# Scratch 3.0で

# WeDo 2.0を動かそう!







Learning Systems

## 目

次

はじめに	
Scratchについて	1
WeDo 2.0を使う準備	
Scratch Linkをインストール	2
WeDo 2.0スマートハブをScratchと接続	10
プログラムをつくろう	
モーターを回してみよう	11
モータを使ったモデルを作ろう	13
センサーを使ったモデルを作ろう	18
「人を感知して動く扇風機(せんぷうき)」を作ってみよう!	18
「強さを調整できる扇風機(せんぷうき)」を作ってみよう!	28
扇風機モデル組み立て説明図	
扇風機組み立て図	2
速度調整つまみサンプル組み立て図	7

LEGO, the LEGO logo, and WeDo are trademarks of the LEGO Group. ©2019 the LEGO Group.

ScratchはMIT Media LabのLifelong Kindergarten Groupによって開発されました。 http://scratch.mit.eduを参照して下さい。

## はじめに

#### Scratch(スクラッチ)について

Scratch (スクラッチ)は、アメリカのMIT (マサチューセッツ工科大学)メディアラボのライフロン グキンダーガーデングループのプロジェクトで、無償で提供されているビジュアルプログラミング ツールです。命令の書かれたブロックを並べてプログラムを作り、自分で描いたキャラクターな どを動かしたりゲームを作ったりすることができます。制作した作品はオンラインコミュニティで世 界中のユーザーと共有することもできます。

バージョン2からはインストールの必要がないオンライン版となりWEBブラウザーさえあれば使用できるようになりました。さらに、Scratchだけではパソコンの画面の中だけで閉じていましたが、様々な電子デバイスとの連携が可能で、現実世界の物を動かすことができることが可能となり、作品作りの幅が広がります。

そのScratchと連携できる電子デバイスの中にLEGO® WeDo 2.0があります。パソコンの画面上のアニメーションやゲームとWeDo 2.0で作ったモデルを連携させて楽しい作品を作りましょう!

ScratchでWeDo 2.0を使用するための条件

- 1. PCがBluetooth 4.0に対応していること
- 2. OSが最新のサービスパックを適応したWindows 10またはMacOS Xであること
- 3. WEBブラウザーがEdge, Safari, Chrome, FireFoxなどの最新の物であること

8 scratch.mit.edu/pro	ojects/ed X +			
(←) → ℃ @	① ① A https://scratch.mit.edu/projects	/editor/?tutorial=getStarted	… 🛛 🗘 🔍 検索	<u>+</u> III 🗉 =
<b>6:14:16</b> -	ファイル 編集 🌾 チュート <b>リアル</b>			Scratchに参加しよう サインイン
📰 🗆 – F 🚽 :	コスチューム (1)音			
WeDo 2.0	ク−・ を 1 機能などす。 μ	50 0/#		
	ター・ をオンにする ター・ をオフにする	<ul> <li>(50)まで繰り返す</li> <li>をオンにする</li> </ul>		
	ター・ のパワーを 100 にす ター・ の方向を こちら向き	2		×
	の色を 50 にする <・ 50 のとき		27546	7=-3
	uvojas Calutza		スプライト 表示する ② Ø	↔ x         0         ‡ y         0           大⋶さ         向き
	にわめ)向き - に破いた - 方向の極き			1
=				



## WeDo 2.0を使う準備

Scratch Linkをインストール

ScratchでWeDo 2.0を使うために少々準備が必要です。 まずは、WEBブラウザーを起動してScratchのホームページを開きます。 次のURLをブラウザーのアドレス欄に入力するか、下のリンクをクリックします。

https://scratch.mit.edu/

Scratchのホームページ(下の図)が開きましたか?



