

TUTORIEL COZMO POUR LE CYCLE 4 PROJET COZMO FOOT !



Un tutoriel Génération Robots

Tutoriel Cozmo pour le cycle 4 : projet Cozmo Foot ! (complet)

Dans ce tutoriel pédagogique, nous allons vous montrer comment faire vos premiers pas avec l'interface **Code Lab** de Anki, qui vous permettra de programmer Cozmo.

Au programme : repérage dans un plan (x, y), conditions, fonctions et variables !

Activité COZMO FOOT

- **Durée :** 2-3 séances de 2h
- **Public :** 4ème/3ème
- **Discipline :** mathématiques, technologie, programmation
- **Thématique :** programmation
- **Niveau :** intermédiaire
- **Notions utilisées :** repérage dans l'espace, repérage d'un point ou d'une ligne dans un plan (mathématiques), conditions (programmation), variables (programmation), utilisation de capteurs (robotique, programmation)

Présentation rapide du robot Cozmo et de son application

Cozmo est un petit robot doté d'une puissante IA, de bras élévateurs, d'une tête très sympathique qui intègre un petit écran, d'une paire de chenilles motorisées très précises et de 3 cubes interactifs !



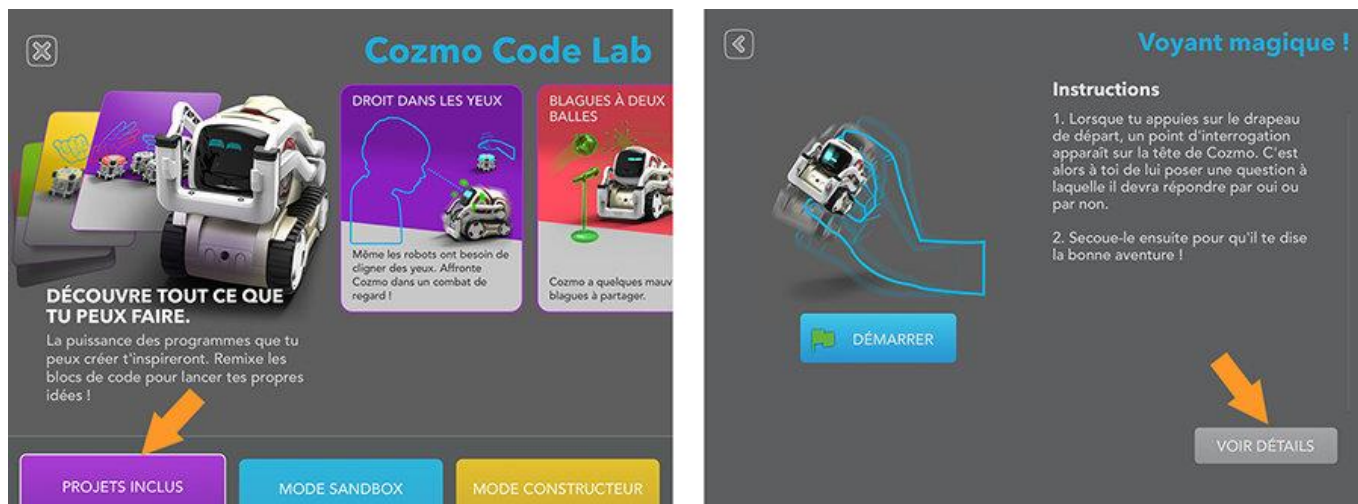
Quand vous achetez Cozmo, il vous faut ensuite télécharger l'application Cozmo (gratuite, disponible pour [iOS](#) et [Android](#)), qui vous permettra d'interagir avec Cozmo, mais aussi de le programmer.

Dans cette appli Cozmo, vous allez trouver Code Lab : une version de **Scratch Blocks** disponible en deux niveaux : **Sandbox** (cycle 2, 3, début du cycle 4) et le mode **Constructeur** (fin du cycle 4, lycée).

Code Lab : le mode constructeur

Le mode constructeur de Code Lab, qui est l'interface avancée de programmation de Cozmo, permettra à vos élèves de **créer des programmes complexes, de manière très progressive**.

Quand vous vous trouvez dans Code Lab, vous avez également accès à une bibliothèque de jeux et applications pour Cozmo (PROJETS INCLUS).

