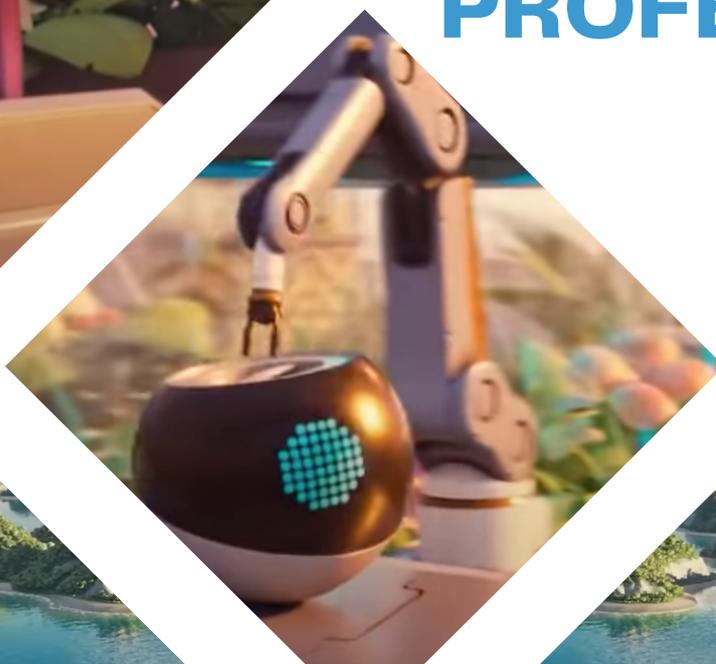
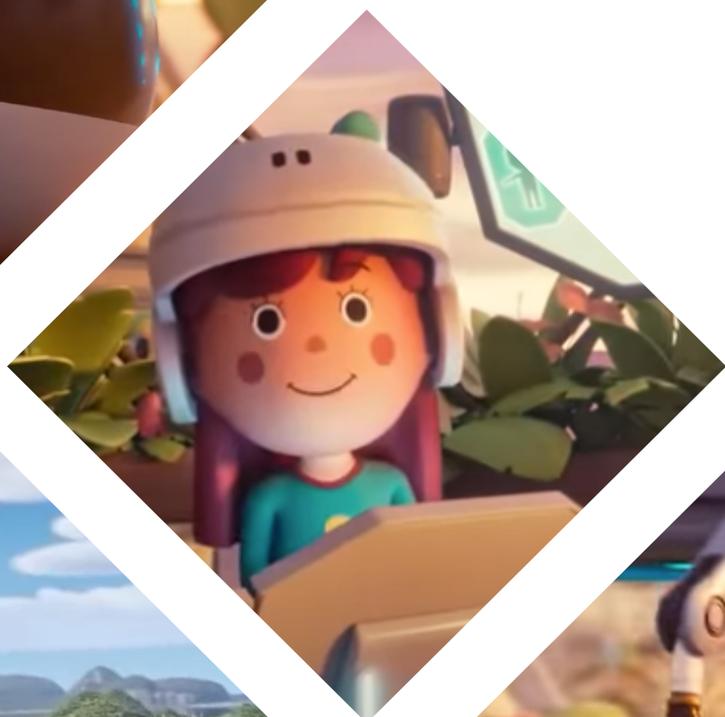


winky



LIVRET DU PROFESSEUR

Robot Winky



HEY WINKY



Grâce à son design moderne, Winky est un sympathique robot mais il n'en est pas moins un vrai robot et un véritable concentré de technologie. Il est équipé d'une tête rotative, de 4 boutons, d'un haut-parleur, de plusieurs LED pour les yeux, d'un gyroscope et d'un détecteur de mouvement et de distance.

vitta
science

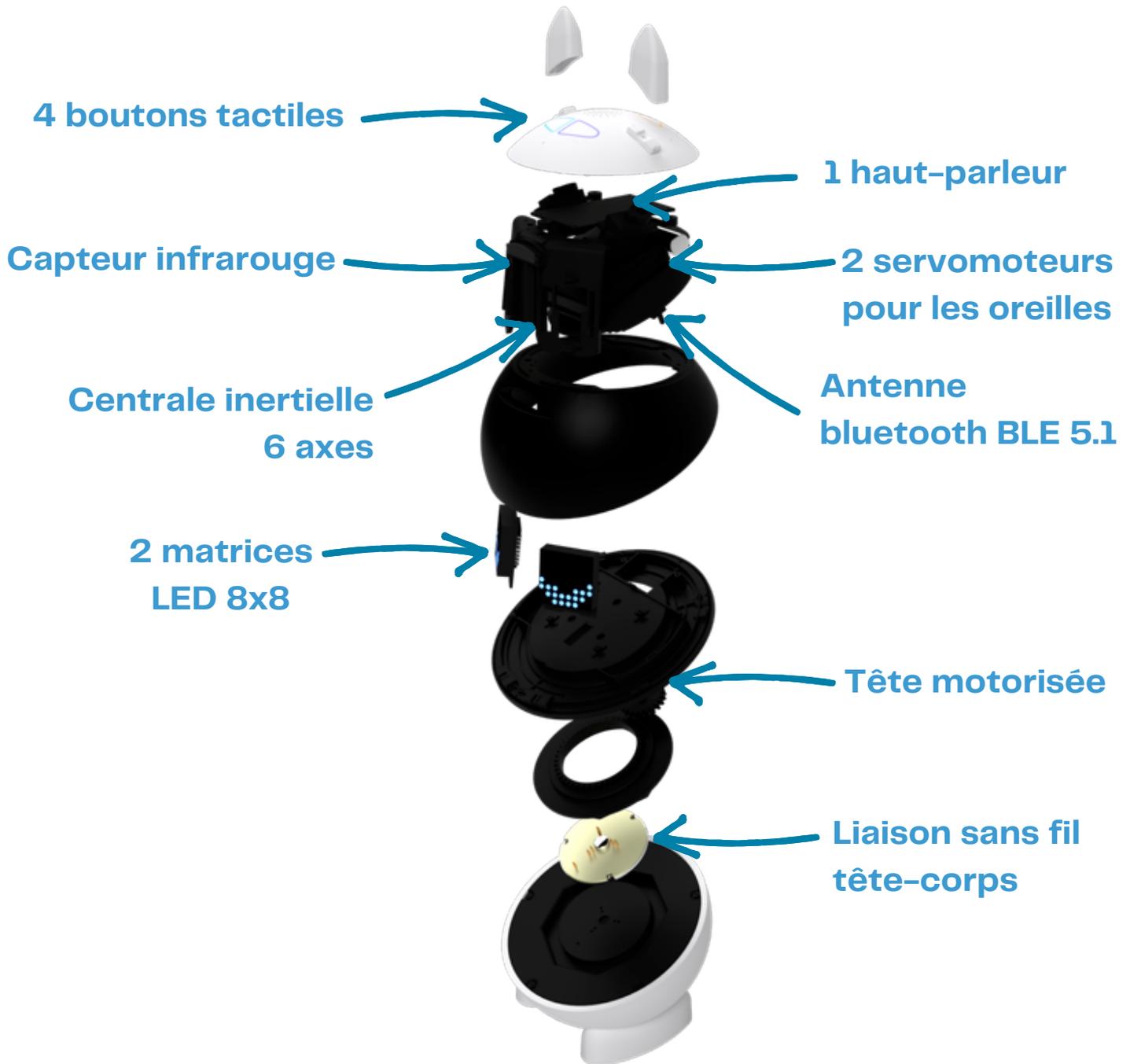
A l'aide de l'interface développée par Vittascience, vous allez amener vos élèves à réaliser plusieurs missions. La programmation sera facilitée grâce à l'usage des blocs où chaque instruction correspond à une action.

Auteur : Damien Vallot

LES COMPOSANTS DU ROBOT

Le robot Winky est équipé de nombreux éléments : des capteurs et des actionneurs permettant une interaction avec son environnement. Chaque élément peut être programmé par les élèves.

