

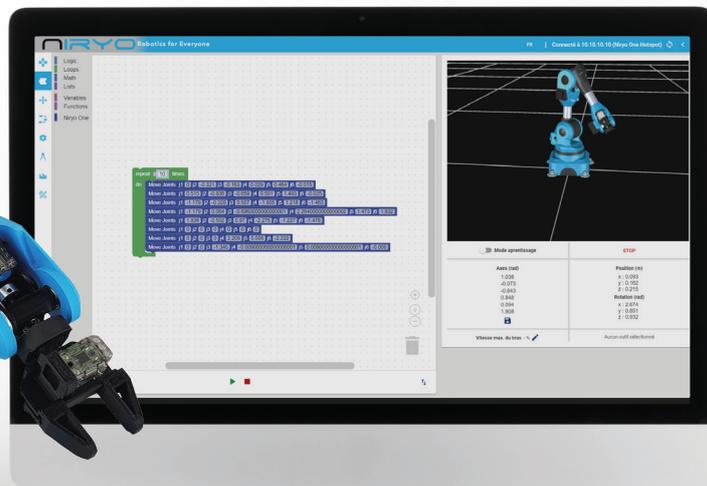
# NIRYO

HUMAN - MOTION - ROBOT

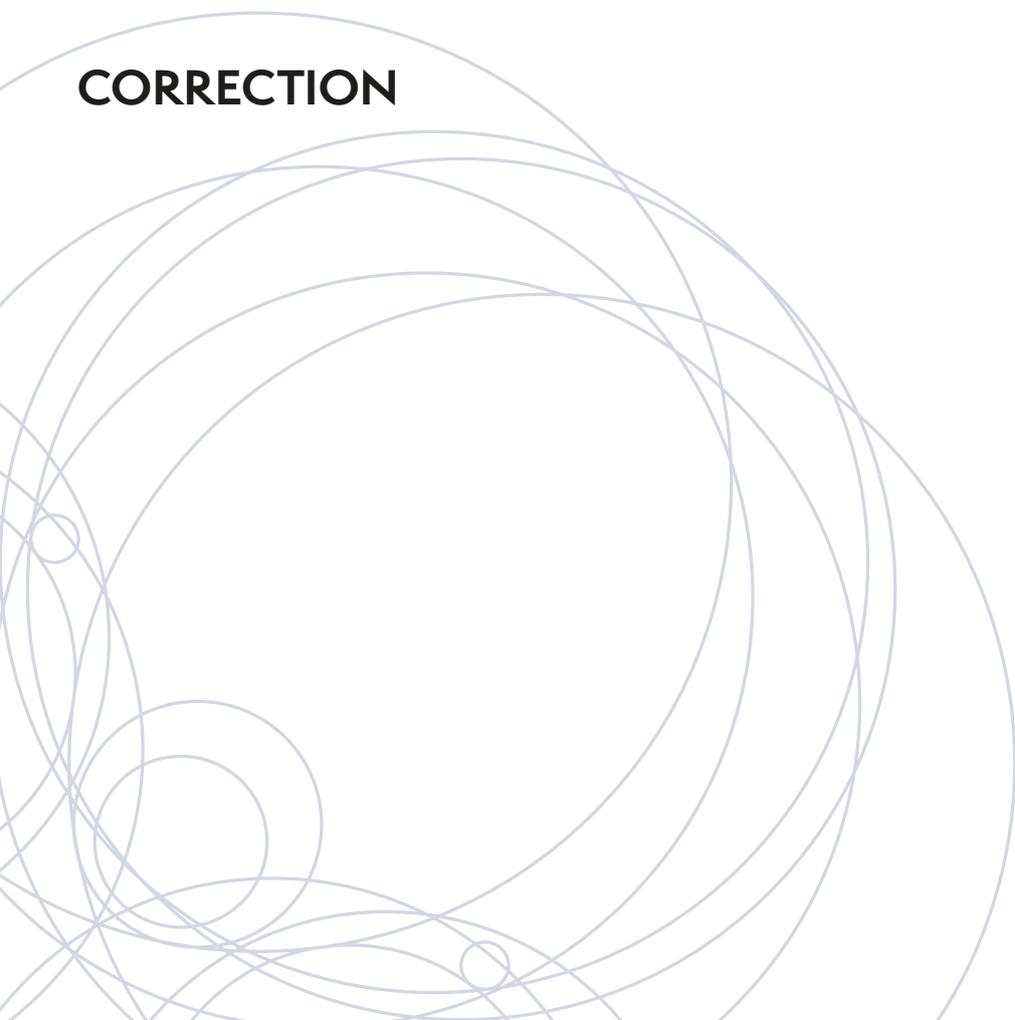
# BLOCKLY

# 1

Programmer le Niryo One avec  Blockly



<b>OBJECTIFS</b>	<b>3</b>
<b>PRÉREQUIS</b>	<b>3</b>
<b>CE DONT VOUS AUREZ BESOIN</b>	<b>3</b>
<b>PROGRAMMER LE NIRYO ONE AVEC LE MODE APPRENTISSAGE</b>	<b>3</b>
PRÉSENTATION DU MODE APPRENTISSAGE	3
EXERCICE	4
<b>PROGRAMMER LE NIRYO ONE AVEC BLOCKLY</b>	<b>5</b>
PRÉSENTATION DE BLOCKLY	5
L'ENVIRONNEMENT DE PROGRAMMATION	5
LES BLOCS DE PROGRAMMATION	7
EXERCICE	8
<b>ANNEXE</b>	<b>11</b>
<b>CORRECTION</b>	<b>16</b>



# OBJECTIFS

- Se familiariser avec les notions de base de programmation robotique
- Commander le robot en utilisant Niryo One Studio et Blockly
- Être capable de créer des algorithmes pour commander le robot

# PRÉREQUIS

- Starter guide

# CE DONT VOUS AUREZ BESOIN

- Ordinateur équipé du wifi
- Niryo One Studio (interface de programmation graphique du Niryo One).
- Un Niryo One

# PROGRAMMER LE NIRYO ONE AVEC LE MODE APPRENTISSAGE

## PRÉSENTATION DU MODE APPRENTISSAGE

Le « **mode apprentissage** » (ou « **learning mode** ») est une méthode de programmation avec laquelle l'opérateur humain **place manuellement le robot dans une position** puis l'enregistre afin que le robot puisse l'exécuter par la suite.