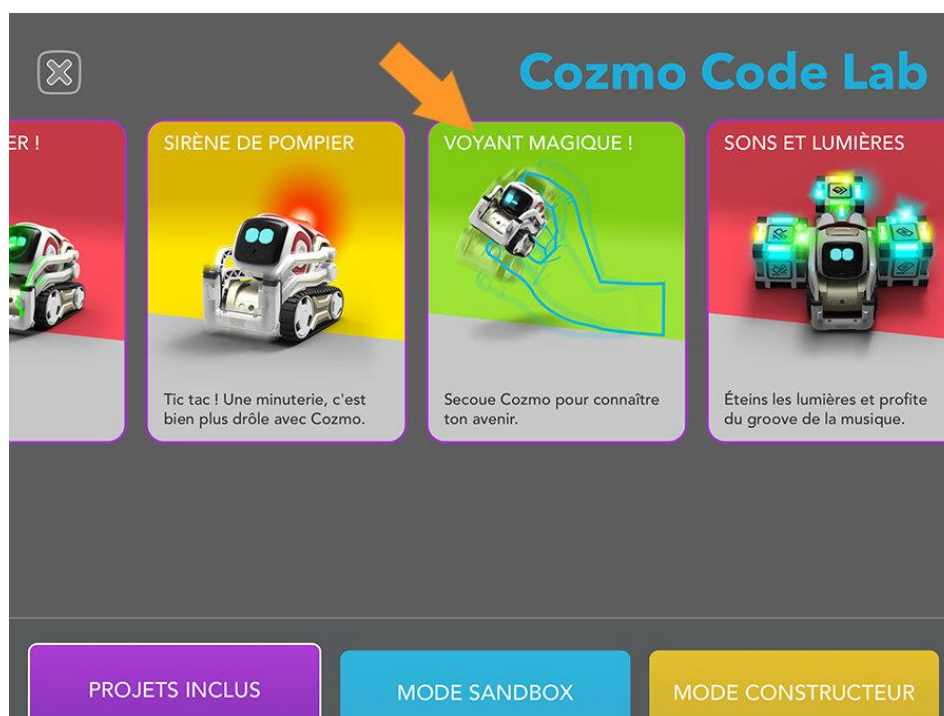


Si vous sélectionnez un jeu dans cette collection de projets, vous constatez que vous pouvez soit lancer le jeu, soit voir à quoi ressemble le code du jeu (VOIR DETAILS).

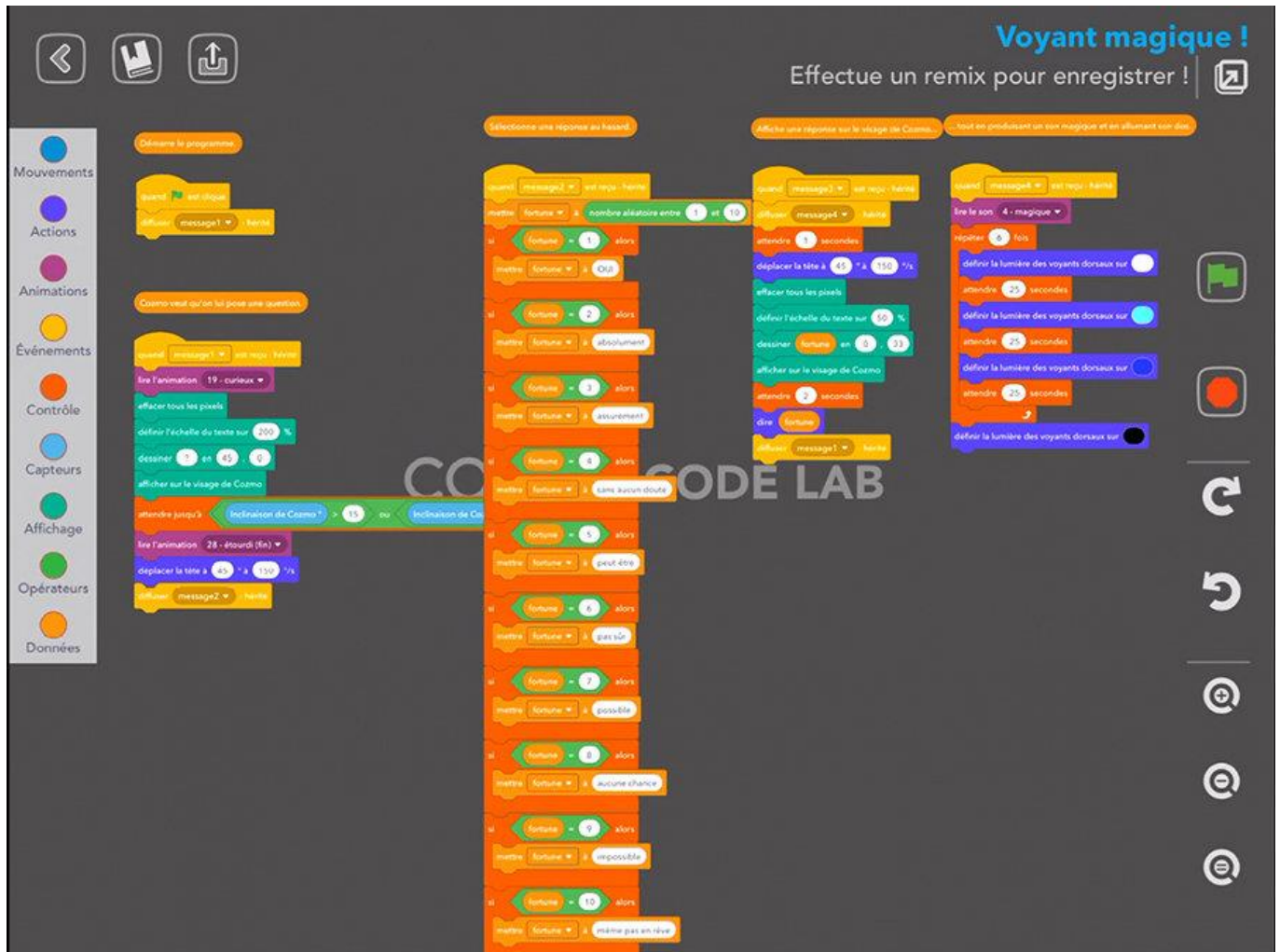
Une manière simple et très ludique d'apprendre à se servir du mode constructeur de Code Lab est de **partir d'un jeu déjà existant et de le modifier**.

