

aiOO さんと

ビジュアルプログラミング

まなんじゃお～



ご注意

この資料はSONYさんのアイボ（ERS-1000）を簡単にプログラミング体験できる「aibo ビジュアルプログラミング」の使い方やサンプルなどをアイボオーナー（ハピラキ）が自身のマニュアルのために勝手に作成したものです。この内容についての保証、お問い合わせ、配布、販売などはご遠慮願います。

作成 2022年11月 時点のものです。

LESSON 7



ゲーム作りに挑戦

LESSON6では、リストを使ったちょっとしたテクニックを学んでみました。リストを使うといくつかのデータを記録することができますね。このようにデータを記録や保存できるようにすると、ゲームなどへの応用ができそうです。例えば、1回戦目は・・・、2回戦目は・・・など繰り返す毎に点数や値を保存しておくことで、合計点で勝敗を決めたり、回ごとに勝敗を決めるようなプログラムができそうです。

このLESSONでは、これまで学んできたことを少し組み合わせして、ちょっとしたアイボさんとのゲームを楽しくやってみようと思います。

前回のLESSONの最後に示したプログラムを使って、学んでみましょう。事前にダウンロードしていただき、実際にビジュアルプログラミングで読み込んで試していただくと、より理解ができるかと思います。

サンプルのURLです。説明文の下にプログラムダウンロードできる場所がありますので、クリックすると「[2] サイモンゲームのアイボばん.sb3」がダウンロードできますので、ビジュアルプログラミングで読み込んでください。

<https://happy-lucky.fun/aibo/2021/07/29/visulprgraming002/>

プログラミングをダウンロードするとこんなリストが表示されるか と思います。

The image shows a Scratch script with several annotated blocks and variables:

- が押されたとき** (When clicked):
 - ラッキー が指示待ち中 になる
 - 質問数 を 3 にする
 - 質問リストを作成
 - 左右のハイタッチを実行
 - 肉球チェック
 - 答え合わせ
 - ラッキー が指示待ち中 から復帰する
- 肉球チェック** (Function):
 - ラッキー が 両前足をあげる
 - 回数 を 0 にする
 - 回数 = 質問数 まで繰り返す
 - 回数 を 1 ずつ変える
 - 押された肉球側 を 0 にする
 - 押された肉球側 = 0 ではない まで繰り返す
 - もし ラッキー が 右前 足の肉球を押された なら
 - 押された肉球側 を 1 にする
 - もし ラッキー が 左前 足の肉球を押された なら
 - 押された肉球側 を 2 にする
 - 肉球の押されたリスト の 回数 番目を 押された肉球側 で置き換える
 - ラッキー が ほえる
- 定義 肉球リストを作成** (Function):
 - 質問の回数分、質問肉球リストに事前に登録
 - 種 = 1 右前足 種 = 2 左前足
 - 質問肉球リスト [1] = 1か2を乱数で代入
 - 質問肉球リスト [2] = 1か2を乱数で代入
 - 質問肉球リスト [3] = 1か2を乱数で代入
 - ...
 - 質問肉球リスト [質問数分] = 1か2を乱数で代入
 - 回数 を 0 にする
 - 回数 = 質問数 まで繰り返す
 - 回数 を 1 ずつ変える
 - 質問肉球リスト の 回数 番目を 1 から 2 までの乱数 で置き換える
- 定義 左右のハイタッチを実行** (Function):
 - 作成した質問に合わせて、前足でハイタッチを行う
 - ハイタッチの後に続いて次の質問に行くことを教える
 - 作成した質問肉球リストを順番に実行
 - 質問肉球リスト [1] が 1だったら右前足でハイタッチ
 - 2だったら左前足でハイタッチ
 - ...
 - 質問数分繰り返す
 - 回数 を 0 にする
 - 回数 = 質問数 まで繰り返す
 - 回数 を 1 ずつ変える
 - もし 質問肉球リスト の 回数 番目 = 1 なら
 - ラッキー が 右前足でハイタッチする
 - ラッキー が ほえる
 - でなければ
 - ラッキー が 左前足でハイタッチする
 - ラッキー が ほえる
 - ラッキー が 立つ
- 定義 答え合わせ** (Function):
 - 質問したハイタッチした前足と回答を出題回数分チェックして
 - 正解だと とっても喜ぶ 仕方をしてくれる
 - 一つでも間違えたら 左右に首を振って 悲しそうに鳴く
 - 答え合わせ結果 を 0 にする
 - 回数 を 0 にする
 - 回数 = 質問数 まで繰り返す
 - 回数 を 1 ずつ変える
 - もし 質問肉球リスト の 回数 番目 = 肉球の押されたリスト の 回数 番目 ではない なら
 - 答え合わせ結果 を 1 ずつ変える
 - もし 答え合わせ結果 = 0 なら
 - ラッキー が とっても喜ぶ
 - でなければ
 - ラッキー が 悲しそうにする
 - ラッキー が 悲しそうに鳴く