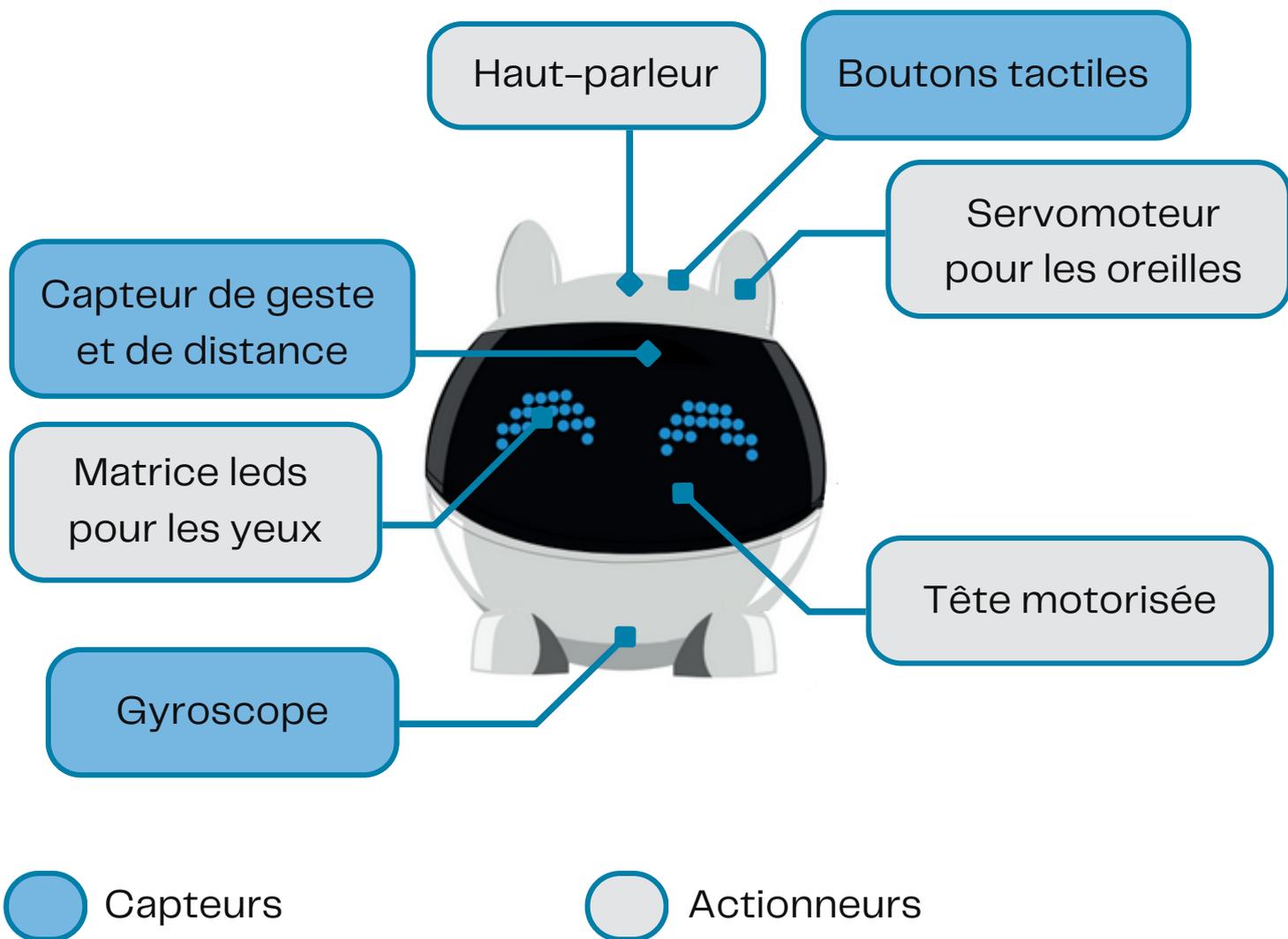
L'éclaté ci-dessous permet de repérer les composants du robot.



Les capteurs intégrés (boutons, gyroscope, capteur de mouvement et de distance) sont les dispositifs permettant de capter un phénomène physique et de le restituer sous forme d'un signal électrique qui sera analysé par la carte électronique interne au robot.

Les actionneurs (haut-parleur, LED, servomoteurs) sont quant à eux les dispositifs qui reçoivent un apport d'énergie et le convertissent en mouvement ou en signal lumineux ou sonore.

Retrouvez à la fin du livret la fiche élève avec les éléments à compléter. Invitez-les à colorier les légendes qui représentent des capteurs et des actionneurs.



Le robot Winky intègre d'autres éléments comme la batterie pour le faire fonctionner et un port USB de recharge mais aussi une antenne bluetooth pour la connexion à l'ordinateur.

Pour mettre en veille et réveiller Winky, il suffit d'appuyer et de maintenir le doigt sur le bouton  qui se trouve sur la tête, jusqu'à ce que le robot émette un son et que les yeux s'allument. Cela prend quelques secondes.

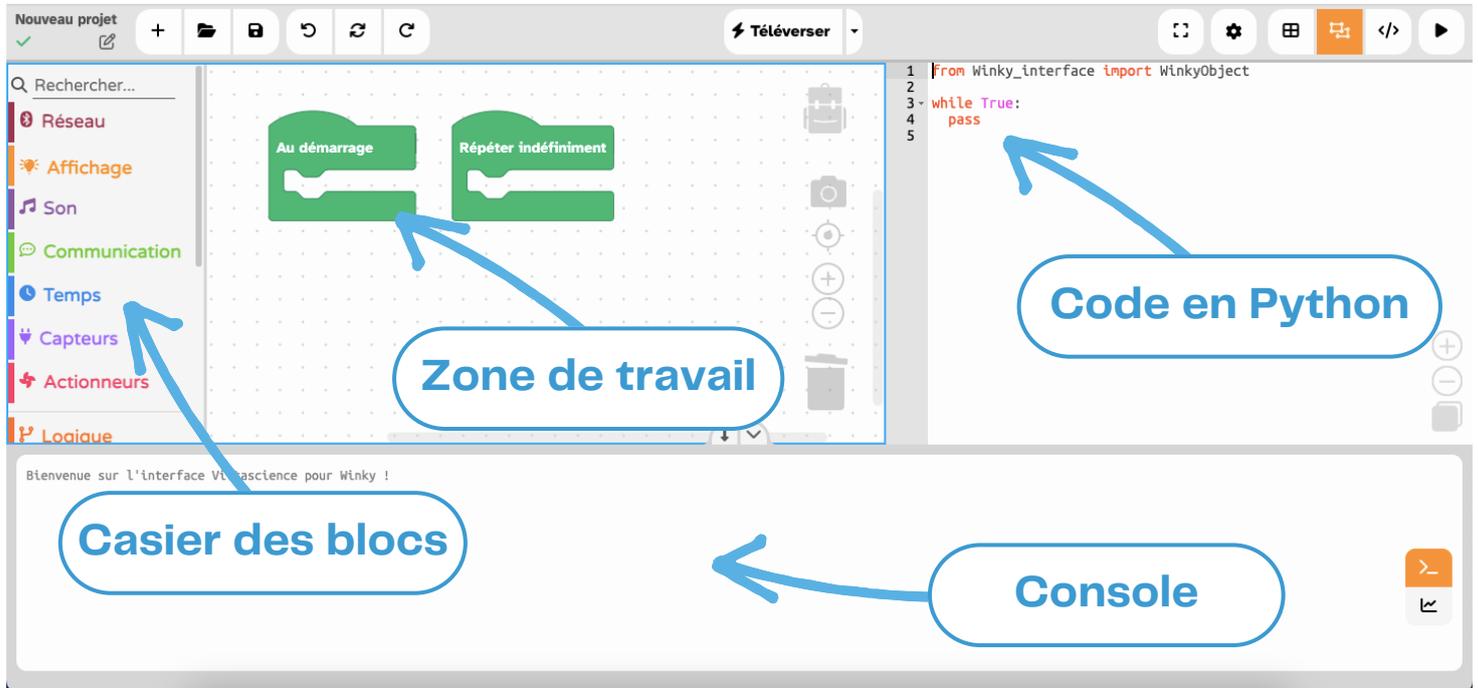
Pour l'éteindre complètement, utiliser le bouton on/off dans le corps du robot.



