

夏休み自由研究 スクラッチ教科書プログラミング

(Ver. 0.9 2018/08/01)

夏休みの自由研究などを想定したプログラミングヒント集です。

「No.1 スクラッチ教科書プログラミング」は、国語・算数・理科・社会・図工・英語・総合的な学習の時間(この資料では“総合”と略します)などの、教科書に載っているような内容でプログラミング又はプログラミングしながらレポートなどを作成するものです。

[プログラミング内容 (1)]

1. 物語を作ろう。 [国語・社会・総合]
デジタル絵本やデジタル絵日記を作りましょう。
2. 写真クイズ。 [理科・社会]
自分でとった写真を使って、クイズを作りましょう。
3. 飛び出す地図。 [社会・総合]
近所のようなすを紹介する面白いデジタル地図を作りましょう。
4. 写真アニメ。 [図工]
沢山の写真をつなげてアニメーション作りましょう。
5. 三角形・多角形。 [算数]
三角形や多角形をプログラミングして、その性質を考えましょう。
6. 防犯ライト。 [理科]
防犯ライトの仕組みから、自動的に動くものを作りましょう。
7. 物の分類。 [理科]
物の特徴に注目して、いろいろなものを分類・判断してみましょう。
8. 速さシミュレーション。 [算数]
プログラミングで「速さ」とは何か考えていきましょう。
9. 空気の粒シミュレーション。 [理科]
プログラミングで「空気の粒」の動きを考えていきましょう。



[プログラミング内容 (2)]

10. 都道府県を比べよう。 [社会]

統計データをもとに、都道府県を比べてみよう

11. スクラッチ動物園を救え。 [図工] (「Why!プログラミング」より)

自分だけの動物園を作ってみよう。

12. 自分だけの楽器をつくれ。 [音楽] (「Why!プログラミング」より)

いろいろな楽しい楽器を作ってみよう。最後は自動演奏に挑戦。

13. AI 英会話。 [英語]

コンピュータが自動的に会話するプログラムを作ってみよう。

[共通テクニック]

- スマホから写真を取り込もう。写真を加工しよう。
- リストを使おう。

[参考: スクラッチ スタジオ]

スクラッチ内の「夏休み自由研究」スタジオ内に、もともになるプログラムや参考プログラムが入っています。必要におうじてアクセスしてみてください。



(検索したら、スタジオを見てね)



(スタジオには完成版のプログラムも多くありますが、できるだけ見ないでやってみよう。わからなくなったら見てもいいかも)

1.物語を作ろう。

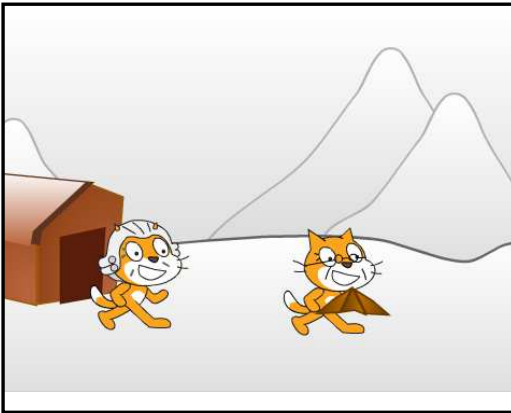
([国語・社会・総合]) レベル:2~6年生 (初級)

スクラッチを使っていろいろな、物語をデジタルで作ってみよう。

例:

- ・夏休みの思い出のデジタル絵日記
- ・昔話などのデジタル絵本
- ・歴史の一場面の紹介
- ・学校や住んでいる場所の紹介
- ・自分で考えた物語のデジタル絵本

[参考プログラム]

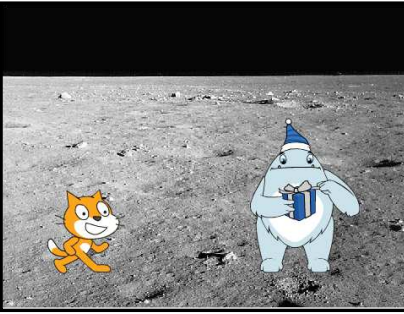


プログラム名: 物語

昔話の「かさじぞう」をデジタル絵本にしたプログラムです。

- ・参考プログラムの中身を見て、物語をスクラッチでどんなテクニックを使っているか確認してみてください。
- ・次のカードからは、「物語」プログラムで使っている基本的なテクニックを簡単なプログラムを作りながら説明しています。

物語用の基本テクニック



・プログラムを作りながら確認しましょう。

ステップ 1: キーを押すと、話したり、動いたり
ネコ(Sprite1)のスク립ト



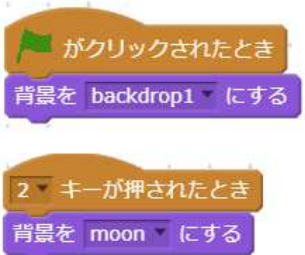
旗をクリックして開始した時は
隠れます。

1 のキーが押されると、表示さ
れ、言葉を言った後に、移動し
ます。

ステップ 2: キーを押すと背景が変わる。

・新しい背景を追加します。(例では、moon)

