À présent, tu vas écrire le script de ce lutin, pour lui dire ce qu'il doit faire.



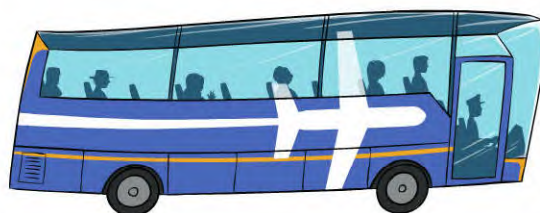
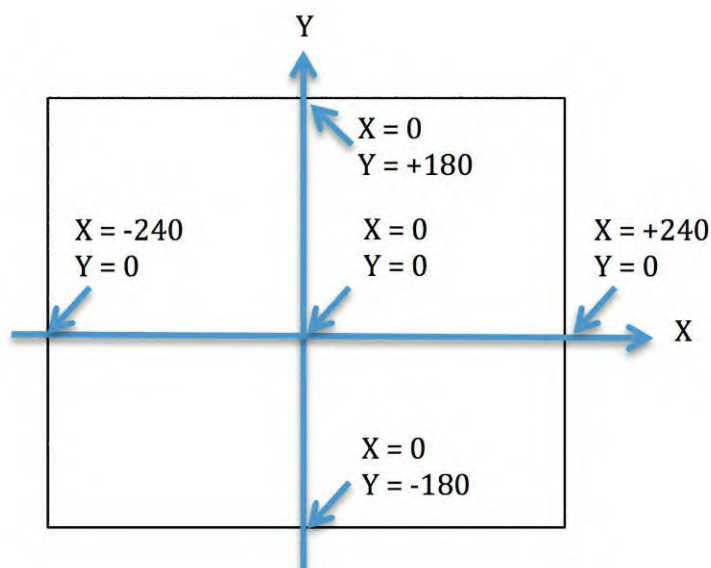
Rends-toi dans l'onglet Scripts. Il contient tous les blocs d'instructions possibles, rangés dans des palettes de différentes couleurs, auxquelles tu accèdes par un clic sur leur nom : le bleu foncé pour les instructions de mouvements, le marron caramel pour les blocs d'évènements, le violet clair pour le son... Pour écrire un script, appelé aussi programme, rien de plus simple : il suffit de sélectionner certains blocs, en cliquant dessus avec la souris, et de les déposer (en maintenant le bouton de la souris enfoncé) dans la zone grise à droite. Par exemple, grâce au bloc , le script démarrera lorsqu'on cliquera sur le drapeau vert en haut à droite de la scène. En emboîtant les blocs les uns dans les autres, on forme un programme complet.

À toi de jouer maintenant ! Crée le programme **1** en assemblant les blocs à partir des palettes Évènements, Apparence, Mouvement et Sons. Ne t'inquiète pas si les blocs n'ont pas exactement les mêmes valeurs, on va les régler ensemble et voir à quoi ils servent.



Comment se repérer et se déplacer ?

Attention, concentration ! Au centre de la scène, les coordonnées sont $x = 0$, $y = 0$. La largeur totale de la scène est de 480. Complètement à gauche, $x = -240$ et complètement à droite, $x = +240$. La hauteur totale de la scène est 360. Complètement en bas, $y = -180$ et complètement en haut, $y = +180$.



Pour déplacer le bus de gauche à droite, on fait passer x de -300 au début à +350 à la fin. On déborde un peu car il faut tenir compte de la taille du bus. Comme la route est un peu de travers, on fait « monter » le bus de $y = -80$ à $y = -40$. On utilise les blocs **aller à x: -300 y: -80** et **glisser en 4 secondes à x: 350 y: -40**. Tu peux changer les valeurs des coordonnées x et y , en cliquant dans la zone blanche du bloc et en tapant les valeurs au clavier. Essaie avec différentes valeurs pour x et y , puis clique sur le drapeau vert en haut à droite de la scène pour voir le mouvement du bus.

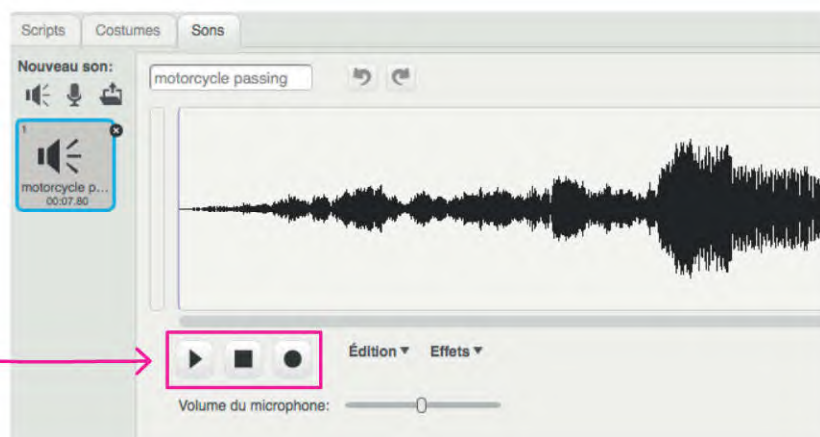
Contrôle le son et les images

Pour le bloc **jouer le son**, choisis le son que tu veux en cliquant sur l'onglet Sons de la zone des scripts. Tu peux choisir un son à partir de la bibliothèque , l'enregistrer à partir du micro de ton ordinateur , ou ouvrir n'importe quel fichier de son . Ici, choisis le son « motorcycle passing » dans la bibliothèque. Écoute-le !

Une fois le son téléchargé depuis la bibliothèque, choisis-le dans le menu déroulant du bloc **jouer le son** en cliquant sur la petite flèche noire.



Avec ces boutons, tu peux lire, arrêter ou enregistrer un son.

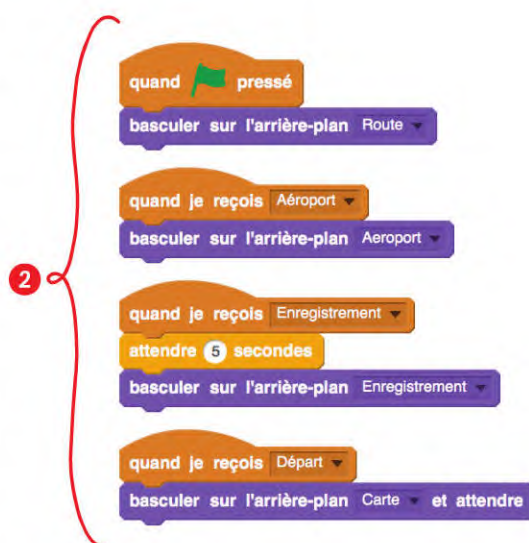


Les blocs **montrer** et **cacher** permettent de montrer ou cacher les lutins : au début du programme, on positionne le bus sur la scène, on le montre, puis on le cache une fois arrivé à destination.

Pour contrôler le déroulement des différents événements du programme, on utilise des messages. Ici, une fois que le bus a traversé la scène, il envoie à tous les lutins le message **Aéroport** qui lance automatiquement la séquence suivante. Pour créer ce message, clique sur la flèche noire à droite du bloc **envoyer à tous**, choisis Nouveau message, écris « Aéroport » et clique sur OK.

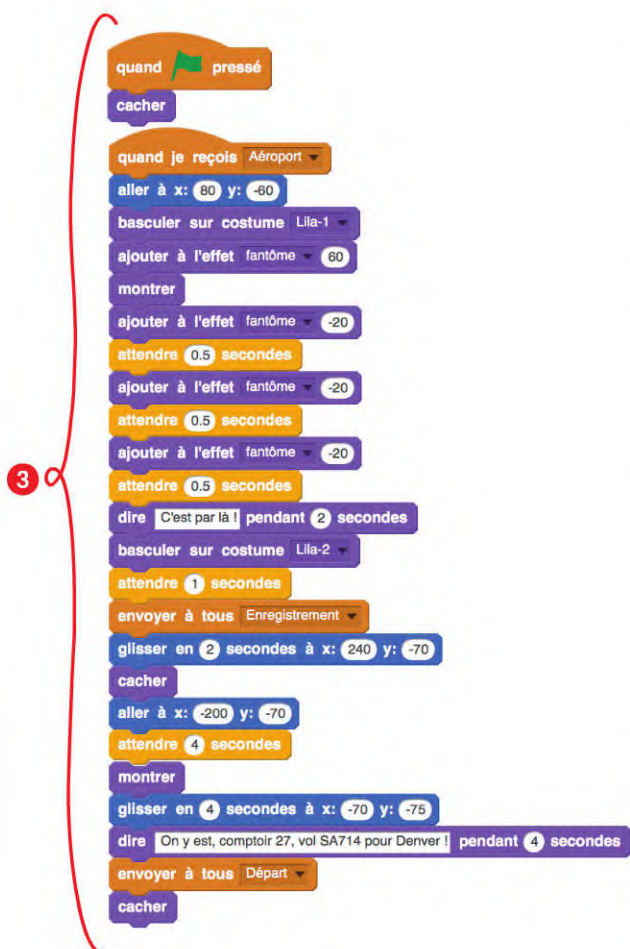
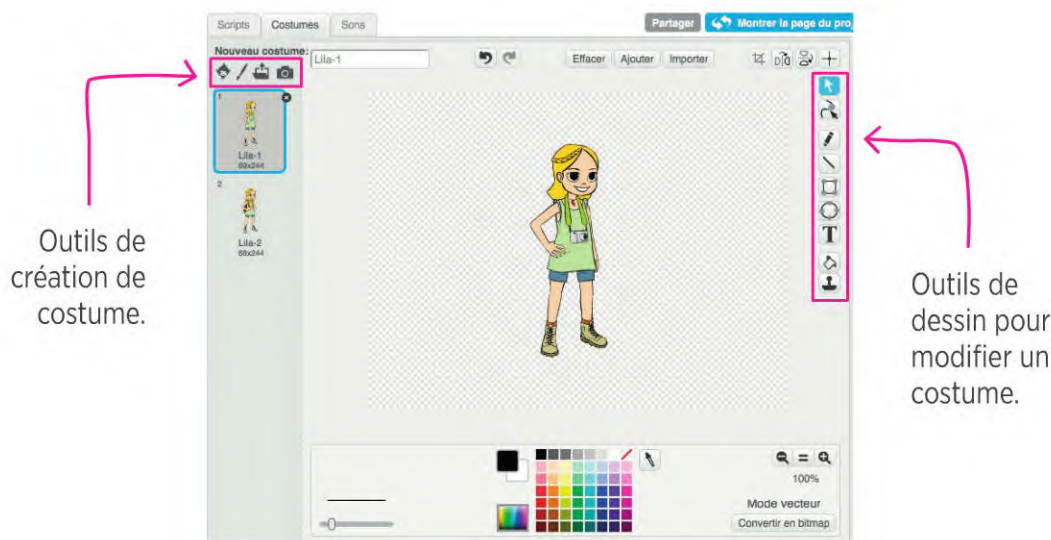
Changement de décor

Tu vas maintenant définir le changement de décor des 4 séquences. Clique sur le lutin Scène. Dans l'onglet Scripts, glisse les blocs des palettes Événements, Apparence et Contrôle pour écrire le programme 2. Comme pour le message **Aéroport**, crée cette fois les messages **Route**, **Enregistrement** et **Départ**. Chaque message reçu déclenche un changement d'arrière-plan. Ainsi, quand tu lances le programme avec le drapeau vert, tu bascules sur l'image de la route. Et quand le bus a fini de traverser l'écran et que le message **Aéroport** a été envoyé, c'est l'image de l'aéroport qui s'affiche.



Lila entre en scène...

À partir du menu Nouveau lutin de la liste des lutins, ouvre le fichier Lila.sb2 des ressources de l'Étape 1 et clique sur l'onglet Costumes : tu vois que Lila a déjà 2 costumes. Dans Scratch, les costumes sont tous les dessins associés à un lutin. Et quand on change rapidement le costume d'un personnage, on peut l'animer !



Tu vas maintenant programmer les mouvements de Lila. Clique sur l'onglet Scripts, puis assemble les blocs des palettes Événements, Mouvement, Apparence et Contrôle pour créer le programme 3.

Au début de l'animation, le lutin Lila est caché (elle est dans le bus). Puis, à la fin de la séquence du bus, quand le message **Aéroport** est reçu, Lila est positionnée sur l'écran (x = 80, y = -60) avec le costume Lila-1 et elle apparaît progressivement grâce aux blocs **mettre l'effet fantôme** et **montrer**. Il y a plein d'autres effets disponibles en cliquant sur la petite flèche noire au bout du bloc **ajouter à l'effet** : teste-les !



Petite astuce

La valeur à droite du bloc **mettre l'effet fantôme** à **0** fixe le degré de transparence du lutin. Si elle vaut 0, le lutin est totalement visible ; si elle vaut 100, il a entièrement disparu !

Le bloc **dire C'est par là ! pendant 2 secondes** fait parler Lila dans une bulle. Lila prend alors son sac à dos (on bascule sur le costume Lila-2) et s'en va vers le hall d'enregistrement grâce au bloc **glisser**. Le message **Enregistrement** envoyé à tous les lutins déclenche la séquence suivante.

Nous voici dans le hall d'enregistrement. Lila apparaît à gauche de la scène (**aller à x: -200 y: -70**, **montrer**) et se déplace jusqu'au comptoir d'enregistrement (**glisser en 4 secondes à x: -70 y: -75**) pour dire « On y est, comptoir 27, vol SA714 pour Denver ! ». Lila envoie alors le message **Départ** et l'arrière-plan bascule sur la séquence finale (programme **4**). Lila est maintenant cachée. Teste à présent ton script en cliquant sur le drapeau vert.

... suivie par Tom !

Au tour de Tom maintenant. Crée le lutin Tom à partir du menu Nouveau lutin en ouvrant le fichier Tom.svg. Les programmes de Tom ressemblent beaucoup à ceux de Lila. Glisse dans la zone des scripts les blocs pour obtenir le programme **4**.



Au début de l'animation, le lutin Tom est caché. On utilise les mêmes blocs que pour Lila pour faire apparaître et parler Tom. On change juste les valeurs de x et y pour que Tom soit à côté de Lila. Le lutin attend le message **Enregistrement** pour se déplacer vers la droite (glisser en 4 secondes à x: 240 y: -80) dans l'aéroport, puis réapparaître en bas à gauche (aller à x: -240 y: -80, montrer) et se déplacer vers la droite dans le hall d'enregistrement (glisser en 3 secondes à x: -145 y: -80). Quand il reçoit le message **Départ**, Tom est caché.

Ça y est, on a décollé ! Le message **Départ** envoyé par Lila fait apparaître la carte du parcours.

Retrouve le quiz et les bonus de cette étape dans le dossier Bonus !



Bravo, tu as gagné ton badge de **niveau 1** !



Découvre ce qu'est
une variable. Apprends
à utiliser les boucles dans un
programme et à le faire démarrer
si une condition est remplie.
Et deviens incollable pour
reconnaître des traces
d'animaux !

Étape 2

Sur les traces des animaux sauvages

Tom et Lila sont arrivés dans le parc de Yellowstone. Aujourd'hui, ils partent pour leur première randonnée. Ils sont très impatients d'observer des animaux dans leur milieu naturel. Tom met Lila au défi de reconnaître leurs traces...