SERVICE PES 見る アイデア Sci

「Scratchに参加しよう」を選択すると、Scratchアカウントが作成でき、サインインするとコミュニ ティに参加でき自分が作った作品を世界中のユーザーに見てもらうことができ、コメントももらえ るようになります。興味があったらアカウントを作成してみてください。プログラムを作るのはアカ ウントがなくても作れますし、パソコンにプログラムのファイルを保存したり開いたりできます。

Scratchに参加しよう サインイン



「作る」メニューをクリックすると、Scratchでプログラムを作る画面(下の図)に移動します。

チュートリアルが表示されているのを「閉じる」をクリックして閉じてしまいましょう。



チュートリアルは興味があったら見てみてください。ここでは、省略します。 次に、画面左下の青いボタン(拡張機能を追加)をクリックします。



追加できる拡張機能を選択するページ(下の図)が開きます。

2段目の一番右にEV3が表示されていますね。そうです、ScratchではEV3のプログラムも作れ ちゃうんです。



画面をスクロールしてもっと下の方を表示します。WeDo 2.0が表示されました。WeDo 2.0を選択

します。



デバイス(WeDo 2.0スマートハブ)を検索する(探す)画面が開くので「検索を開始」をクリック



デバイスの検索が始まりますが、エラーで次の様な画面が出ます。「ヘルプ」をクリックします。





ScratchでWeDo 2.0と接続するためのツール(プログラム)「Scratch Link」のダウンロードをするページ(下の図)が開きます。



画面をスクロールしてScratch Linkを「直接ダウンロード」をクリックしてダウンロードします。

ファイル(E) 編集(E) 表示(⊻) 履歴( ß scratch.mit.edu/projects/ed ×	<u>S)</u> ブックマーク( <u>B</u> ) ツール(I) ヘルプ( <u>H</u> ) <u>多</u> Scratch - WeDo 2.0 × +		
← → ⊂ ŵ	🛈 👽 🔒 https://scratch.mit.edu/wedo	・・・ 🖂 🌪 🔍 検索	
<u>e</u>	min 作る 見る アイデア Scratchについて Q 検	索 Scratchに参加しよう サインイン	^ 
OS&	選んでください: 🚝 Windows 💣 macOS		
-	しよう.	ください。ツールバーにあるはずです。	
	Get it from Microsoft		E
	Microsoft		
		يد م الله مي 2:39 PM برج م الله ه (بر مار) 2:39 PM برج م الله (بر مار) 2:39 PM	
t	あ、始めましょう		
We	Do 2.0をScratchに接続する		
U	2		
https://downloads.scratch.mit.edu/li	nk/windows.zip	My Blocks point In direction 60	-

リンクをクリックすると次の様な画面(下図はWindowsの場合)が出るので、ファイルを保存する をチェックしてから「OK」をクリックします。



ファイルを適当なフォルダに保存します。お使いのブラウザの設定によっては、下の様な画面が 出ずに自動的にあらかじめ決められたフォルダに保存されます。どこのフォルダに保存されるの かは、お使いのブラウザの設定をご確認ください

整理 ▼ 新しいフォ)	レダー			•
▲☆ お気に入り	名前	更新日時	種類	サイズ
■ デスクトップ	E 🔒 🖉	11 March 11	1.1.1.1.1	
🕘 最近表示した場所	🔰 🔒 kata miner Ja	in Principal 4	7 m 7 m	
🚺 ダウンロード		en de contra c	20 A.	
<b>Oreative</b> Cloud Fi		in faile of t	100 C - 100	
	and March			
▲ 詞 ライブラリ	Apple 1 Provide	a province.		
ト ドキュメント	Jan Harrison Contraction	and the second	2010/02	
▶ 📄 ピクチャ	🔒 i likan 🛛 🕹	de la fai de la secola de la se	$\mathcal{O}(\mathcal{O}(\mathcal{O}(\mathcal{O}(\mathcal{O}(\mathcal{O}(\mathcal{O}(\mathcal{O}($	
▶ 📑 ビデオ				
ファイル名(N): win	dows.zip	,		
ファイルの種類(T): ZIP	ファイル (*.zip)			
▲ フォルダーの非美示			保存(6)	キャン
				12