背景のスク립ト

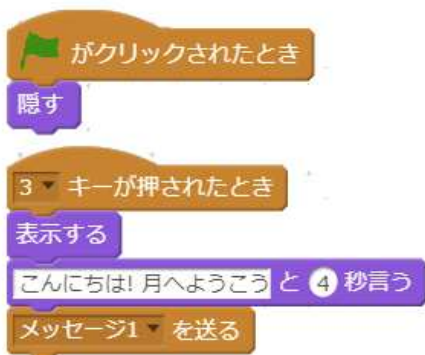


旗をクリックして開始した時の背景を指定し
ます。

2 のキーが押された時の、背景を指定しま
す。

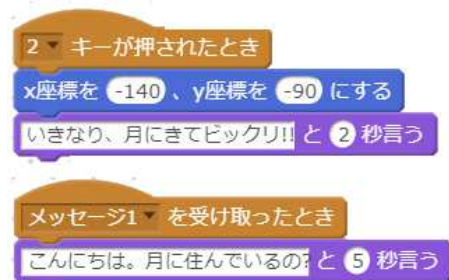
ステップ3: 新しいスプライトの追加と、会話のやりとり。

- ・新しいスプライトを追加します。(例では、Creature1)
- 新しいスプライト(Creature1)のスク립ト



旗をクリックして開始した時は隠れます。
3のキーが押されると、表示され、言葉を言った後に、メッセージを送ります。

ネコ(Sprite1)へのスク립トの追加



2のキーが押されると、言葉を言います。

メッセージを受け取ると、言葉を言います。

[参考プログラム(完成版)] プログラム名: 物語基本

チャレンジ：起承転結



[参考プログラム] プログラム名：起承転結 テクニック

- ・話が自動的に進むようにしてみよう。
- ・クリックすると話が進むようにしてみよう。
- ・いろんな、仕掛けを入れてみよう。

話の構成

- ・オリジナルでしっかりした構成の物語を作ってみよう。

多くの物語は、起承転結の4つの部分から作られています。

起：話の始まり。ここで興味をひく

承：話の深めて核心につなげる部分。

転：話の中心場面。

結：話の結末がどうなったのか。まとめ。

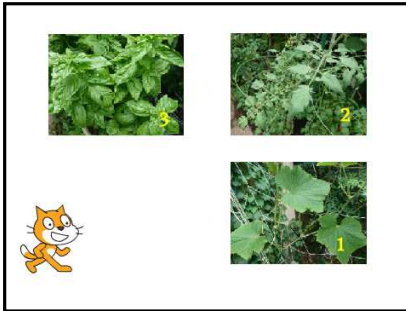
2.写真クイズ。

([理科・社会]) レベル:2~6年生 (初級)

いろいろな写真をとって、3択クイズを作ってみよう。

参考プログラムを見て、自分の写真に変えてみよう。

[参考プログラム] プログラム名: 写真クイズ

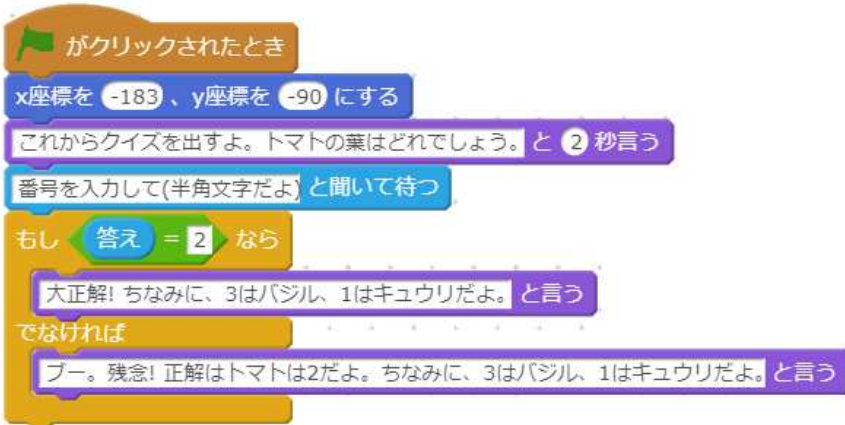


スプライトとして、3枚の写真を使っています。

ネコのスプライトが問題を出して回答を判断しています。

(写真の撮り込み方は、[共通テクニック]a.スマホから写真を取り込もう。写真を加工しよう。)

ネコ(Sprite1)のスク립ト



チャレンジ:

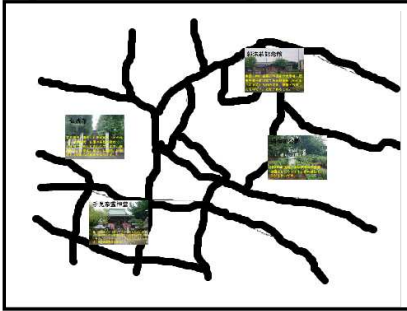
・問題を増やして、みよう。この場合、写真を入れ替えたりする必要があるよ。(複数の背景に問題の写真の写真を予めいれておくと、プログラム自体は簡単かもしれない)

3. 飛び出す地図。

([理科・社会]) レベル:2~6年生 (初級)

近所のデジタル地図を作ってみよう。自分で近所の写真を撮ったり、調べて作ろう。参考プログラムは写真が飛び出す地図の例です。

[参考プログラム] プログラム名: 飛び出す地図



プログラムの動き:
地図上の写真をクリックすると、その拡大表示されます。
地図の道は背景に書いてあるよ。
(写真の撮り込み方は、[共通テクニック]a.スマホから写真を取り込もう。写真を加工しよう。)