C'est parti !

Sélectionnez le projet VOYANT MAGIQUE ! et cliquez sur VOIR DETAILS.



Vous avez maintenant devant vous le code du jeu.



Nous allons partir de ce code (VOYANT MAGIQUE !), qui transforme Cozmo en “magic 8 ball” et le modifier petit à petit jusqu’à obtenir un nouveau programme, où Cozmo aura le rôle de pronostiqueur de la coupe du monde !

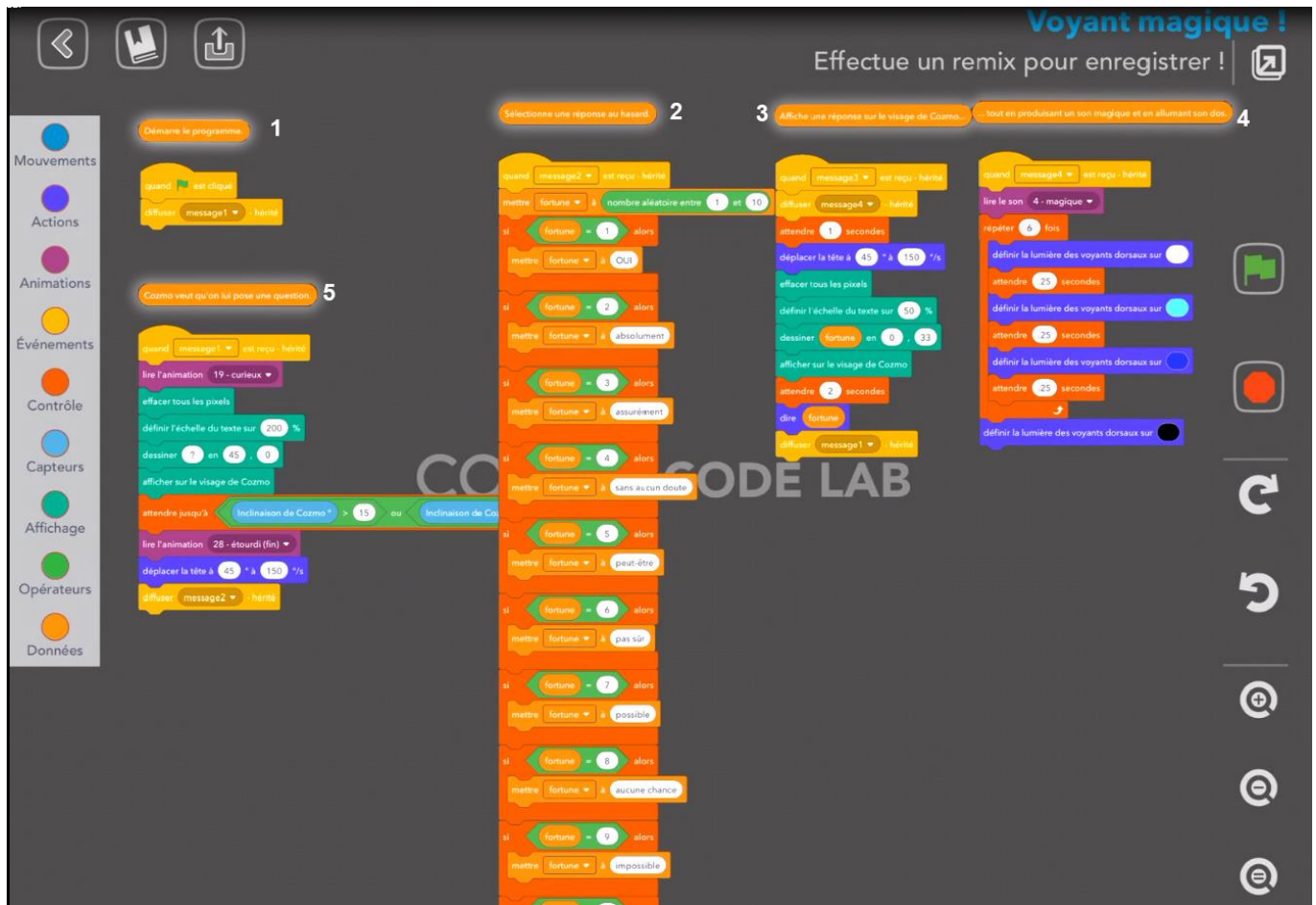


Comme vous le voyez, chaque bloc de couleur contient une instruction, qu’il est possible de modifier ou de combiner avec d’autres pour créer un programme.

1. Trois astuces avant de vous lancer

Commençons notre tutoriel... Avant toute chose, démarrez le projet VOYANT MAGIQUE ! et testez-le en posant des questions à Cozmo. Puis, sur votre tablette, cliquez sur VOIR DETAILS.

Astuce 1. Tout d’abord, regardez les blocs oranges situés en haut de l’interface (par exemple, le bloc “Affiche une réponse sur le visage de Cozmo”. Ces blocs sont là pour **décrire les portions de code qui se trouvent au-dessous**.



Ils sont l’équivalent des commentaires en programmation textuelle. Lorsque l’on programme, il est conseillé de bien commenter son code. Les commentaires (ci-dessous en vert) permettent de facilement comprendre le code (très utile si plusieurs développeurs travaillent sur le même code) et de pouvoir le modifier rapidement.

```
/**
 * Simple HelloButton() method.
 * @version 1.0
 * @author john doe <doe.j@example.com>
 */
HelloButton()
{
  JButton hello = new JButton( "Hello, wor
  hello.addActionListener( new HelloBtnList