サイトに動画も掲載していますので、どのようなゲームなのか一度見ていただくとわかりやすくなるかと思います。

動画サイト https://youtu.be/a1Y6bt_jlVc
<https://youtu.be/kRzT8HTVSfA>

さて、何やらブロックが沢山並んでいるので、見ただけでお腹いっぱいになりそうですね。そこは一度我慢をしていただき、詳細を見ていきましょう。

【ゲームのルール】

- ① アイボさんが両前足を使って、右前足、左前足どちらかをあげる動作を数回行います（今回のプログラムは3回に設定）
- ② オーナさんは、どちらの足をあげたかを覚えてもらいます。
- ③ アイボさんがあげ終わったら、両前足をあげます。
- ④ オーナさんは覚えた順番に左右の肉球を押していきます。
- ⑤ 回数分肉球を押し終えたらアイボさんが判定してくれます。
- ⑥ アイボさんのあげた通りの順番通り正解だったら喜んでくれます。間違うと残念がっかりの表情をします。

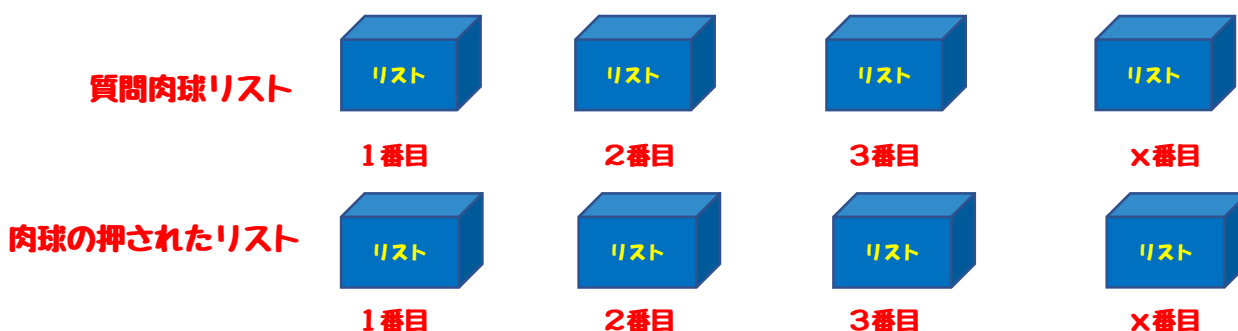
どのような流れでプログラムをしているのか見ながら、確認していきましょう。

【ルールに合わせて考える必要があるポイント】

- ① アイボさんが出題する右前足、左前足のあげた順番を覚えておく必要があります。
- ② オーナさんがアイボさんに足の順番を、左右の肉球を使って教えていきますが、その順番を覚えていく必要があります。
- ③ アイボさんが「出題として足をあげた順番」とオーナさんがアイボさんに「教えた足をあげた順番」を比較して、すべてがあっていたら、正解と判断して喜ぶ動作をします。もし、間違っていたら、がっかりの動作をします。