各写真(スプライト)のスク립ト

がクリックされたとき
x座標を **-102**、y座標を **-78** にする
大きさを **15** % にする

旗をクリックして開始した時は、縮小して地図の特定の場所に表紙します。

このスプライトがクリックされたとき
x座標を **-2**、y座標を **-9** にする
大きさを **100** % にする
スペース キーが押された まで待つ
x座標を **-102**、y座標を **-78** にする
大きさを **15** % にする

スプライトがクリックされると拡大して、画面いっぱいに表示されます。
スペースキーが押されるまで待ちます。
もとの位置に縮小して表示します。

チャレンジ:

- ・名所マップ以外に、防災・危険な場所マップなど作ってみよう。
- ・いろいろな仕組みをマップに組み込んでみよう。

4.写真アニメ。

([理科・社会]) レベル:2~6年生 (初級)

静止画の写真を連続表紙してアニメーションを作ってみよう。

参考プログラムを見て、プログラムは簡単だよ。

[参考プログラム] プログラム名: 写真アニメ



ホチキスが口を開いたり、閉じたりしながら移動します。

背景のスク립ト



([0.2秒待つ]の待ち時間を調整してみてください)

チャレンジ:

- ・写真にいろいろ、絵を描き加えてみよう。
- ・いろいろな物を使ってみよう。
- ・粘土を使うと、本格的なクレイアニメになるよ。Youtubeにあるクレイアニメも参考にしな。

写真アニメの基本テクニック

固定したカメラやスマホで、写真をたくさんとって使います。写真の撮り方には次のような方法があります。

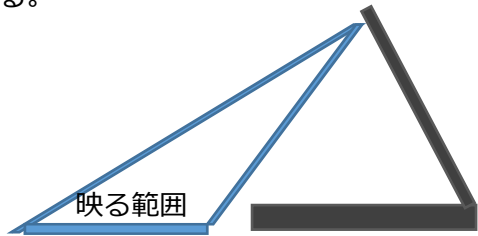
方法 1: カメラやスマホでとってパソコンに持ってくる。

三脚などでカメラやスマホを固定します。

(写真の撮り込み方は、[共通テクニック]a.スマホから写真を取り込もう。写真を加工しよう。)

方法 2: PC 内蔵のカメラを使用する。

右の図は、ノートパソコンを半開きにした状態です。パソコンの画面の上にはほうについている内蔵カメラでもとることができます。



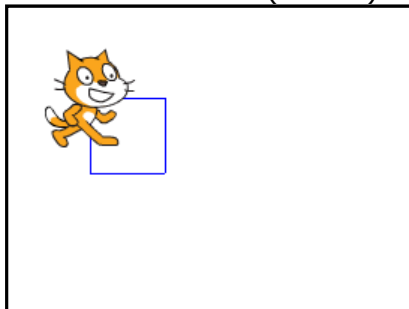
PC 内蔵カメラでのうつし方は、書籍「小学生からはじめるわくわくプログラミング 2」に詳しく出ていますので、そちらも参考にしてください。

5. 三角形・多角形。

([算数]) レベル:2~6年生 (初級), 教科書:5年生

正方形(四角形)、三角形、多角形をプログラミングしながら、これらの図形の性質について考えましょう。そして、それらについてレポートにまとめましょう。

ステップ 1: 正方形(四角形)を書く



正方形を描くプログラムは、もう作った人も多いかもしれないね。

[参考プログラム 1] プログラム名: 四角形

ステップ 2: 三角形を絵が

今度は三角形を描くプログラムを作ってみよう。

[参考プログラム(完成版)] プログラム名: 三角形

ステップ 3: 五角形、六角形、・・・を描く

さらに、五角形、六角形、・・・を描くプログラムをつくってみよう。

ステップ 4: 法則を見つけて、 x 角形を描く

x 角形を描く場合の法則を見つけてみよう。

見つけたら、 x を入力して x 角形を描くプログラムを作ってみよう。

ヒント: すこし難しいけど、「外角と内角」が多角形を描くヒントになるよ Webなどで検索してみよう。

[参考プログラム(完成版)] プログラム名: 多角形

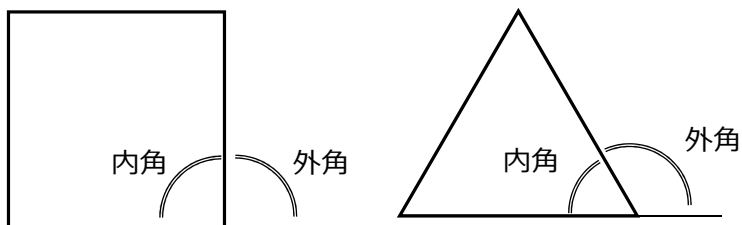
レポートのまとめ方

x 角形の描く法則を中心にステップ 1 から 4 をレポートにしよう。

チャレンジ:

・多角形を参考に、色や大きさも変えて、綺麗な模様を作ってみよう。

ヒント： 外角と内角

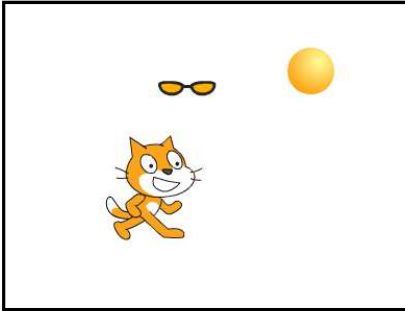


多角形を描くとき、外角の合計はいくつになるか考えてみよう。
(三角形の場合は、内角・外角はそれぞれ3個。四角形の場合は、それぞれ4個になります)