  // use the JFrame type until support for t
  // new component is finished
  JFrame frame = new JFrame( "Hello Button"
  Container pane = frame.getContentPane();
  pane.add( hello );
  frame.pack();
  frame.show();           // display the fra
}
```

Astuce 2. Une autre bonne pratique est d’enregistrer dès maintenant ce projet en tant que “remix” de VOYANT MAGIQUE ! Il vous suffit de cliquer sur l’icône « Effectue un remix pour enregistrer ! » située en haut à droite de l’écran. Toutes vos modifications seront désormais sauvegardées automatiquement dans votre nouveau remix, que vous pourrez renommer.

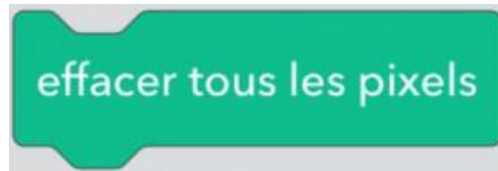
Astuce 3. Enfin, dernière astuce, pour supprimer un bloc ou un ensemble de blocs, faites le glisser en dehors de l’écran, sur la gauche (du côté où se trouve votre bibliothèque).

2. Modifier l'affichage sur l'écran de Cozmo : dessin et texte

Dans le jeu VOYANT MAGIQUE !, un point d'interrogation s'affiche sur l'écran de Cozmo lorsque l'utilisateur pose sa question au robot. **Nous allons transformer ce point d'interrogation en ballon de foot !**

Tous les blocs gérant l'affichage de l'écran sont verts et sont classés dans la catégorie "Affichage".

Lorsque vous souhaitez modifier ce que Cozmo affiche sur son écran, **commencez par effacer ce qui peut se trouver sur l'écran** (yeux, texte), grâce au bloc "effacer tous les pixels".



Attention, pour que ce bloc fonctionne, il est nécessaire de **rajouter le bloc "afficher sur le visage de Cozmo"** à votre code.