Il est possible de programmer le robot Niryo One en utilisant cette méthode de programmation. Pour activer ou désactiver ce mode, cliquer sur le bouton « **Mode apprentissage** » sur l'interface de Niryo One Studio.

## EXERCICE

- Activer le mode d'apprentissage du robot
- Bouger le robot avec la main vers une position de votre choix
- Maintenir le robot dans cette position et cliquer sur « **ACTUALISER LES VALEURS** » avant de relâcher le robot.



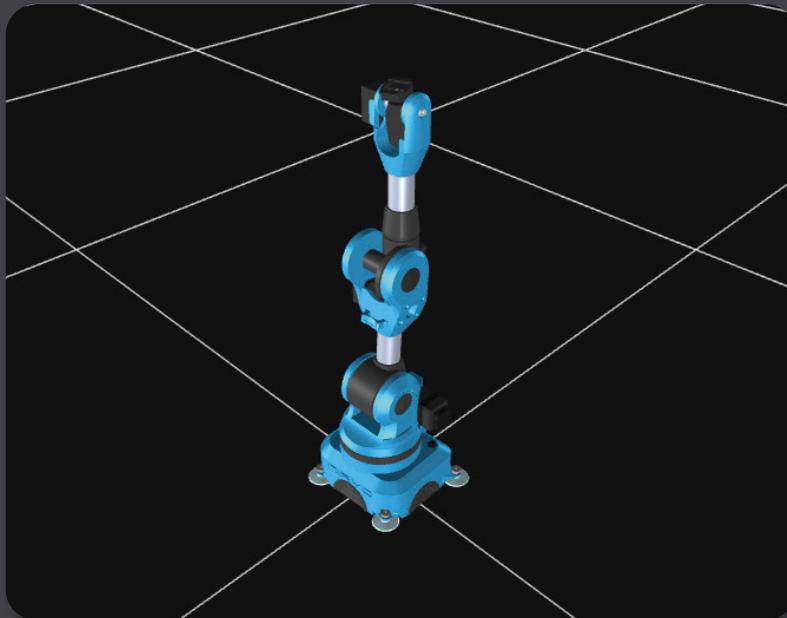
*Vous pouvez également appuyer sur le bouton supérieur du robot pour enregistrer la valeur.*

- Désactiver le « **mode apprentissage** »
- Exécuter un mouvement vers cette position en cliquant sur « **DÉPLACER AXES** »

### Le robot a-t-il effectué le mouvement voulu ?

---

- On désire mettre le robot dans une position « TOP » comme montrée dans la figure ci-dessous :



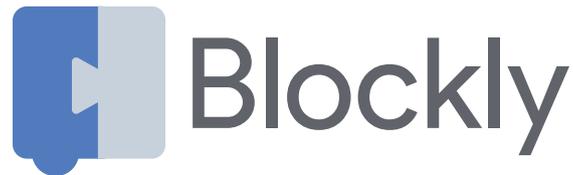
### Quel est l'axe qui va faire son angle maximal ?

- Activer le « **mode apprentissage** » et bouger le robot avec la main vers cette position
- Maintenir le robot dans cette position, sélectionner « **ACTUALISER LES VALEURS** » puis l'enregistrer sous le nom « TOP »
- Sélectionner cette position et l'exécuter en cliquant sur « **DÉPLACER AXES** »