6. 防犯ライト。

([理科]) レベル:4~6年生 (中級), 教科書:6年生

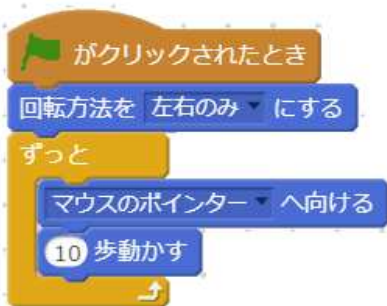
人が近づくと、自動的にライトが点灯する防犯ライトをスクラッチ上で再現してみます。防犯ライトのプログラムを作った後に、そこで使ったセンサーの機能を使って、いろいろなプログラムを作ってみましょう。



センサー(めがね)にネコが近づくと、ライトがつくよ。

・プログラムを作りながら確認しましょう。

ステップ 1: ネコのスクリプト



おなじみのプログラムで、ネコがマウスを追いかけて移動します。

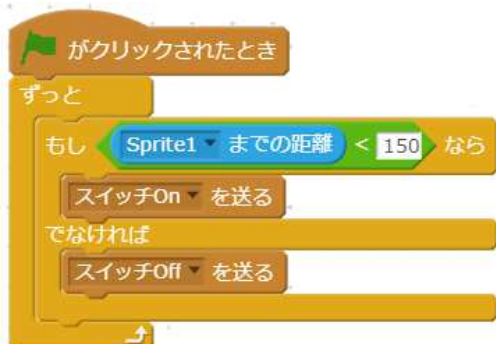
補足: フィジカルコンピューティングに挑戦。

PCの画面上のプログラミングだけでなく、実際にセンサーを使って、電気を点けるなどしてみよう。

「自由研究プログラミング No.3 micro:bit プログラミング」も見てね

ステップ 2: センターのスプライトの追加

ネコを検出するセンサーのスプライトを追加します。例ではメガネのスプライトを追加しています。



ネコとの距離を判断しています。
スイッチ on/off をメッセージとして出しています。

ステップ 3: ライトのスプライトの追加

ライトのスプライトを追加します。電気が点いた状態と消えた状態の2種類のコスチュームが必要です。例では、ボールで、一つのコスチュームを黒くして使用しています。



メッセージを受け取った時に、コスチュームを変更して、電気が点いたり消えたりを表現しています。

[参考プログラム(完成版)] プログラム名: 防犯ライト

チャレンジ:

- ・本物の防犯ライトに動作を近づけてみよう(近づくと、ある一定時間電気がついていますよね)
- ・このようなセンサーを使った身の回りのものを作ってみよう。
- ・「3.飛び出す地図」にセンサーの仕組みをいれてみよう。

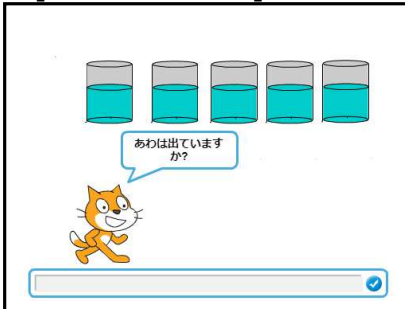
7. 物の分類。

([理科]) レベル:4~6年生(上級), 教科書:5・6年生

物の特徴に注目して、いろいろなものを分類・判断してみましょう。プログラムとしては、質問していくと何か判断するものです。

例 1: 水溶液の判断

[参考プログラム] プログラム名: 水溶液分類



プログラムの中を見て、どのように判断しているか確認しましょう。

チャレンジ:

- ・他の、いろいろな水溶液の組み合わせを判断できるようにしてみよう。
- ・他の物(例えば、動物や植物)を分類するプログラムを作ってみよう。

ヒント:

プログラムを作る前に、分類する時の特徴などを表にまとめておくと分かりやすいよ。レポートを作る場合も、この表をのせましょう。

例:

	あわ	溶ける:鉄	溶ける:アルミニウム	蒸発:白い粉
炭酸水	○			
薄い塩酸		○		
薄い水酸化ナトリウム水溶液			○	
食塩水				○
普通の水				

例 2: 雲の判断

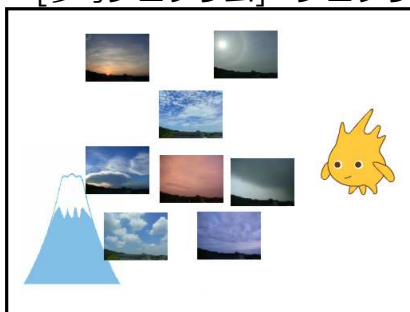
[参考プログラム] プログラム名: 雲(テキスト)

1. 直線状やかぎ状の雲です
2. モクモクとわきあがった雲です
3. 暗く全天をおおう雲です
4. いくつかの塊からできる雲です
5. 層になった、または霧のような雲です



背景に分類の質問を表示しています。
(背景は別のソフトで作成しました)

[参考プログラム] プログラム名: 雲(写真)