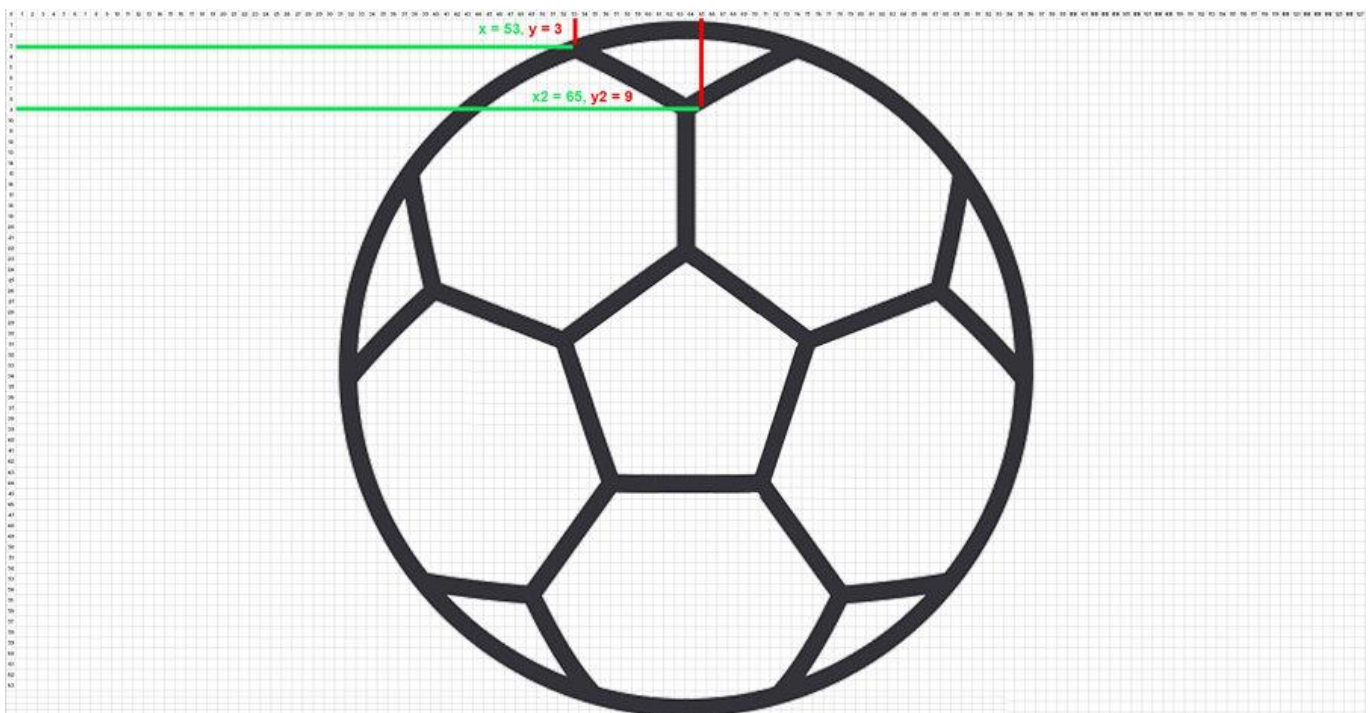
Déterminez ensuite la taille de votre dessin ou de votre texte grâce au bloc "définir l'échelle du texte sur [X] %". La taille par défaut est 100%, vous pouvez aller au-delà et en-deçà. **N'hésitez pas à tester différentes tailles !**

Pour dessiner le ballon de foot, nous allons afficher un cercle (bloc "dessiner un cercle en [X],[Y] de rayon [Z]") et rajouter les lignes qui forment les coutures du ballon. **Changez les valeurs X, Y et Z afin d'ajuster le dessin à l'écran.**

Passons au dessin des coutures... **Nous allons tracer 20 lignes** en indiquant pour chacune les coordonnées de son point de départ et de son point d'arrivée. Les **notions de repères et coordonnées dans un plan** sont utilisées ici (cours de 4ème).

L'écran de Cozmo mesure **128 pixels de largeur sur 64 pixels de hauteur**. Nous avons donc créé une grille numérotée de 128 carrés de large sur 64 carrés de hauteur, sur laquelle nous avons superposé le dessin vectorisé de notre ballon de foot. Il est maintenant simple d'obtenir les coordonnées X et Y pour chaque point, et de tracer les lignes avec le bloc "dessiner une ligne de [X],[Y] à [X2],[Y2]".

Vous pouvez télécharger le modèle en cliquant sur l'image ci-dessus.



C'est parti ! Commencez par vous débarrasser du bloc qui affiche le point d'interrogation et du bloc d'échelle du texte. **Laissez de côté le bloc d'affichage, on s'en servira plus tard !**

Choisissez le bloc “dessiner une ligne de [X],[Y] à [X2],[Y2]” dans la bibliothèque. Modifiez les coordonnées de la première ligne comme expliqué ci-dessus et **répéter l’opération 19 fois, en modifiant les coordonnées à chaque fois**. Ensuite, nous allons **ajouter le cercle** pour compléter notre ballon de foot.

Ajouter un bloc « *définir l’échelle du texte sur [X] %* » et ajustez la taille du cercle aux coutures et à l’écran (200%). **Pour finir, raccordez la fin du code au nouveau tronçon !**

Votre ballon de foot devrait s’afficher sur l’écran de Cozmo à présent.