Le jeu

Pour chaque animal, trouve son empreinte.
Si tu cliques sur les bonnes paires en moins
de 10 essais, tu as gagné !



Crée tes premières variables

Alors, prêt à créer ton premier jeu ? Tu vas commencer en douceur avec seulement 2 animaux et leurs empreintes, l'ours et le bison. Mais tu verras à la fin de cette étape comment jouer avec 7 animaux.

Pour écrire le programme de ce jeu, tu vas utiliser des variables. Une variable, c'est une sorte de petite boîte contenant une donnée : un nombre, une lettre, un mot... Elle associe donc un nom (celui de la boîte) à une valeur, qui peut éventuellement varier au cours du temps, comme le score d'une partie de jeu.

Lance Scratch, clique en haut sur Créer, puis sur la palette Données dans la zone des scripts. Clique en dessous sur le bouton Créer une variable, tape « Animal », puis clique sur OK. Fais la même chose pour créer les variables Empreinte, Paire1, Paire2 et Essais. Si la case à gauche du nom de la variable est cochée, la variable est visible sur la scène. Pour l'instant, garde les variables visibles : ce sera très pratique pour vérifier ton programme.



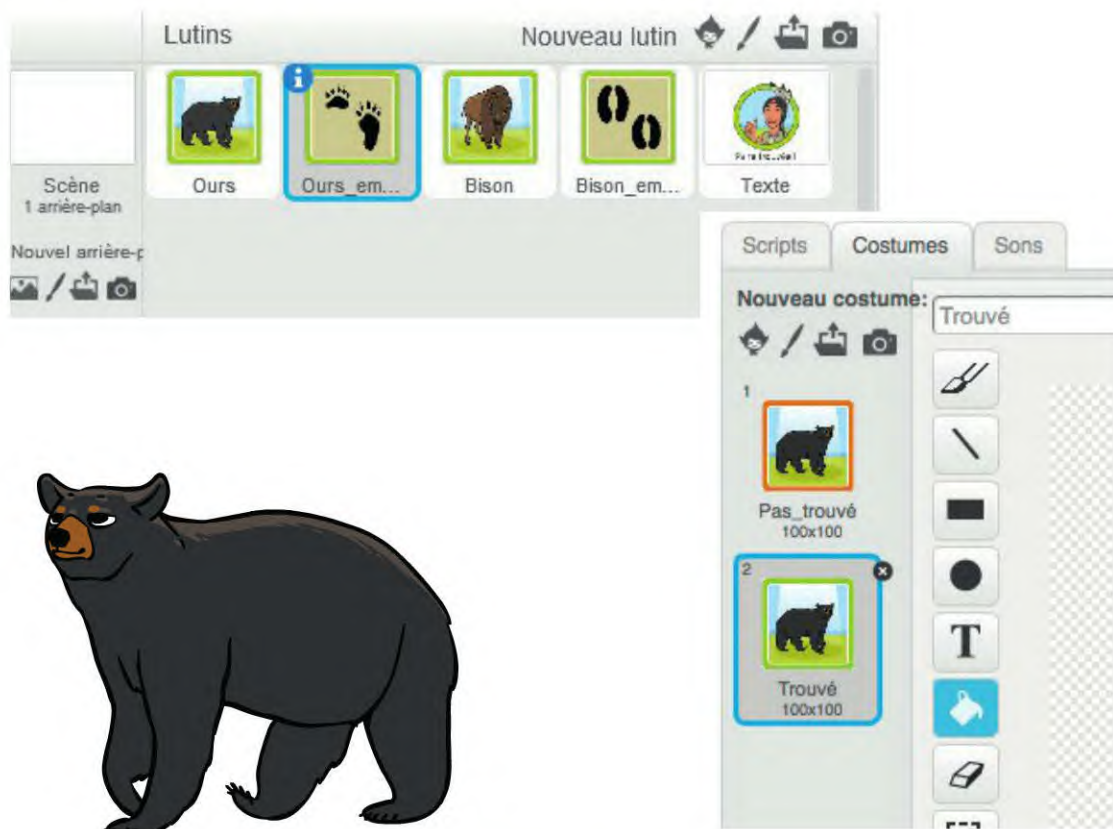
Dans ce jeu, chaque animal et chaque empreinte sont associés à un numéro : 1 pour l'ours et son empreinte, 2 pour le bison et son empreinte. Quand on clique sur une image d'animal, son numéro est stocké dans la variable Animal. De la même manière, quand on clique sur une image d'empreinte, son numéro est stocké dans la variable Empreinte.

Tu vas écrire un programme qui compare les numéros stockés dans les variables Animal et Empreinte. Si ce sont les mêmes, une paire a été trouvée ; sinon, les variables sont remises à 0, et le joueur peut réessayer tant qu'il lui reste des essais.

La paire ours

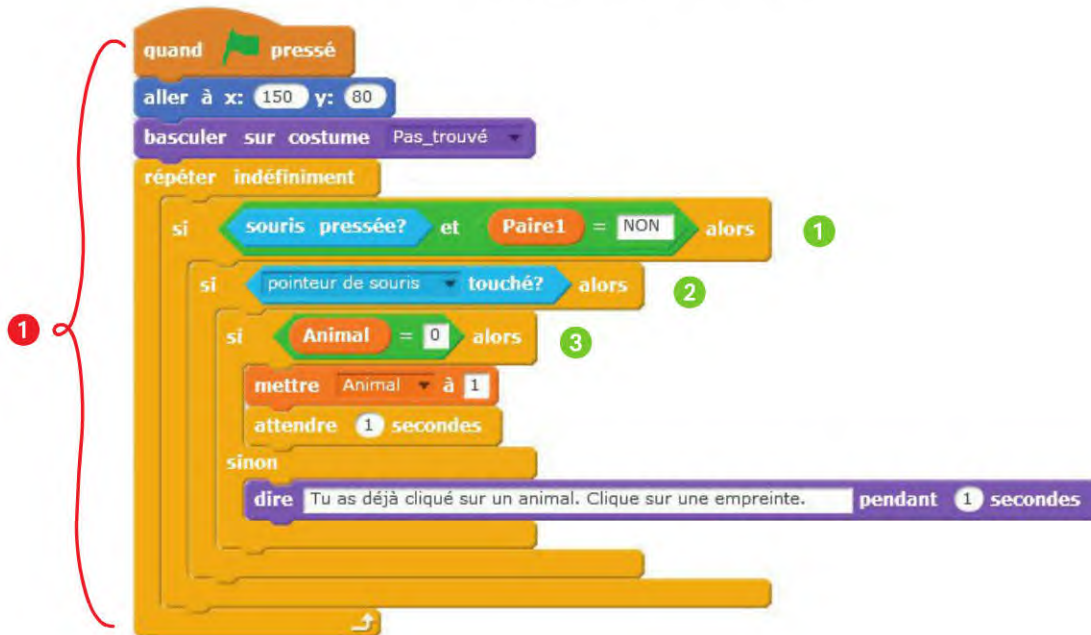
Dans le menu Fichier en haut, clique sur Importer depuis votre ordinateur et ouvre le projet Etape2.sb2 du dossier de l'Étape 2. Ce fichier contient tous les lutins et les costumes du jeu, mais pas les scripts puisque c'est toi qui vas les écrire !

Regarde les lutins Ours et Ours_empreinte, qui vont former la paire ours. En cliquant sur leur onglet Costumes, tu remarques que chacun d'eux possède 2 costumes, Pas_trouvé et Trouvé. Quand la bonne paire aura été trouvée, tu rendras visibles les costumes Trouvé.



Le script du lutin Ours

On va maintenant écrire ensemble le programme du lutin Ours. Clique sur ce lutin, puis sur l'onglet Scripts. Ce programme **1** utilise les blocs des palettes Événements, Mouvement, Apparence et Contrôle, que tu connais déjà, et ceux de trois nouvelles palettes : Données, Capteurs et Opérateurs.



Commence par placer les 3 premiers blocs du programme, en choisissant Pas_trouvé dans le menu déroulant du bloc **basculer sur costume Pas_trouvé**. Ainsi, tu positionnes l'ours sur la scène à x = 150 et y = 80, en l'affichant avec son costume Pas_trouvé.

Puis ajoute la boucle **répéter indéfiniment**, qui est un bloc en forme de C. Grâce à cette boucle, le groupe de blocs à l'intérieur du C va se répéter jusqu'à la fin du jeu. Sur le programme, tu vois que dans le C, il y a 2 blocs **si ... alors** et 1 bloc **si ... alors ... sinon** qui sont imbriqués les uns dans les autres. Chacun de ces blocs en forme de C va vérifier si la condition écrite dans la zone en forme de losange est vraie. C'est seulement dans ce cas que les instructions à l'intérieur du C seront effectuées.

Dans ce script, tu testes 3 conditions l'une après l'autre. Si on clique avec la souris et que la paire du lutin Ours, appelée Paire1, n'a pas encore été trouvée (**1**), alors tu testes si le pointeur de la souris touche le lutin Ours (**2**). Si cette condition est vraie, alors tu vérifies que la variable Animal est bien égale à 0 (**3**), ce qui veut dire qu'on n'a pas encore cliqué sur un animal.

Si cette dernière condition est vraie, alors tu donnes la valeur 1 à la variable Animal (souviens-toi, c'est le numéro de la paire ours). Sinon, si elle est fausse, cela veut dire qu'on avait déjà cliqué avant sur un animal, et qu'on a cliqué à nouveau sur un animal au lieu de cliquer sur une empreinte. Dans ce cas, tu exécutes le bloc

dire Tu as déjà cliqué sur un animal. Clique sur une empreinte. **pendant 1 secondes**

Crée ton premier test « si... alors »

Pour créer la première condition, fais glisser dans la zone des scripts un bloc **si** **alors**, puis les blocs **et** et **=** de la palette Opérateurs, et enfin les blocs **souris pressée?** de la palette Capteurs et **Paire1** de la palette Données.

Pour remplir le rectangle de gauche du bloc **=**, approche le bloc **Paire1** : le tour du rectangle devient blanc. Relâche alors la souris : c'est magique, le bloc s'est inséré ! Pour le rectangle de droite, écris dedans « NON ».

En utilisant la même méthode, insère ensuite les blocs **Paire1 = NON** et **souris pressée?** dans le bloc **et**. Puis ajoute ce dernier dans le bloc **si** **alors**. Bravo, tu as créé ta première boucle !

Assemble ensuite les autres blocs du programme 1. Vérifie bien qu'il est comme celui de la page 22. Puis lance-le en cliquant sur le drapeau vert. Si tu cliques sur l'image de l'ours, alors la variable Animal devient égale à 1. Si tu recliques sur l'ours, alors le message « Tu as déjà cliqué sur un animal. Clique sur une empreinte. » apparaît. Ça fonctionne !

Le script du lutin Ours_empreinte

Tu vas maintenant écrire le script du lutin Ours_empreinte. C'est presque le même que celui du lutin Ours.



Petite astuce

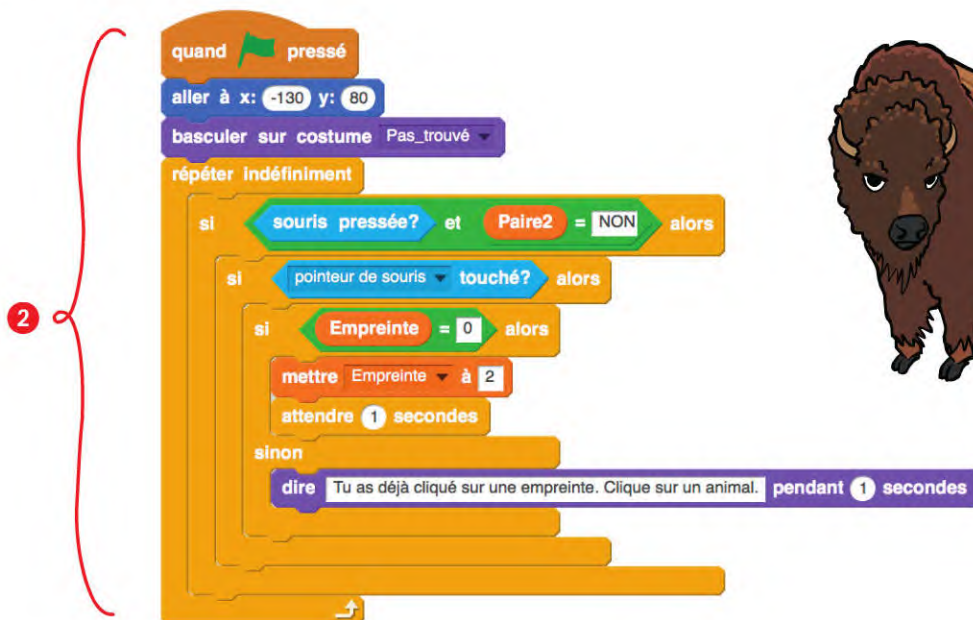
Pour éviter de recopier tous les blocs, clique sur le bloc **quand pressé** du lutin Ours et fais glisser tout le script sur le lutin Ours_empreinte dans la liste des lutins. Relâche alors la souris. Si tu cliques maintenant sur l'onglet Scripts du lutin Ours_empreinte, tu verras que tout le script a été copié ! Génial, non ?

Une fois le script copié, tu dois remplacer la variable Animal par Empreinte et changer le message par « Tu as déjà cliqué sur une empreinte. Clique sur un animal. ». Enfin, dans le bloc **aller à**, modifie les valeurs de x et y pour que les nouveaux lutins apparaissent sur 2 colonnes : une à droite pour les empreintes, une à gauche pour les animaux.

La paire bison

Maintenant que tu t'es occupé de la paire ours, fais pareil pour la paire bison, en écrivant les 2 scripts des lutins Bison et Bison_empreinte. C'est presque comme pour la paire ours, sauf qu'il faut remplacer Paire1 par Paire2, et donner la valeur 2 aux variables Animal et Empreinte lorsqu'on clique sur le bison ou son empreinte. Modifie aussi les valeurs de x et y du bloc **aller à**.

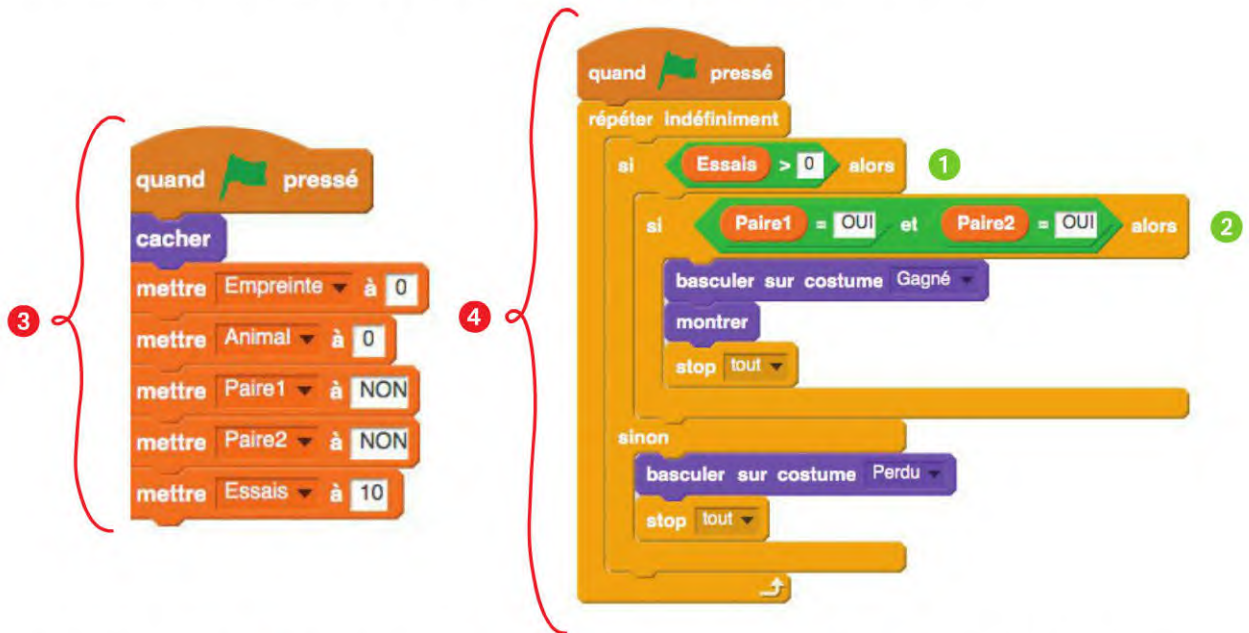
Quand tu as fini, le script du lutin Bison_empreinte doit être identique au programme 2. Vérifie bien les valeurs des variables !