写真を使った分類・判断しています。
(雲の上下の位置を実際に雲が見える高さにする工夫をしています)

チャレンジ:

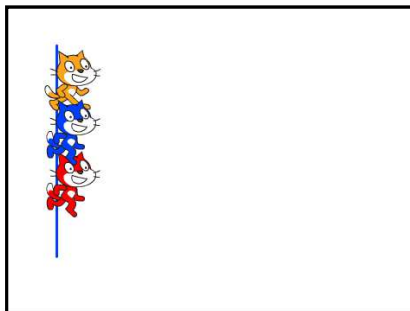
- ・自分でとった写真や、グラフィック、文字を組み合わせて、雲の分類のプログラムを作ってみよう。
- ・他の物(例えば、動物や植物)を分類するプログラムを、写真、グラフィック、文字で質問して作ってみよう。

8. 速さシミュレーション。

([算数]) レベル:3~6年生 (初級~上級), 教科書:5年生

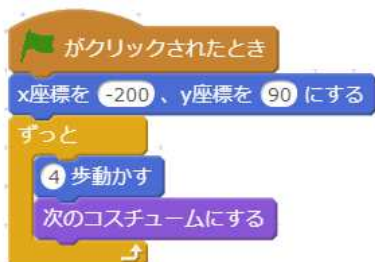
プログラミングしながら「速さ」とは何か考えていきます。プログラムを作りながら、それを動かし見つけた内容をレポートしてみましよう。

実験 1: 動いている時間

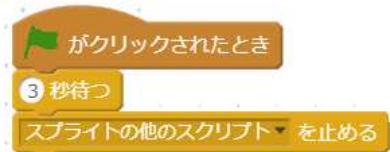


スタートラインに、3つのスプライトを並べて配置します。例では、色違いのネコのスプライトを3つ使用してます。

スプライトのスク립ト



旗をクリックして開始した時は、スタートラインの場所について、ずっと横に移動します。

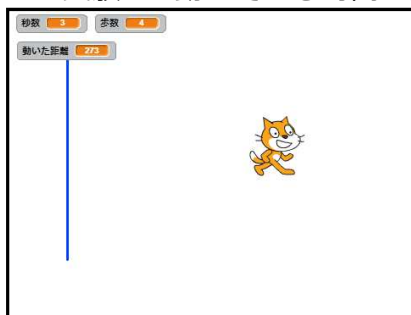


指定秒数たったら(左の例では、3秒)、動きを止めます。3つのスプライトでこの秒数は違った数を設定してます。

・[()秒待つ]の指定した秒数の違いでネコはどこまで進みますか? 秒数の値と進んだ距離はどんな関係にあるでしょう。

[参考プログラム(完成版)] プログラム名: 速度 01

実験 2: 動いている時間・歩数・歩いた距離



一匹のネコを移動します。
プログラムでは、移動する秒数と歩数を入力しています。
変数の内容も表示しています。

スプライトのスク립ト



[参考プログラム(完成版)] プログラム名: 速度 02

レポートのまとめ方

歩数・秒数と移動した距離の関係についてまとめてみよう。速度とは中に考え見よう。例えば、一秒間に移動する距離を計算してみよう。

チャレンジ:

- ・歩数・秒数・移動した距離。一秒に移動する距離などを、表やグラフにしてレポートにしてみよう。
- ・歩数・秒数・移動した距離などをグラフで表す、スクラッチのプログラムを作ってみよう。

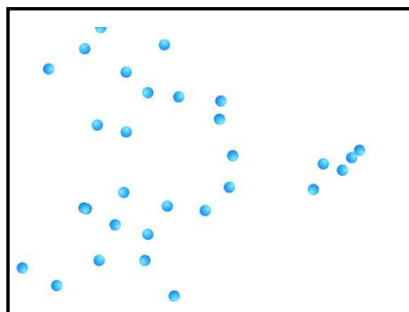
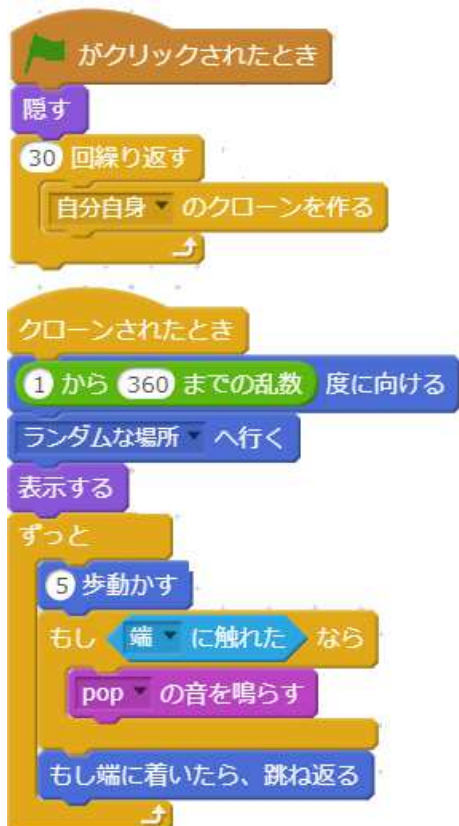
[参考プログラム(完成版)] プログラム名: 速度 03