3. Faire parler Cozmo : rajouter les dialogues dans le bout de code “Cozmo veut qu’on lui pose une question”

Nous allons maintenant nous occuper de la partie dialogue.

Dans le projet VOYANT MAGIQUE !, l’animation se déroule ainsi :

Cozmo affiche un point d’interrogation sur son écran, l’utilisateur pose sa question et le secoue, ce qui l’étourdit, puis Cozmo énonce sa réponse (et celle-ci apparaît sur l’écran).

Nous allons supprimer cette animation et la remplacer l’animation suivante :

Cozmo énonce les instructions, l’utilisateur pose sa question : “est-ce que l’équipe de France va battre l’Uruguay ?” et appuie sur le cube de cristal pour obtenir la réponse.

A partir de la 2ème question, le dialogue changera, car Cozmo n’aura pas besoin d’énoncer les instructions de nouveau. Il demandera simplement “Souhaites-tu connaître un autre pronostic ?”.

Le **bloc dialogue** se trouve dans la catégorie “Actions” de la bibliothèque. Rajouter le premier bloc d’instruction “Dire” et **changer le texte** avec le clavier de la tablette.

Déplacez le bloc “Animation” et remplacer l’animation “28 – étourdi” par “2 – vainqueur”.

- Texte du premier bloc : “Grâce à moi, remporte tous les paris sportifs !”
- Texte du second bloc : “Demande moi si l’équipe que tu soutiens va remporter son prochain match. Ensuite, touche mon cube de cristal pour connaître mon pronostic !”

Pour tester votre code, appuyez sur le drapeau vert. **Attention, pensez bien à raccorder tous les blocs d’une partie de code !** (par exemple tous les blocs de la partie “Cozmo veut qu’on lui pose une question”).

4. Modification de l’événement déclencheur

Dans cette partie, nous allons faire un peu de **programmation événementielle** en abordant les **conditions** (‘si’... ‘alors’...) !

Pour rappel, l'action sous condition ('si'... 'alors'...), est l'un des piliers de la programmation, il est très important de bien comprendre ce concept qui peut être résumé ainsi :

SI (proposition-à-vérifier est vraie ou fausse) ALORS :

...
on fait ça
...
SINON
...
on fait ça
...
FINI

Pour l'instant, l'événement qui déclenche la réponse de Cozmo est : **"si l'inclinaison de Cozmo est supérieure à 15° ou inférieure à -15°"**. Je souhaite changer cela. Mon nouvel événement déclencheur sera **"si l'utilisateur touche le cube 1"**, alors Cozmo déclenche l'animation réponse.

Supprimer les blocs opérateurs/capteurs "Inclinaison de Cozmo" mais conservez le bloc "attendre jusqu'à <>".

Dans la catégorie "Opérateurs", choisir le bloc « ([X] = [Y]) ». Ce bloc est notre condition :

- [X] sera remplacé par le bloc capteur "le cube [#] a-t-il été touché »
- [Y] sera remplacé par le texte "True"



Ce bloc se traduit par 'si' le cube 1 est touché, 'alors'...

Note : le cube 1 ressemble à un trombone, le cube 2 ressemble à une lampe ou à un cœur et le cube 3 ressemble à un bébé dans son siège.

5. Peaufinage des interactions homme-machine

Pour finir la partie "Cozmo veut qu'on lui pose une question", il reste deux choses à faire :

1. Déplacer l'avant dernier bloc d'instruction "déplacer la tête à [45]° à [150] °/s » en deuxième position, sous le bloc "quand message1 est reçu – hérité". De cette manière, Cozmo commencera son interaction avec l'utilisateur en le regardant dans les yeux !

2. Enfin, créer le cube de cristal de Cozmo ! (qui lui permet de prédire les résultats)

C'est très facile, il vous suffit d'aller dans la catégorie "Actions" et de sélectionner le bloc "sur le cube [#], mettre [tous les voyants/le voyant #] en [couleur]".

Choisissez le cube 1 (car déjà sélectionné dans notre bloc "condition" (attendre jusqu'à – le cube 1 a-t-il été touché = True). Pour la couleur, vous êtes libre ! Notre choix s'est porté sur le bleu clair, qui pourrait rappeler la teinte du verre d'une boule de cristal...



6. Modification des variables pour l'animation COZMO FOOT et nouveau format de conditions

Comme les conditions, les variables sont des éléments essentiels à tout programme. **Une variable est un emplacement virtuel qui va servir à stocker des informations qui seront utilisées plus tard.**

Dans le projet VOYANT MAGIQUE !, la variable "Fortune" a été créée. Elle correspond aux **différentes prédictions qui seront tirées aléatoirement**. En fonction du chiffre qui sortira (pour l'instant, de 1 à 10), Cozmo énonce une prédiction au hasard.