L'INTERFACE DE PROGRAMMATION

La programmation du robot Winky s'effectue à l'aide de l'interface Vittascience.

Pour y accéder, il faut se rendre sur : <https://fr.vittascience.com/winky/>



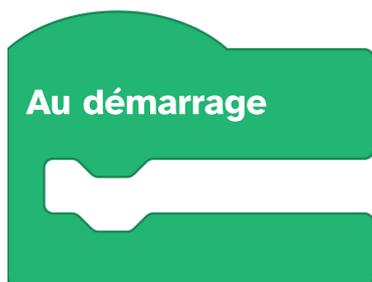
L'interface Vittascience

Cette interface gratuite et accessible sans installation depuis un navigateur internet permet de programmer le robot.

Elle se compose de différentes parties :

- au centre, la zone de travail permet la création du programme. Elle accueille les différents blocs placés par l'utilisateur pour construire le programme.
- le casier des blocs, situé à gauche, regroupe l'ensemble des instructions permettant de créer un programme.
- la zone située à droite, correspond au programme créé qui s'affiche sous la forme d'un texte, on parle de code. Le langage utilisé est spécifique pour chaque robot, ici c'est du Python.
- la console, en bas de l'interface, permet à l'utilisateur de connaître l'état des tâches envoyées au robot connecté à l'ordinateur.

Les blocs "Au démarrage" et "Répéter indéfiniment" sont les éléments clés du fonctionnement du robot. Ils sont présents dès l'ouverture de l'interface Vittascience.

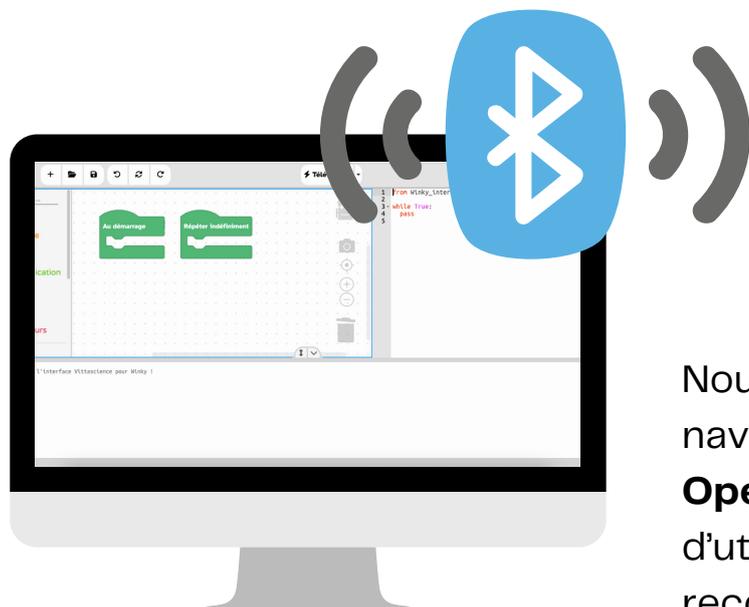
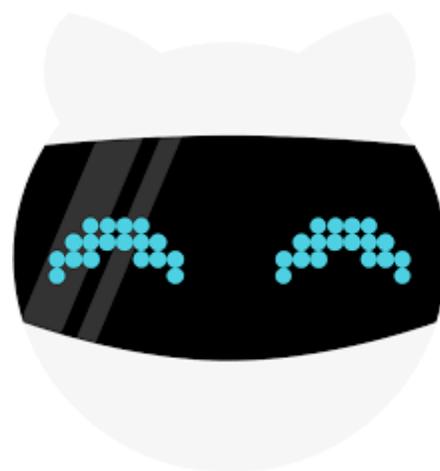


Les instructions placées dans le bloc "Au démarrage", seront exécutées dès le lancement du programme sur le robot.

Alors que les instructions du programme placées dans le bloc "Répéter indéfiniment", seront exécutées après le démarrage et en continue jusqu'à ce que le robot soit arrêté.

ENVOYER UN PROGRAMME AU ROBOT

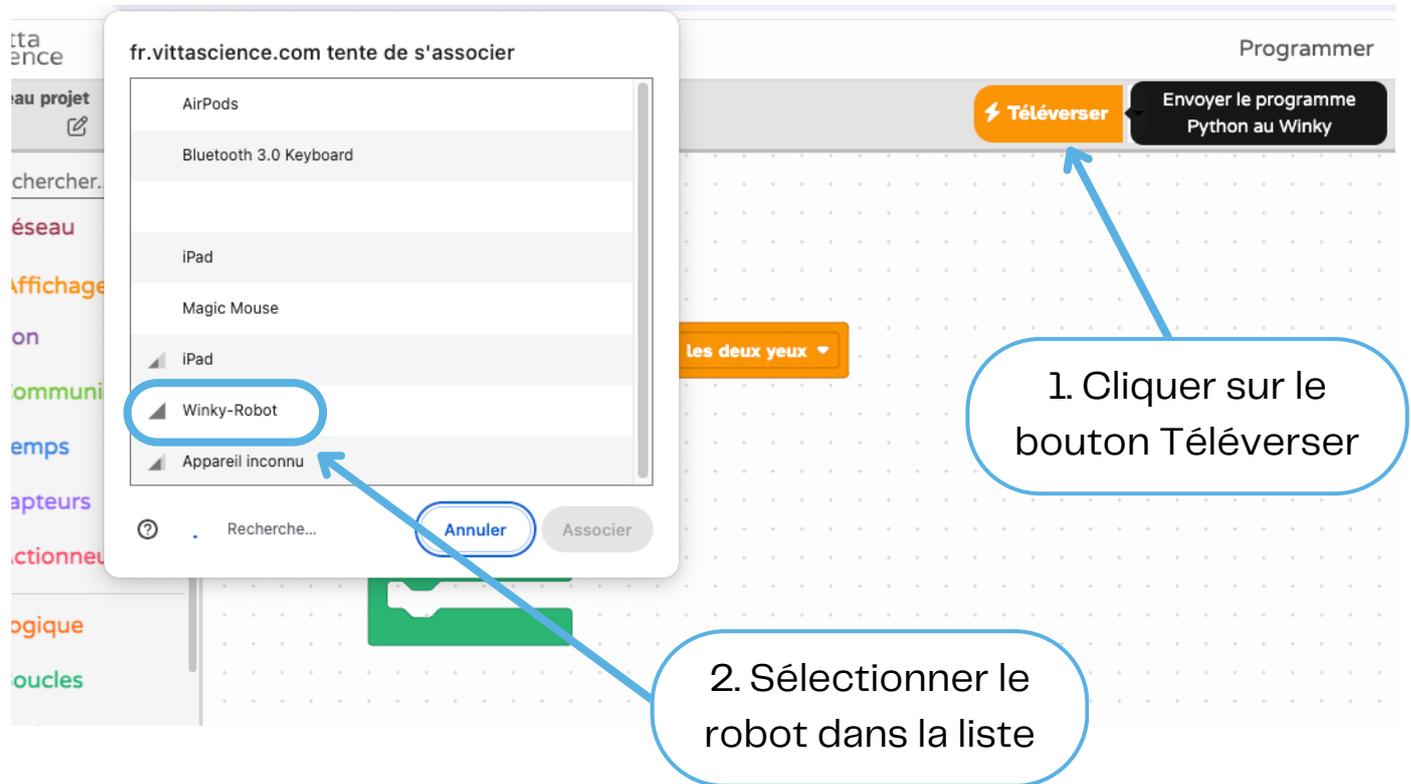
Le robot Winky va être connecté à l'ordinateur à l'aide du Bluetooth pour recevoir les différents programmes.