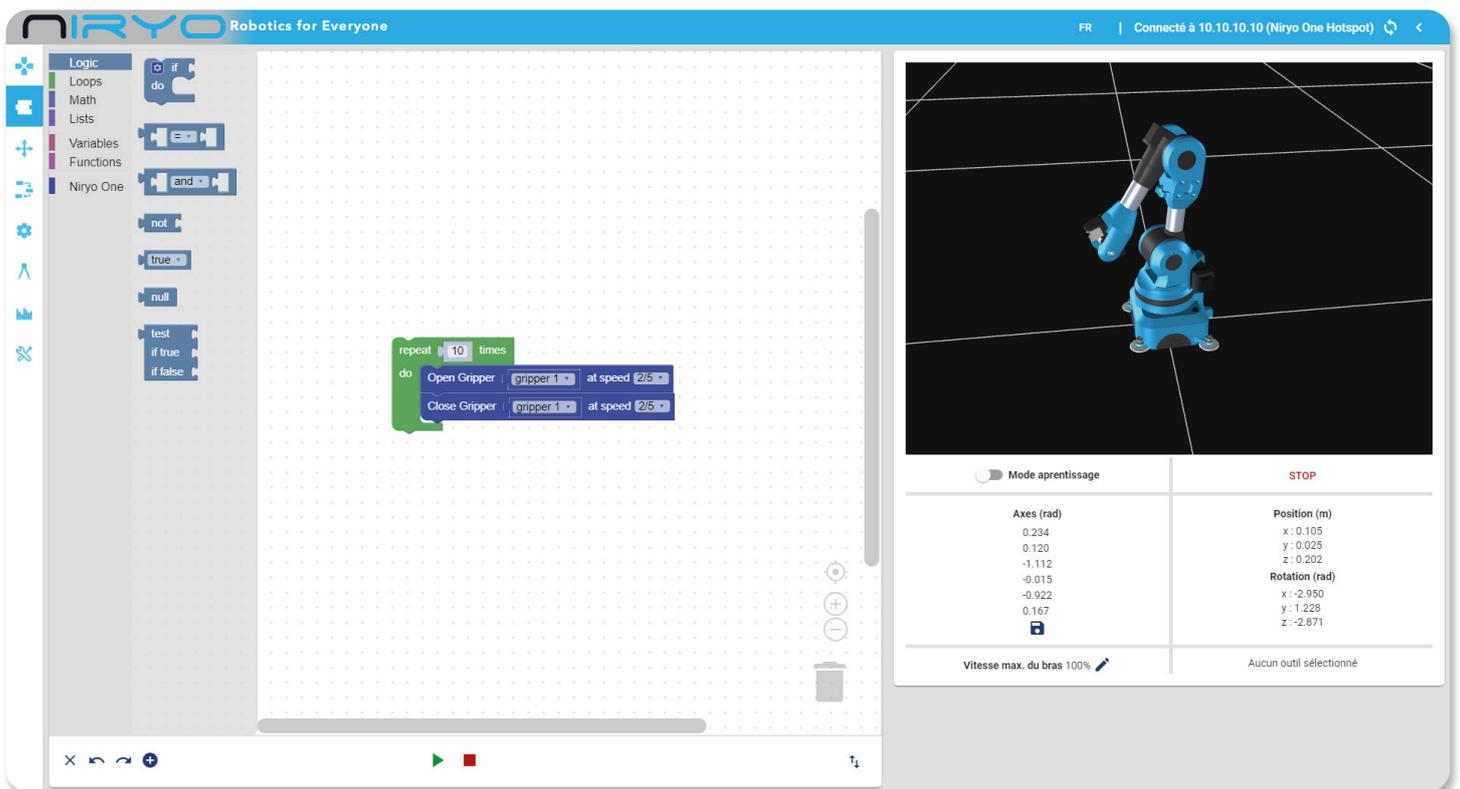
# PROGRAMMER NIRYO ONE AVEC BLOCKLY

Dans cette partie, nous nous intéressons aux fonctions de base du logiciel de programmation visuelle, basé sur Blockly.

## PRÉSENTATION DE BLOCKLY

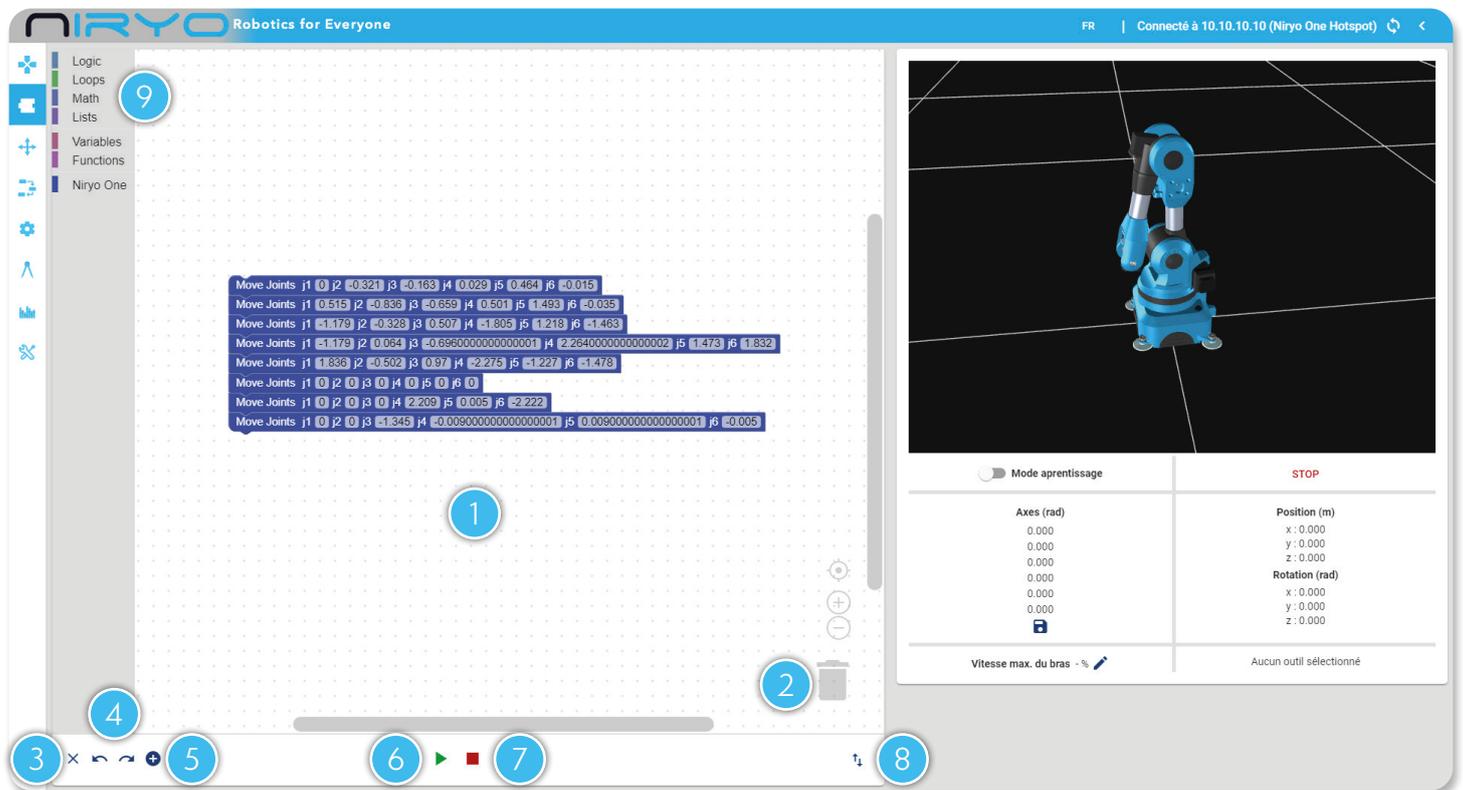


L'interface visuelle de Niryo One Studio est basée sur Blockly, une bibliothèque Google, qui est également utilisée pour le projet Scratch réalisé par le MIT.



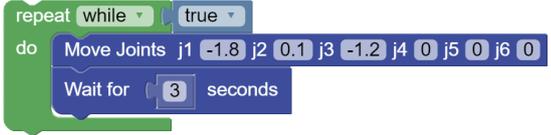
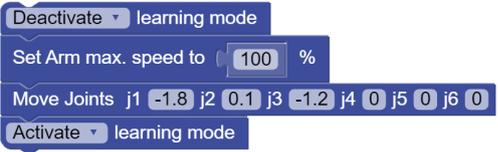
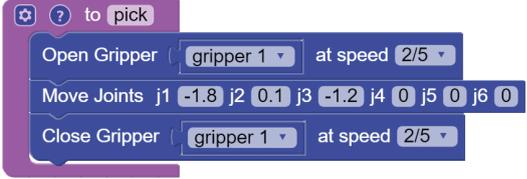
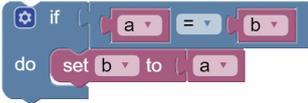
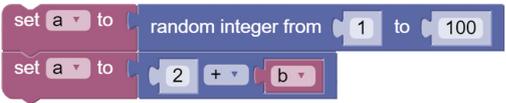
## L'ENVIRONNEMENT DE PROGRAMMATION