OK, mais quand est-ce qu'on gagne ?!

Dans la liste des lutins, clique sur le lutin Texte, puis sur l'onglet Costumes. Tu vois qu'il a déjà 4 costumes : Bravo, Erreur, Gagné et Perdu. Ils servent à indiquer si la paire sélectionnée est correcte ou non, et si tu as gagné ou perdu le jeu. Si tu veux, tu peux les modifier avec les outils de dessin, situés dans la colonne d'outils à droite de la zone des costumes.

Commence par écrire les 2 programmes 3 et 4 pour ce lutinTexte.



Le programme 3 n'est effectué qu'une fois au début du jeu puisqu'il n'a pas de boucle. Il permet de masquer le lutin Texte et de donner des valeurs aux variables au début du jeu. Comme on n'a pas encore cliqué sur un animal ou une empreinte, les variables Animal et Empreinte valent 0. La valeur NON pour les variables Paire1 et Paire2 indique qu'aucune des bonnes paires n'a été trouvée. Enfin, le joueur a droit à 10 essais pour trouver toutes les paires.

Le programme 4 utilise le bloc **répéter indéfiniment**, 1 bloc **si ... alors** et 1 bloc **si ... alors sinon**. Le bloc 1 permet de vérifier s'il reste des essais au joueur. S'il en reste, le bloc 2 contrôle si toutes les paires ont été trouvées ou pas. Si oui, il affiche le costume Gagné et stoppe tous les scripts. S'il ne reste plus d'essais, alors il affiche le costume Perdu et stoppe tous les scripts.

Le script du lutin Texte

Écris le grand programme 5 de la page 26, qui est plus simple qu'il en a l'air.

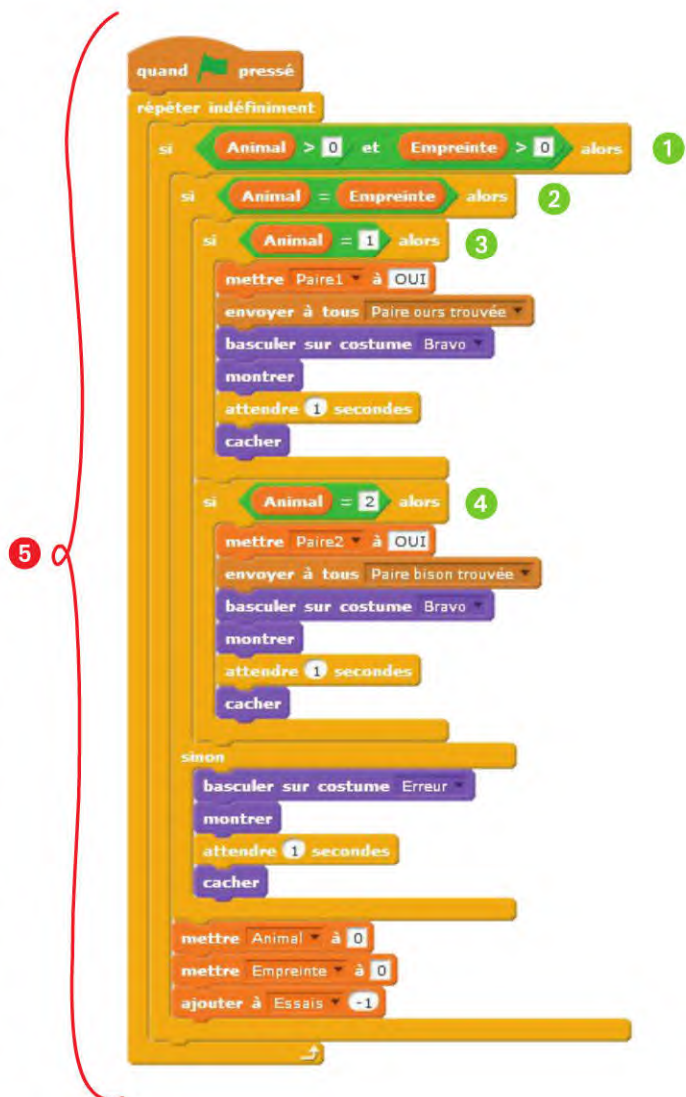
Ce script commence par vérifier pendant tout le jeu si le joueur a bien cliqué sur un animal et sur une empreinte (1), donc si les variables Animal et Empreinte ont toutes deux des valeurs différentes de 0. Si c'est le cas, le script contrôle alors si ces variables ont la même valeur (2), donc si une bonne paire a été trouvée.

Si une paire a été trouvée, les 2 blocs **si ... alors** (3 et 4) vérifient laquelle. Si la variable Animal vaut 1, alors c'est la paire ours ; si elle vaut 2, alors c'est

la paire bison. Dans chaque cas, le programme donne la valeur OUI à la variable Paire correspondante, envoie un message à tous les lutins indiquant que la paire a été trouvée et affiche le costume Bravo. Tu te souviens (voir page 14) que pour créer un message, il faut cliquer sur la petite flèche noire à droite du bloc **envoyer à tous** Paire ours trouvée et choisir Nouveau message.

Sinon, si les variables Animal et Empreinte ont des valeurs différentes, cela veut dire que l'empreinte cliquée n'est pas celle de l'animal cliqué, alors le costume Erreur est affiché pendant 1 seconde.

Enfin, le programme **5** remet à 0 les variables Animal et Empreinte, et donne au joueur un essai de moins avec le bloc **ajouter à Essais** -1. On peut alors cliquer à nouveau sur un animal et une empreinte pour tenter sa chance !



Un dernier effort !

Il ne te reste plus qu'à écrire les 4 programmes permettant d'afficher le costume Trouvé pour les lutins d'animaux et d'empreintes. Ajoute le programme **6** dans les scripts des lutins Ours et Ours_empreinte, et le programme **7** pour les lutins Bison et Bison_empreinte.



Tu peux maintenant tester ton jeu. Si tout fonctionne, cache les variables en décochant leur case dans la palette Données.

Le dossier de l'Étape 2 contient les fichiers pour créer les lutins pour 5 nouvelles paires. Maintenant que tu as écrit le programme pour 2 paires, ça ne devrait pas être trop difficile de le modifier pour en ajouter des nouvelles ! Si tu es coincé, la solution complète du jeu avec 7 paires est aussi dans ce dossier.



J'ai fait plein de photos de ma première journée. Regarde ce que j'ai vu dans le parc !

Le carnet de Lila



Antilopes

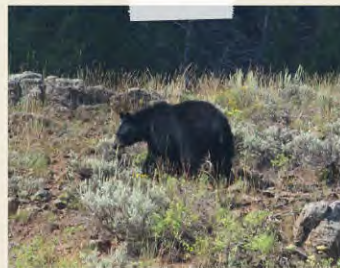


Empreinte de loup

Le parc de Yellowstone a été créé en 1872 : c'est le plus ancien des États-Unis, l'un des plus grands (presque 9 000 km²) et certainement l'un des plus beaux. On peut y voir de nombreuses espèces de grands mammifères : bisons, ours noirs, grizzlis, lynx, cougars, wapitis, daims, antilopes, chèvres de montagne, mouflons...



Bison dans la vallée de Hayden



Ours noir en lisière de forêt

Le bison est l'animal symbolique du parc : presque totalement disparu autour de 1900, il est aujourd'hui représenté par plusieurs milliers d'individus répartis en deux troupes. Il est facile à observer dans les prairies des vallées de Lamar et de Hayden.



Bravo, tu as gagné ton badge de **niveau 2** !



Étape 3

Le secret du geyser

Aujourd'hui, Tom et Lila vont aller observer l'éruption du Vieux Fidèle, le plus célèbre geyser de Yellowstone. Ses éruptions sont assez régulières pour qu'on puisse les prévoir à 10 minutes près !

Découvre comment donner une valeur au hasard à une variable. Apprends à enregistrer l'heure ou une valeur tapée au clavier. Familiarise-toi avec les opérateurs mathématiques.



Le jeu

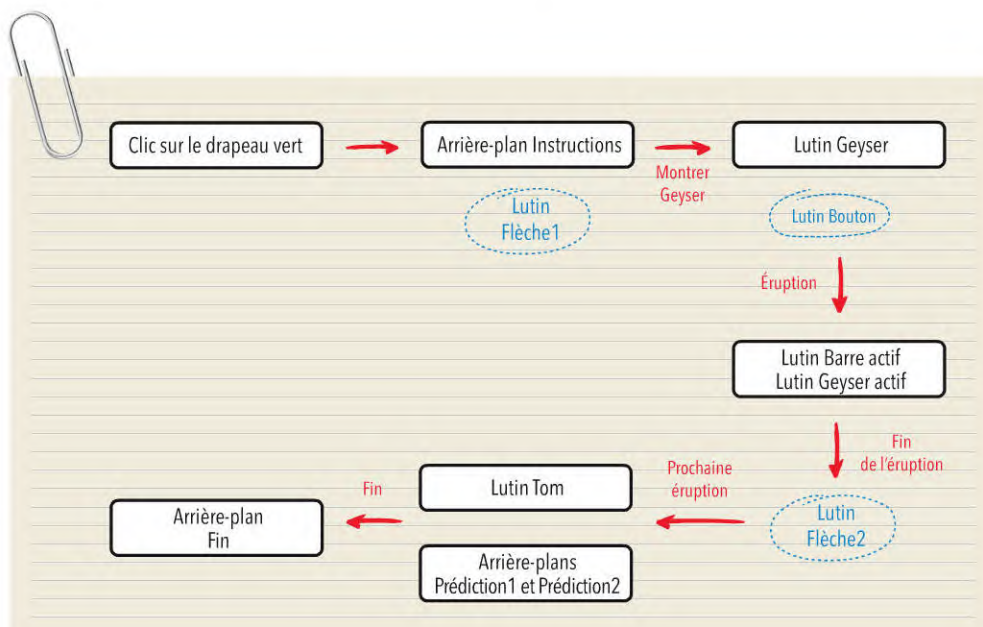
Il y a deux parties. Tu vas d'abord simuler l'éruption d'un geyser, puis résoudre un petit problème pour prédire l'heure de la prochaine éruption !



Demandez le programme !

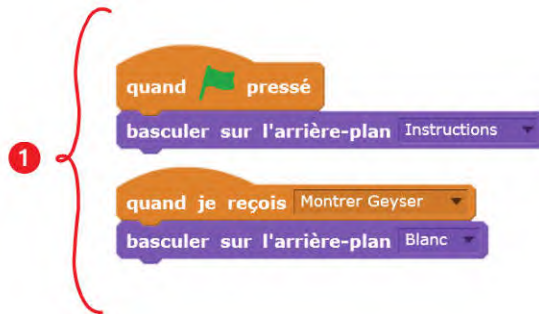
Dans Scratch, clique en haut sur Créer, puis dans le menu Fichier, choisis Importer depuis votre ordinateur et ouvre le fichier Etape3.sb2 du dossier de l'Étape 3. Ce fichier contient déjà tous les lutins dont tu auras besoin.

Le schéma ci-dessous décrit l'ordre des étapes du grand programme que tu vas créer. Quand on clique sur un lutin en bleu, le programme envoie un message à tous les lutins, ce qui les fait apparaître/disparaître et déclenche certaines actions.



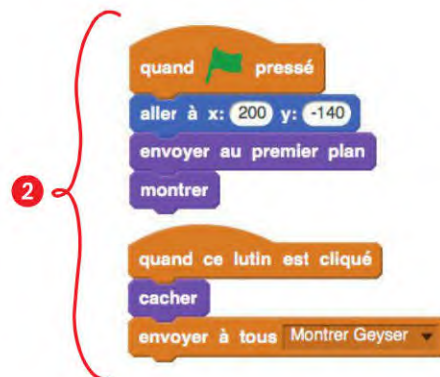
Clique sur le lutin Scène. Il possède 5 arrière-plans (Instructions, Blanc, Prédiction1, Prédiction2 et Fin), qui vont apparaître les uns après les autres. Ajoute le programme 1 pour ce lutin.

Lorsqu'on clique sur le drapeau, l'arrière-plan des instructions apparaît. Quand le lutin reçoit le message **Montrer Geyser**, il bascule sur l'arrière-plan Blanc, qui est juste un fond blanc sur lequel apparaîtra le lutin Geyser.



Ton premier lutin cliquable !

Clique maintenant sur le lutin Flèche1 et ajoute-lui le programme 2. Grâce au bloc **quand ce lutin est cliqué**, un clic sur le lutin Flèche1 envoie le message **Montrer Geyser** à tous les lutins, ce qui permet de passer à l'étape suivante du grand programme.

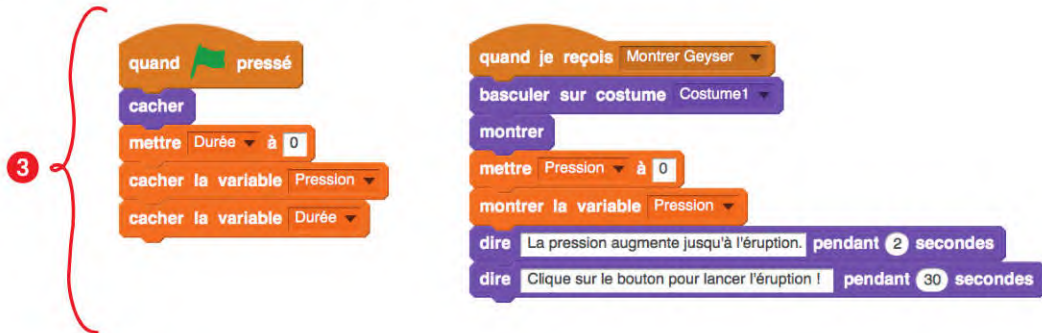


Anime le geyser

Clique sur le lutin Geyser. Au début du jeu, il doit être caché, puis apparaître quand le message **Montrer Geyser** est reçu. Pour cela, tu vas avoir besoin de plusieurs variables.

À partir de la palette Données, crée les variables Pression, Durée, Heure, Minutes, Début et Prochain. Cache toutes ces variables, sauf la variable Pression, en décochant leur petite case. Comme c'est l'augmentation de la pression qui déclenche l'éruption du geyser, on affiche en effet l'indicateur Pression.