Nous vous conseillons de choisir un navigateur comme **Google Chrome**, **Opera** ou **Microsoft Edge** afin d'utiliser facilement le robot car nous recourons à la technologie : Web Bluetooth API

Pour connecter le robot à l'ordinateur, il faut cliquer sur le bouton 'Téléverser'. Une fenêtre s'ouvre dans le navigateur, affichant la liste des éléments détectés qui utilisent une connexion Bluetooth. Il suffit alors de sélectionner le robot.

Une fois cette étape effectuée, le robot affiche le logo Bluetooth sur la matrice LED des yeux : 



fr.vittascience.com tente de s'associer

- AirPods
- Bluetooth 3.0 Keyboard
- iPad
- Magic Mouse
- iPad
- Winky-Robot**
- Appareil inconnu

Recherche...

Programmer

Envoyer le programme Python au Winky

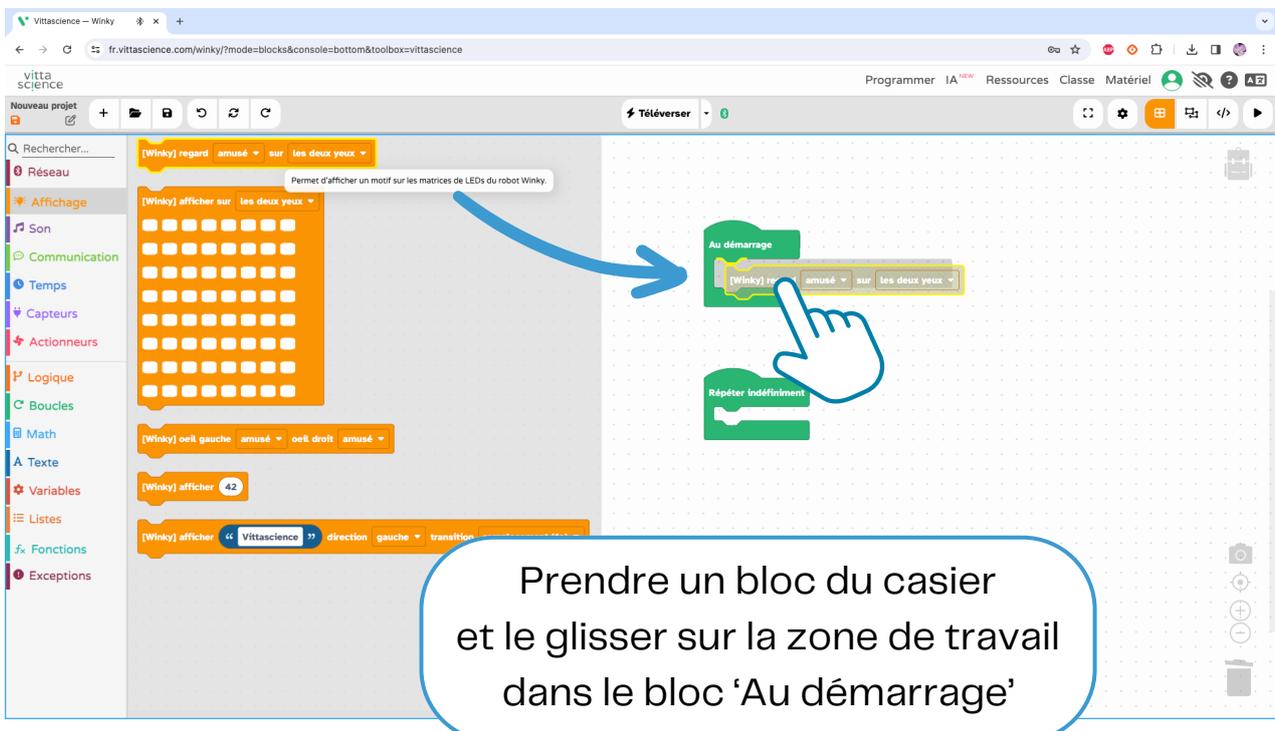
les deux yeux ▾

1. Cliquer sur le bouton Téléverser

2. Sélectionner le robot dans la liste

Se connecter au robot en Bluetooth

Passons maintenant à la création des programmes !



fr.vittascience.com/winky?mode=blocks&console=bottom&toolbox=vittascience

Programmer IA^{NEW} Ressources Classe Matériel

Nouveau projet +

Rechercher...

[Winky] regard amusé sur les deux yeux

Permet d'afficher un motif sur les matrices de LEDs du robot Winky.

[Winky] afficher sur les deux yeux

Au démarrage

[Winky] regard amusé sur les deux yeux

Répéter indéfiniment

[Winky] œil gauche amusé œil droit amusé

[Winky] afficher 42

[Winky] afficher "Vittascience" direction gauche transition

Prendre un bloc du casier et le glisser sur la zone de travail dans le bloc 'Au démarrage'

MISSIONS À RÉALISER

L'élève va devoir jouer le rôle d'un opérateur pour le contrôle qualité du robot Winky qui vient de sortir de l'usine. Il doit réaliser les différentes missions pour valider son bon fonctionnement.

• Mission n°1 : le test des hauts-parleurs

Objectif : émettre un son au démarrage.

Algorithme :

Au démarrage
Lancer un son 'amusé'

Programmation :



Plusieurs sons préenregistrés sont disponibles. Il faut cliquer sur la flèche du menu déroulant pour afficher la liste.

Une fois le bloc placé, cliquer sur le bouton 'Téléverser' et écouter le robot. Recommencer en testant les différents sons.

• Mission n°2 : le test des yeux

Objectif : animer les yeux, faire afficher des figures aléatoires, utiliser une pause pour créer une animation.

Algorithme :

Au démarrage
Afficher un regard 'amusé' sur 'les deux yeux'

Programmation :



Il est possible de sélectionner l'oeil droit, l'oeil gauche ou les deux yeux en cliquant sur la flèche de sélection.

Une fois le bloc placé, cliquer sur le bouton 'Téléverser' et observer les yeux sur le robot. Recommencer en testant les autres regards préenregistrés.

Utiliser un bloc de temporisation :

Algorithme :

Au démarrage

Effacer les deux yeux

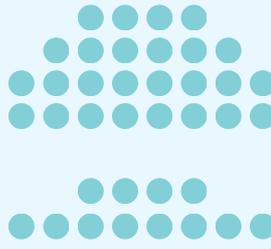
Répéter indéfiniment

Afficher sur l'oeil gauche :