Le bloc "nombre aléatoire entre [X] et [Y]" permet de **renvoyer un nombre pris au hasard** entre [X] et [Y]. Dans le projet VOYANT MAGIQUE !, X = 1 et Y = 10.

Pour COZMO FOOT nous n'aurons besoin que de trois prédictions, je change donc mes valeurs pour X = 1 et Y = 3.

Je change également le texte de ces prédictions.

- Texte 1 : "Oui, tu peux compter là-dessus !"
- Texte 2 : "Hmmm... J'ai de sérieux doutes."
- Texte 3 : "Haha, ça n'arrivera jamais !"

Là aussi, vous pouvez tout à fait entrer les textes de votre choix.

Nous venons de créer l'animation suivante :

Cozmo choisit un chiffre au hasard, compris entre 1 et 3. A chaque chiffre est associée une action différente (affichage d'un certain texte).

Pour lancer telle ou telle actions, je fais de nouveau appel aux conditions, cette fois avec le bloc "si (condition) alors [A]".

7. Les fonctions : ajout d'une fonction supplémentaire

Qu'est-ce qu'une fonction ? C'est un ensemble d'instructions, un petit bout de code, qui sera regroupé sous une fonction. Si on prévoit d'**utiliser un bout de code plusieurs fois** dans un programme, on le range dans une fonction. Chaque fois que j'aurais besoin de d'ajouter ce bout de code dans mon programme, « **j'appellerai** » **la fonction à la place**. Comme ça, je n'aurai pas besoin de retaper mon bout de code à chaque fois.

La fonction présente deux gros avantages :

- Je n'ai **pas besoin de retaper mon bout de code** plusieurs fois
- **Mon code sera plus facile à lire** (une fonction qui tient sur une ligne peut remplacer un bout de code de plusieurs dizaine de lignes)

Dans Code Lab, pour créer une fonction, il suffit de placer le bloc d'instruction « *quand [message] est reçu* » au début de la suite d'instructions que vous souhaitez réutiliser.

A chaque fois que vous souhaitez « appeler » cette fonction dans votre programme, il vous suffira simplement de placer le bloc « *diffuser [message]* ».

Exemple :

Création de la fonction : « *quand [dialogue1] est reçu* »

A chaque fois que vous souhaitez « appeler » cette fonction dans votre programme, il vous suffira simplement de placer le bloc « *diffuser [dialogue1]* ».

Ici, nous allons créer une fonction pour le code qui gèrera les nouvelles questions de l'utilisateur (car nous ne voulons pas que Cozmo répète les instructions de COZMO FOOT ! à chaque fois, il dira juste « Veux-tu connaître un autre pronostic ? »).