Les différentes parties de l'environnement et leurs fonctionnalités sont :



- 1 — L'espace de travail qui va contenir votre code
- 2 — Pour supprimer un bloc, le faire simplement glisser jusque la corbeille  
*Il est également possible de le sélectionner puis appuyer sur la touche « Suppr » du clavier*
- 3 — Effacer l'espace de travail actuel
- 4 — Annuler / Rétablir
- 5 — Ajouter un bloc de position
- 6 — Exécuter la séquence affichée dans l'espace de travail
- 7 — Arrêter l'exécution de la séquence en cours
- 8 — Importer ou exporter une séquence
- 9 — Les différents blocs de programmation

# LES BLOCS DE PROGRAMMATION

CONCEPT	EXPLICATION	EXEMPLE
<b>Variables</b>	Les blocs variables permettent de créer des variables et les utiliser dans le programme. <i>Une variable est un symbole qui associe un nom à une valeur.</i>	
<b>Listes</b> (Lists)	Les blocs listes permettent de stocker puis d'avoir accès à une liste de nombres et de chaînes de caractères. <i>Une liste est une structure permettant de regrouper des données de manière à pouvoir y accéder librement. Elle peut être considérée comme un tableau.</i>	
<b>Boucles</b> (Loops)	Les blocs de boucles peuvent être utilisés pour une itération (répétition d'une série d'instructions). <i>Une boucle est une structure de contrôle de programmation qui permet de répéter l'exécution d'une séquence d'instructions.</i>	
<b>Niryo One</b>	Les blocs de la catégorie Niryo One permettent de contrôler le robot, ses outils ainsi que ses interfaces d'entrée/sortie.	
<b>Fonctions</b> (Functions)	Les blocs fonctions permettent de créer et appeler des fonctions et procédures. <i>Une fonction est une entité informatique qui encapsule une portion de code effectuant un traitement spécifique qui peut être réutilisé dans le programme.</i>	
<b>Logiques</b> (Logics)	Les blocs logiques permettent soit d'engager une action suivant une condition soit de manipuler les variables logiques.	
<b>Maths</b>	Les blocs Maths permettent d'utiliser les opérations mathématiques.	

Se référer à l'annexe pour plus de détails sur les différents blocs.

# EXERCICE

Nous désirons déplacer le Niryo One aux deux positions P1 et P2 avec :

- $P1 = [ x = -0.03 ; y = -0.156 ; z = 0.48 ; roll = -0.58 ; pitch = -0.58 ; yaw = -0.145 ]$
- $P2 = [ x = -0.136 ; y = -0.133 ; z = 0.255 ; roll = -0.081 ; pitch = 0.744 ; yaw = -2.535 ]$

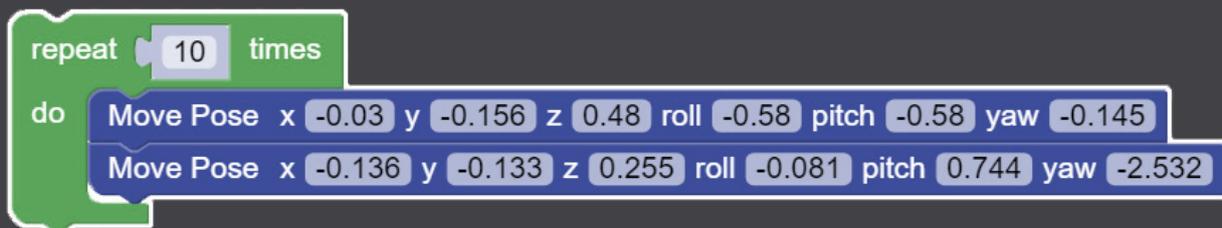
Après avoir pris connaissance de l'annexe, déterminer le bloc qui permet de déplacer le robot à une position avec des coordonnées cartésiennes x y et z.

- Copier le code ci-dessous dans Niryo One Studio

```
Move Pose x -0.03 y -0.156 z 0.48 roll -0.58 pitch -0.58 yaw -0.145
```

```
Move Pose x -0.136 y -0.133 z 0.255 roll -0.081 pitch 0.744 yaw -2.532
```

- Désactiver le « mode apprentissage » et exécuter le programme
- Ajouter le bloc sur votre séquence comme le montre l'illustration ci-dessous :



- Désactiver le « mode apprentissage » puis exécuter cette nouvelle séquence

Expliquer la différence entre la première et la seconde séquence et conclure sur le rôle du bloc ajouté.

On désire automatiser la tâche d'activation et de désactivation du « mode apprentissage ».

Quel bloc convient-il d'utiliser pour effectuer cette opération ? Ajouter ce bloc au début et à la fin de votre programme, puis exécuter ce dernier.

Une séquence est une suite d'instructions. Elle représente le programme à faire exécuter par le robot.

En gardant le programme développé dans la section précédente, cliquez sur le bouton « import/export ».