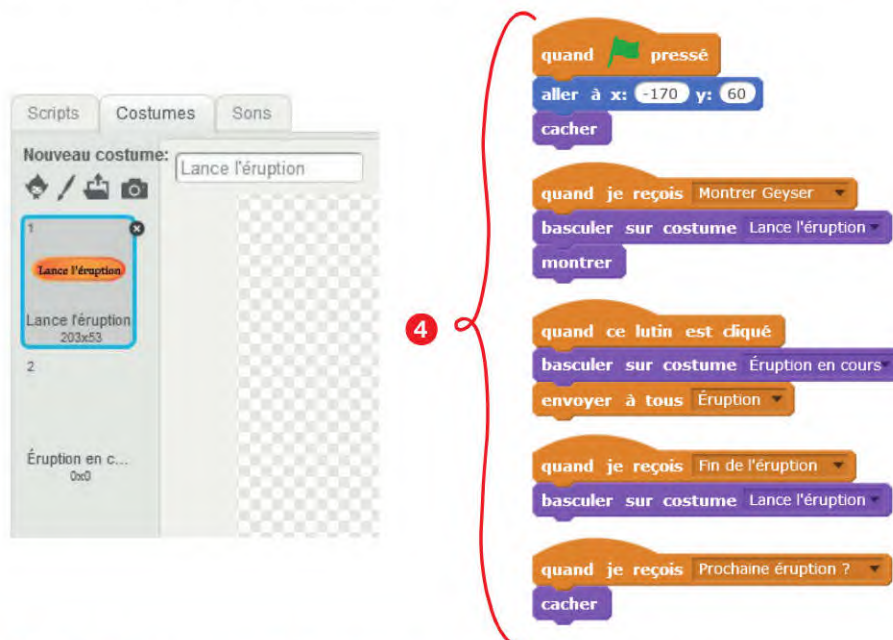
Écris le programme 3 pour le lutin Geyser. Il donne la valeur 0 aux variables Durée et Pression au début du grand programme, montre le geyser au repos et affiche les instructions pour lancer l'éruption.



Le lutin Bouton

Clique sur le lutin Bouton, puis sur l'onglet Costumes. Ce lutin a 2 costumes : « Lance l'éruption » quand le geyser est au repos et « Éruption en cours » pendant l'éruption. Écris le programme 4 qui est très simple ; il montre/cache le lutin et choisit son costume en fonction des messages qu'il reçoit.

Au début, le lutin est caché, puis quand il reçoit le message **Montrer Geyser**, il est montré avec son costume « Lance l'éruption ». Lorsqu'on clique dessus, il bascule sur son costume « Éruption en cours » et envoie à tous le message **Éruption**. Ce message démarre le programme de l'éruption pour le lutin Geyser.



Lorsque le message **Fin de l'éruption** est reçu, le lutin rebascule sur le costume « Lance l'éruption », et il disparaît quand il reçoit le message **Prochaine éruption ?**.

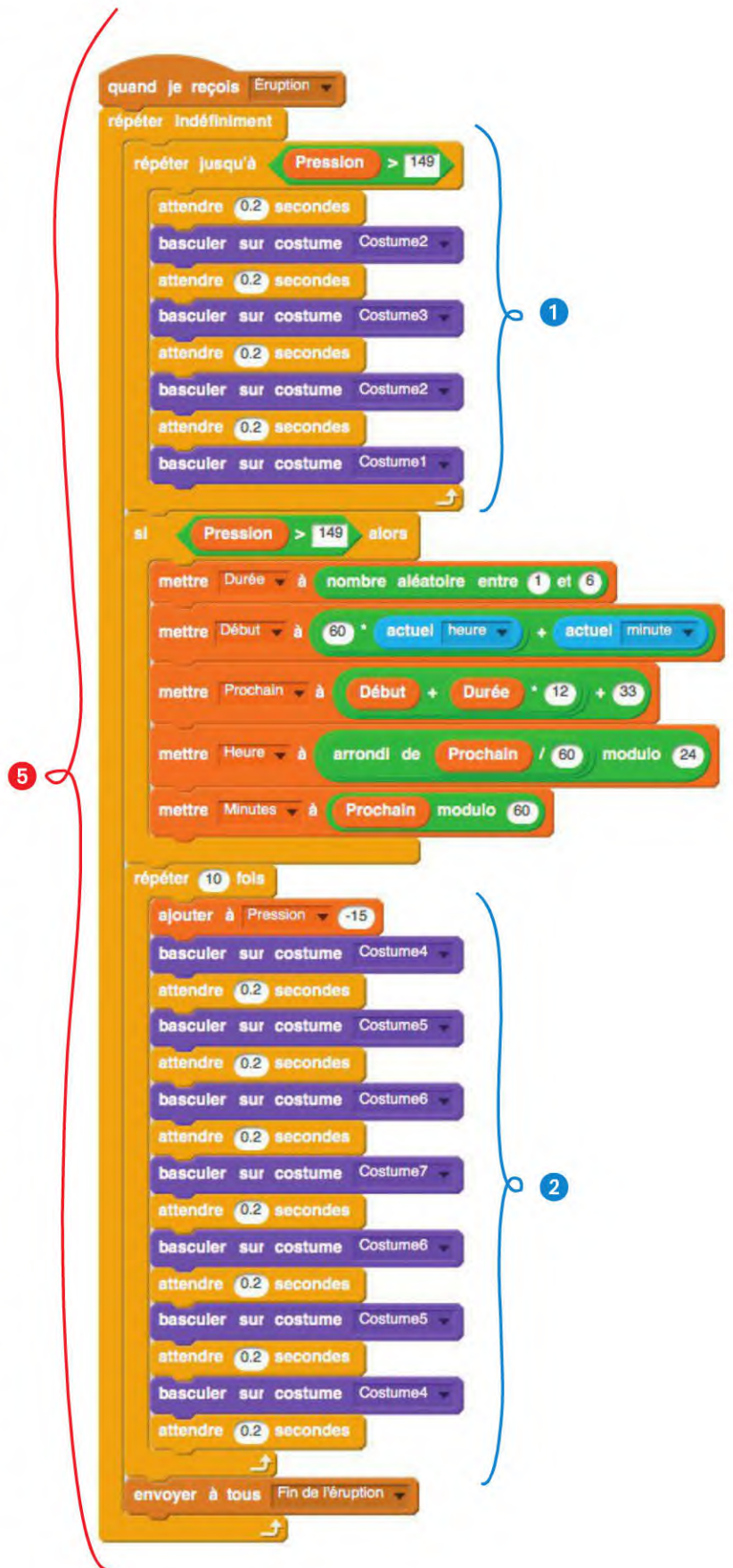
Éruption !!!

Retourne sur le lutin Geyser et écris à présent le long script **5** qui anime l'éruption.

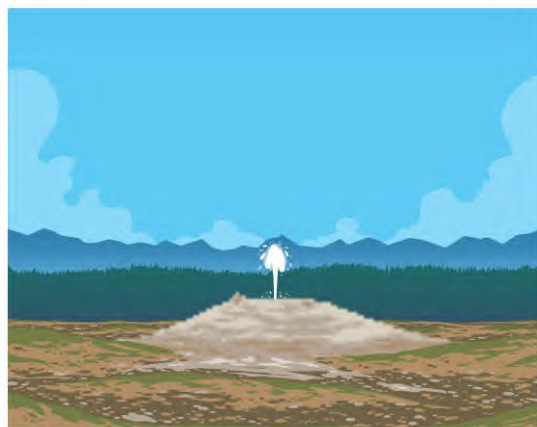
Le message **Éruption** déclenche l'animation de l'éruption. Souviens-toi, il est envoyé quand on clique sur le lutin Bouton.

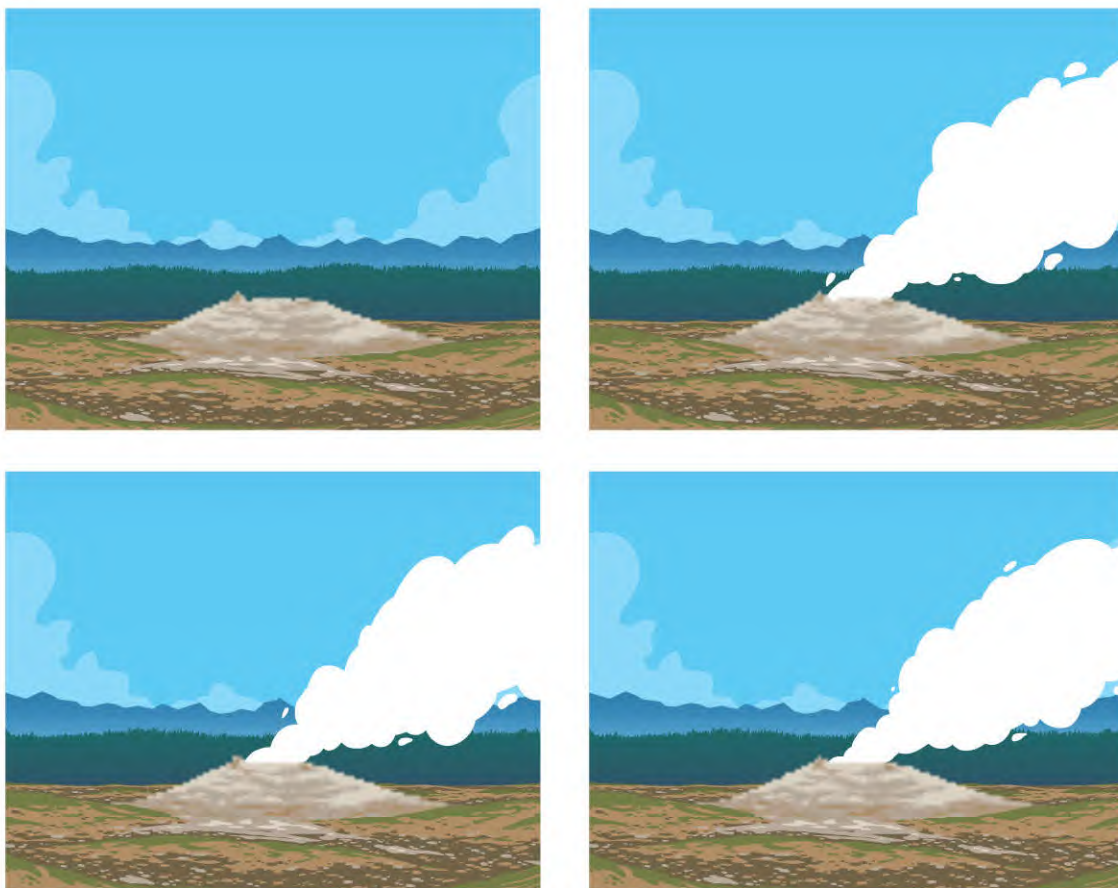
L'animation du lutin Geyser utilise plusieurs blocs de la palette Contrôle. Tant que la pression n'a pas atteint 150, on utilise la boucle **1** **répéter jusqu'à** qui change rapidement les costumes 1 à 3 pour faire fumer le geyser. Quand la pression dépasse 150, l'éruption commence : l'eau et la vapeur jaillissent ! On bascule alors sur les costumes 4 à 7.

L'éjection de ces milliers de litres d'eau et de vapeur fait peu à peu baisser la pression. Dans le script, la pression retombe progressivement à 0 grâce au bloc **ajouter à Pression -15** de la boucle **2** **répéter 10 fois**, ce qui revient à soustraire 150 en tout.



À la fin de l'éruption, le lutin envoie le message **Fin de l'éruption** à tous les lutins. Tu remarques que dans ce programme, aucun bloc ne fait augmenter la pression : c'est le lutin Barre qui va s'en charger.





La formule magique

Les scientifiques qui étudient ce geyser ont trouvé une règle pour connaître l'heure de sa prochaine éruption. Si on mesure avec un chronomètre la durée D (en minutes) de l'éruption, on peut calculer le temps T (en minutes) à attendre avant la prochaine éruption grâce à la formule $T = (D \times 12) + 33$. Si on note l'heure de l'éruption en cours, on peut donc savoir quand sera la prochaine !

La partie 2 du script 5 permet de calculer le nombre de minutes jusqu'à la prochaine éruption. D'abord, on choisit au hasard la durée de l'éruption, entre 1 et 6 minutes, grâce au bloc **nombre aléatoire entre 1 et 6**. C'est comme si tu lançais un dé !

Puis le programme utilise les blocs **actuel heure** et **actuel minute** de la palette Opérateurs, qui permettent de donner l'heure. Il enregistre l'instant (en minutes) du début de l'éruption dans la variable Début. Si elle a lieu à 14 h 05 par exemple, la variable Début prend la valeur $(14 \times 60) + 5 = 845$ minutes.

On peut alors calculer l'heure de la prochaine éruption grâce à la formule $\text{Prochain} = \text{Début} + (\text{Durée} \times 12) + 33$. Par exemple, si $\text{Durée} = 3$ minutes, alors $\text{Prochain} = 845 + (3 \times 12) + 33 = 914$.

Pour convertir ce résultat en une heure qu'on peut lire sur une montre, utilise les opérateurs diviser, arrondi et modulo. Tu obtiens l'heure en divisant Prochain par 60 et en arrondissant le résultat. L'opérateur modulo donne le reste de la division de Prochain par 60 : ce sont les minutes.

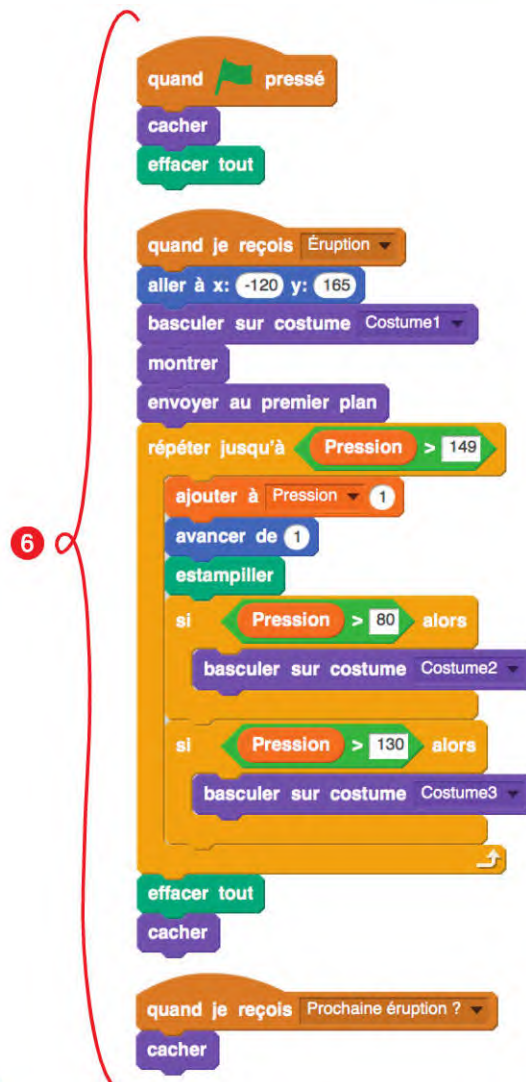
Dans notre exemple, cela donne : $914 / 60 = 15,23333$, qu'on arrondit à 15. L'heure de la prochaine éruption est donc prévue à 15 heures. Pour obtenir les minutes, on prend le reste (0,23333) qu'on multiplie par 60 : on obtient alors 14 minutes. La prochaine éruption aurait donc lieu à 15 h 14.