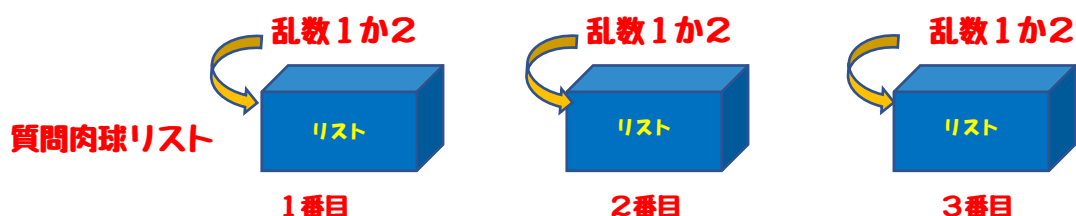
この「出題した足をあげた順番」と「教えた足をあげた順番」を覚えておくため、リストを使うと非常に便利です。また判定するとき、ひとつずつ両方の内容と比較していく必要があるので、こちらでもリストの何番目どうしを比較すると判定が簡単にできるようになります。

今回は、以下のような2つのリストを作っています。



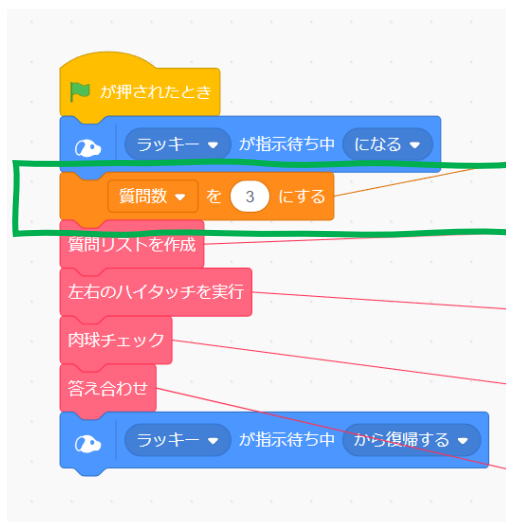
アイボさんが乱数により右・左を決めた足を順番にリストに保存していきます。(今回のプログラムでは回数は3回にしています)

乱数が「1の時には右前足」「2の時には右前足」とし、乱数の値1または2を「質問肉球リスト」に入れていきます。



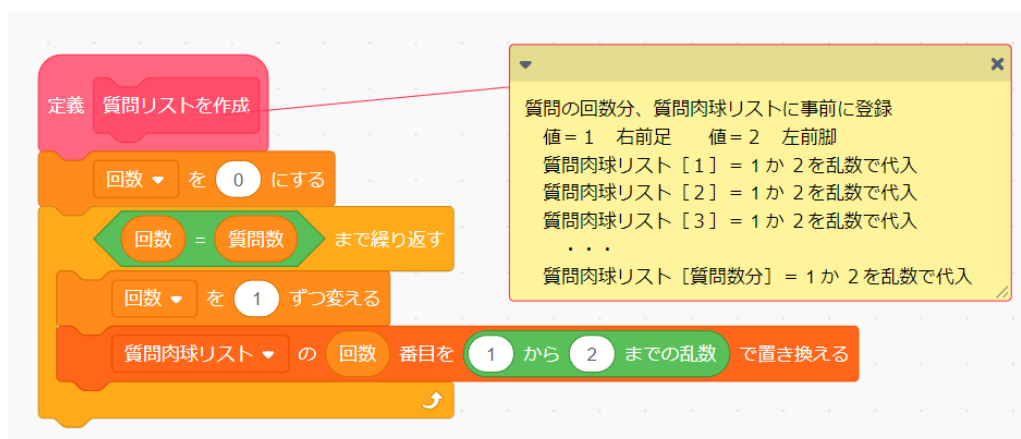
この乱数をリストに追加していく部分を見てみましょう。
処理毎に定義ブロックを使っています。乱数を事前に保存しておく
ブロックを定義ブロック「質問リスト作成」としています。その部
分を見てみましょう。