← CRÉER UNE NOUVELLE SÉQUENCE

Nom \_\_\_\_\_

Description \_\_\_\_\_

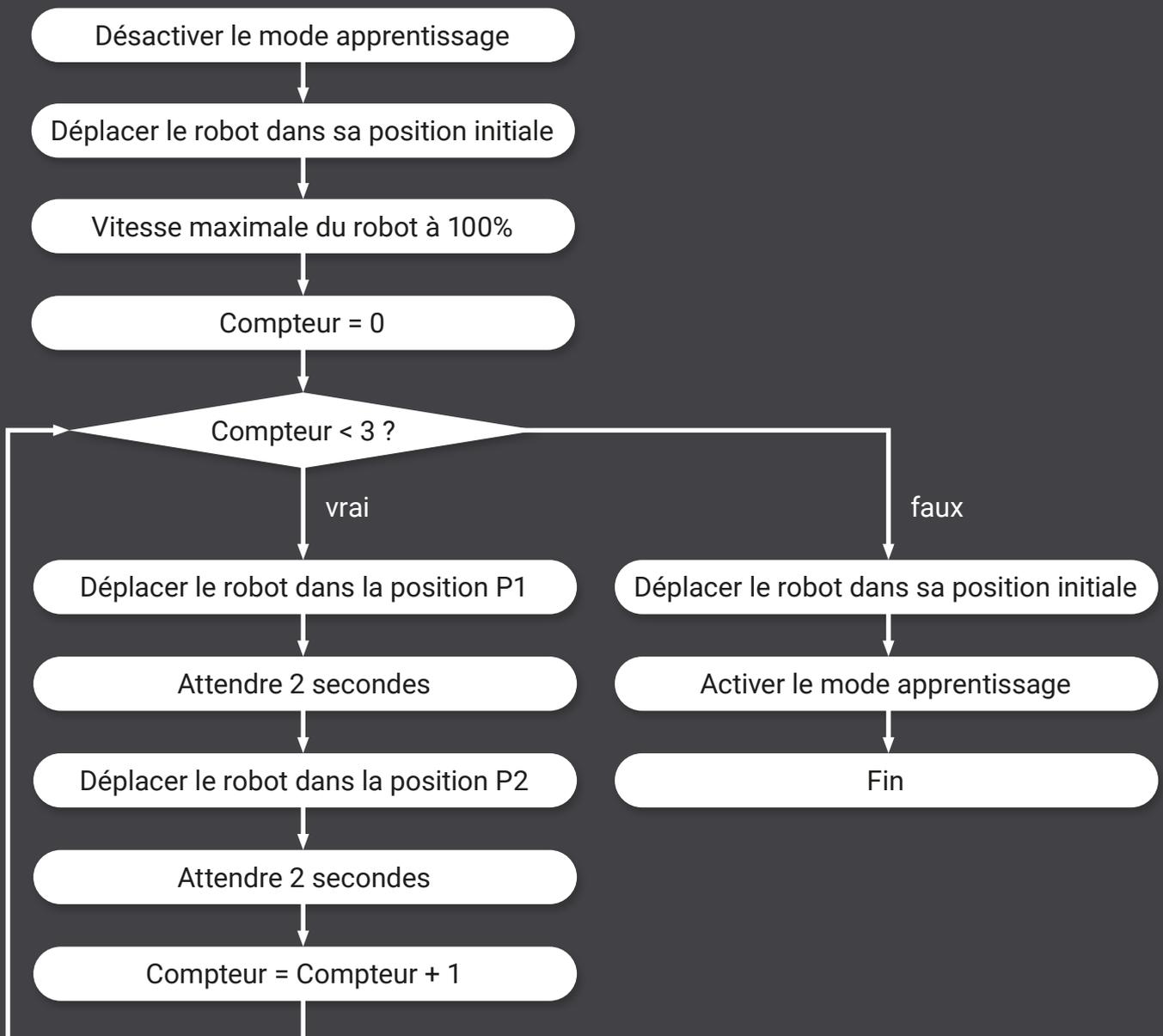
XML Blockly  
 <xml xmlns="http://www.w3.org/1999/xhtml"><block type="niryo\_one\_activate\_learning\_mode" id="2H-LaSy+N:i'w+\*a?r" x="-770" y="-710"><field name="LEARNING\_MODE\_VALUE">0</field><next><block type="controls\_repeat\_ext" id="Q9PmGur00P6[Y|=i%WSf"><value name="TIMES"><shadow

SAUVEGARDER

Une séquence est caractérisée par :

- Son nom (par défaut, celui-ci sera « sequence » + ID)
- Sa description

On désire exécuter le programme suivant :



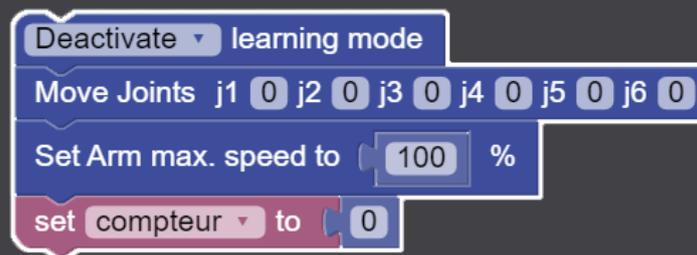
Sachant que :

- Les positions du robot sont en coordonnées articulaires (joints)
- La position initiale du robot est [ 0 ; 0 ; 0 ; 0 ; 0 ; 0 ]
- $P1 = [-1.215 ; -0.333 ; -0.867 ; -3.053 ; -0.014 ; 0.127 ]$
- $P2 = [-1.377 ; 0.22 ; -0.281 ; -3.053 ; -0.031 ; 1.549 ]$

Traduire le diagramme précédent en texte.

---

- Reproduire le programme ci-dessous :



Choisir le bloc qui permet de faire la boucle du programme présenté ci-dessus.

À l'aide de l'annexe, compléter la séquence Blockly afin qu'elle reproduise le diagramme précédent.

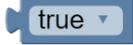
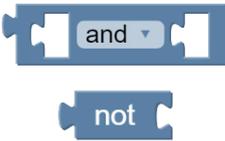
---

Considérons la variable compteur et le bloc de boucle. Proposer un bloc permettant de remplacer ces deux blocs.

- Remplacer les blocs de variable et de boucle par ce nouveau bloc
- Exécuter la séquence

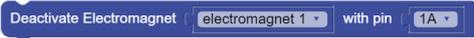
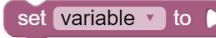
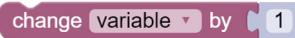
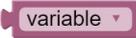
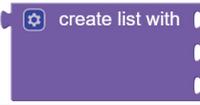
Commenter.

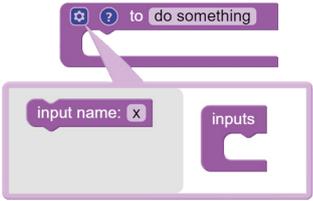
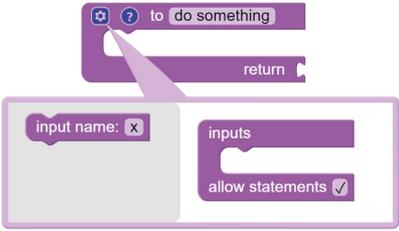
# ANNEXE