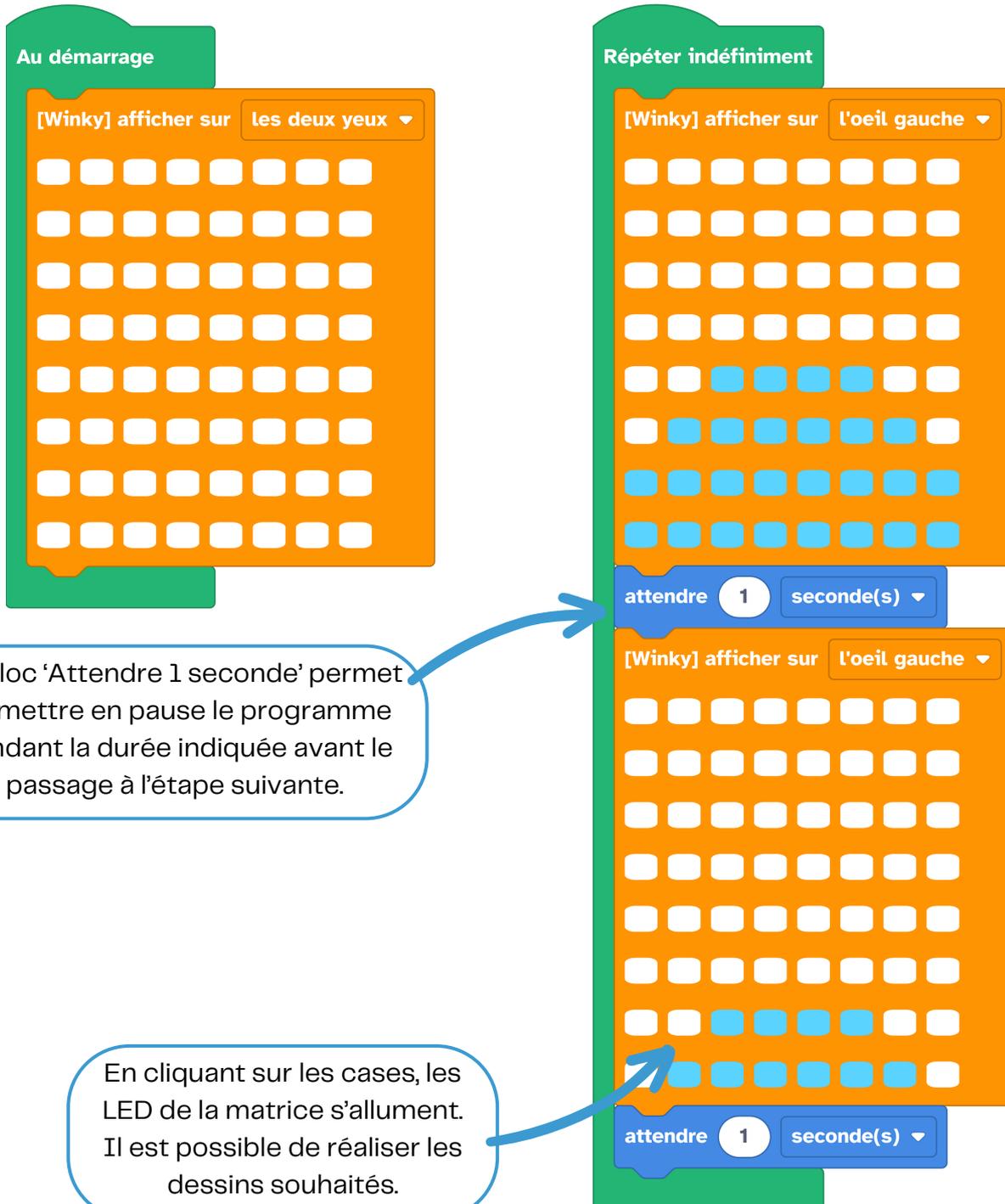
Attendre 1 seconde

Afficher sur l'oeil gauche :

Attendre 1 seconde



Programmation :



Le bloc 'Attendre 1 seconde' permet de mettre en pause le programme pendant la durée indiquée avant le passage à l'étape suivante.

En cliquant sur les cases, les LED de la matrice s'allument. Il est possible de réaliser les dessins souhaités.

• Mission n°3 : le contrôle des oreilles

Objectif : faire bouger les oreilles du robot avec les blocs préenregistrés puis avec les degrés d'angle.

Algorithme :

Au démarrage

Mettre les deux oreilles en position dressée

Programmation :



Trois blocs sont disponibles :
oreilles baissées, oreilles dressées
ou oreilles en arrière

Utiliser les degrés d'angle :

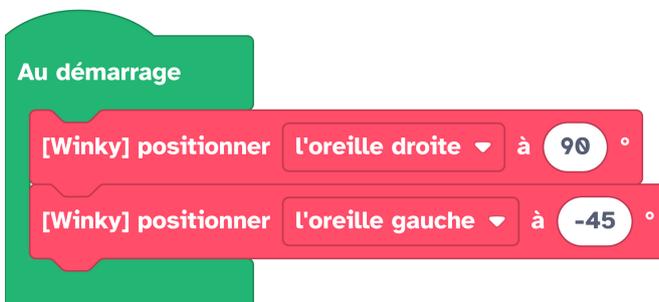
Algorithme :

Au démarrage

Positionner l'oreille droite à 90°

Positionner l'oreille gauche à -45°

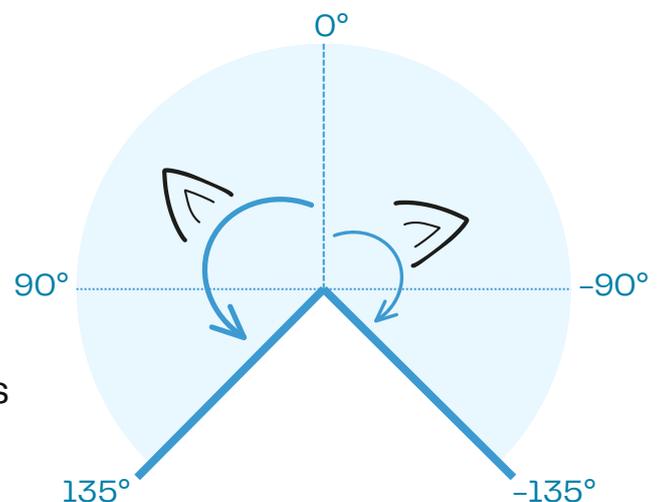
Programmation :



Les servomoteurs des oreilles ont une amplitude de rotation de 135° à -135° .

Le sens positif de rotation est antihoraire tandis que le sens négatif de rotation correspond à une rotation dans le sens des aiguilles d'une montre.

Si vous faites tourner les deux oreilles en même temps, elles tournent donc dans la même direction.



• Mission n°4 : le contrôle de la tête

Objectif : faire tourner la tête du robot puis définir un degré de rotation.

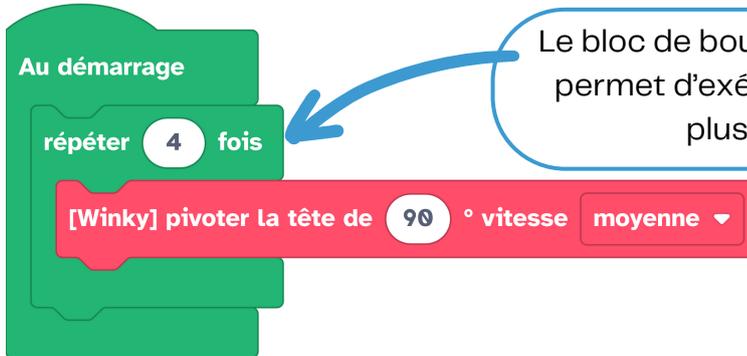
Algorithme :

Au démarrage

Répéter 4 fois

Pivoter la tête de 90° vitesse moyenne

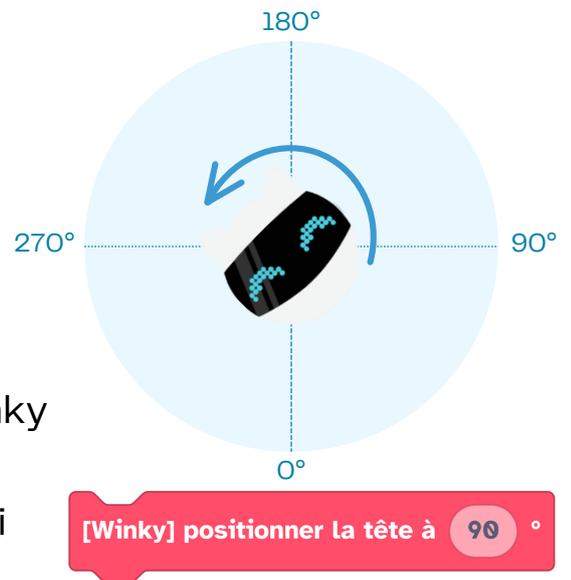
Programmation :



Le moteur de la tête peut tourner de manière infinie grâce au système de liaison sans fil tête corps.

