

スクラッチで猫を動かしてみよう

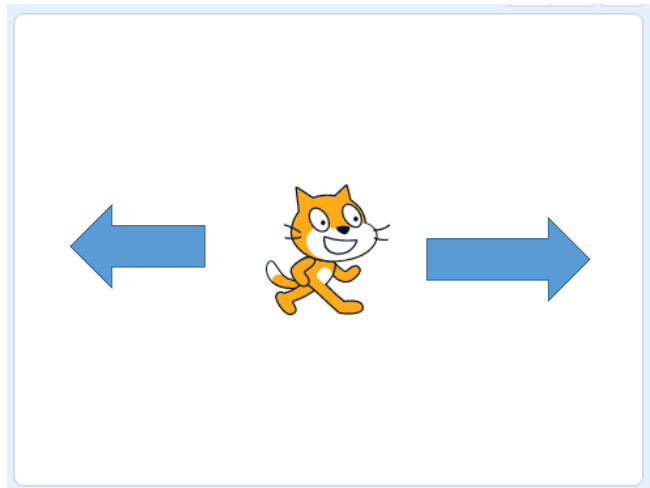
1

最初にスクラッチを開くと、猫がいるよ。



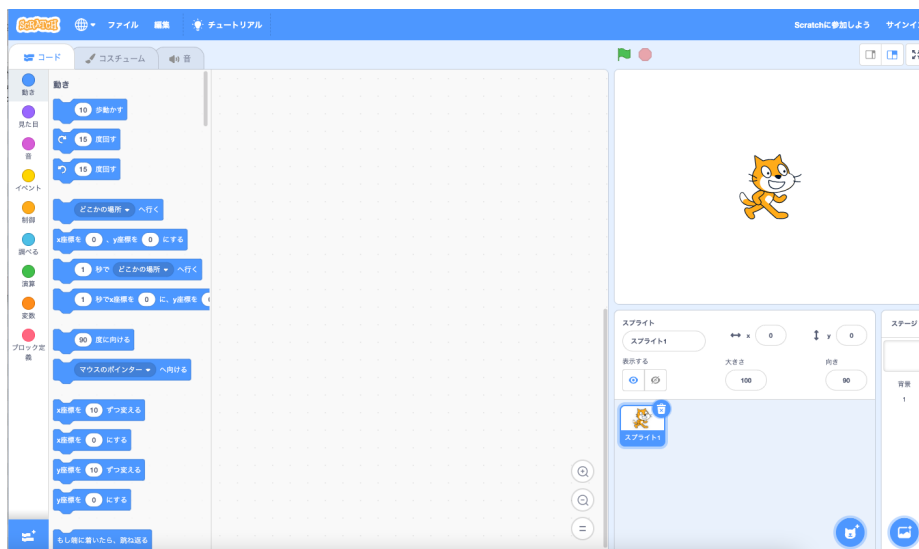
2

この猫を左右にずっと動かしたい！

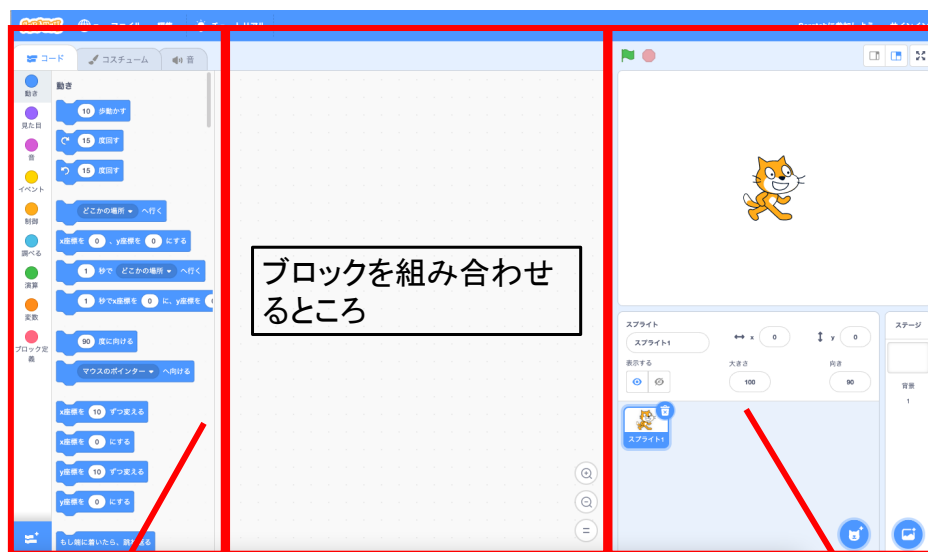


3

まずは画面の説明から！



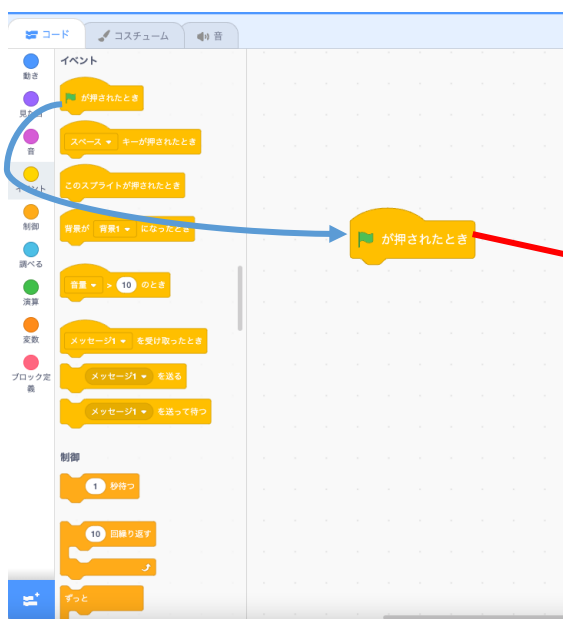
4



動きなどブロックを置いてあるところ

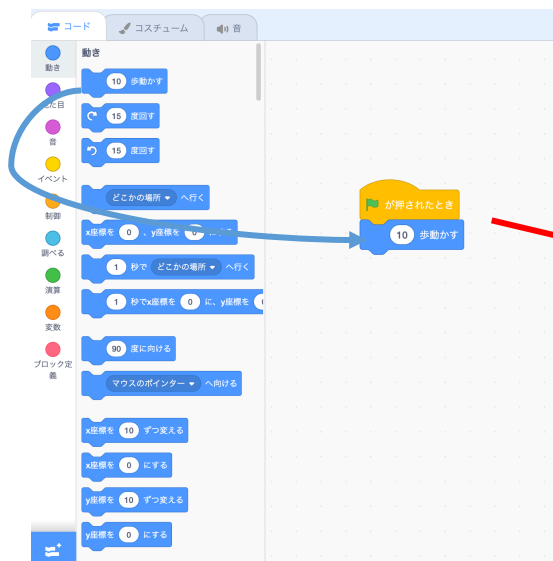
見た目を確認できるところ

5



「緑旗が押されたとき」をドラッグして、まんなかのエリアに持って行く

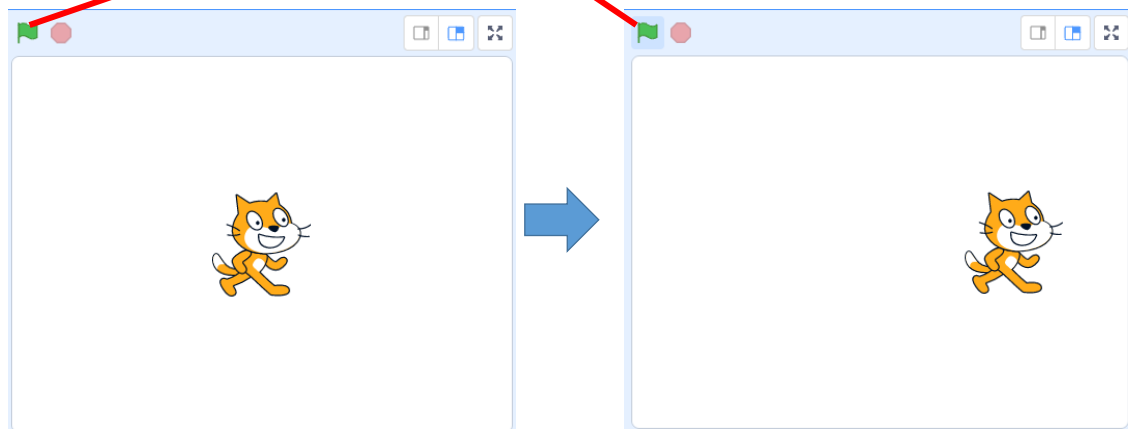
6



「10歩動かす」をドラッグして、
まんなかのエリアに持っていき
「緑旗が押されたとき」の下にくっつける