ダウンロードしたZIPファイルを展開します。

Windowsの場合

 1.展開したいファイルを右ク
リック
 2.表示されたメニューの中から
「すべて展開」を選択

展開したフォルダを開き、ScratchLinkSetup.msiファイルをダブルクリックしてインストールします。画面の表示に従って進めます。



Scracth Linkが起動したときに下の様な画面が出た場合は、「プライベートネットワーク」にチェッ クを入れて「アクセスを許可する」をクリックします。

	۲	このアプリ ています	の機能のいくこ	つかが Windows Defender ファイアウォールでブロックされ		
	すべてのパブリック ネットワークとプライベート ネットワークで、Windows Defender ファイアウォールにより Scratch Link の機能のいくつかがブロックされています。					
		<b>@</b>	名前(N):	Scratch Link		
	_		発行元(P):	不明		
チェックを入れ	る		パス(H):	C:¥program files¥scratch link¥scratchlink.exe		
	Scratch Li	nk にこれらの	ネットワーク上でのコ	通信を許可する:		
	<b>⊘</b> プ∋	iイベート ネット	∿ワ−ク (ホ−ム ネッ)	トワークや社内ネットワークなど)(R)		
	レル (この	ブリック ネットワ のようなネット!	フーク(空港、喫茶) フークは多くの場合。	店など) (非推奨)(U) 、セキュリティが低いかセキュリティが設定されていません)		
	アプリにファ	<u>イアウォールの</u>	経由を許可すること	との危険性の詳細		
				アクセアを許可する(A) キャンセル		

Scracth Linkが起動していることを確認します。

画面右下タスクバーの「ヘ」マークをクリックすると、バックグラウンドで動いているタスクが表示 されます。その中に「Scratch Link」があれば大丈夫です。





Scratch Linkが動作していることが確認できた

ここにScracth Linkが表示されていない場合 は、起動できていないのでメニューから起動 します(左図)。

これでScratchでWeDo 2.0スマートハブを接続する 準備ができました。

#### WeDo 2.0スマートハブをScratchと接続

WeDo 2.0スマートハブを検索する画面に戻ります。スマートハブに電池を入れて緑のボタンを押しながら、「検索を開始」をクリックします。



接続できると「ピロピロピロ」と音がして青色のLEDが点灯し、Scratchの画面に接続成功したことが表示されます(下の図)。

でしたした を は は た は た は た し た し た し た し た し た し た し た し た し た し た し た し た	? ヘルプ III LEGO Education WeDo 2.0 ×
またした をつう をつう をつう をつう をつう をつう をつう をつう	
接続されました	
接続されました	
	接続されました
切断 する エディターへ行く	<b>切</b> 筋する よう エディターへ行く
279	2774

## プログラムをつくろう

#### モーターを回してみよう

先ずは、動作を確認するためにスマートハブにモーターだけを接続して回転させてみましょう。

その前に、Scratchの画面について簡単に説明しておきます。

画面は左から順番に見ます。



①コマンドカテゴリー

プログラムのコマンド(命令)のカテゴリー(種類)を選びます

2コマンドブロック

選択したカテゴリー(種類)のコマンドブロック(命令ブロック)を選びます

#### ③コードキャンバス

コマンドブロック(命令ブロック)を並べてプログラムを作る場所です

④ステージ

「スプライト」と呼ばれるイラストを動かす舞台(ステージ)です

プログラムを作る時は、①カテゴリを選び ②コマンドを選んで ③キャンバスにドラッグ&ドロップ これを繰り返してコマンドブロックを並べて行きます。

#### Learning Systems

では、モーターを回転させてみましょう。

- WeDo 2.0カテゴリを選択
- ② 「モーターをオンにする」ブロックを選択
- ③ キャンバスにドラッグ&ドロップ



WeDo 2.0をえらんで・・・

今は、モーターを回転させるだけなので、これで終わりです。 では、プログラムを動かします。キャンバスに置いたブロックをダブルクリックして下さい。



どうなりましたか?モーターは回転していますか?モーターの回転を止めるにはステージの上 にある緑の旗の隣にある赤丸「●」をクリックします。



#### モーターを使ったモデルを作ってみよう

モーターを回転させることができるようになったので、モーターで動くモデルを作って、プログラム を作り動かしてみましょう!