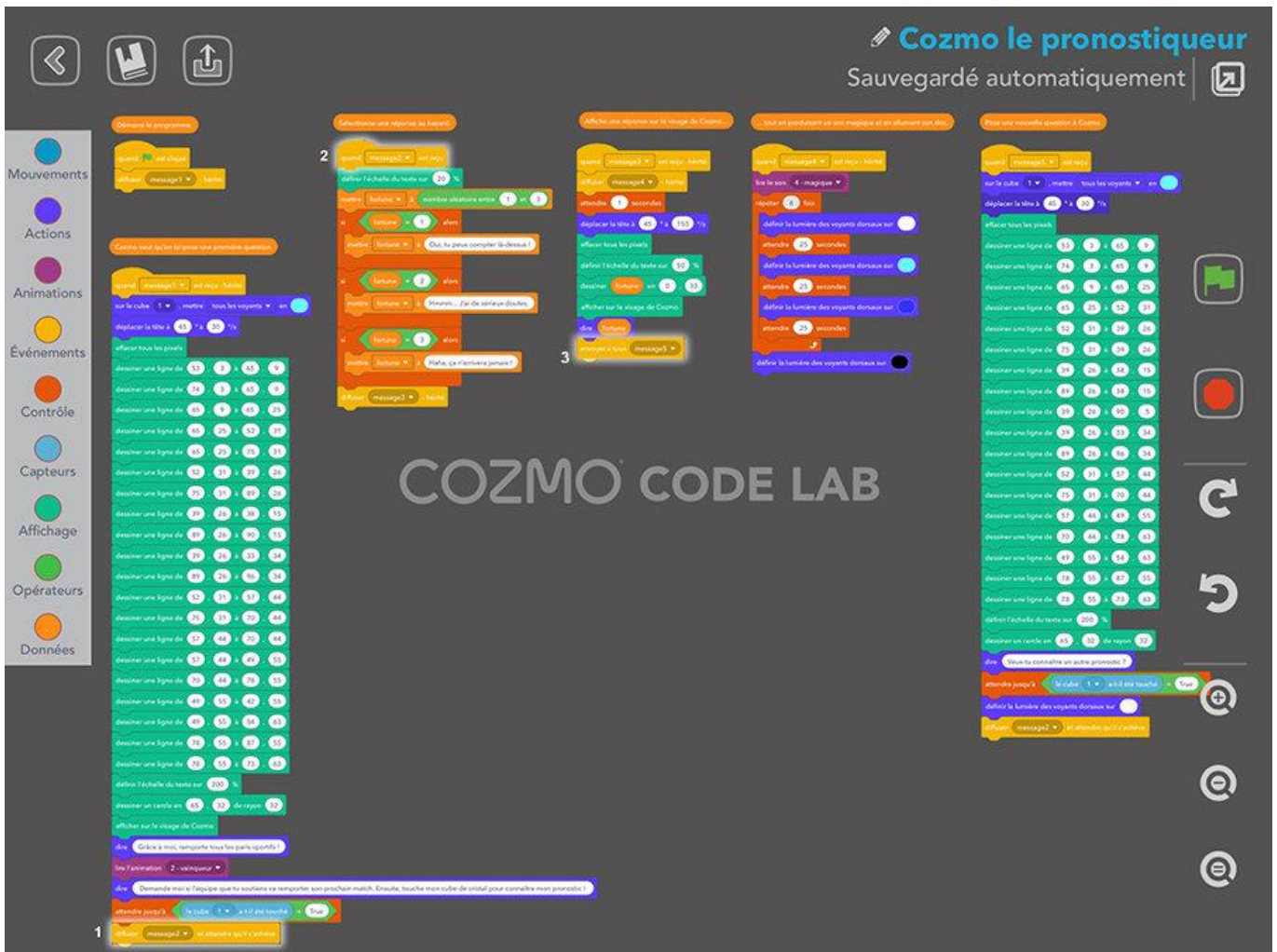
1. Commencez par nommer ce nouveau bout de code en ajoutant un commentaire (dans Code Lab, cela se passe dans « Données » > « Créer une nouvelle variable »). Ici, nous nous servons du bloc « Variable » comme d'une ligne de texte descriptive. Appelez ce nouveau bout de code « Pose une nouvelle question à Cozmo ».
2. Vous allez maintenant écrire le même code que pour la partie « Cozmo veut qu'on lui pose une question » (qui est la fonction « message1 ») à quelques différences près :
 1. Nommer votre fonction « message5 » (quand [message5] est reçu)
 2. Après « afficher sur le visage de Cozmo », le code n'est plus le même que pour la fonction « message1 ». Il vous faut remplacer les trois blocs « Dire », « Lire l'animation » et « Dire » par un seul bloc « Dire [Veux-tu connaître un autre pronostic ?] ».
 3. Le dernier bloc doit être « diffuser [message2] » car une fois notre animation achevée, le programme doit lancer la fonction « message2 », qui est le choix aléatoire d'une prédiction.

Note : dans cet exemple, la fonction « message5 » n'est utilisée qu'une seule fois, mais pourrait tout à fait imaginer que ce soit plusieurs fois, c'est tout l'intérêt d'une fonction !

Attention, vous avez peut-être remarqué que les fonctions déjà présentes dans le projet VOYANT MAGIQUE ! portent la mention « – hérité », mais que cette mention est absente des blocs de fonction que vous avez rajoutés. Cela est dû au fait que le projet VOYANT MAGIQUE ! a été créé avec la **version précédente de Code Lab**.

Dans cette version précédente, les messages ne pouvaient porter que les noms « message1 », « message2 », etc. Ces noms étaient sélectionnés dans une liste déroulante. Une mise à jour a été faite, qui permet de nommer les fonctions comme on le souhaite. Par contre, on ne peut pas utiliser des nouvelles fonctions et des anciennes fonctions dans un même bout de code, il y a une incompatibilité de version.

Pour éviter cela, il suffit juste de remplacer les blocs « – hérités » par de nouveaux blocs de fonction. Pour plus de facilité, nous vous montrons **ci-dessous**, quels sont les blocs à remplacer (et par quoi les remplacer).



Il ne vous reste plus qu'à remplacer le premier commentaire « Cozmo veut qu'on lui pose une question » par « Cozmo veut qu'on lui pose une première question » !

Vous venez de finir votre premier tutoriel pour Code Lab. Nous vous y avons présenté des **fondamentaux de la programmation** (conditions, variables, fonctions) et vous avons expliqué comment les utiliser dans Code Lab. **A présent, vous en savez suffisamment pour créer votre propre jeu ou activité !**