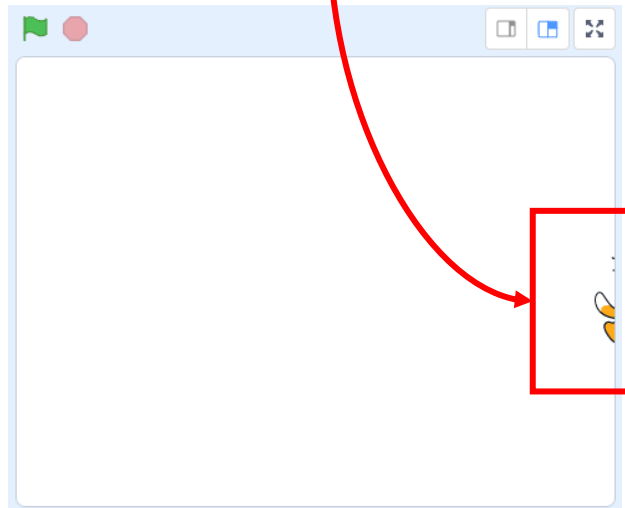
7

旗を押すとうごくように！



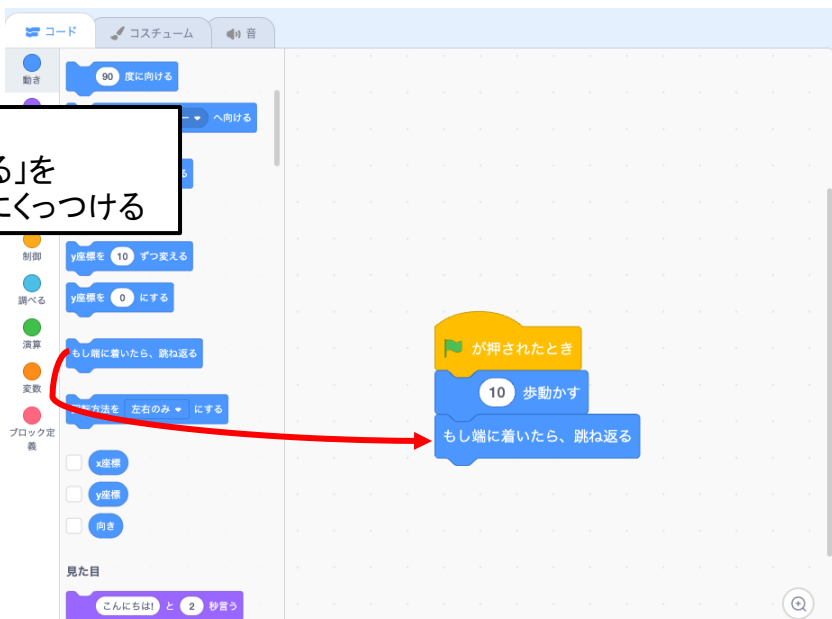
8

猫が端までいくと、はねかえらずに埋まってしまう。。。



9

動きのブロックの中から、
「もし端に着いたら、跳ね返る」を
「10歩動かす」ブロックの下にくっつける



10

ねこが端までいくと、跳ね返るようになった！
けど、方向といっしょに猫も逆さまに。



11

反対になっている人は、
猫の向きを直すために、「向き」を90に変更



12

「動き」から「回転方法を左右のみにする」ブロックをくっつける

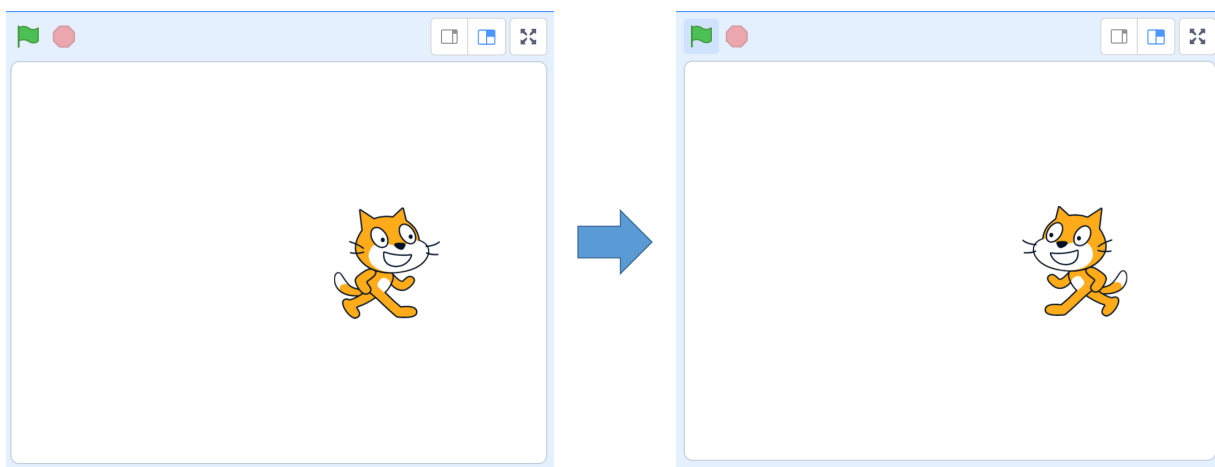
The screenshot shows the Scratch code editor interface. On the left, the 'Code' palette is visible with various block categories. The main workspace contains a script area with the following blocks:

- When green flag clicked (yellow)
- Move 10 steps (blue)
- If touching edge, bounce back (blue)
- Rotation method: Left and Right (blue)

A red arrow points from the 'rotation method' dropdown in the 'If touching edge, bounce back' block to the 'rotation method' dropdown in the 'When green flag clicked' block, indicating the process of copying the block.