9. 空気の粒シミュレーション。

([算数]) レベル:4~6年生 (中級~上級), 教科書:4年生

プログラミングしながら「空気の粒」の動きを考えていきましょう。プログラムを作りながら、それを動かし見つけた内容をレポートしてみましょう。

実験 1: 単純な空気の粒の動き



Ball のスプライトを空気の粒と考えています。

クローンを使って、空気の粒を 30 個作っています。

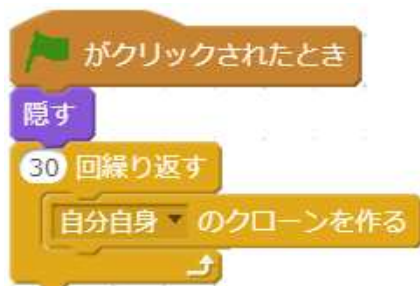
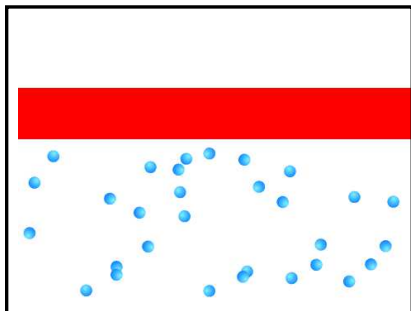
個々の ball のスプライトは、ランダムな位置に表示された後、動きます。端に触れたら、音が出ます。

- ・空気の粒とはどのように動くか考えてみましょう。
- ・プログラムの[()歩く]の歩数の変えることは、どんな意味を持つか考えてみましょう。

[参考プログラム(完成版)] プログラム名: 空気の粒 01

実験 2: 空気の粒の動く空間の大きさを変える

ball(空気の粒)のスク립ト



赤い横棒のSprite

・赤い横棒のSpriteを作成します。

赤い横棒のスク립ト



[参考プログラム(完成版)] プログラム名: 空気の粒 02

・赤い横棒を上下に動かして、ball(空気の粒)の動きがどうなるか考えてみましょう。

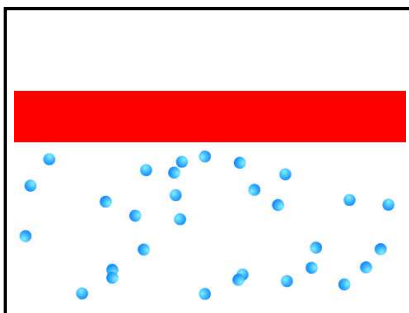
チャレンジ 1:

・端や、赤い横棒に、どのぐらい ball(空気の粒)の粒が当たったか、カウントできるようにしてみよう。カウントするためには、変数の追加がひつようだよ。

[参考プログラム(完成版)] プログラム名: 空気の粒 03

実験 3: 空気の粒が当たると力が働く

チャレンジ 2:



強い風があると、ものが倒れたりします。空気の粒があたると押す力が働きます。プログラ身で、空気の粒があたった時に力が働く状態を再現してみましょう。赤い棒に ball(空気の粒)が当たった場合に少し上に移動するようにします。

[参考プログラム(完成版)] プログラム名: 空気の粒 04

実験 4: 空気の粒が当たっても、赤い棒の位置を一定にする

チャレンジ 3:

チャレンジ 2 で作ったプログラムは、どんどん赤い棒が上に行ってしまうのが、矢印キーで下に戻してやる必要がありました。自動的に、赤い棒を下に下げないようにして、空気の粒があたっても、一定の位置に赤い棒があるようなプログラムを作ってみよう。

[参考プログラム(完成版)] プログラム名: 空気の粒 05

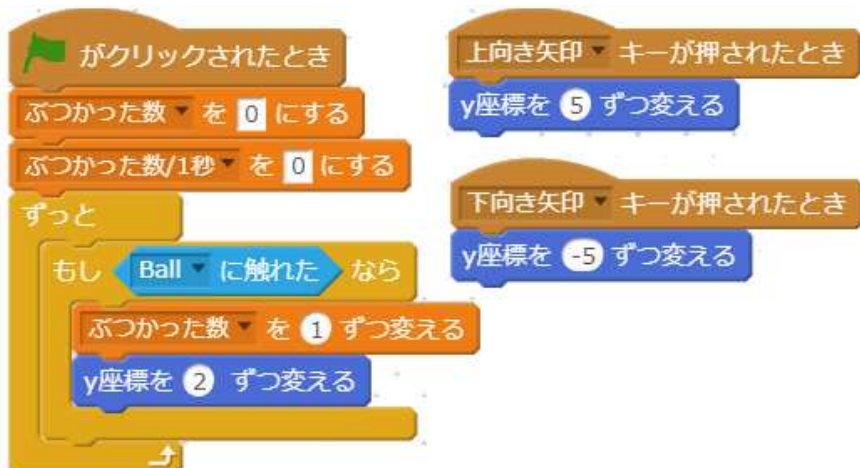
レポートのまとめ方

次のような視点からレポートをまとめてみよう。

- ・空気の粒が遅く動いたり、早く動いたりするのはどんなことか。
- ・空気が端や赤い棒に当たる回数が増えるのはどんなことか。
- ・赤い棒を一定の位置にとどめておくのは、実際にはどんな現象と対応するか。

ヒント:

[参考プログラム(完成版)] プログラム名: 空気の粒 04 の内容
赤い棒のスク립ト



Scratch script for the red stick:

- When clicked (green flag icon)
- Set hit count to 0
- Set hit count per second to 0
- Forever loop:
 - When up arrow key is pressed:
 - Change y-coordinate by 5
 - When down arrow key is pressed:
 - Change y-coordinate by -5
 - When Ball is touched:
 - Change hit count by 1
 - Change y-coordinate by 2



Scratch script for the red stick:

- When clicked (green flag icon)
- Forever loop:
 - Wait 1 second
 - Set hit count per second to hit count
 - Set hit count to 0

10. 都道府県を比べよう。

([社会]) レベル:3~6年生(上級), 教科書:4・5年生

あらかじめ都道府県の統計情報の入ったプログラミングをリミックスして、都道府県の情報をスクラッチで比較してみよう。

統計情報の入ったプログラムの利用方法:

[元になるプログラム] プログラム名: 都道府県統計データ

都道府県 可住地面積:h a	人口 森林面積:h a	65歳以上人口 年平均気温:℃	総面積:h a 降水量(年間):mm
総生産額:百万 総生産額(卸売・小	総生産額(農業):百万 総生産額(情報通信)	総生産額(林業):百万 総生産額(サービス業)	総生産額(水産業):百万円 総生産額(製造業):百万円
労働力人口	第1次産業就業者数	第2次産業就業者数	第3次産業就業者数
平均余命(男)	平均余命(女)	病院数	保育所数
1 75.17	1 86.3	1 566	1 827
2 77.26	2 85.34	2 97	2 465
3 78.53	3 85.86	3 91	3 361
4 79.65	4 86.39	4 141	4 370
5 78.22	5 85.93	5 70	5 261
6 79.97	6 86.28	6 68	6 248
7 78.84	7 86.05	7 127	7 310
8 79.09	8 85.83	8 179	8 531
+ 長さ: 47	+ 長さ: 47	+ 長さ: 47	+ 長さ: 47

「都道府県統計データ」には、いくつかの都道府県別の統計データがリストに入っています。

リミックスした後、必要な情報だけ残したり表示したりして利用しましょう。

(リストについては、[共通テクニック]b. リストを使おう。を参考にしてください。)

The screenshot shows a Scratch project window. On the left, there is a list of 47 prefectures with various statistics. On the right, the 'Scripts' area shows a 'Data' block selected, and the 'Costumes' area shows a list of variables and objects, including '都道府県' (Prefecture) which is checked.

リスト名のチェックを外すことで、表示しなくなります。

またリスト自体を右クリックして削除してもいいです。

チャレンジ 1:

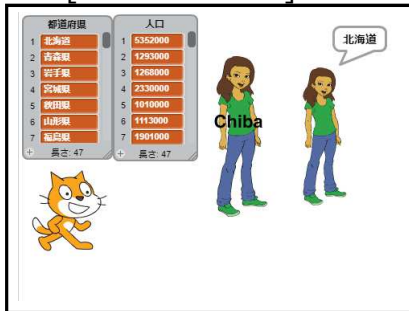
- 都道府県の違いを、スクラッチを使って表現してみよう。

チャレンジ 2:

- 都道府県の違いを、スクラッチを使ってグラフで表現してみよう。

サンプルプログラム:

[参考プログラム] プログラム名: 都道府県人口比較



都道府県の人口を比較して、千葉県に対してどのぐらいの大きさになるか絵の大きさを示しています。

Abby2(各都道府県の大きさを表現する)のスク립ト



11. スクラッチ動物園を救え。 NHK 「Why!プログラミング」より ([図工]) レベル:2~6 年生 (初級から中級)

NHK のスクラッチプログラミング番組の「Why!プログラミング」を見ながら(Web でみ見れるよ)、自分の楽しい動物園をつくってみよう。



(作者 NHK_for_School)

URL:

http://www.nhk.or.jp/sougou/programming/?das_id=D0005180310_00000

“Why!プログラミング”で検索して、「スクラッチ動物園を救え」を選んでもいいよ。

「スクラッチ動物園を救え」で使うプログラムのスタジオ:

スタジオ名: Why_programming_9(作者 NHK_for_School)

チャレンジ:

- ・自分だけの xxx 園、xxxx 博物館を作ってみよう。

12. 自分だけの楽器をつくれ。 NHK 「Why!プログラミング」より ([音楽]) レベル:2~6年生 (初級から中級)

NHKのスクラッチプログラミング番組の「Why!プログラミング」を見ながら(Webでみられるよ)、自分の楽しい楽器をつくってみよう。



(作者 NHK_for_School)

URL:

http://www.nhk.or.jp/sougou/programming/?das_id=D0005180311_00000

“Why!プログラミング”で検索して、「自分だけの楽器をつくれ」を選んでもいいよ。

「スクラッチ動物園を救え」で使うプログラムのスタジオ:

スタジオ名: Why_programming_10(作者 NHK_for_School)

チャレンジ:

- ・作った楽器に自動演奏されてみよう。

13. AI 英会話。

([英語]) レベル:4~6 年生 (上級), 教科書:5・6 年生

スクラッチで人工知能的に自然な英会話をパソコンとするプログラムを作ってみよう。

ベースになるプログラムの利用方法:

[元になるプログラム] プログラム名: AI 英会話



英文の文集を単語に分けて、リストに格納するカスタムブロックが定義されています。

(リストについては、[共通テクニック]b. リストを使おう。を参考にしてください。)