「扇風機(せんぷうき)」を作ってみよう!



「扇風機(せんぷうき)」の組み立て図をみながら左図の様な扇 風機を組み立てて扇風機を回しましょう。

今回は、ブロックをダブルクリックするのではなくて別の方法でプログラムをスタートさせてみま しょう。ステージの左上にある「緑の旗」をクリックしたらプログラムがスタートするようにします。



今回は、ブロックをダブルクリックするのではなくて別の方法でプログラムをスタートさせてみましょう。ステージの左上にある「緑の旗」をクリックしたらプログラムがスタートするようにします。





#### 「自動的にとまる」ようにプログラムを作ってみよう!

扇風機を止めるのにPC画面で「●」をクリックするのも面倒ですね・・・一定の時間動いたら自動 的に止まるようにプログラムしてみましょう。

カテゴリで「制御(せいぎょ)」を選択します。一番上にある「1秒待つ」をドラッグしてきて、プログ ラムの一番下へドロップしてくっつけます。



次に、一定時間経ったら扇風機を止めたいので、モーターをオフにする必要がありますね。 カテゴリで「WeDo 2.0」を選び、「モーターをオフにする」をドラッグして、プログラムの一番下に並 べます(次頁の図)。



最後に、停まるまでの待ち時間が今は「1秒」になっています。1秒ではとても短すぎますので、 「1分」回してから止めるようにしたいと思います。「1秒待つ」ブロックの数字の部分をクリックして 「60」と変更します。



プログラムが出来上がったら動かしてみましょう。ステージ左上の「緑の旗」をクリックします。 扇風機が回り出し、1分たったら止まりましたか? 回っている時間を変えたりして何度かやってみましょう。 ー度止まった扇風機を動かすためにその都度「緑の旗」をクリックして動かすのも面倒ですね… 扇風機を再び回すのも止まった後に一定の時間がたってから回すことにしてみましょう。 今度は、自分で考えて作ってみてください。

どうですか?できましたか?

扇風機の動きを単純に考えてみると・・・

「回る」⇒「一定時間待つ」⇒「止める」⇒「一定時間待つ」⇒「回る」⇒「一定時間待つ」⇒「止める」・・・・となりますから、下の図のようなプログラムにすれば、「緑の旗」をクリックしなくても、止まった後に一定時間経つとまた自動的に扇風機は回り出しますね。

でも、ちょっとまって?これでは3回動いたら完全に止まってしまって、また「緑の旗」をクリックしないと動いてくれませんね。一度動かしたら(「緑の旗」をクリックしたら)ずっと「緑の旗」をクリックせずに自動的に動かし続けたい時はどうしましょう・・・?



同じ命令をもっとたくさん並べればいいでしょうか?それも一つの方法ですが、と~~~っても 長いプログラムになってしまって見にくいですね・・・

コンピューターのプログラムの一つの特徴として、同じプログラムを何度も繰り返して実行できる というものがあります。そのためにプログラムの命令の中には「ここから、ここまでを繰り返す」と いう命令も用意されています。もちろんScratchにも。 「ここからここまでを繰り返す」という命令を使って、永久に動き続けるプログラムに変えてみま しょう。

1回目の部分だけを残して、2回目、3回目の部分を消しておきます。

カテゴリから「制御(せいぎょ)」を選んで、その中から「ずっと」という命令をドラッグしてきます。 「ずっと」ブロックを「旗が押されたとき」と「モーターをオンにする」のブロックの間くらいにドラッグ してくると、「ずっと」ブロックのかげが自動的に繰り返す部分(ここからここまで)を決めてはさみ こみます。1回目の部分がかげにはさみこまれたらそこでドロップします。



プログラムでは同じ動きをする部分はこの様に「繰り返し」の命令を使って効率よく作ることが分 かりやすいプログラムを作るコツになりますすので覚えておきましょう!