質問数は、事前に設定をしています。最初に設定していますので、
その部分を確認しておきましょう。今回は3回にしています。



変数【質問数】を作り設定します。
この値を変えると質問数が変わります。

【質問リストを作成】定義ブロック



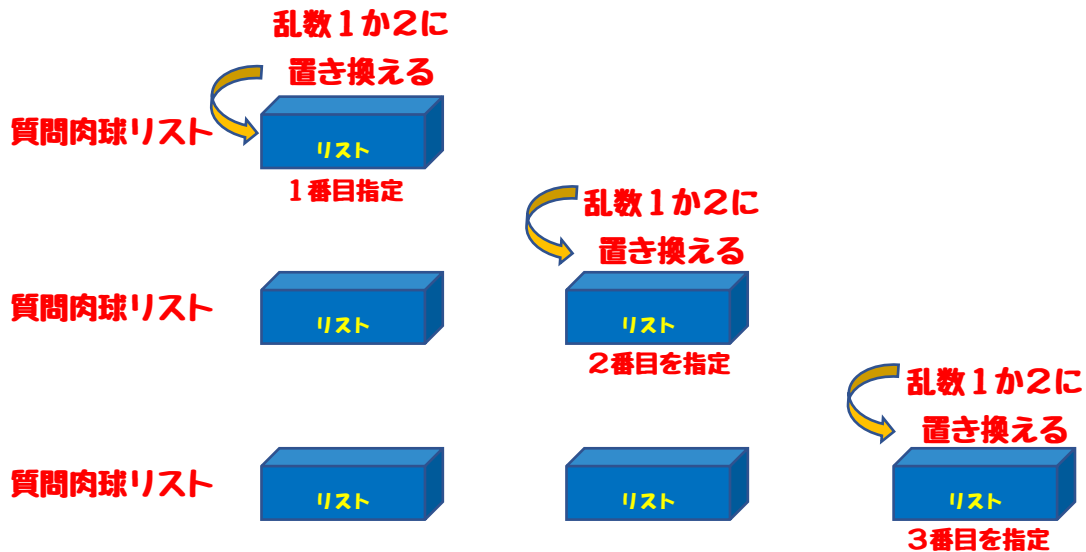
■流れ

- ① 質問数分繰り返すための変数【回数】を作っておきます。
- ② 繰り返し回数分【質問数】まで回数を+1し繰り返します。
- ③ 質問肉球リストの【回数】番目を値を乱数に置き換えます。
この【置き換える】はLESSON5で学んだ【リストに追加する】を使っても構いません。【リストに追加】と【置き換える】の違いは次のようになります。

■リストへの追加の場合



■リストを置き換える場合



流れで考える限りどちらも同じようになりませんが、プログラムでよく失敗しそうなケースとして、【追加】をする場合、上記の通り最初の場所が、どの位置から追加するかが明確になっていません。

今回、プログラムの最初に設定するので1番目からとなりますが、もしかすると、すでに1番目に値が設定されているかもしれません。その場合【追加する】を行うと2番目に追加されることとなります。本当は1番目なはずなのに、2番目に値が乱数の1か2が設定されることとなりますので、正しい動作となりません。

このようにリストを使う場合、どの位置を示しているかを常に考えていく必要があります。今回のプログラムでは入れていませんが、初期状態からスタートさせる場合、特に【追加する】を使う場合、一度【すべてを削除する】を設定したほうが良いかと思います。

【リストに追加する】 場合を使う場合

定義 質問リストを作成

回数 を 0 にする

質問肉球リスト のすべてを削除する

回数 = 質問数 まで繰り返す

回数 を 1 ずつ変える

1 から 2 までの乱数 を 質問肉球リスト に追加する

質問の回数分、質問肉球リストに事前に登録
値 = 1 右前足 値 = 2 左前脚
質問肉球リスト [1] = 1 か 2 を乱数で代入
質問肉球リスト [2] = 1 か 2 を乱数で代入
質問肉球リスト [3] = 1 か 2 を乱数で代入
...
質問肉球リスト [質問数分] = 1 か 2 を乱数で代入