Le 0° degré d'angle correspond à la tête de Winky face à l'utilisateur.

La rotation s'effectue dans le sens antihoraire si les degrés augmentent de manière positive.



• Mission n°5 : le test des 4 boutons

Objectif : programmer les 4 boutons pour réaliser une action différente.

Algorithme :

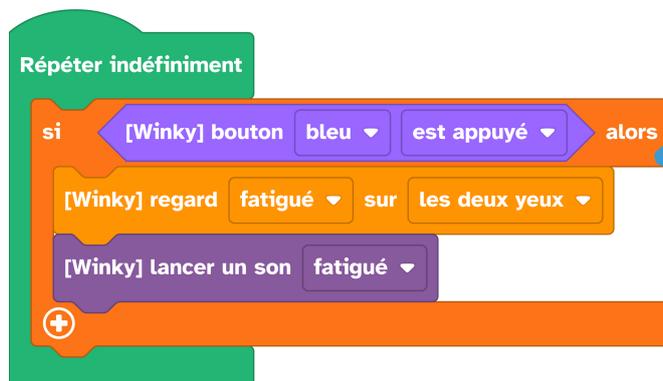
Répéter indéfiniment

Si bouton bleu est appuyé **alors**

Winky regard 'fatigué' sur 'les deux yeux'

Lancer un son 'fatigué'

Programmation :



Le bloc "si... alors" permet de tester si la condition est vraie et lance la réalisation des instructions

Tester une nouvelle instruction conditionnelle :

Algorithme :

Répéter indéfiniment

Si bouton 'bleu' est appuyé **alors**

Winky regard 'fatigué' sur 'les deux yeux'

Lancer un son 'fatigué'

Sinon si bouton 'violet' est appuyé **alors**

Winky regard 'endormi' sur 'les deux yeux'

Lancer un son 'endormi'

Programmation :



• Mission n°6 : la détection des obstacles

Objectif : utiliser le capteur de distance et créer une alerte visuelle.

Algorithme :

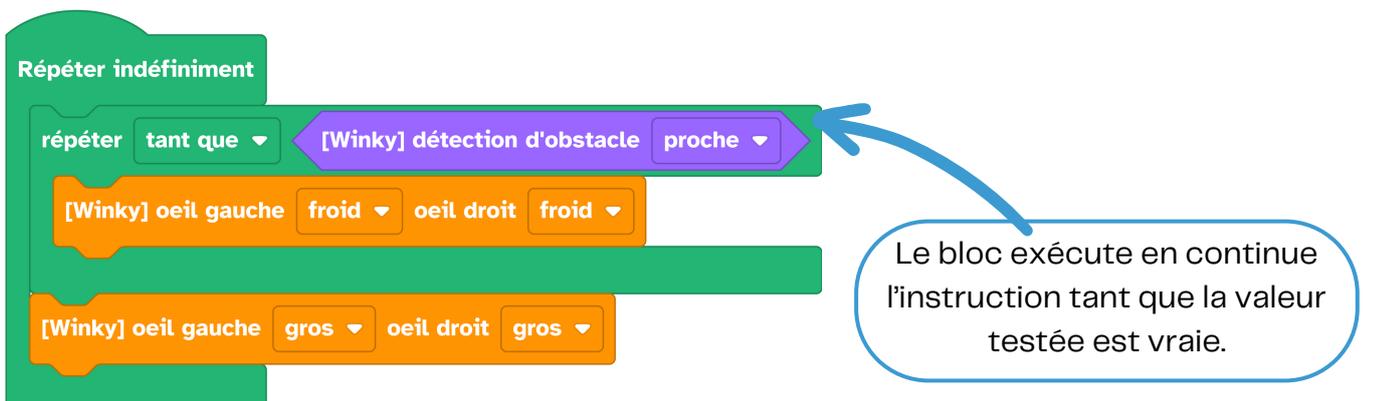
Répéter indéfiniment

Répéter 'tant que' détection d'obstacle 'proche'

Winky oeil gauche 'froid', oeil droit 'froid'

Winky oeil gauche 'gros', oeil droit 'gros'

Programmation :



• Mission n°7 : la détection du mouvement

Objectif : programmer une animation suivant le mouvement détecté.

Algorithme :

Répéter indéfiniment

Si détection de mouvement 'vers la droite' **alors**

afficher sur 'les deux yeux' 

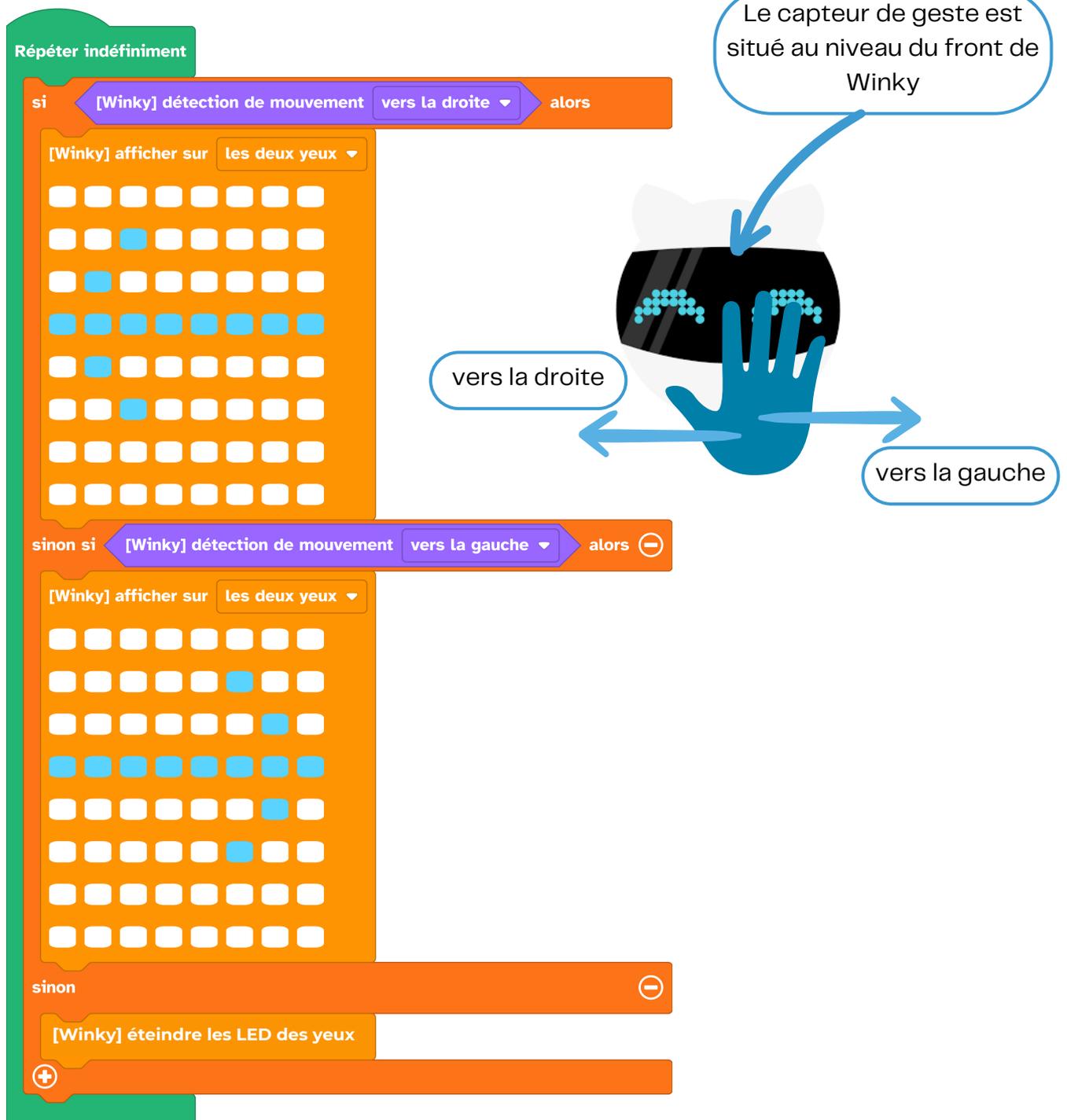
Sinon si détection de mouvement 'vers la gauche' **alors**

afficher sur 'les deux yeux' 