60) 秒待つ

60) 秒待つ

をオフにする

#### センサーを使ったモデルを作ってみよう

次はセンサーを使って、モーターをコントロールするモデルを作って、プログラムを作り動かして みましょう!

#### 「人を感知して動く扇風機(せんぷうき)」を作ってみよう!

次は、人が来たことを感知して動き、人がいなくなると自動的に停まる扇風機を作りましょう。 扇風機に「人が来た」、「人がいなくなった」というのをどうしたら知らせることができるでしょう か?WeDo 2.0には、「モーションセンサー」というセンサーが入っています。「モーションセン サー」は、センサー正面から15cm以内に何か物がある時に、距離はどれくらいか(近いか、遠 いか)を測ることのできるセンサーです。

では、センサーがどのような値をコンピュータに伝えているのかを調べてみましょう。ステージに 表示されているネコにセンサーの値をしゃべらせます。ネコに何かをしゃべらせるには、「見た 目」カテゴリーの「(こんにちは!)と言う」ブロックを使います。



「旗」をクリックして、プログラムを実行してみましょう。 ネコが「こんにちは!」と言っています。 (こんにちは!)の部分をクリックして、他の言葉に変えて実行 してみましょう。ネコがしゃべる言葉が変わりましたね。 ここにセンサーから伝わる値を入れればネコがセンサーの値 をしゃべってくれます(画面で値を確認できます)。



Learning Systems

扇風機にモーションセンサーを取り付けま しょう。羽根の回転の邪魔にならないよう に、人が来たことを感じやすいように工夫 してセンサーを取り付けて下さい。



モーションセンサーの値をネコにしゃべらせます。WeDo 2.0カテゴリを選択します。「距離(きょり)」ブロックをドラッグして来て、(こんにちは!)の部分にはめ込み(ドロップし)ます。



WeDo 2.0カテゴリを選ぶ

プログラムが完成したら「旗」をクリックしてプログラムを実行 してみましょう。ネコが「100」と言ってますので、モーション センサーの前に何もないか、十分に遠ければセンサーの値 は「100」であることが分かります。

センサーの前に何か物を置いて近づけてみましょう。ネコが しゃべっている数字が小さくなりました。

モーションセンサーは、十分遠い時は100と言う値を出し、物との距離が小さいほど小さな値を出すことが分かりました。

それも、100から10単位で変化しているようです。一番小さい値は0ですから物までの距離を0~100までの10段階で近さを表しているということですね。

近さを表す値がどれくらいなのかを調べればそこに物がある かどうか、人がいるのかどうかということが調べられます。 では、人が近づいたら回るようにプログラムを変えましょう。



「~と言う」ブロックを「ずっと繰り返す」ブロック の中から取り出して削除してしまいます。



「ずっと繰り返す」ブロックをドラッグして「旗が押されたとき」ブロックと離します。 離れた「旗が押されたとき」ブロックを削除します。



「ずっと繰り返す」ブロックをドラッグして「旗が押されたとき」ブロックと離します。 離れた「旗が押されたとき」ブロックを削除します。 「WeDo 2.0」カテゴリを選択して、「距離く50のとき」イベントブロックをドラッグしてきて、「ずっと 繰り返す」ブロックの上にドロップします。







最後に、「距離<50のとき」イベントブロックの「5 0」の部分をクリックして人が近づいたと感じる距離 を調整します。

モーションセンサーの値がここで設定した数値より も小さくなったとき

「距離(モーションセンサーの値) く設定値」 となったときに「人が近づいた」と判断させることに なります。



そこで、距離をネコにしゃべらせた時のことを思い出して、または同じことをやってみて、どのくらいの値になった時に近づいたとするかを決めて、それよりも大きな数字を設定します。例では、「30」よりも小さい値になれば十分近づいたことになるとして、「30」を設定しています。

このような値のことを「しきい値」と言います。

しきい値が設定したら、扇風機にMiniFig(レゴ人形)を近づけてみましょう、思ったところまで近づ いたら扇風機が回り出せば成功です!