TYPE	CONCEPT	BLOC	EXPLICATION
Logic	Instructions conditionnelles		Effectuer différents calculs ou actions en fonction de l'évaluation d'une condition.
Logic	Opérateurs de relation ou comparaison		<p>Pour comparer certaines valeurs :</p> <ul style="list-style-type: none"> <li>• Le test d'égalité est effectué avec l'opérateur = tandis que le test de différence est effectué avec l'opérateur ≠</li> <li>• Les comparaisons strictes sont effectuées avec les opérateurs &lt; et &gt;</li> <li>• Les comparaisons larges sont effectuées avec les opérateurs &lt;= et &gt;=.</li> </ul>
Logic	Types booléens		Le type booléen définit deux états : l'état vrai et l'état faux. Les mots clés associés sont donc true (vrai) et false (faux).
Logic	Opérateurs logiques booléens		<ul style="list-style-type: none"> <li>• And : retourne «vrai» si les deux valeurs correspondent à «vrai», sinon retourne «faux»</li> <li>• Or : retourne «vrai» si au moins une des deux valeurs correspond à «vrai»</li> <li>• Not : retourne l'inverse du booléen.</li> </ul>
Logic	Valeur		Une variable est « null » si elle est nulle ou ne contient aucune valeur.
Loops	Itération (boucle)		Répéter une séquence d'instructions x fois, x étant alors le nombre d'itérations.
Loops	Itération (boucle)		Créer une boucle qui exécute une instruction tant qu'une condition n'est pas vérifiée.
Loops	Itération (boucle)		<p>Répéter un certain nombre de fois une suite d'instruction.</p> <p><i>Exemple : count i from 1 to 10 by 1 (cette instruction permet de compter la variable i de 1 à 10 en incrémentant par 1).</i></p>

TYPE	CONCEPT	BLOC	EXPLICATION
Loops	Itération (boucle)		Répéter certaines instructions pour chaque élément d'une liste.
Loops	Itération (boucle)		Sortir de la boucle actuelle (for/repeat/count) et passer directement à l'instruction suivante.
Math	Paramétrage		Spécifier un nombre pour une fonction.
Math	Opérateur		<ul style="list-style-type: none"> <li>• + : Ajouter deux nombres</li> <li>• - : Soustraire le deuxième nombre du premier nombre</li> <li>• * : Multiplier les deux nombres</li> <li>• / : Diviser le premier nombre par le second nombre.</li> </ul>
Math	Opérateur		<p>Vérifier si un nombre est :</p> <ul style="list-style-type: none"> <li>• Even : un nombre pair (exactement divisible par 2)</li> <li>• Odd : un nombre impair (n'est pas exactement divisible par 2)</li> <li>• Prime : un nombre premier (2, 3, 5...)</li> <li>• Whole : un nombre entier</li> <li>• Positive : <math>\geq 0</math></li> <li>• Negative : <math>\leq 0</math></li> <li>• Divisible by : un nombre divisible par...</li> </ul>
Math	Opérateur		Renvoyer l'entier inférieur (round down), supérieur (round up) ou l'entier le plus proche (round).
Math	Opérateur		<p>Renvoyer :</p> <ul style="list-style-type: none"> <li>• Sum : la somme d'une liste</li> <li>• Min : le minimum d'une liste</li> <li>• Max : le maximum d'une liste</li> <li>• Average : la moyenne d'une liste</li> <li>• Median : la médiane (la valeur centrale d'une liste)</li> <li>• Mode : la valeur la plus fréquente de la liste</li> <li>• Standard deviation : l'écart-type d'une liste</li> <li>• Random item : un nombre aléatoire de la liste.</li> </ul>

TYPE	CONCEPT	BLOC	EXPLICATION
Math	Opérateur		Calculer le reste dans une division de nombres entiers.
Math	Opérateur		Renvoyer un entier aléatoire dans l'intervalle.
Niryo One	Mouvement		Bouger les axes de robot suivant les angles spécifiés.
Niryo One	Mouvement		Bouger le robot à la position et l'orientation spécifiées.
Niryo One	Mouvement		Décaler une coordonnée (orientation ou position) à une valeur.
Niryo One	Calibrage		Calibrer le robot automatiquement.
Niryo One	Calibrage		Calibrer le robot manuellement.
Niryo One	Paramétrage		Régler la vitesse maximale du robot.
Niryo One	Programmation		Mettre en pause l'exécution pendant le nombre de secondes indiqué.
Niryo One	Programmation		Activer ou désactiver le mode apprentissage
Niryo One	Préhenseurs		Changer le type de préhenseur.
Niryo One	Préhenseurs		Une variable qui contient les préhenseurs disponible pour Niryo One.
Niryo One	Préhenseurs		Détacher le préhenseur.
Niryo One	Préhenseurs		Ouvrir le gripper à une vitesse spécifique.
Niryo One	Préhenseurs		Fermer le gripper à une vitesse spécifique.
Niryo One	Préhenseurs		Aspirer l'air avec la pompe à vide pour attraper un objet.
Niryo One	Préhenseurs		Vider l'air de la pompe à vide pour lâcher l'objet.
Niryo One	Préhenseurs		Configurer l'électroaimant en spécifiant le pin de connexion.
Niryo One	Préhenseurs		Activer l'électroaimant en spécifiant le pin de connexion.

TYPE	CONCEPT	BLOC	EXPLICATION
Niryo One	Préhenseurs		Désactiver l'électroaimant en spécifiant le pin de connexion.
Niryo One	Entrées/Sorties		Une variable qui contient les pins disponibles sur le Niryo One.
Niryo One	Entrées/Sorties		Mettre le pin numérique à HIGH ou LOW.
Niryo One	Entrées/Sorties		Configurer le pin : • INPUT : entrée, • OUTPUT : sortie.
Niryo One	Entrées/Sorties		Mettre le pin digital à HIGH ou LOW.
Niryo One	Entrées/Sorties		Renvoyer la valeur du pin.
Niryo One	Entrées/Sorties		Régler le commutateur (SW1/ SW2) à 1 (HIGH) ou à 0 (LOW).
Niryo One	Programmation		Ajouter un commentaire
Niryo One	Programmation		Ajouter un break point permet de mettre le programme en pause. Pour reprendre l'exécution, il faut appuyer sur le bouton « play ».
Variables	Variables		Créer et nommer une nouvelle variable. Lorsque vous créez une variable, les blocs correspondant à la variable s'affiche.
Variables	Variables		Attribuer une valeur à la variable.
Variables	Variables		Changer la valeur de la variable.
Variables	Variables		Renvoyer la variable.
Lists	Listes		Créer une liste vide.
Lists	Listes		Créer une liste en ajoutant des éléments.
Lists	Listes		Créer une liste avec un élément répété x fois.
Lists	Listes		Retourner la longueur de la liste.