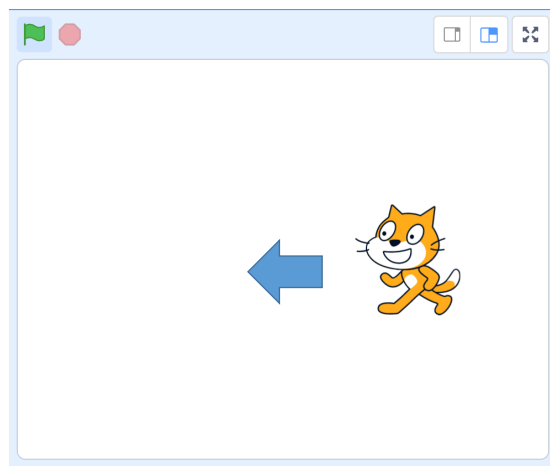
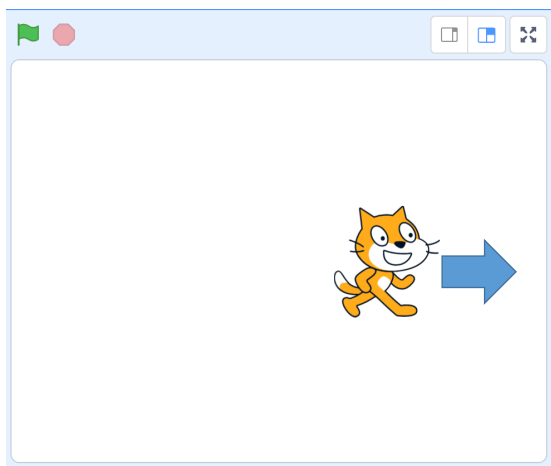
13