人が近づいたら自動で回転し始める扇風機を作ることに阿h成功しましたが、離れても止まりま せん・・・もっと便利な扇風機にするために、人がいなくなったら(離れたら)回転を止めるようにし ましょう。どうしたらよいか・・・少し自分で考えてみてください。



どうですか?考えられましたか?

モーターをオンにした後、人が離れるまでそのまま待っていて、人が離れたらモーターをオフにすればよさそうです。

「制御」カテゴリから、「~まで待つ」ブロックをドラッグして来て、「モーターをオンにする」の下にドロップします。



「~まで待つ」の「~まで」が「人が離れるまで」と言う条件になります。では、人が離れたかどう かはどうやって知ることができるでしょうか? 人が近づいてきたのをどうやって知ることができたかを思い出して下さい。

# 【自分の考えを書いてみよう】

そうです、モーションセンサーの距離の値がある値(しきい値)より小さくなった時に人が近づいたことにしたわけですから、今度はその反対ですね。

「モーションセンサーの距離の値>しきい値」

となったときが人が離れた時とすればよいわけです。

「演算」カテゴリを選択して「〇>50」(比較ブロック:大なり)をドラッグして来て、「~まで待つ」ブロックの条件部分(空欄部分)にドロップします。



比較ブロックの左側の空欄(「>」記号の左側)に「距離ブロック」をドラッグ&ドロップします。 しきい値を調整します(例では30に設定しています)。



人が離れたら回転を止めるので、モーターをオ フにするブロックをその下にドロップします。



さあ、これで完成!・・・?この状態で実行してみましょう。回転は止まりましたか? 止まりませんね。なぜでしょう?

【自分の考えを書いてみよう】

プログラムは「ずっと繰り返す」となっているので、モーターをオフにしてもすぐに先頭に戻って、 モーターがオンになってしまうので止まらないのですね。では、どうしたらよいでしょう?

自分の考えを書いてみよう】	

モーターが回転するときは、人が近づいている時ですから、オフにしたらまた人が近づいてくるまで待っていればよいですね。

再び人が近づくのを待って回転させるために、「~まで待つ」ブロックをドロップして、条件部分に 「人が近づいたら」の条件を作ります。



では、実行させてみましょう。扇風機にMiniFig(レゴ人形)を近づけると回転が始まります。 MiniFigを遠ざけると・・・・止まりましたか?

再び近づけると・・・・回転が始まりましたか?

成功ですね!!これで人が居る時に回転して、居ないと自動で止まるスマート扇風機が完成し ました!!

このプログラムを作った考え方を整理しておきましょう。

① 人が値がづいたら(センサーの距離が30より小さくなったらプログラムを実行する

- ② モーターを回転させる
- ③ 人が離れる(センサーの距離が30より大きくなる)まで待つ
- ④ 人が離れたら(センサーの距離が30より大きくなったら)モーターを止める
- ⑤ 人が近づく(センサーの距離が30より小さくなる)まで待つ
- 2へ戻って繰り返す

という考え方で作りました。

実は、違う考え方でも同じ動作をするプログラムを作ることができます。その考え方と言うのは、 次の様な考え方です。

もし、人がいる(センサーの距離が30より小さい)なら

モーターを回転させる

でなければ

モーターを止める

これをずっと繰り返す

という考え方です。

では、この考え方でプログラムを作ってみましょう!