Petite astuce

Pour vérifier et comprendre les différentes opérations, affiche sur la scène les variables **actuel heure** et **actuel minute** en cochant leur case dans la palette Capteurs, et les variables que tu as créées en cochant aussi leur case dans la palette Données.

La pression monte...

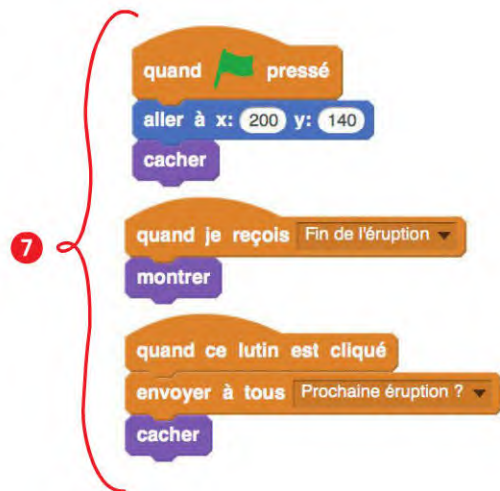


Clique sur le lutin Barre, qui va indiquer la montée de la pression jusqu'à la valeur limite de 150, sous la forme d'une barre qui grandit et change de couleur. Il possède 3 costumes qui sont affichés selon la valeur de la pression. Écris son programme **6**.

Au début du programme, le lutin est caché, puis il apparaît en haut à gauche avec son costume 1 lorsque le message **Éruption** est reçu. Dans la boucle **répéter jusqu'à** **Pression > 149**, le bloc **ajouter à Pression** **1** permet d'augmenter progressivement la pression jusqu'à la valeur maximale. Le lutin est alors déplacé de 1 vers la droite (bloc **avancer de 1**) et recopié grâce au bloc **estampiller** de la palette Stylo : à chaque fois que ce bloc est lu, le lutin est redessiné à l'identique mais à droite du précédent. Quand la pression dépasse 80, alors on le dessine en orange, et si elle est plus grande que 130, on le dessine en rouge.

Le lutin Flèche2

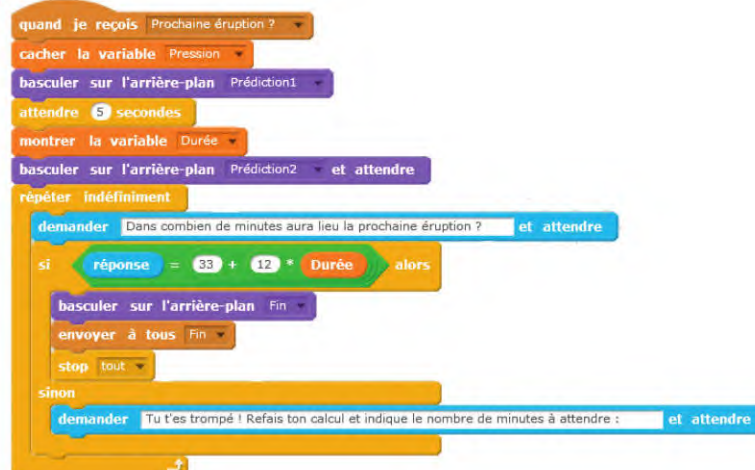
Clique sur le lutin Flèche2 qui va permettre de passer à la deuxième partie du jeu. Ajoute-lui le script **7** pour qu'il apparaisse à la fin de l'éruption. Lorsqu'on clique sur le lutin, il envoie à tous le message **Prochaine éruption ?**.



Retour sur la scène

Retourne sur la scène et ajoute le script **8**, qui se déclenche à la réception du message **Prochaine éruption ?**. Les arrière-plans **Prédiction1** puis **Prédiction2** sont alors affichés. Le bloc **demande** **et attendre** attend une réponse tapée au clavier et l'enregistre dans la variable **réponse**. Si la valeur entrée au clavier est la bonne, c'est gagné : l'arrière-plan bascule sur **Fin** et le message **Fin** est envoyé à tous. Sinon, il faut refaire son calcul et tenter à nouveau sa chance !

8



Tom apparaît !

Clique sur le lutin Tom et ajoute-lui le script **9**. Tom apparaît suite au message **Prochaine éruption ?**. Lorsque la bonne réponse a été donnée, il dit l'heure de la prochaine éruption dans une bulle. L'astuce ici consiste à utiliser 1 bloc **dire** **pendant 2 secondes** et 4 blocs **regroupe** pour construire une phrase à partir de différents éléments.

9



Tu peux maintenant lancer le jeu pour voir le geyser s'animer et prévoir la prochaine éruption !

Sur le site du parc, tu peux suivre en direct une éruption du Vieux Fidèle !
<http://www.nps.gov/features/yell/webcam/oldFaithfulStreaming.html>



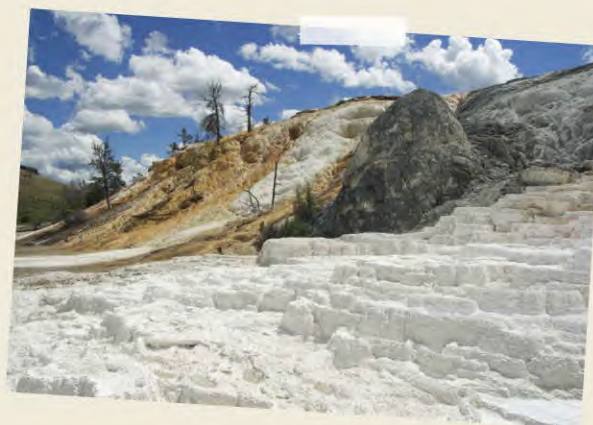
Le carnet de Lila

Le parc de Yellowstone s'étend au-dessus du cratère effondré (appelé caldeira) d'un énorme volcan. Les effets de la chaleur des profondeurs de la terre sont visibles partout, sous forme de geysers, fumerolles, sources chaudes et fontaines de boue bouillonnante. Si le Vieux Fidèle est le plus célèbre des geysers du parc, il en existe plus de 300 en tout, chacun ayant sa personnalité : régulier, ou rare et capricieux, avec des éruptions intenses et bruyantes comme celles du geyser Steamboat, ou gracieuses comme celles du geyser Anémone.



L'eau des sources chaudes atteint parfois plus de 90 °C. Leurs couleurs vives sont dues aux bactéries qui les colonisent.

L'eau très chaude et acide est aussi très riche en sels minéraux qui pétrifient les arbres et s'accumulent petit à petit pour construire des « terrasses », comme ici à Mammoth au nord du parc.



Bravo, tu as gagné ton badge de **niveau 3** !



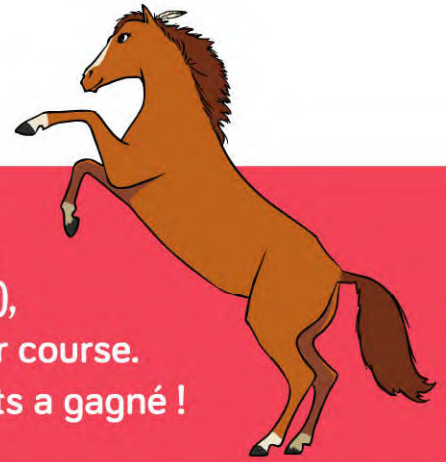
Étape 4

La course de chevaux de Crow Agency

Tom et Lila vont assister aujourd'hui au grand pow-wow des Indiens Crows, un rassemblement avec des concours de danse et de chant, et des épreuves à cheval. Ils arrivent juste à temps pour la grande course de l'après-midi.

Apprends à te servir du chronomètre. Utilise le clavier pour déplacer un lutin. Découvre comment rendre un jeu plus réaliste en faisant défiler le paysage et bouger les personnages.





Le jeu

Chacun des 2 joueurs contrôle un cheval et doit attraper le maximum d'objets (plumes et étoiles), tout en évitant les obstacles qui ralentissent leur course. Au bout d'une minute, celui qui a le plus de points a gagné !

Plante le décor

Dans Scratch, ouvre le fichier Etape4.sb2 qui contient déjà tous les lutins pour le jeu.

Clique sur la scène, qui a 6 arrière-plans. Pour les faire apparaître les uns après les autres, écris le script **1**. D'abord, on voit les instructions du jeu, puis il y a un compte à rebours et la course commence. Le message **Départ** est alors envoyé à tous les lutins. Remarque les 2 blocs de la palette Sons qui permettent d'ajouter des effets sonores : un hennissement dans sa totalité, puis le bruit du galop à l'intérieur d'un bloc **répéter indéfiniment** pour qu'on l'entende durant toute la course.



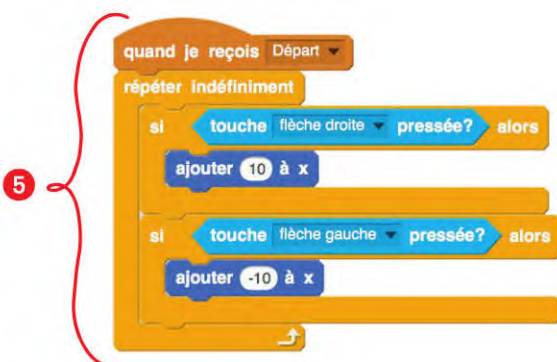
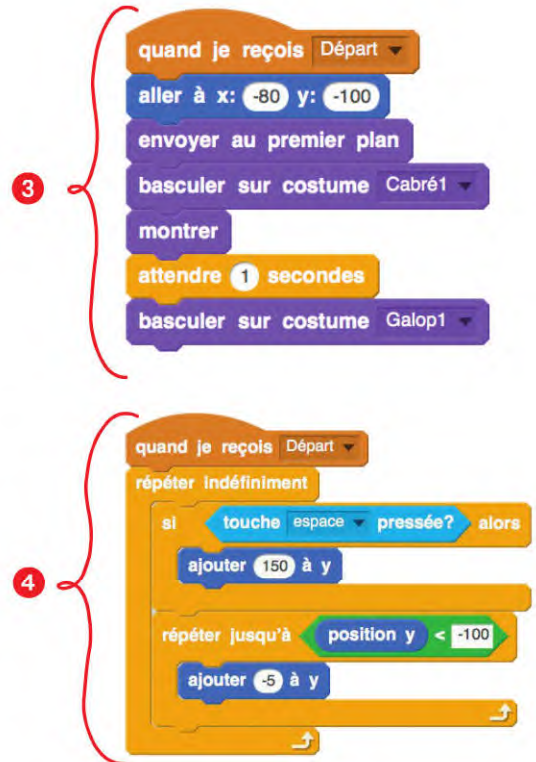
Ajoute aussi le programme 2, qui démarre lorsqu'il reçoit le message **Départ**. Il utilise le chronomètre que tu trouveras dans la palette Capteurs. Le bloc **réinitialiser le chronomètre** met le chronomètre à 0 au moment du départ. Après 60 secondes, la course est terminée, la scène envoie à tous le message **Fin**.

À cheval ! yiiiihaaa...

Clique sur le lutin Cheval1 et ajoute-lui les programmes 3 à 5. Grâce au 3, il est caché au début du jeu, puis apparaît quand il reçoit le message **Départ**, d'abord avec son costume Cabré1 puis Galop1.

Le programme 4 fait sauter le cheval lorsque le joueur appuie sur la barre d'espace. Si on la presse, alors le cheval est déplacé vers le haut (**ajouter 150 à y**). Puis il redescend progressivement grâce au bloc **répéter jusqu'à** de la palette Contrôle. Tant que la variable de position y du lutin est supérieure à -100 (ce qui correspond au niveau du sol), on lui enlève 5 pour le faire descendre.

Pour faire bouger le cheval à gauche ou à droite, c'est simple grâce au programme 5 : si on appuie sur la touche Flèche droite du clavier, on ajoute 10 à x ; si on appuie sur la flèche gauche, on ajoute -10 à x. Clique ensuite sur le lutin Cheval2, qui a presque les mêmes costumes que Cheval1. Ajoute-lui les mêmes scripts, mais en programmant cette fois la touche X pour sauter, et les touches A et Z pour se déplacer. Bien sûr, tu peux choisir d'autres touches si tu veux !



Et souviens-toi de l'astuce de l'Étape 2 pour copier un script...

Le paysage défile

Pour donner l'impression que les chevaux avancent, tu vas utiliser un vieux truc de cinéma : faire bouger le décor ! Les lutins Piste1 et Piste2 vont t'aider.

Écris le programme **6** pour le lutin Piste1, en créant la nouvelle variable Défilement à partir de la palette Données.

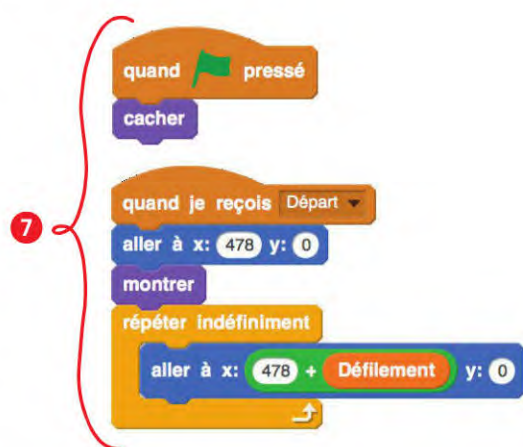
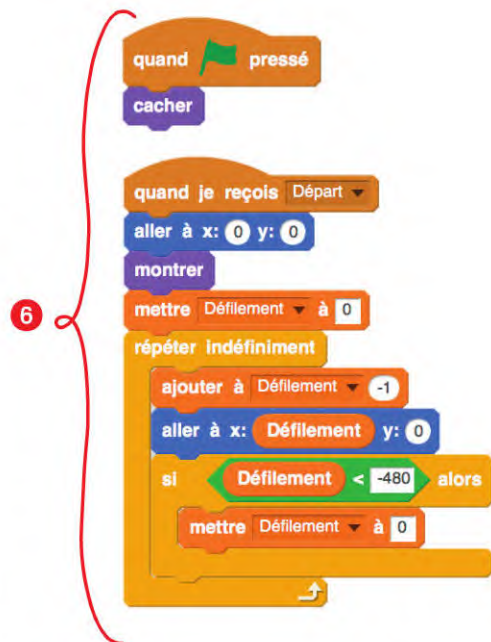
Le lutin apparaît à la position $x = 0$, $y = 0$ lorsqu'il reçoit le message **Départ**. La variable Défilement est alors mise à 0. Une boucle **répéter indéfiniment** permet de faire défiler lentement la piste. Dans cette boucle, tu enlèves 1 à la variable Défilement, puis tu déplaces le lutin avec le bloc **aller à x: Défilement y: 0**. Remarque que la valeur de y reste à 0 pendant tout le programme (la piste ne décolle pas !). Puis, lorsque la variable Défilement atteint -480, tu la remets à 0.

Pourquoi -480 ? La scène va de $x = -240$ à $x = +240$, donc elle a une largeur totale de 480. Au début, la variable Défilement vaut 0, donc la valeur -480 est atteinte quand le lutin sort de la scène par la gauche.

Clique sur le drapeau vert et observe le déplacement du lutin et l'évolution de la variable. Ça marche ! Le paysage défile, comme si le cheval se déplaçait ! Mais quelque chose cloche : un espace vide apparaît à droite quand le lutin défile. Pour régler ce problème, tu vas utiliser le lutin Piste2.