回転の方法を制限するブロックを足したことにより、猫が逆さまにならないようになった！



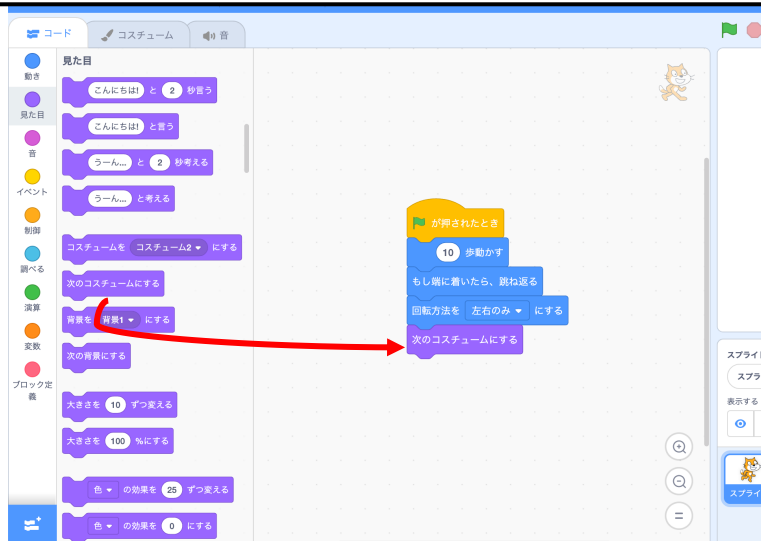
14

今はねこが滑って動いているように見えるので、歩いているように見せるブロックを足します。



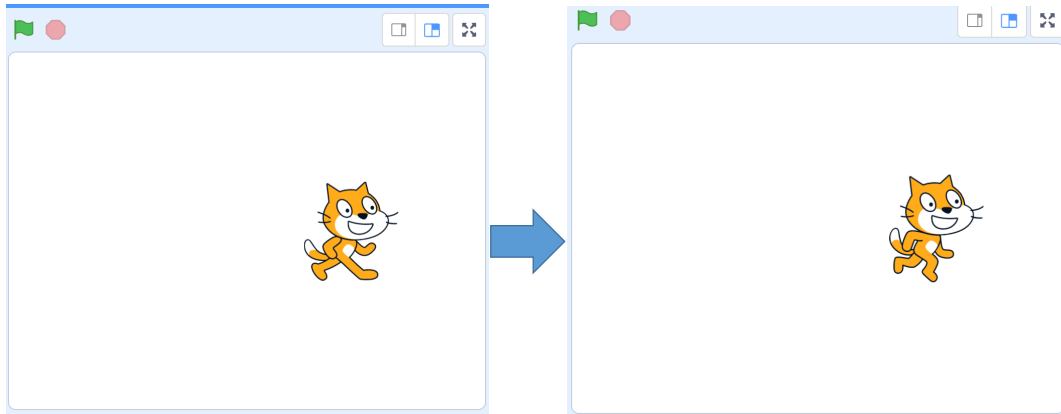
15

見た目のブロックから「次のコスチュームにする」をつける。



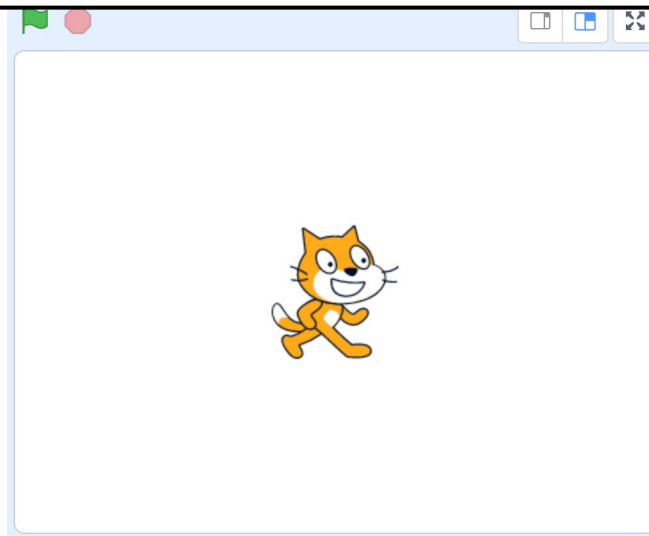
16

旗を押すたびに次のコスチュームになり、
ねこが歩いているような動きに！



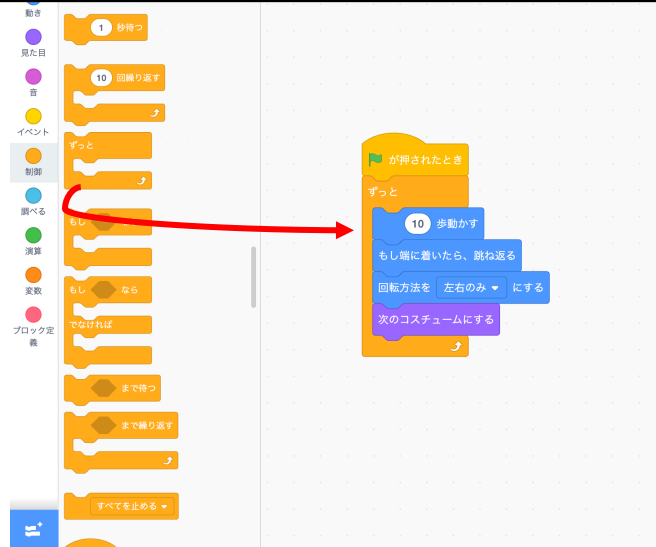
17

旗をずっとクリックするのも疲れるので、
一回旗を押したらずっと動いてくれるようにする。



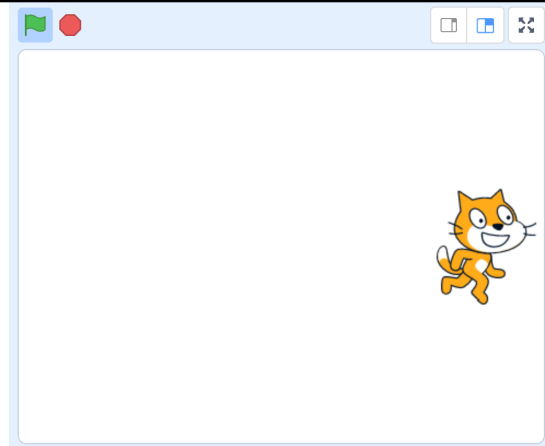
18

制御のブロック群から「ずっと」のブロックを
いままでのブロックに食べさせるようにつける。



19

猫が赤い丸(止める)ボタンを押すまで
動くように！



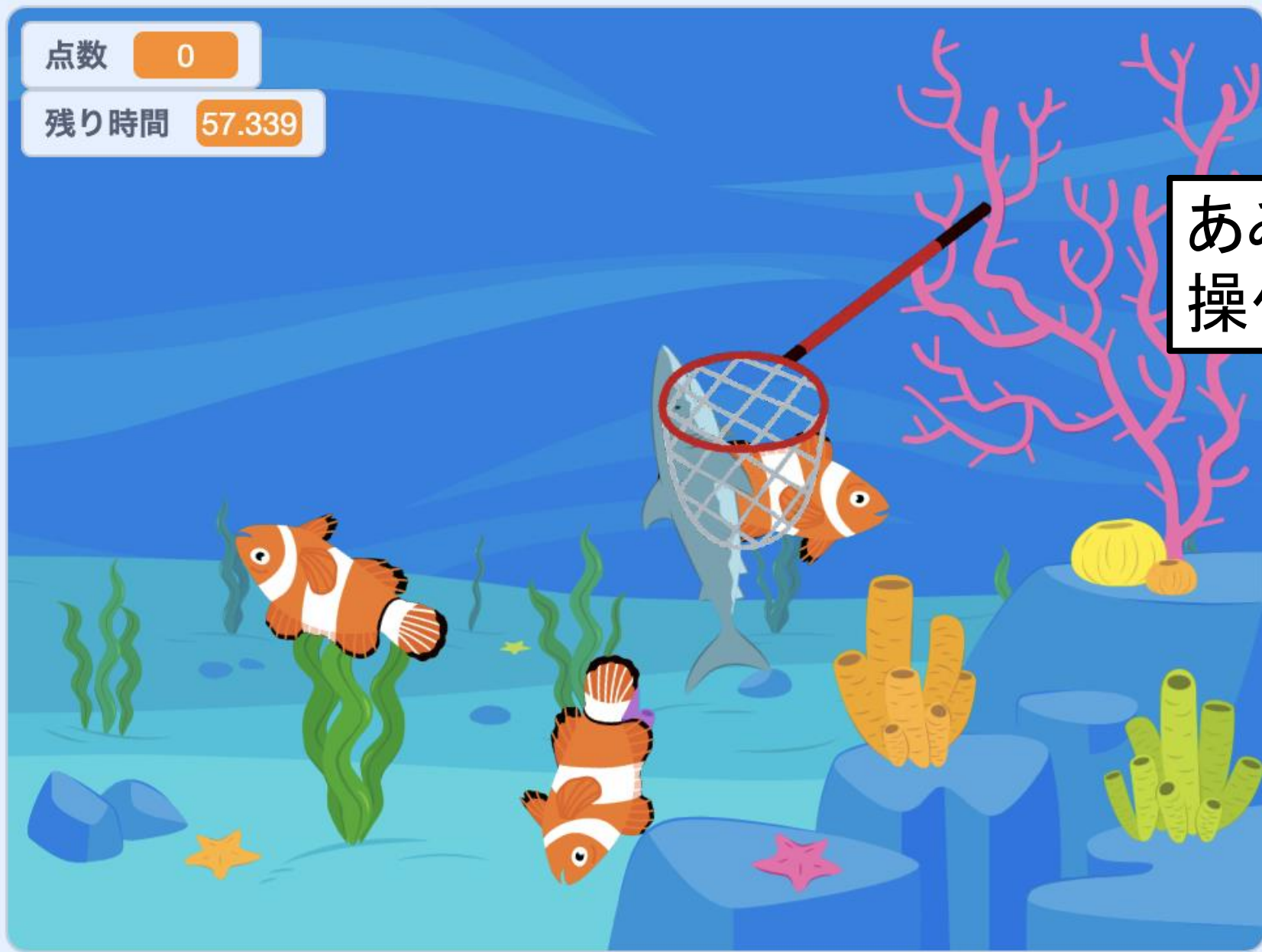
20

さかなすくいゲームを作ろう

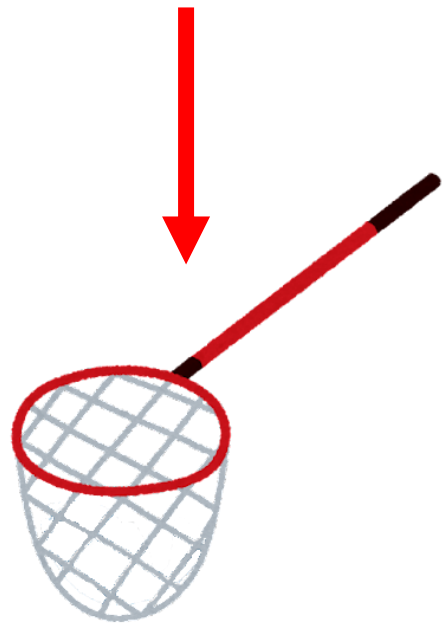


旗マークを押してゲームを開始してみよう

点数 0
残り時間 57.339



あみを操作したいけど、操作できない。。。。



スクラッチの位置指定について



The diagram illustrates the Scratch coordinate system overlaid on a game scene. A red crosshair is centered on a fishing net. The vertical axis is labeled 'y座標' (y-coordinate) with a '+' sign at the top and a '-' sign at the bottom. The horizontal axis is labeled 'x座標' (x-coordinate) with a '-' sign on the left and a '+' sign on the right. In the top-left corner, the score is '点数 0' and the remaining time is '残り時間 53.432'.

スプライト
あみ
表示す

← x 0
→

↑ y 0
↓

ステージ