「旗が押されたとき」ブロックと「ずっと繰り返す」ブロックをドロップしておきます。 「制御」カテゴリから「もし~なら でなければ」ブロックをドラッグして、「ずっと繰り返す」ブロック の中にドロップします。



次に、「もし~(条件)なら」の部分の条件を作ります。条件は

「モーションセンサーの距離くしきい値」 でしたね。しきい値は自分で適当に設定して下さい。 もし 1日 距離 < 30 なら

「もし距離くしきい値なら」の下は、人がいる時なの で、モーターを回転させます。 「モーターをオンにする」ブロックをドロップします。



プログラムを実行してみましょう!

MiniFig(レゴ人形)を近づけると・・・回転しますね。 遠ざけると・・・・止まりましたね。 前のプログラムとは違う考え方でも同じ動作をするプログラムを作ることができました。

プログラムやその作り方には「正解」はありません。色々な考え方があるからで す。自分が実現したい動作をどうすれば実現できるかを考えること、どんな命令を どう並べれば実現できるか(アルゴリズム)、プログラムの流れ(フロー)を考えるこ とがプログラムを作る(プログラミングする)ということです。実際にプログラムを作 る(書く)作業のことを「コーディング(Coding)」とも言います。「コーディング」は「プ ログラミング」の作業の一部でしかなく、あまり重要な部分ではありません。プログ ラミングで一番重要なのは「プログラムの流れフロー」を考えるところです。フロー が間違えていれば思った通りの動作は実現できません。どんな条件でどんな動作 をさせるのか・・・どんな計算をどんな順番ですればいいのか・・・よ~く考えて、自 分なりのフローを作ってプログラミングを楽しんで下さい!!

WeDo 2.0にはもう一つセンサーが入っています、「チルトセンサー(傾きセンサー)」です。つぎ は、このチルトセンサーを使ってみましょう。 WeDo 2.0にはもう一つセンサーが入っています、「チルトセンサー(傾きセンサー)」です。つぎ は、このチルトセンサーを使ってみましょう。

#### 「強さを調整できる扇風機(せんぷうき)」を作ってみよう!

チルトセンサーを風力調整つまみとして使います。右に一回傾けると1段階強くなり、左に傾けると弱くなるようにします。Oページからの組み立て図を見ながら、風力調整つまみを組み立て て扇風機に取り付けましょう。



組み立てられましたか?では、まずは傾きセンサーを左右に傾けた時、ネコが傾いた方向を しゃべるようにして見ましょう。プログラムの考え方(流れ;フロー)を考えていきましょう。 センサーが左に傾いたなら「左」と言う、右なら「右」と言うプログラムを作りたいので・・・ 簡単ですね・・・このようになります。

もし センサーが左に傾いた なら

ネコが「左」としゃべる

でなければ

もし センサーが右に傾いた なら

ネコが「右」としゃべる

よく見ると、もうすでにスクラッチのプログラムと同 じになっていますね。では、プログラムを作って実 行してみましょう。



どうでしょうか?つまみを右に回すと「右」と言いましたか?左に回すと「左」と言いましたか?





逆でしたね。つまみを「右」に回すと、ネコは「左」と言いました。人の側(操作する側)から見ると 右でしたが、センサー側からみれば左だったのです。機械を動かす様なプログラムを作る時に は、機械側に立った考え方も必要になります。さて、ではネコの言っている「右」と「左」を逆にし てもう一度確認しましょう。



今度は「(人間側からみた)右」につまみを回すと、ネコも右と言いましたね。 これで、つまみを右に回した時のプログラム、左に回した時のプログラムをどこに作ればいいか わかりました。では、それぞれの場合のプログラムの流れを考えていきましょう。

もし 左に傾いた なら

モーターのパワーを10上げる

でなければ

もし 右に傾いた なら

モー<mark>ターのパワーを10下げる</mark>

でよさそうですね。モーターのパワーを設定するブロックを見てみましょう。



パワーの値を指定して設定することが できるようです。 次に、「パワーを10上げる」「パワーを10さげる」を実現するにはどうすればよいででしょう? 「パワーを10上げる」とは、つまり