Écris le programme **7** pour ce lutin. Il va défiler avec une position x toujours décalée de 479 par rapport au lutin Piste1. Le problème est réglé : il n'y a plus d'espace vide ! Les 2 lutins Piste1 et Piste2 défilent côte à côte en permanence, couvrant toute la largeur de la scène.

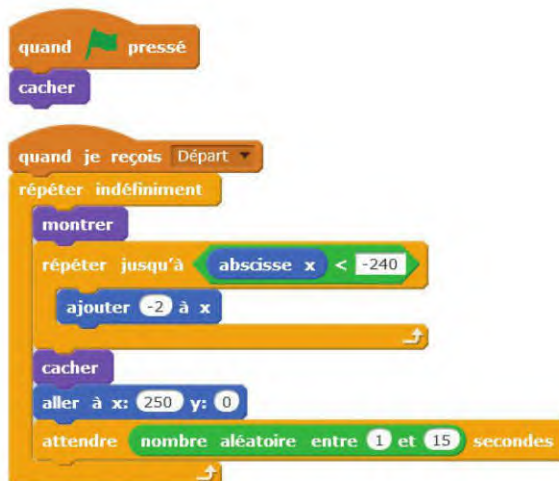
Vérifie à présent que la piste défile correctement ! Tu peux la redessiner comme tu veux, ajouter des collines, des montagnes, des arbres...



Ajoute un peu de verdure

Clique maintenant sur le lutin Arbre et ajoute-lui le programme **8** qui fait défiler des arbres sur le bord de la piste. Le bloc **répéter indéfiniment** fait apparaître le lutin à droite de l'écran, puis le fait défiler en diminuant sa position x de 2 jusqu'à ce qu'il sorte de la scène. Le lutin est alors caché pendant une durée au hasard entre 1 et 15 secondes (bloc **attendre nombre aléatoire entre 1 et 15 secondes**) avant de réapparaître et de recommencer à bouger.

8



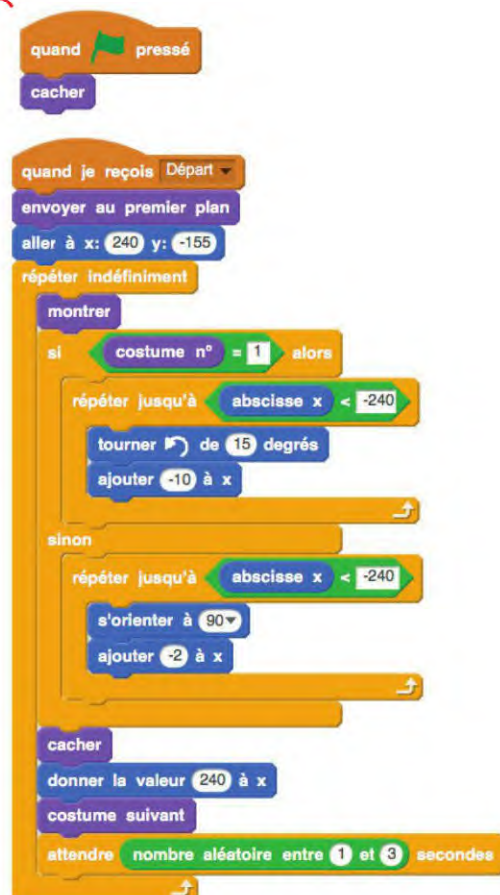
Attention aux obstacles !

Clique sur le lutin Obstacles. Il possède 3 costumes : Buisson, Rocher et Troncs. Le rocher et les troncs vont défiler comme les arbres. Pour le buisson, tu vas le faire en plus tourner sur lui-même, comme s'il était roulé par le vent.

Écris le programme **9**.



9

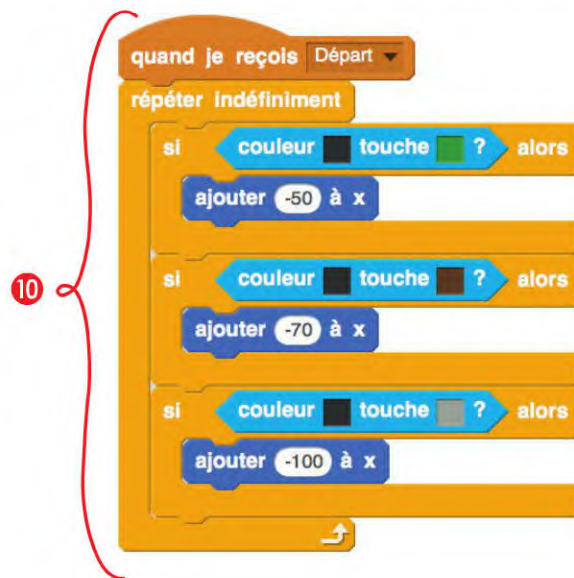


Si le lutin porte le costume n° 1 (Buisson), il défile rapidement jusqu'au bord de la scène, en tournant sur lui-même grâce aux blocs **ajouter -10 à x** et **tourner de 15 degrés** à l'intérieur d'un bloc **répéter jusqu'à**. Sinon, si le lutin porte le costume Rocher ou Troncs (costume n° 2 ou 3), il défile plus lentement avec le bloc **ajouter -2 à x**. Le bloc **s'orienter à 90** oriente les costumes Troncs et Rocher dans le bon sens. Ensuite, le lutin change de costume grâce au bloc **costume suivant** (quand il arrive au costume n° 3, il retourne au n° 1, et ainsi de suite). Enfin, le bloc **attendre nombre aléatoire entre 1 et 3 secondes** choisit au hasard le moment où apparaît le prochain obstacle. Les obstacles vont ainsi défiler chacun leur tour pendant toute la course !

Maintenant que tu as programmé les obstacles, il faut que tu écrives un script pour qu'à chaque fois qu'un cheval touche un obstacle, il recule sur la gauche. Place-toi sur le lutin Cheval1, tu vas lui ajouter le programme 10.

Commence par récupérer le bloc **couleur touche ?** dans la palette Capteurs. Il va te permettre de savoir si le cheval touche un obstacle. Clique dans une de ses petites cases, puis sur un point de l'écran (lutin, scène, ou ailleurs) : la case prend la couleur du point cliqué ! Ici, clique sur le bout du sabot d'un cheval pour colorer la case de gauche en noir, et sur le buisson pour passer l'autre case en vert. Ainsi, si ces deux couleurs se touchent, alors les chevaux reculent de 50. Si un cheval touche les troncs (marron), alors il recule de 70. Lorsqu'il touche un rocher (gris), il recule de 100.

Écris le reste du programme, puis ajoute le même au lutin Cheval2.



À qui les bonus ?

Pour gagner à ce jeu, il faut attraper le plus d'objets possible qui apparaissent pendant la course, car ils donnent des bonus.

Clique sur le lutin Bonus. Il a 2 costumes, Plumes et Étoile, qui correspondent à des bonus d'1 et 3 points. Écris pour ce lutin le programme 11, qui va permettre de compter les points des 2 joueurs. Pour cela, crée les variables Score1 et Score2.

Dans le jeu, un cheval marque 1 point s'il touche le lutin Bonus au costume Plumes, et 3 points s'il touche celui au costume Étoile. Pour vérifier quel cheval attrape quel bonus,

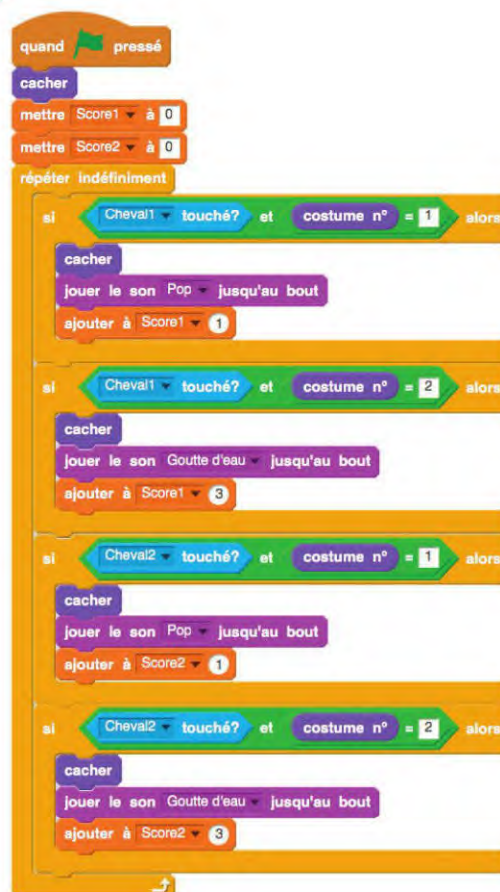


le programme utilise 4 blocs



Au début du programme, les scores sont mis à 0, puis on leur ajoute 1 ou 3 selon les bonus gagnés. On entend le son Goutte d'eau quand un cheval attrape un bonus Étoile, et le son Pop quand c'est un bonus Plumes. Souviens-toi, dans l'Étape 1 du voyage, tu as vu comment choisir n'importe quel son à partir de l'onglet Sons.

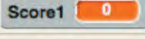
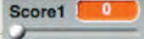
11



Écris à présent le script 12 de la page 48 qui permet de contrôler le moment et l'endroit où apparaissent les bonus.



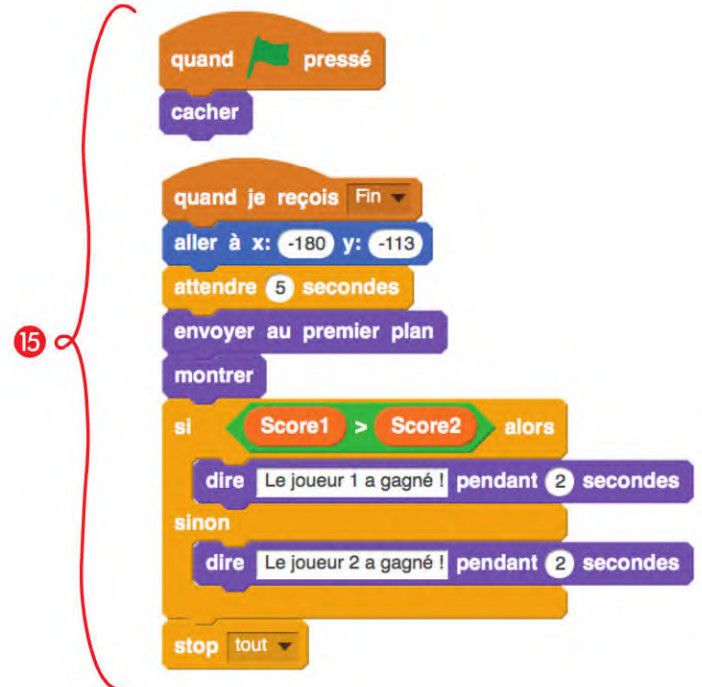
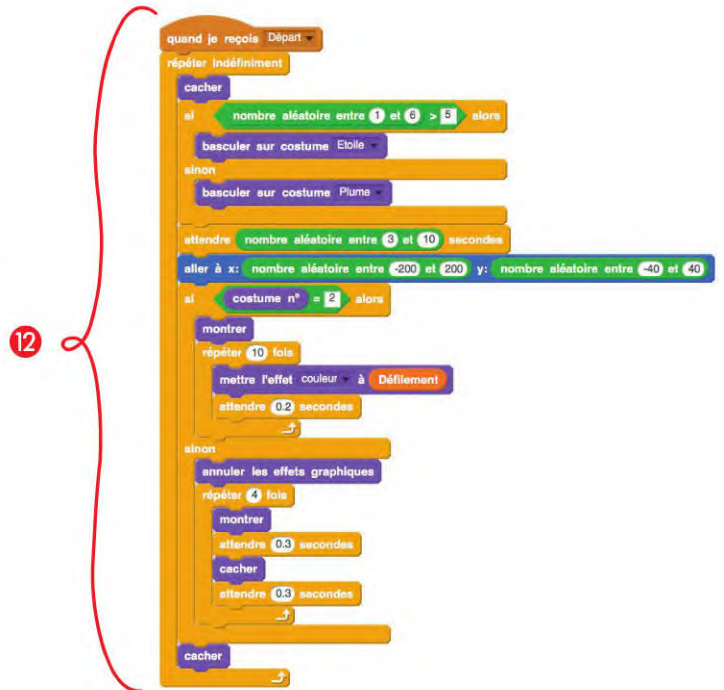
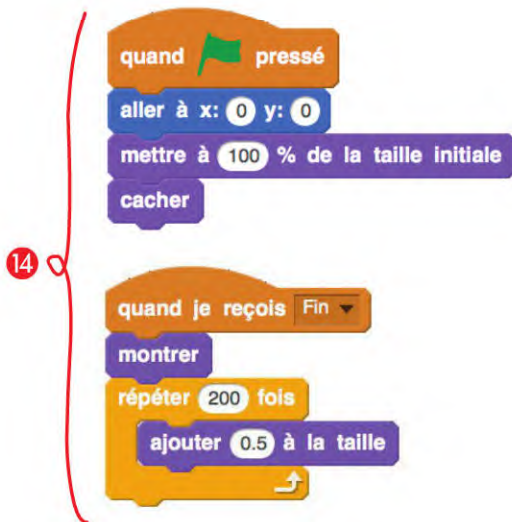
Petite astuce

En double-cliquant sur les variables visibles sur la scène, tu peux changer leur aspect. Par exemple, pour la variable Score1, tu peux avoir les affichages ,  et .

Ce programme utilise 4 fois le bloc **nombre aléatoire entre** et **et**. Une première fois pour faire apparaître l'étoile seulement 1 fois sur 6 (comme si tu lançais un dé et que tu affichais l'étoile seulement lorsque tu fais un 6), une deuxième fois pour attendre entre 3 et 10 secondes avant de faire apparaître le lutin Bonus, et les deux dernières fois pour choisir les positions x et y au hasard. Libre à toi de modifier tous ces paramètres !

Pour rendre le jeu encore plus vivant, on a ajouté des effets visuels. Dans le script, le bloc **mettre l'effet couleur à Défilement** fait changer rapidement la couleur du bonus Plumes. Et le bloc **répéter 4 fois**, qui contient les blocs **montrer**, **cacher** et **attendre 0.3 secondes**, fait clignoter le bonus Étoile.

Et le gagnant est...



Rappelle-toi, à la fin de la course, le lutin Scène envoie le message **Fin** dans le programme 1. Pour que le jeu s'arrête, ajoute le programme 13 aux lutins Cheval1, Cheval2, Piste1, Piste2, Arbre, Obstacles et Bonus. Il permet de cacher ces lutins et de stopper tous leurs scripts lorsqu'ils reçoivent le message **Fin**.

Clique sur le lutin Arrivée et ajoute-lui le programme 14. Ici, tu augmentes progressivement la taille du lutin pour donner l'impression qu'on se rapproche de l'arrivée.

Utilise pour cela une boucle **répéter 200 fois** pour augmenter la taille de 0,5 à chaque fois. Tu peux modifier ces valeurs : selon celles que tu choisis, l'arrivée sera plus ou moins rapide !

À la fin du jeu, c'est Lila qui va annoncer le vainqueur. Clique sur le lutin Lila et écris le programme **15**.