Sinon

Eteindre les LED des yeux

Programmation :



Le capteur de geste est situé au niveau du front de Winky

vers la droite

vers la gauche

Répéter indéfiniment

si [Winky] détection de mouvement vers la droite alors

[Winky] afficher sur les deux yeux

sinon si [Winky] détection de mouvement vers la gauche alors

[Winky] afficher sur les deux yeux

sinon

[Winky] éteindre les LED des yeux

• Mission n°8 : le test du capteur gyroscopique

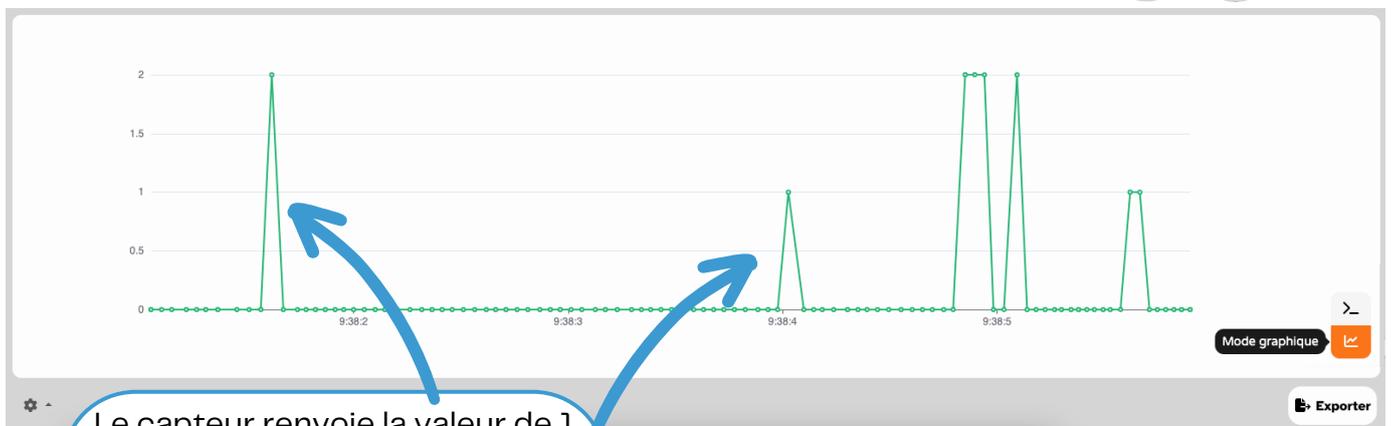
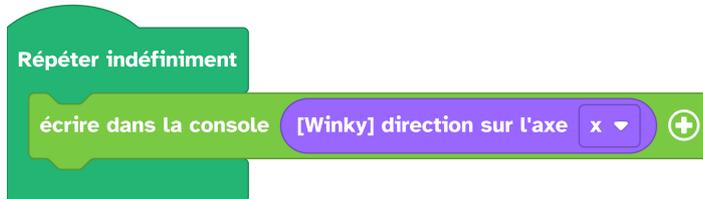
Objectif : écrire dans la console de l'ordinateur les valeurs du capteur, émettre un son d'avertissement lorsque le robot est renversé.

Algorithme :

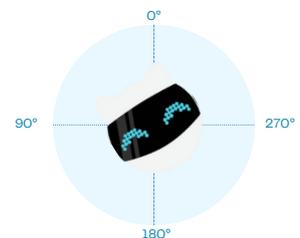
Répéter indéfiniment

Ecrire dans la console 'direction sur l'axe x'

Programmation :



Le capteur renvoie la valeur de 1 si le robot est basculé à droite et 2 s'il est basculé à gauche.



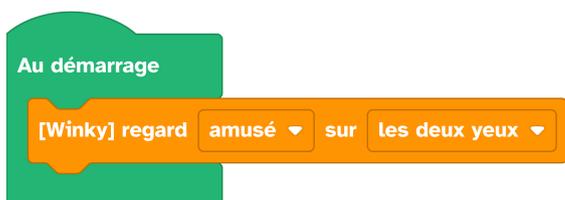
Utiliser les degrés d'angle autour d'un axe :

Algorithme :

Répéter indéfiniment

si angle autour de l'axe x $\geq 20^\circ$ et angle autour de l'axe x $\leq 340^\circ$ alors exécuter la fonction 'alerte'

Programmation :



• Mission n°9 : Je me présente : Winky le robot

Objectif : utilisation des listes comme mémoire du robot.

Algorithme :

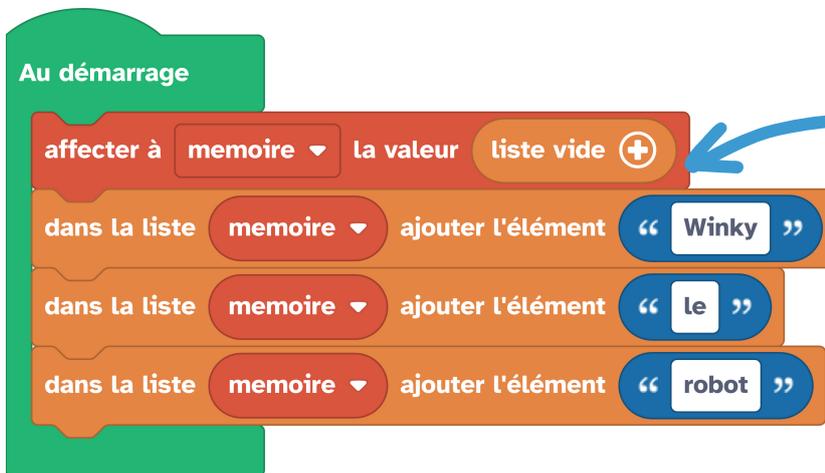
Au démarrage

 affecter à une variable 'memoire' la valeur liste vide
 dans la liste 'memoire' ajouter l'élément texte "Winky"
 dans la liste 'memoire' ajouter l'élément texte "le"
 dans la liste 'memoire' ajouter l'élément texte "robot"

Répéter indéfiniment

 pour chaque élément 'mot' dans la liste 'memoire'
 afficher 'mot' direction 'gauche'

Programmation :



Ce bloc permet d'initialiser une variable de type 'liste'.



Le bloc permet d'itérer sur les éléments de la liste.

• Mission n°10 : le code du robot, un programme à compléter

Objectif : utilisation des listes et des boutons pour créer un code à deviner.