コード コスチューム 音

- 動き
- 見た目
- 音
- イベント
- 制御
- 調べる
- 演算
- 変数
- ブロック定義
- micro:bit

音量

イベント

- が押されたとき
- スペース キーが押されたとき
- このスプライトが押されたとき
- 背景が ゲーム中 になったとき
- 音量 > 10 のとき
- ゲームオーバー を受け取ったとき
- ゲームオーバー を送る
- ゲームオーバー を送って待つ

まずは「イベントから側が押されたとき」を持ってくる



が押されたとき

- 最前面 へ移動する
- 表示する
- コスチュームを かまえる にする
- ずっと
 - もし なら
 - コスチュームを すくう にする
 - すくう を送る
 - 0.5 秒待つ
 - コスチュームを かまえる にする

スコア表示

点数 0

残り時間 53.432

スプライト

あみ

表示する

Fish Shark

制御

1 秒待つ

The image shows the Scratch code editor interface. The top menu bar includes 'Scratch', 'ファイル', '編集', and 'チュートリアル'. Below the menu, there are tabs for 'コード', 'コスチューム', and '音'. The left sidebar shows various block categories: 動き, 見た目, 音, イベント, 制御, 調べる, 演算, 変数, ブロック定義, and micro:bit. The main workspace contains two scripts. The first script starts with a 'が押されたとき' block, followed by a '1 秒待つ' block, and then a 'もし< >なら' block. A red arrow points from the 'もし< >なら' block to a text box. The second script starts with a 'が押されたとき' block, followed by a '最前面へ移動する' block, a '表示する' block, a 'コスチュームをかまえるにする' block, a 'ずっと' loop containing a 'もし< >なら' block with a 'すくうを送る' block and a '0.5 秒待つ' block, and another 'コスチュームをかまえるにする' block.