Le carnet de Lila

Le parc de Yellowstone est une terre indienne. Les tribus Sheepeater, Crow, Shoshone, Blackfeet, Kiowa et d'autres y vivaient ou y séjournèrent durant les périodes de chasse. Désormais, la réserve de la Wind River dans le Wyoming est une des plus grandes réserves de tous les États-Unis et le cœur des tribus Shoshone et Arapaho. Dans le Montana, la réserve des Indiens Crows compte le site de la bataille de Little Big Horn, la plus grande victoire des Sioux et des Cheyennes contre les tunique bleues du général Custer. Tous les ans, un grand pow-wow a lieu à Crow Agency la troisième semaine d'août. C'est l'occasion pour les communautés et les familles de renforcer leurs liens. Plusieurs centaines de larges tipis sont montés en cercle autour de l'endroit où sont organisés des concours de danse et de chant, ainsi qu'un grand rodéo.



Bravo, tu as gagné ton badge de **niveau 4** !



Apprends à
programmer un lancer de
dé. Utilise des variables sous
forme de listes et des clones
de lutins. Découvre comment
créer tes propres blocs
d'instructions.

Étape 5

Sur la piste de l'Oregon

Aujourd'hui, Tom et Lila partent randonner au sud du Wyoming, en suivant la piste qui a mené plus de 400 000 pionniers vers les terres accueillantes de l'Oregon. Il ne faut pas qu'ils oublient leur pique-nique !



Le jeu

Tu es un pionnier qui essaie de rejoindre l'Oregon. Pour y arriver, tu devras survivre à 4 étapes et autant de bivouacs. Il faudra que tu aies une bonne stratégie... et de la chance aux dés !



En route !

Ouvre le fichier Etape5.sb2. Regarde le lutin Scène ; il a 9 arrière-plans qui vont donner les instructions du jeu et annoncer les événements arrivant au cours du voyage. Dans le fichier, il y a 8 autres lutins dont tu vas écrire les scripts.



Si le jeu te semble compliqué, lis la règle détaillée qui est dans le dossier Etape5 des ressources du livre.

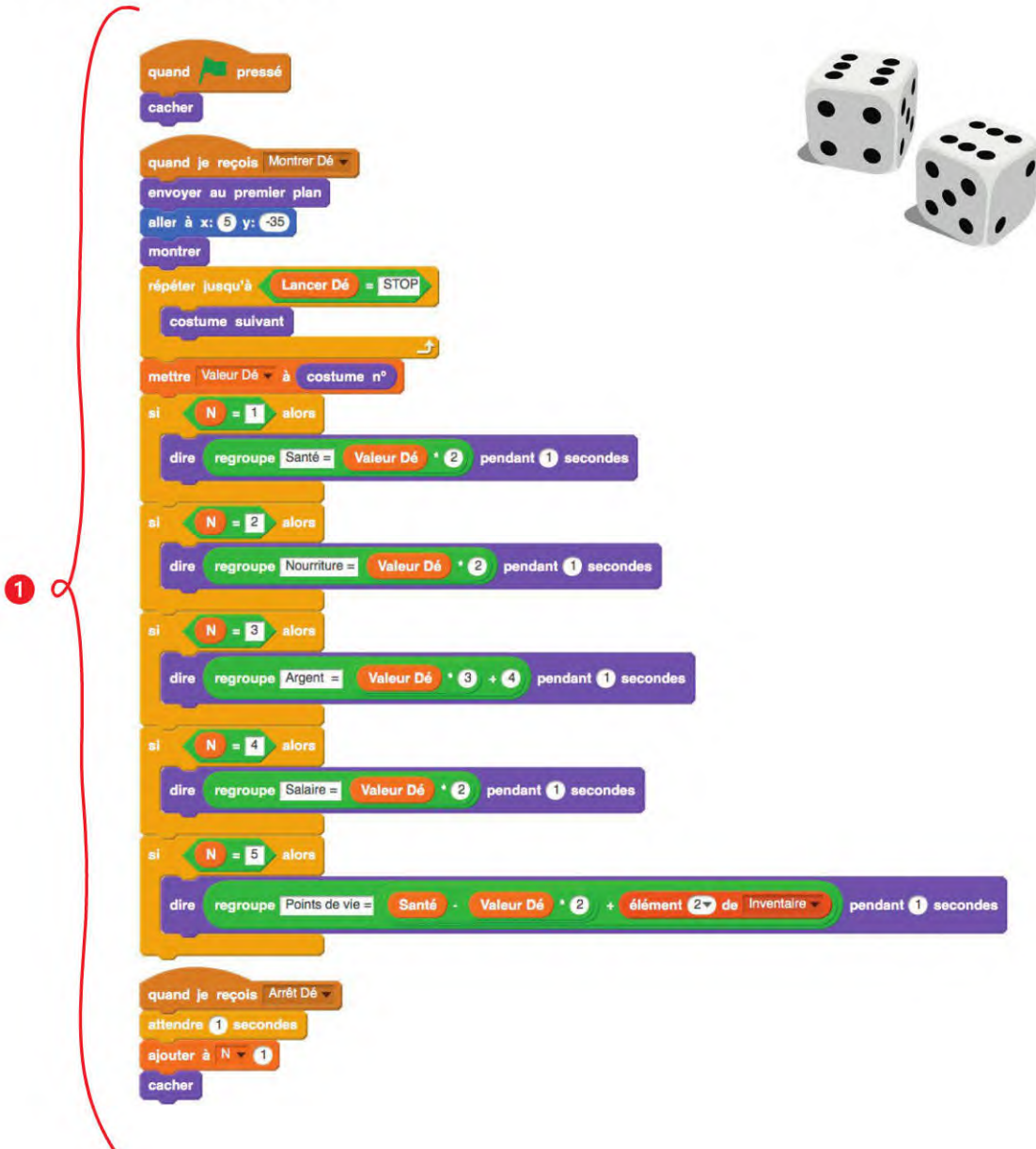
Au début du jeu, les points de nourriture, de santé et d'argent sont fixés par un lancer de dé que tu vas programmer. Après, tu prends la route qui sera semée d'aventures ! À la fin de chaque étape, le joueur doit se reposer et se nourrir au bivouac, et il peut aller au magasin pour acheter de la nourriture et des médicaments. S'il arrive au bout du voyage, il a gagné !



Programme le dé

Place-toi sur le lutin Dé, qui a 6 costumes, un par numéro de 1 à 6. Ajoute-lui le script 1.

Dans ce programme, le lutin Dé apparaît quand il reçoit le message **Montrer Dé** envoyé par le programme principal (c'est le 11) que tu écriras plus tard. Il change alors de costume (et donc de numéro) tant que la variable Lancer Dé n'est pas égale à STOP. Pour le faire changer de costume, on utilise le bloc **costume suivant**, que tu as découvert page 46.



Quand la variable Lancer Dé vaut STOP, la variable Valeur Dé prend comme valeur le numéro du costume, donc un chiffre de 1 à 6.

La variable N indique le nième lancer de dé en cours (1 si le dé a été lancé 1 fois, 2 s'il a été lancé 2 fois...). Les 5 blocs si alors **si** **alors** qui suivent servent à annoncer le résultat obtenu.

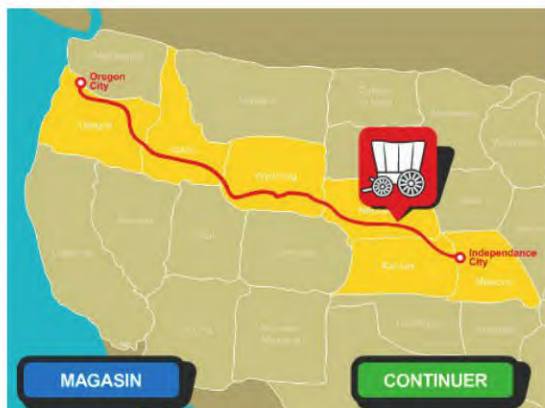
Écris ensuite le script **2** pour le lutin Stop, qui est un bouton permettant d'arrêter le lancer de dé en cliquant dessus. Le message **Arrêt Dé** est alors envoyé à tous les lutins. On emploie aussi la variable Lancer Dé. Avant chaque lancer de dé, cette variable est mise à GO dans le programme **9** (tu l'écriras plus tard). Lorsque tu cliques sur le bouton Stop, tu fais passer cette variable à STOP, ce qui arrête le dé.

Suis le chariot sur la carte

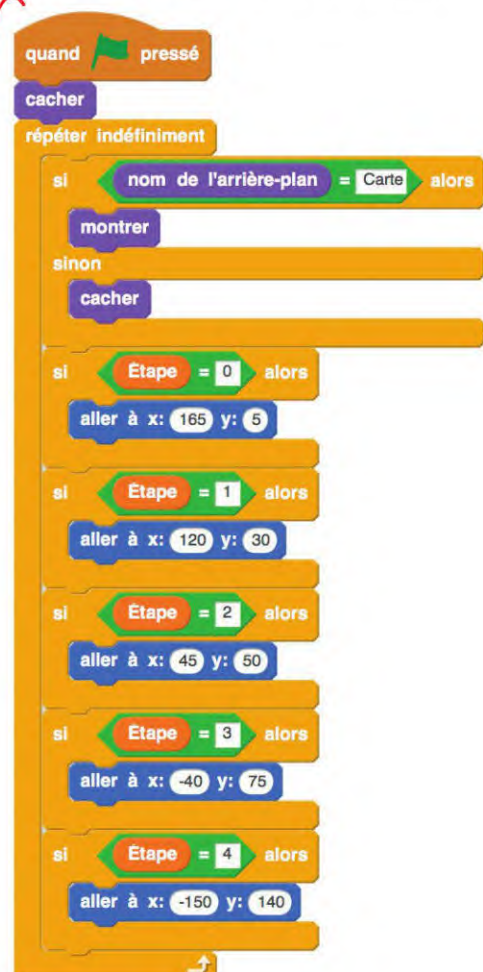
L'arrière-plan Carte montre le parcours du joueur, à partir du Missouri jusqu'en Oregon. Sur l'image tu vois 3 autres lutins : les boutons Magasin et Continuer, et le lutin Chariot qui indique la position du chariot sur la piste.



Place-toi sur le lutin Chariot et ajoute-lui le programme **3** qui est très simple. Il permet de rendre visible le chariot quand l'arrière-plan est la carte. À chaque étape du parcours, il le fait apparaître à un endroit différent sur la piste.



3

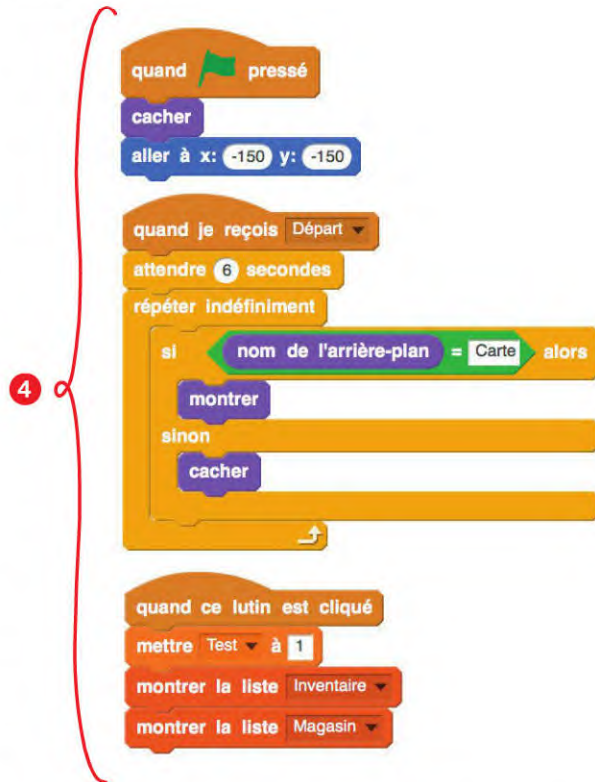


Un tour au magasin

Place-toi sur le lutin Magasin, qui est un bouton apparaissant quand l'arrière-plan est la carte. Regarde son script **4** avant de l'écrire.

Au début du jeu, ce bouton est caché pendant que le joueur tire au dé ses points de nourriture, de santé et d'argent. Puis il apparaît quand il reçoit le message **Départ** envoyé par le script **10** (tu verras plus loin, c'est lorsque le joueur clique sur le bouton Jouer).

Quand on clique sur le bouton Magasin, la variable Test passe à 1, ce qui permet à la scène de faire apparaître son arrière-plan Magasin (voir le script **10**). Le clic sur le bouton Magasin rend aussi visibles les listes Magasin et Inventaire. Les listes sont des tableaux qui peuvent contenir plusieurs variables. Tu les crées comme des variables à partir de la palette Données, en cliquant sur le bouton Créer une liste. Tu es prêt maintenant pour écrire le programme **4** !



Clique sur le + pour ajouter un élément à la liste.

Tu vois les listes sur la scène en cochant leur petite case dans la palette Données. La liste Magasin montre les marchandises disponibles dans le magasin. La liste Inventaire montre ce que tu possèdes. L'élément 1 correspond à la nourriture, le 2 aux médicaments, et le 3 à l'argent.

Écris maintenant le script **5** pour le lutin Bouton. Ce bouton + doit apparaître 2 fois, à droite des éléments Nourriture et Médicaments de la liste Magasin. Un clic dessus permet d'acheter de la nourriture ou des médicaments.



Dans le premier morceau de code, on fait apparaître ce lutin à droite de l'élément Nourriture quand la variable Test vaut 1, ce qui est le cas quand on a cliqué sur le lutin Magasin (voir script **4**). Ici, tu as besoin de montrer un second bouton +, mais c'est possible avec un seul lutin ! Il suffit d'utiliser le bloc **créer un clone de moi-même**, en choisissant « moi-même » dans le menu déroulant. Ainsi, tu recopies l'image de ce bouton, et les programmes écrits pour le lutin Bouton seront également valables pour son clone. Très pratique !

Le bloc **quand je commence comme un clone** permet alors d'annoncer les instructions données au clone. Si la variable Test vaut 1 (donc si on est dans le magasin), alors le clone est visible en face de la ligne Médicaments (**aller à x: 185 y: 122**).

Lorsqu'on clique sur le lutin ou son clone, le programme vérifie que le joueur a assez d'argent (ici, au moins 3 dollars) pour ajouter +2 à ses points de nourriture (l'élément 1 de la liste Inventaire), ou bien +1 à ses points de médicaments (l'élé-

ment 2 de la liste Inventaire). Dans les 2 cas, on enlève 3 dollars à l'élément 3 de la liste Inventaire. Pour savoir lequel des 2 boutons + a été cliqué (le lutin ou son clone ?), le script teste la position y de la souris au moment du clic avec les blocs **si** **alors**. Et si le joueur n'a pas assez d'argent, un message apparaît.

Pour sortir du magasin, il faut cliquer sur le lutin OK. Écris le script **6** pour ce lutin qui apparaît quand la variable Test vaut 1, donc quand on a cliqué sur le bouton Magasin (voir script **4**).