Programmation à compléter :

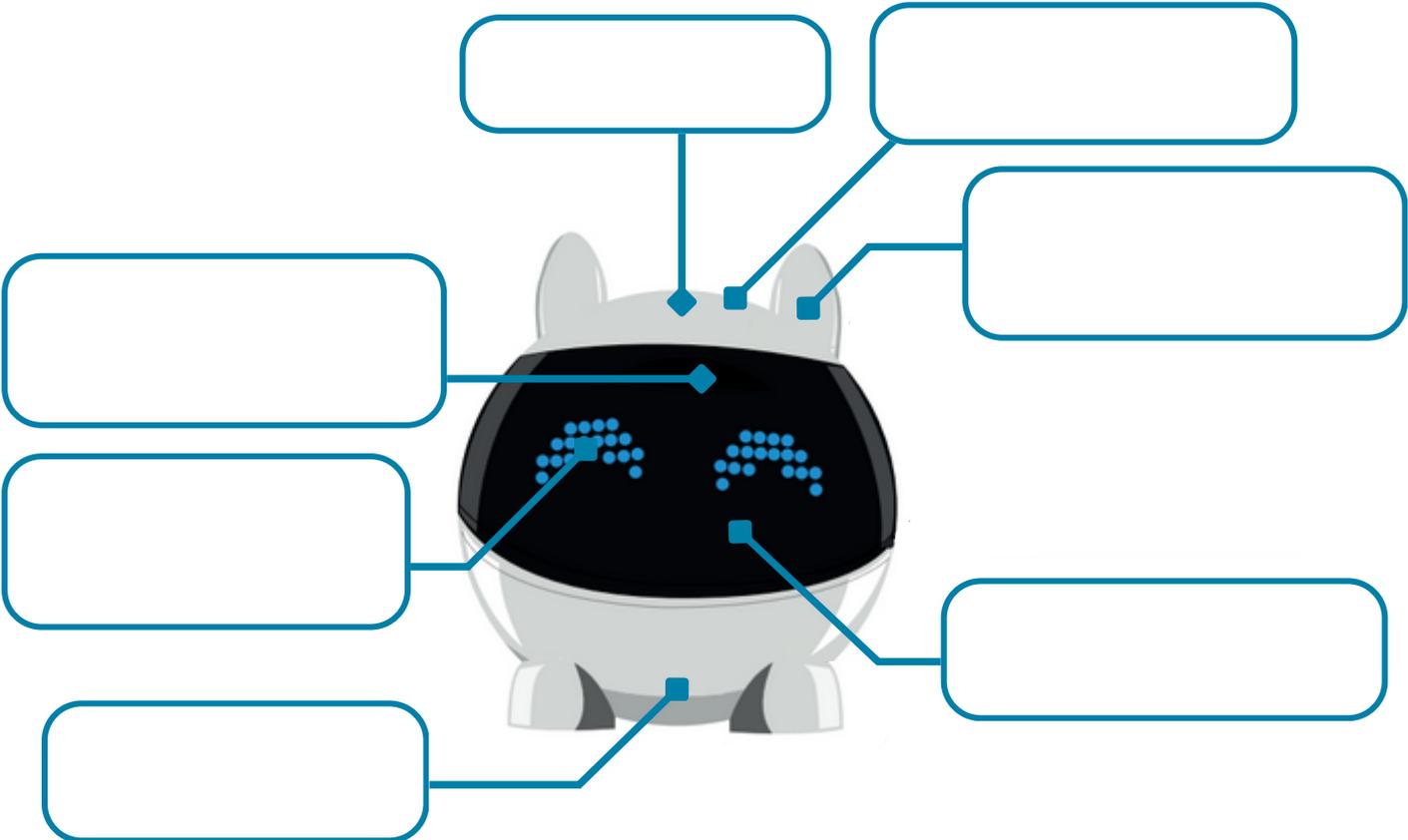
The image displays a Scratch script for a password guessing game, divided into three main sections with annotations:

- Au démarrage (Start):** A list is initialized with the colors "bleu", "rouge", "jaune", and "violet".
- verification_code (verify_code):** A function that takes a "code" and an "essai" (attempt) as input. It loops through the length of the "code" to compare each character with the corresponding element in the "essai" list. If all characters match, it returns "vrai" (true); otherwise, it returns "faux" (false).
- saisir_mot_de_passe (guess_password):** A function that repeatedly prompts the user to enter a color. It uses a loop of 4 iterations. For each iteration, it checks if a specific color button (bleu, violet, rouge, jaune) is pressed. If pressed, the color is added to a list. After 4 iterations, the list is returned.

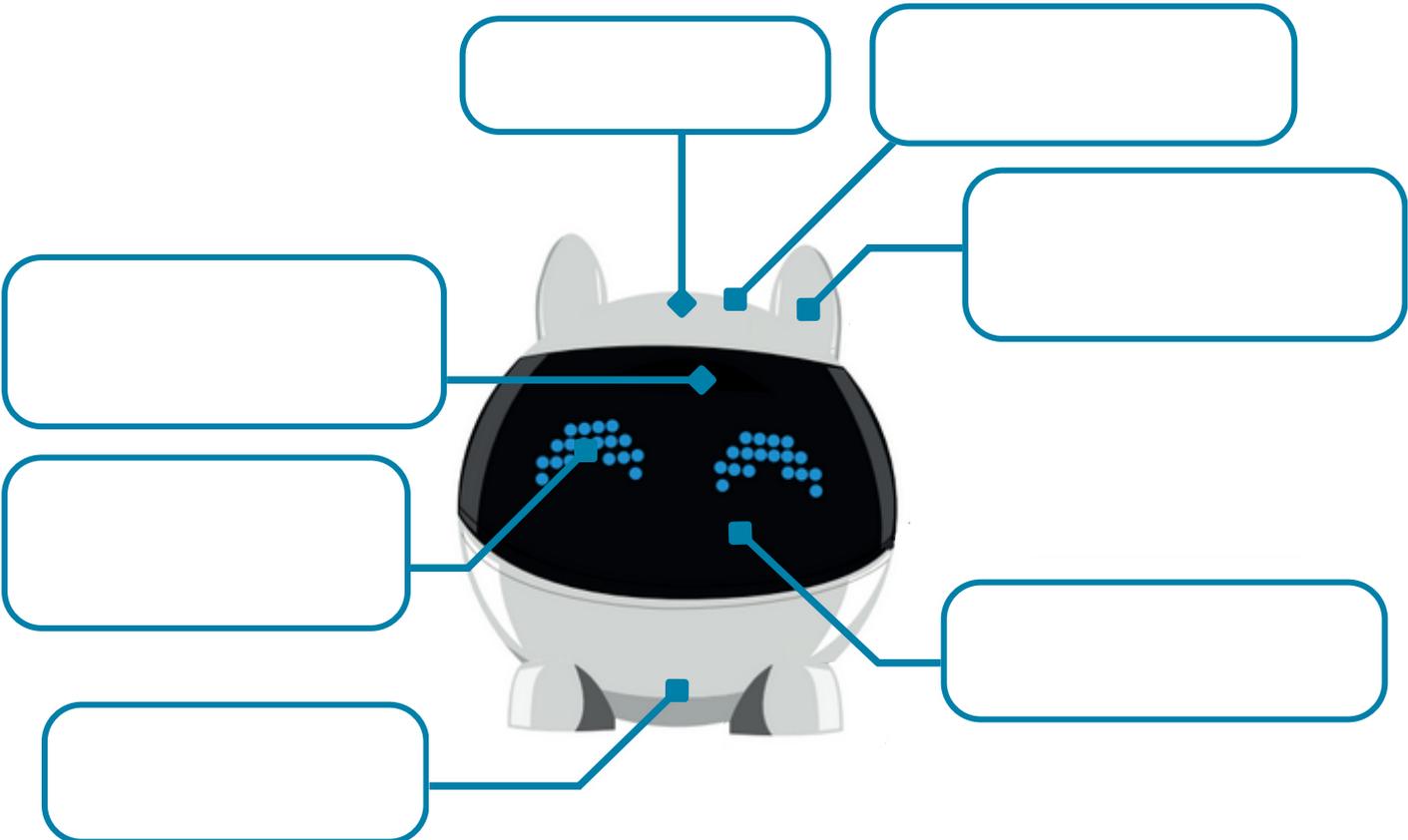
Annotations (callouts) highlight completion points:

- Éléments à compléter = séquence couleur à retrouver**: Points to the "ajouter l'élément" blocks in the "Au démarrage" section.
- Éléments à compléter = son à émettre**: Points to the "[Winky] lancer un son" blocks in the "verification_code" section.
- Éléments à compléter = variables à définir**: Points to the "ajouter l'élément" blocks in the "saisir_mot_de_passe" section.

LES COMPOSANTS DU ROBOT



LES COMPOSANTS DU ROBOT





Tour Montparnasse
33, avenue du Maine
75015 PARIS
contact@mainbot.me