次に「x座標を〇〇ずつ変える」ブロックに制御にある「もし< >なら」ブロックを被せます。

コード コスチューム 音

調べる

動き

見た目

音

イベント

制御

調べる

演算

変数

ブロック定義

micro:bit

調べる

マウスのポインターに触れた

色に触れた

色が色に触れた

マウスのポインターまでの距離

あなたの名前は何ですか?と聞いて待つ

答え

スペースキーが押された

マウスが押された

マウスのx座標

マウスのy座標

ドラッグできるようにする

音量

タイマー

タイマーをリセット

もしスペースキーが押されたなら

x座標を10ずつ変える

もし右向き矢印キーが押されたなら

コスチュームをすくうを

0.5秒待つ

コスチュームをかまえる

もしなら

コスチュームをすくうを

0.5秒待つ

コスチュームをかまえる

表示する

コスチュームをかまえる

最前面へ移動する

押されたとき

点数

残り時間

▼を押すと、キーボードの押すところを選べます。

- スペース
- 上向き矢印
- 下向き矢印
- ✓ 右向き矢印
- 左向き矢印
- どれかの
- a
- b
- c

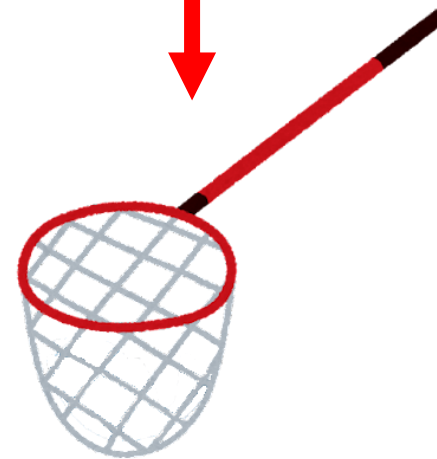


旗マークを押してゲームを開始してみよう

点数 0

残り時間 57.339

動くと思ったが、
動かない。。。。



ゲーム中

“ずっともし右キーが押されたらx座標を10ずつ変える”

にしたいため、「ずっと」のブロックをもしのブロックに囲うように入れる

The image shows the Scratch code editor interface. On the left, a sidebar contains various block categories: Control (orange), Cycles (blue), Operators (green), Variables (red), and Block Definition (pink). The main workspace displays two scripts. The first script, titled 'が押されたとき' (When pressed), contains a 'ずっと' (Forever) loop block. Inside this loop is a 'もし' (If) block with the condition '右向き矢印 キーが押された なら' (Right arrow key is pressed). The 'もし' block contains an 'x座標を 10 ずつ変える' (Increase x coordinate by 10) block. A red arrow points from the 'ずっと' block in the left sidebar to the 'ずっと' block in the script. The second script, also titled 'が押されたとき', contains a '最前面へ移動する' (Move to front) block, a '表示する' (Show) block, a 'コスチュームを かまえる にする' (Switch costume) block, a 'ずっと' (Forever) loop block, a 'もし' (If) block with an empty condition, a 'コスチュームを すくう にする' (Switch costume) block, a 'すくう を送る' (Send message) block, a '0.5 秒待つ' (Wait 0.5 seconds) block, and another 'コスチュームを かまえる にする' (Switch costume) block.