TYPE	CONCEPT	BLOC	EXPLICATION
<b>Lists</b>	Listes		Retourner la position : <ul style="list-style-type: none"> <li>• First : de la première occurrence de l'élément dans la liste</li> <li>• Last : de la dernière occurrence de l'élément dans la liste.</li> </ul>
<b>Lists</b>	Listes		Retourner <ul style="list-style-type: none"> <li>• # : l'élément n dans la liste</li> <li>• # from end : l'élément n dans la liste à partir de la fin de celle-ci</li> <li>• First : le premier élément de la liste</li> <li>• Last : le dernier élément de la liste</li> <li>• Random : élément aléatoire de la liste.</li> </ul>
<b>Functions</b>	Fonctions		Créer une procédure avec des arguments (paramètres d'entrée).
<b>Functions</b>	Fonctions		Créer une fonction avec des arguments (paramètres d'entrées) et un sortie (un retour).

Retrouvez tous nos supports pédagogiques sur

**[www.niryo.com](http://www.niryo.com)**

# MODE APPRENTISSAGE : CORRECTION

- Activer le mode d'apprentissage du robot
- Bouger le robot avec la main vers une position de votre choix
- Maintenir le robot dans cette position et cliquer sur « **ACTUALISER LES VALEURS** » avant de relâcher le robot.



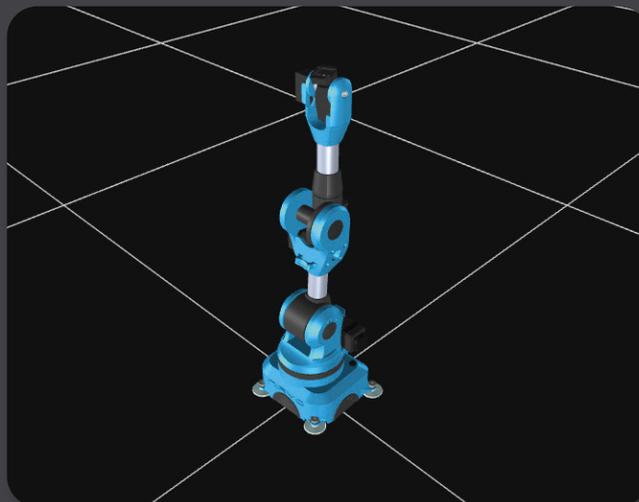
*Vous pouvez également appuyer sur le bouton supérieur du robot pour enregistrer la valeur.*

- Désactiver le « **mode apprentissage** »
- Exécuter un mouvement vers cette position en cliquant sur « **DÉPLACER AXES** »

**Le robot a-t-il effectué le mouvement voulu ?**

**Oui, le robot a effectué le mouvement voulu. En effet, le mode apprentissage est conçu pour enregistrer des positions et séquences déterminées manuellement par l'opérateur afin qu'elle puissent être réutilisées par la suite.**

- On désire mettre le robot dans une position « TOP » comme montrée dans la figure ci-dessous :



**Quel est l'axe qui va faire son angle maximal ?**

**Il s'agit de l'axe 3, qui va atteindre un angle de 90°.**

- Activer le « **mode apprentissage** » et bouger le robot avec la main vers cette position
- Maintenir le robot dans cette position, sélectionner « **ACTUALISER LES VALEURS** » puis l'enregistrer sous le nom « TOP »
- Sélectionner cette position et l'exécuter en cliquant sur « **DÉPLACER AXES** »

# BLOCKLY : CORRECTION

Nous désirons déplacer le Niryo One aux deux positions P1 et P2 avec :

- $P1 = [x = -0.03 ; y = -0.156 ; z = 0.48 ; roll = -0.58 ; pitch = -0.58 ; yaw = -0.145]$
- $P2 = [x = -0.136 ; y = -0.133 ; z = 0.255 ; roll = -0.081 ; pitch = 0.744 ; yaw = -2.535]$

Après avoir pris connaissance de l'annexe, déterminer le bloc qui permet de déplacer le robot à une position avec des coordonnées cartésiennes x y et z.

```
Move Pose x 0 y 0 z 0 roll 0 pitch 0 yaw 0
```

- Copier le code ci-dessous dans Niryo One Studio

```
Move Pose x -0.03 y -0.156 z 0.48 roll -0.58 pitch -0.58 yaw -0.145  
Move Pose x -0.136 y -0.133 z 0.255 roll -0.081 pitch 0.744 yaw -2.532
```

- Désactiver le « mode apprentissage » et exécuter le programme
- Ajouter le bloc sur votre séquence comme le montre l'illustration ci-dessous :

```
repeat 10 times  
do  
  Move Pose x -0.03 y -0.156 z 0.48 roll -0.58 pitch -0.58 yaw -0.145  
  Move Pose x -0.136 y -0.133 z 0.255 roll -0.081 pitch 0.744 yaw -2.532
```

- Désactiver le « mode apprentissage » puis exécuter cette nouvelle séquence

Expliquer la différence entre la première et la seconde séquence et conclure sur le rôle du bloc ajouté.

**Dans le premier code, le robot s'est déplacé vers une première position puis une seconde. Dans le second code, le robot a répété ces mouvements 10 fois.**

On désire automatiser la tâche d'activation et de désactivation du « mode apprentissage ».

Quel bloc convient-il d'utiliser pour effectuer cette opération ? Ajouter ce bloc au début et à la fin de votre programme, puis exécuter ce dernier.

Activate ▾ learning mode

Comme détaillé dans l'annexe, ce bloc permet d'activer ou de désactiver le mode apprentissage.

Une séquence est une suite d'instructions. Elle représente le programme à faire exécuter par le robot.

En gardant le programme développé dans la section précédente, cliquez sur le bouton « import/export ».