チャレンジ:

・元になるプログラムを参考に、スクラッチのプログラムが会話を初めて、できるだけ自然な会話が続くようなプログラムをつくってみよう。

ヒント:

質問に対して、どんな答えが想定して入力されるか考えてみよう。そして、それが、どのような語順でどのような単語になるか考えてみよう。いろいろな入力を想定すると、自然な会話になるよ。

英文を単語にわけけるカスタムブロックの説明:



分解したい英文を引数として「文字を単語にする」定義ブロックを呼び出します。



例えば:

“I like cats”を引数として渡すと。リストには、下に示すように格納されます。



[共通的なテクニック]

a.スマホから写真を取り込もう。写真を加工しよう。

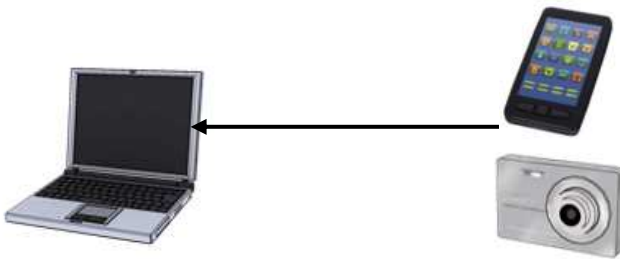
接続方法 1: クラウドストレージを使用する



GoogleDrive(アンドロイド), iCloud(iPhone)などのクラウドストレージを経由して、スマホの写真をパソコンに転送します。

パソコン側では、ブラウザで各クラウドストレージにアクセスすれば写真を取り出せます。

接続方法 2: 直接ケーブルで接続する



デジカメやスマホをケーブルで直接パソコンに接続します。

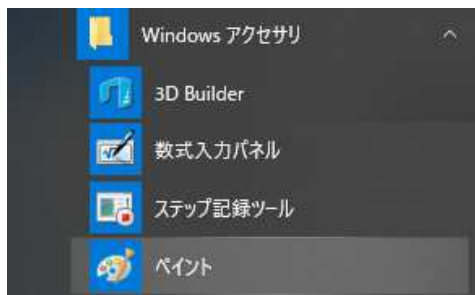
通常 DCIM フォルダの中に写真が入っています。

(スマホで接続する場合、スマホ側で PC への接続許可を行う操作が必要になります)

写真の加工方法:

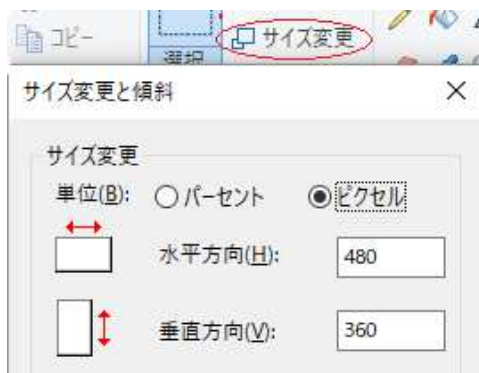
現在使っている Scratch Ver2.0 では、スクラッチの中でグラフィックに日本語を入れることができません。また、写真をそのままスクラッチで使おうとすると大きすぎる場合があります。そんなときは、事前にパソコンで写真を加工したりすると便利です。

使用のお絵かきソフト(Windows の場合):



初めから入っている Windows アクセサリの中のペイントを使うのが簡単です。

○Scratch 用のサイズに変更



スクラッチの背景のサイズ(スクラッチの実行時の画面の大きさ)は 480x360 ピクセルです。あらかじめこの大きさに変更しておく、プログラムで使い易いです。

○文字の入力やその他の加工

ペイントのソフトの中で文字を入力した、加工します。使い方は、スクラッチで描画するときと似ています。

[共通的なテクニック]

b. リストを使おう。

プログラムの中で、数や文字を扱う時に変数を使った人も多いと思います。たくさんの数や文字を使う場合は、リストを使った方がプログラムが簡単になります。

変数とリストの違い:

変数A 32

変数もリストも名前を付けて、複数の変数やリストをプログラムの中で使うことができます。

リストA
1 45
2 32
3 71
+ 長さ: 3

変数は一つの数や文字列を格納することができます。

これに対して、リストは一つのリストの中に複数の数や文字列を持つことができます。この場合、リストの1番目のデータ、2番目のデータなどで区別します。

リストに対する処理:

(数・文字列) を リストA に追加する

1番目 (リストA) を (数・文字列) で置き換える

1番目 (リストA)

リストA の長さ

1番目を リストA から削除する

すべて 番目を リストA から削除する

リストについては、次のような使い方ができます。

- ・データの追加 最後に追加する、
- ・データの置き換え 番号を指定して内容を置き換える
- ・データの取り出し 番号を指定して取り出す
- ・リストの長さ リストに入っているデータの個数を取り出します。
- ・削除 特定の番号やすべてのデータを削除します。

次のリストを使ったプログラムの中身を見て、リストの使い方を確認しましょう。

リストを利用したプログラム例 1:

[参考プログラム] プログラム名: リスト_会話



あらかじめ、リストに会話の内容をいれておいて、ランダムにリストの中の番号を選択しています。

リストを利用したプログラム例 2:

[参考プログラム] プログラム名: リスト_電卓



数を入力して行って、リストの中に記録していきます。最後に全部の数の合計を求めています。