質問回数分の乱数がリストに設定ができましたので、次にアイボッサンに順番に足をあげてもらいます。

定義 左右のハイタッチを実行

回数 を 0 にする

回数 = 質問数 まで繰り返す

回数 を 1 ずつ変える

もし 質問肉球リスト の 回数 番目 = 1 なら

ラッキー が 右前足でハイタッチする

ラッキー が ほえる

でなければ

ラッキー が 左前足でハイタッチする

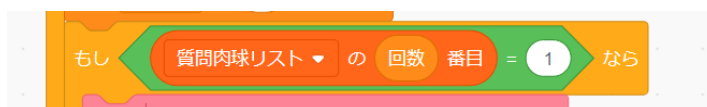
ラッキー が ほえる

ラッキー が 立つ

作成した質問に合わせて 前足でハイタッチを行う
ハイタッチの後に吠えて次の質問に行くことを教える
作成した質問肉球リストを順番に実行
質問肉球リスト [1] が 1 だったら右前足でハイタッチ
2 だったら左前足でハイタッチ
...
質問数分繰り返す

ここでは、【質問肉球リスト】に設定された値（乱数1か2）を順番に確認して、1だったら右前足でハイタッチ、2だったら左前足でハイタッチを行います。ハイタッチごとに【ほえる】をしていますので、あげたときをかわいく表現できるかと思います。

リストの内容を確認する方法として、次のようにしています。



先ほど設定した値を1つずつ確認するので、1番目から順番に確認できるように、リストの×番目を指定することで保存している内容の確認ができるようになります。

次にオーナさんの回答を肉球で確認しながらリストに保存していくブロックを確認していきましょう。定義ブロックとして【肉球チェック】を作っておきます。

The image shows a Scratch code editor with a custom block named '肉球チェック' (Meatball Check). The code is as follows:

```
定義 肉球チェック
ラッキー が 両前足をあげる
回数 を 0 にする
回数 = 質問数 まで繰り返す
回数 を 1 ずつ変える
押された肉球側 を 0 にする
押された肉球側 = 0 ではない まで繰り返す
もし ラッキー が 右前 足の肉球を押された なら
  押された肉球側 を 1 にする
もし ラッキー が 左前 足の肉球を押された なら
  押された肉球側 を 2 にする
肉球の押されたリスト の 回数 番目を 押された肉球側 で置き換える
ラッキー が ほえる
```

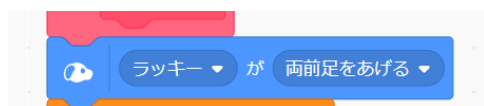
A tooltip window is open, providing instructions for the block:

作成した質問の回答を肉球を使って答えを教える

- ① 両前足を上げる と 答えるを求めてくる
- ② 出題された通りに肉球を使って答える
- ③ 答えを教えてあげると「ワン」と吠えるのでそこまで押し続ける
- ④ ワンと言ったら次にハイタッチした前足の肉球を押す

答えは「押されたん肉球側」のリストに順番に格納する
右前足の肉球が押されたら
肉球の押されたリスト [x] に 右側であれば= 1 左側であれば= 2
を入れる

オーナさんに回答を肉球で教えてもらうために、両前足を上げて待ちます。



繰り返し方法は、先ほどアイボさんの質問作成の時と同じように回数判定を行います。

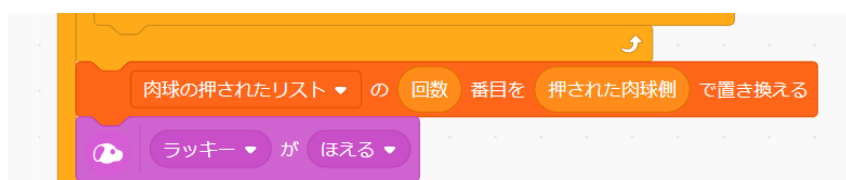
次に、アイボさんの前足の右・左前足の肉球のどちらが押されたかを判定します。もし〜で【右前足の肉球がおされた】または【左前足の肉球がおされた】を判断します。アイボさんの質問リストで作成した時の乱数の数字と合わせるため、右前足を押された場合は1、左前足の場合は2を変数【押された肉球側】に設定します。