今設定されているパワーの値+10を新たにパワーの値として設定する 「パワーを10下げる」とは、

今設定されているパワーの値-10を新たにパワーの値として設定する と言うことですね。「パワーの値」はチルトセンサーが傾くたびに変更される数と言うことになりま す。プログラムの中でこのように計算によって変更されるデータを扱う仕組みとして、「変数」と言 うものがあります。Scratchでももちろんこの変数の仕組みが用意されているのでそれを使って パワーの計算をします。



モーターのパワーは0~100の間で設定できます。そこで、初期設定値として中間の50をプロ グラムの初めに設定しておきます。





次に、左に傾いた(つまみを右に回した)時に パワーを+10上げる部分を作ります。 変数「パワーを〇にする(変数に値を設定する ブロック)」を持ってきます。 演算(えんざん)カテゴリをクリックし、「〇十〇」ブロックを、変数「パワーを〇にする(変数に値を 設定するブロック)」の「〇」の中にドロップします。



「〇十〇」ブロックの左側の「〇」の中にに、変数「パワー」をドロップして、右側をクリックして 「10」を入力します。



同様に、「もし右に傾いたなら パワーを-10する」部分を作ります。



これでつまみを回した時に変数「パワー」の値の設定ができたので、その変数の値をモーターパ ワーの値に設定し、モーターをオンにします。



さて、これで「速度変更つまみを右に回すと早く回るようになり、左に回すとゆっくりになる扇風 機」のプログラムができました、さっそく動かしてみましょう!

うまく思うように動きましたか?モーターのパワーの値は0~100までの値で設定できますので、 ずっと右回していてもある程度の早さ以上にはなりません(100以上を設定しても意味がなく、1 00で打ち止めになる)し、ある程度左に回しているとモーターの速度はゆっくりになり変化しなく なります(0以下を設定しても意味がなく、0で打ち止め。でも止まりません)。



扇風機を題材にして、単純に回転するだけの扇風機、人が近づいたら動き出し、離れていった ら止まる扇風機、そして回転速度を調整することのできる扇風機をモーター、モーションセン サー、チルトセンサーを使って作ってきました。私たちの身近な機械にはこのようにロボットの技 術がたくさん使われています。そして、それをプログラムするためには、どのような動きにしたい のかをよく考えて機械への命令の流れを考えプログラムする必要があります。そして、プログラ ムの流れの基本(プログラムの基本構造)はこれまでにやってきた、「順番に」、「繰り返し」、「も し~なら、でなければ(条件分岐)」の組み合わせでしかありません。この3つのプログラムの基 本構造をしっかりと理解しうまく流れを作ることがプログラミングのコツです。 では、WeDo 2.0を使って色々な機械やロボットを作ってプログラミングして下さい!!

以上でWeDo 2.0+Scratch 3.0プログラミングガイドVol.1を終わります。

# 扇風機(せんぷうき)モデル 組み立て説明図





## 扇風機(せんぷうき)組み立て図















## 速度調整つまみサンプル組み立て図

### 1.





















## 11.







 Scratch 3.0でWeDo 2.0を動かそう! ~レゴ<sub>®</sub>WeDo 2.0 Scratch 3.0プログラミングガイド~
 編著 株式会社ラーニングシステム
 初版発行 2019年5月24日
 発行人 小倉 康司
 発行所 株式会社ラーニングシステム 〒220-0012
 神奈川県 横浜市 西区みなとみらい2-3-2 みなとみらい東急スクエア① 4F TEL 045-232-4391 FAX 045-232-4392
 https://www.mdstorm.com/

LEGO, the LEGO logo, and WeDo are trademarks of the LEGO Group. ©2019 the LEGO Group.

ScratchはMIT Media LabのLifelong Kindergarten Groupによって開発されました。 http://scratch.mit.eduを参照して下さい。 本書中の製品名、ソフトウェア名およびブランド名は各社、団体の商品および登録商標です。 本書の内容を無断で複製使用することを禁じます。