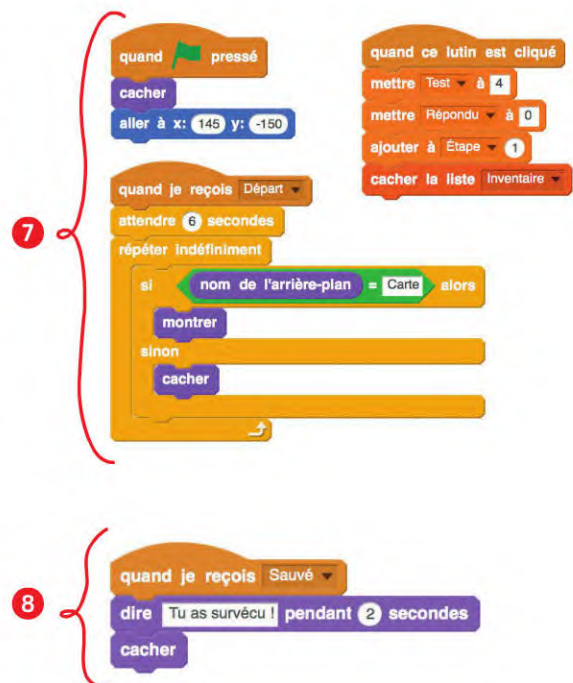
Quand on clique sur le lutin OK, on met la variable Test à 0, ce qui fait disparaître ce lutin ainsi que le lutin Bouton (voir script **5**). Les 2 listes disparaissent aussi et la scène bascule sur l'arrière-plan Carte.



Et le voyage continue...

Ajoute maintenant le script **7** au lutin Continuer, qui est un bouton sur lequel il faut cliquer pour passer à l'étape suivante du parcours (**ajouter à Étape** **1**). Il modifie également la valeur des variables Répondu et Test. On en reparlera...

Place-toi ensuite sur le lutin Lila qui intervient pour commenter les différents événements. Tous ses scripts ressemblent au programme **8**. Lorsque le lutin reçoit un message particulier (ici, « Sauvé ! »), il dit quelque chose (ici, « Tu as survécu ! »), puis il disparaît. Tu trouveras les autres scripts pour chaque message reçu par ce lutin dans la version complète du jeu, mais tu peux aussi inventer les tiens.

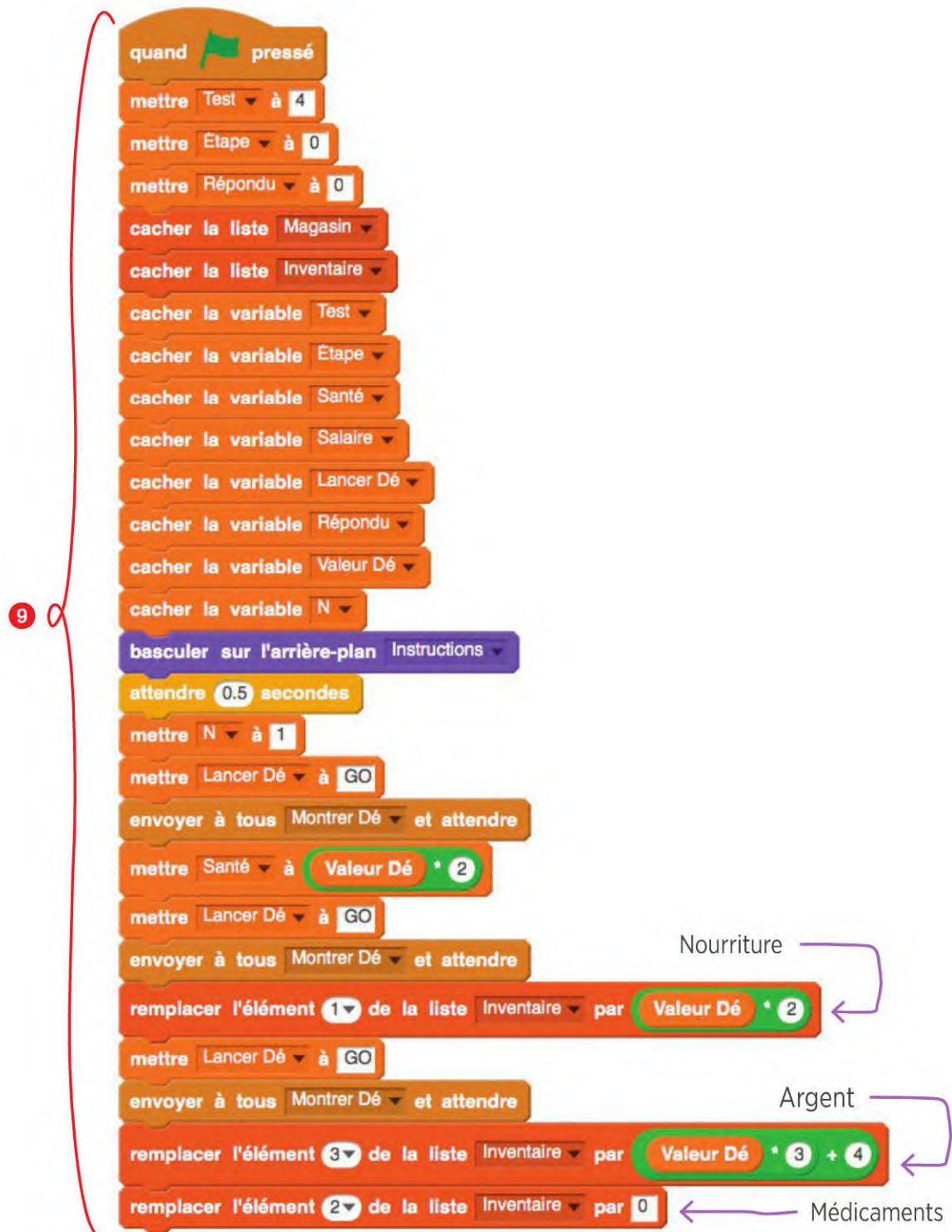


Bravo, mis à part la scène, tu as écrit les scripts de tous les lutins !

La scène finale !

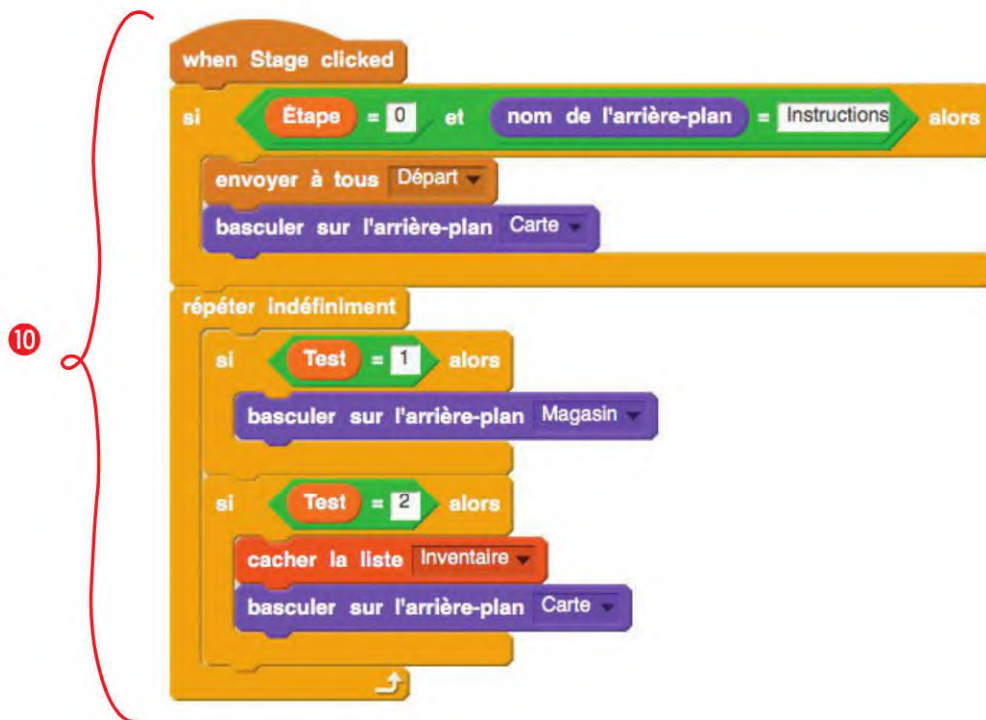
Clique à présent sur la scène, pour laquelle il faut écrire plusieurs programmes : celui qui va donner les valeurs de départ à toutes les variables, celui qui va définir l'ordre des étapes, et ceux qui décrivent le bivouac et chaque étape.

Écris d'abord le programme 9.



Au lancement du jeu, ce script donne une valeur aux variables et les cache. L'arrière-plan Instructions apparaît et le joueur lance 3 fois le dé pour choisir au hasard ses points de santé, de nourriture et d'argent. Au début du jeu, il n'a pas de médicaments.

L'arrière-plan Instructions invite le joueur à cliquer sur le bouton Jouer. Ce clic démarre le programme **10** ; écris-le.



Quand la scène est cliquée, si la variable Étape vaut 0 et que l'arrière-plan est Instructions (c'est le cas au début du jeu), le lutin bascule sur l'arrière-plan Carte et envoie le message **Départ**. Ce message déclenche des programmes pour les lutins Magasin, Continuer et Lila.

Lila t'indique alors qu'il faut faire des provisions avant de partir et que le joueur peut aller au magasin, comme on l'a vu dans les scripts **4**, **5** et **6**. Au cours du jeu, la valeur de la variable Test fait basculer l'arrière-plan sur Magasin ou Carte.

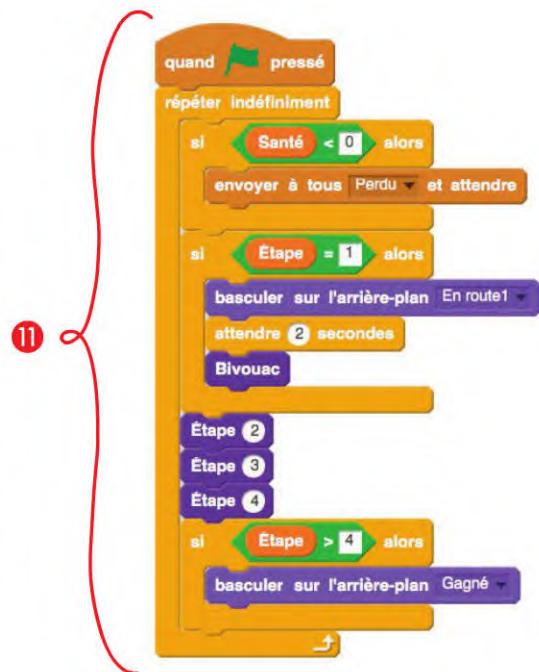


Petite astuce

Fais un clic droit sur un bloc **envoyer à tous message** dans un de tes scripts. Choisis « montrer émetteurs » pour voir quel lutin envoie le message, ou « montrer récepteurs » pour voir quels lutins le reçoivent. Wouah, les lutins concernés apparaissent éclairés dans la liste des lutins !

Maintenant, regarde le programme 11, tu l'écriras plus tard, c'est le cœur du jeu.

Ce script va vérifier pendant tout le jeu que la variable Santé est positive, sinon tu as perdu ! Il fait dérouler les étapes du parcours, l'une après l'autre. Une fois l'étape 1 franchie, l'arrière-plan En route1 apparaît et on utilise le bloc Bivouac. Tu remarques que ce bloc est d'une couleur qu'on n'a encore jamais vue... mystère ? En fait, c'est un bloc que tu vas personnaliser ! Tu vas construire un bloc qui correspond à un programme entier, et que tu pourras réutiliser dans un autre script.



Dans la palette Ajouter blocs, clique sur le bouton Créer un bloc. Tape le nom du nouveau bloc dans la zone rose et clique sur OK. Un nouveau bloc **définir Bivouac** apparaît alors dans la zone des scripts. Tous les blocs que tu vas lui accrocher formeront le bloc **Bivouac**, disponible dans la palette Ajouter blocs. Écris ce nouveau bloc 12.

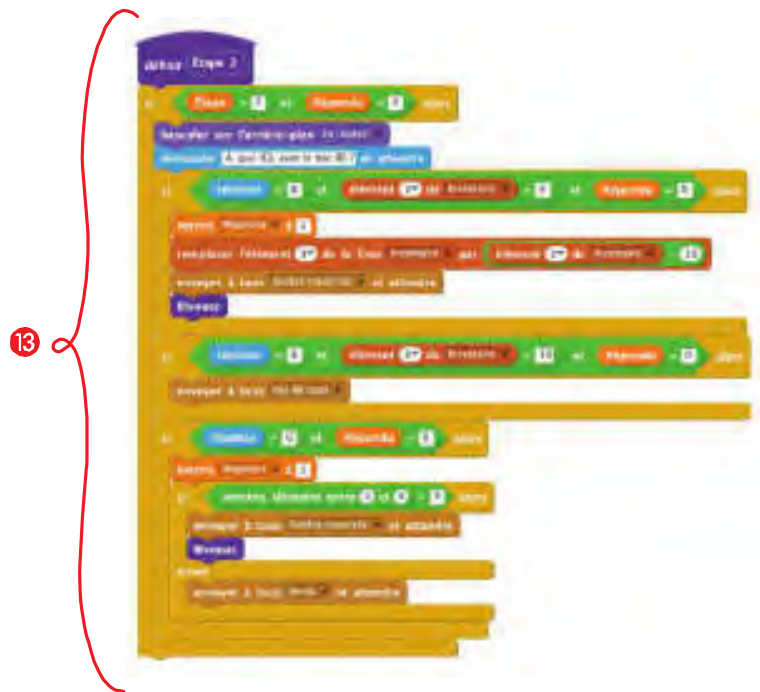


Au bivouac, l'occupation principale est de manger ! Si tu n'as plus de nourriture, ta santé se dégrade (ajouter à Santé -2). Sinon, tu dois choisir entre un repas copieux ou léger. Un repas copieux coûte 2 points de nourriture et rapporte 1 point de santé ; un repas léger coûte 1 point de nourriture et fait perdre 1 point de santé. Les messages envoyés à tous les lutins servent seulement à rendre le jeu plus vivant en déclenchant les commentaires de Lila.



À la fin du script, la variable Test est mise à 2, ce qui fait basculer sur l'arrière-plan Carte (script 10). Il faut alors attendre un clic sur le bouton Continuer (script 7) pour que la variable Test passe à 4. Ce clic ajoute aussi 1 à la variable Étape (script 7). Et donc le voyage continue et on passe à l'étape suivante.

Reviens sur le programme 11. Tu vois qu'il contient aussi un bloc personnalisé Étape 2. Écris ce nouveau bloc 13.



Dans cette étape, tu arrives au bord d'une rivière et tu dois choisir de passer à gué (c'est-à-dire en faisant traverser les chevaux là où l'eau est peu profonde) ou par le bac. L'arrière-plan En route2 est visible. Le programme démarre si on est bien à l'étape 2 et si la variable Répondu vaut 0, ce qui est le cas (voir script 7). Lorsque tu as fait ton choix, cette variable est mise à 1. Comme ça, la question n'est posée qu'une seule fois.

Si tu choisis le bac () et que tu as assez d'argent, ça te coûte 10 points d'argent et tu es sûr de traverser. Si tu choisis de passer à gué (), le programme tire un chiffre au hasard entre 1 et 6. S'il est plus grand que 3, tu traverses sans problème et tu peux te reposer au bivouac. Sinon, tu as perdu !

Tu es maintenant un super Scratcheur ! Tu peux imaginer d'autres étapes du jeu en créant d'autres blocs, ou jouer avec la version complète qui a déjà 4 étapes.



1 000 fois bravo, tu as gagné ton badge de **niveau 5** !