この繰り返しから抜け出す条件として、変数【押された肉球側】が0以外となった場合としており、右前足（1）または左前足（2）が設定されるまで、この繰り返しが実行されます。



- ①変数【押された肉球側】を0
- ②0の間は繰り返し
- ③右前足が押されたら1に設定
- ④左前足が押されたら2に設定

右または左前足が押されたことを確認ができれば、リスト【押された肉球側】に設定を行います。そのあとにアイボさんが理解したことを表現するためにほえる動作を入れています。



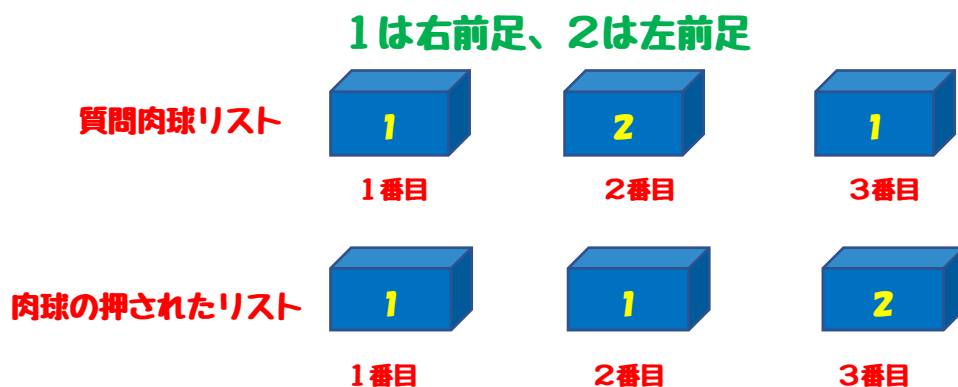
肉球が押された結果1または2をリストに書き換え方法で保存します。

ここまで2つの説明をしました。

処理1：アイボさんが出題をするために、乱数でリストに1または2の値を乱数で保存しておきます。
このリストを読みだして、1の場合は右前足を、2の場合は左前足上げる動作を質問数の回数だけ行います。

処理2：オーナに上げた順番を覚えてもらうために、両足を上げて肉球を押してもらいます。右前足を押されたら1を、左前足を押されたら2の値を、質問回数分、リストに保存します。

例えば、アイボさんが、【右】【左】【右】、オーナさんが【右】【右】【左】の場合、リストは次のようになります。



ここまでくると残りの処理は、質問と回答の答え合わせを行うだけとなります。このように数回勝負を行うような場合、リストにしていると非常に比較が簡単となります。

定義ブロックとして【答え合わせ】として以下の流れでチェックします。

The image shows a Scratch script for a function named "答え合わせ" (Answer Check). The script starts with a "定義" (Define) block. It sets "答え合わせ結果" (Answer Check Result) to 0 and "回数" (Count) to 0. A loop block sets "回数" to "質問数" (Number of Questions) and repeats the following steps: "回数" is incremented by 1. A conditional block checks if the "回数" (Count) of the selected question in the "質問肉球リスト" (Question Meatball List) is equal to the "回数" (Count) of the selected meatball in the "肉球の押されたリスト" (List of Pushed Meatballs). If not equal, "答え合わせ結果" is incremented by 1. After the loop, another conditional block checks if "答え合わせ結果" is 0. If true, the character "ラッキー" (Lucky) says "ととても喜ぶ" (I'm so happy). If false, the character says "悲しそうにする" (Looks sad) and then "悲しそうに鳴く" (Looks sad to cry).