← CRÉER UNE NOUVELLE SÉQUENCE

Nom \_\_\_\_\_

Description \_\_\_\_\_

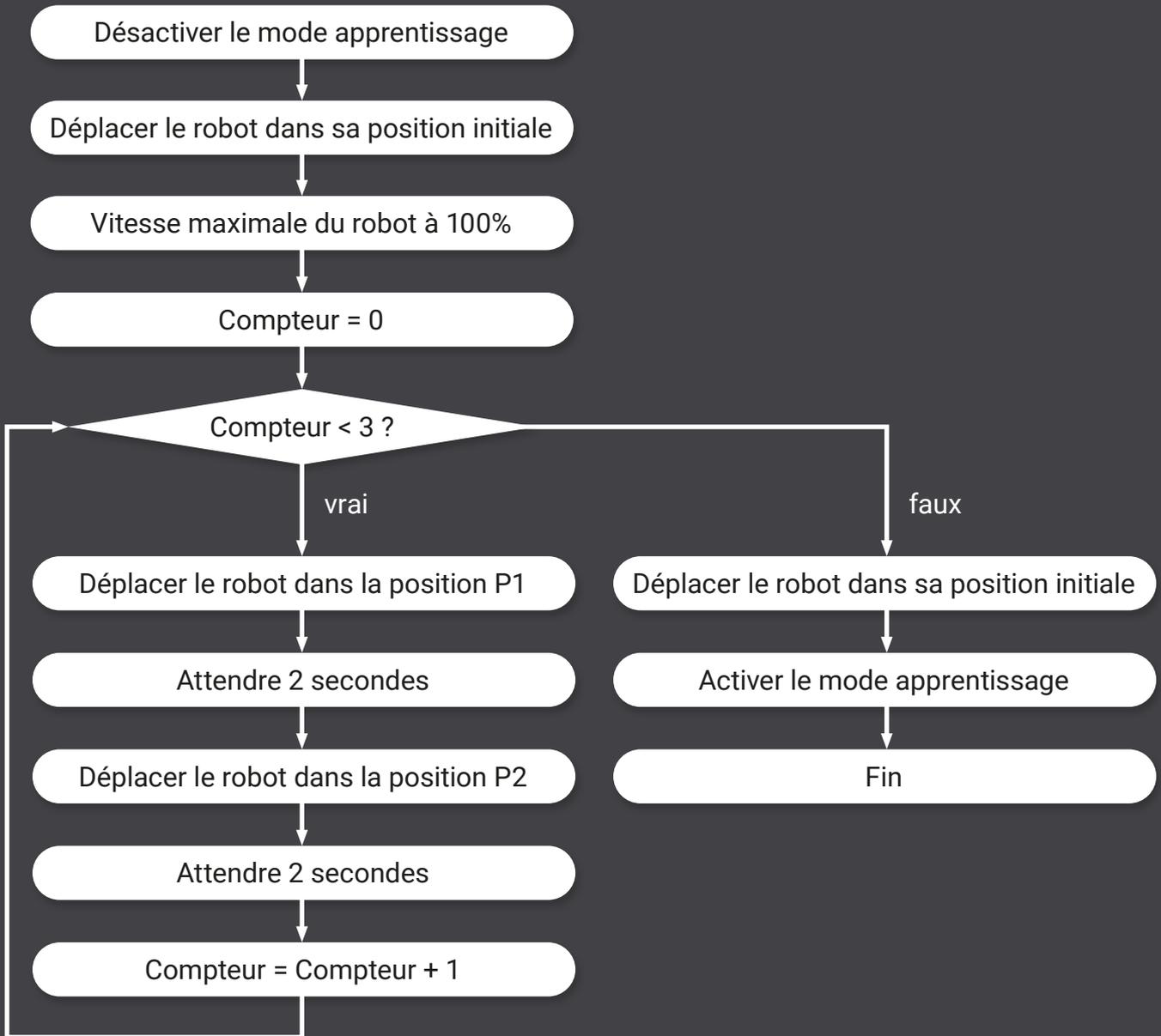
XML Blockly  
<xml xmlns="http://www.w3.org/1999/xhtml"><block type="niryo\_one\_activate\_learning\_mode" id="2H-t,aSy+N:'i'w+\*a?'r" x="-770" y="-710"><field name="LEARNING\_MODE\_VALUE">0</field><next><block type="controls\_repeat\_ext" id="Q9PmGur00P6[Y]=i%WSf"><value name="TIMES"><shadow

SAUVEGARDER

Une séquence est caractérisée par :

- Son nom (par défaut, celui-ci sera « sequence » + ID)
- Sa description

On désire exécuter le programme suivant :



Sachant que :

- Les positions du robot sont en coordonnées articulaires (joints)
- La position initiale du robot est  $[0 ; 0 ; 0 ; 0 ; 0 ; 0]$
- $P1 = [-1.215 ; -0.333 ; -0.867 ; -3.053 ; -0.014 ; 0.127]$
- $P2 = [-1.377 ; 0.22 ; -0.281 ; -3.053 ; -0.031 ; 1.549]$

Traduire le diagramme précédent en texte.

**Tout d'abord, désactiver le mode apprentissage, mettre la vitesse du robot à 100%, et rejoindre la position initiale.**  
**Aller jusque P1, attendre 2 seconds, aller jusque P2, et répéter 3 fois ces mouvements.**  
**Enfin, après ces trois répétitions, retourner à la position initiale puis réactiver le mode apprentissage.**

- Reproduire le programme ci-dessous :

```

Deactivate learning mode
Move Joints j1 0 j2 0 j3 0 j4 0 j5 0 j6 0
Set Arm max. speed to 100 %
set compteur to 0
  
```

Choisir le bloc qui permet de faire la boucle du programme présenté ci-dessus.



À l'aide de l'annexe, compléter la séquence Blockly afin qu'elle reproduise le diagramme précédent.

```

Deactivate learning mode
Move Joints j1 0 j2 0 j3 0 j4 0 j5 0 j6 0
Set Arm max. speed to 100 %
set Compteur to 0
repeat while Compteur < 3
do
  Move Joints j1 -1.215 j2 -0.333 j3 -0.867 j4 -3.053 j5 -0.014 j6 0.127
  Wait for 2 seconds
  Move Joints j1 -1.337 j2 0.22 j3 -0.281 j4 -3.053 j5 -0.031 j6 1.549
  Wait for 2 seconds
  change Compteur by 1
Move Joints j1 0 j2 0 j3 0 j4 0 j5 0 j6 0
Activate learning mode
  
```

Considérons la variable compteur et le bloc de boucle. Proposer un bloc permettant de remplacer ces deux blocs.



- Remplacer les blocs de variable et de boucle par ce nouveau bloc
- Exécuter la séquence et commenter.

La variable Compteur permet de déterminer le nombre de répétitions restant à effectuer. Les blocs de boucle tels que «repeat x times» et «count with» ont la même propriété.