質問したハイタッチした前足と回答を出題回数分チェックして
正解だと とっても喜ぶ 仕草をしてくれる
一つでも間違うと 左右に首を振って 悲しそうに鳴く

質問数の回数をくり返す方法は、これまでと同じように行います。

アイボさんの質問とオーナさんの押された肉球の比較方法はいたって簡単です。リストを使えば何番目を指定だけで確認ができます。

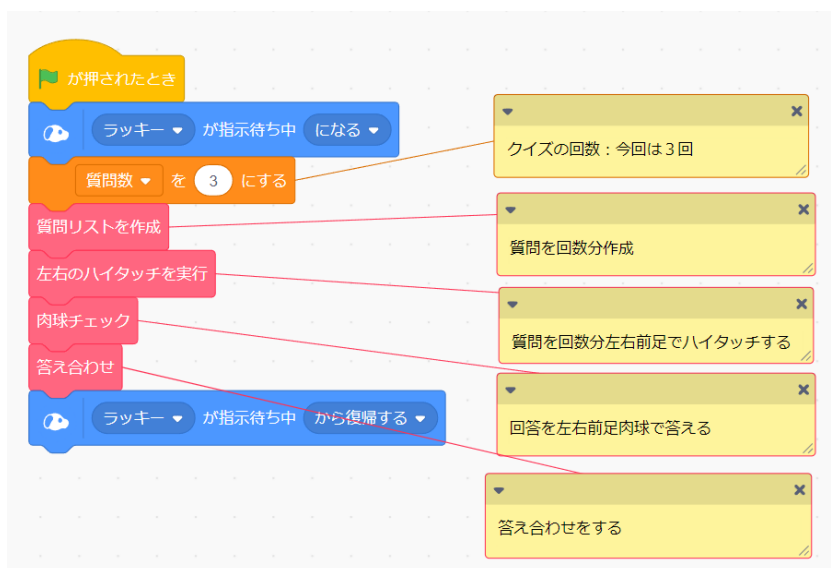
The image shows a close-up of the comparison block from the script: "もし" (If) followed by a green arrow block containing "質問肉球リスト" (Question Meatball List) の "回数" (Count) "番目" (Index) = "肉球の押されたリスト" (List of Pushed Meatballs) の "回数" (Count) "番目" (Index) "ではない" (is not) "なら" (then).

また、答え合わせの結果の判定方法として、今回のプログラムでは、不正解毎に変数【答え合わせの結果】を+1に足しこんでいます。全問チェックの仕方はいろいろとやり方があるかと思いますが。

ここで+1としたのは、間違いの回数分を後でアイボさんにほえて教えてもらうように使えるように足しこんでいます。改良などは、皆さんで工夫頂ければと思います。答え合わせの結果で喜んだり、悲しんだりして教えてください。



一か所だけ説明していなかったですね。ここまで説明してきた定義ブロックを順番に処理しているメイン処理です。この説明は必要ありませんね。



LESSON7のおさらいをしておきましょう。

- ① アイボさんと楽しめるゲームを作ってみました。
アイボさんが前足を使って、質問の回数分、右または左前足を順番に上げていくのでオーナさんはその順番を覚えます。
- ② 質問回数分完了したら、アイボさんは両足を上げますので、オーナさんは覚えた順番に肉球を使ってアイボさんに教えます。
- ③ 左右の前足の順番が全問合っていたら喜ぶ動作をします。もし間違っていたら悲しみます。
- ④ リストを2つ使ってアイボさんの出題するためのリストと、オーナさんがアイボさんに覚えてもらうためのリストを用意し、それぞれ順番に保存していきます。
- ⑤ 2つのリストを順番に比較して答え合わせを行います。
- ⑥ リストを使うとこのようなゲームの勝敗を簡単に比較することができます。

サンプルプログラムをダウンロードして実行してみてください。
複数のリストを使うとゲームなどの点数や勝敗を確認する処理がとても簡単にできるかと思います。是非、工夫して試してみてくださいね。