旗マークを押してゲームを開始してみよう

点数

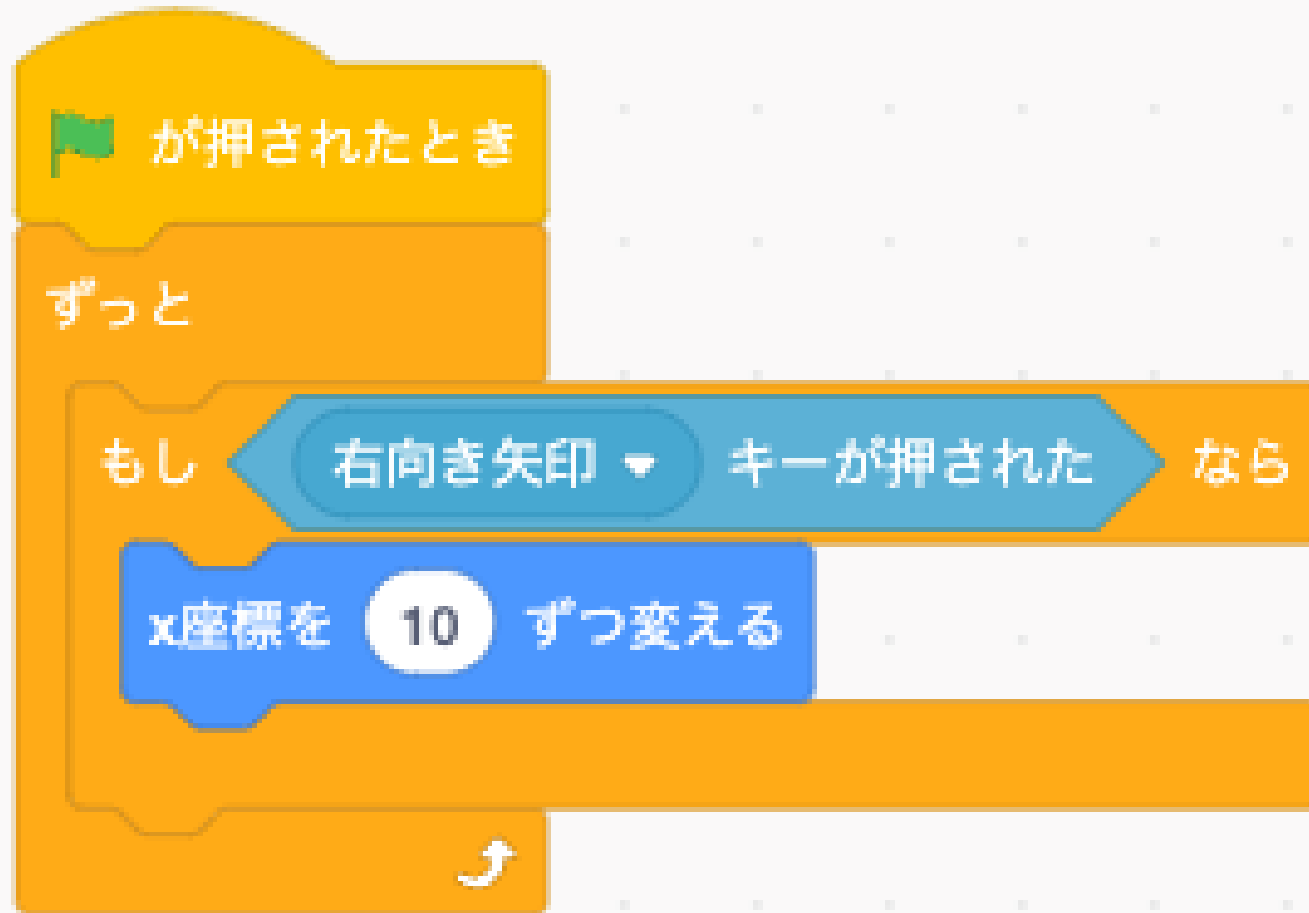
0

残り時間

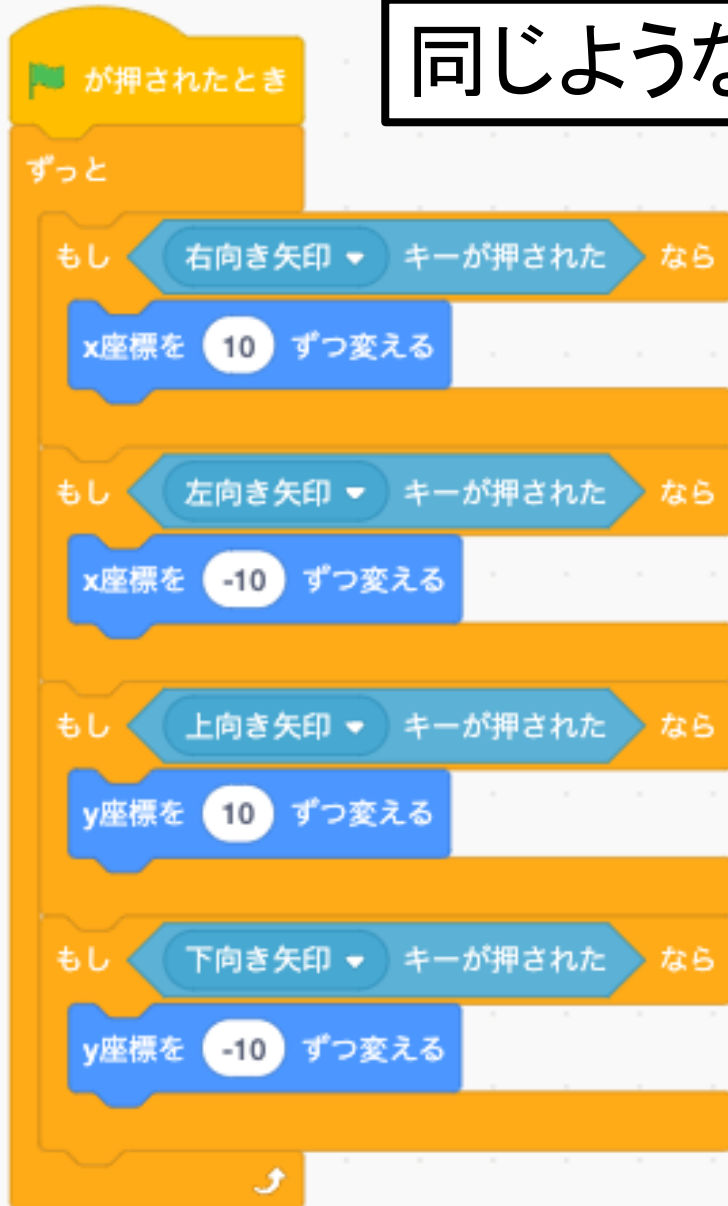
57.339



上下左右に動かすには？



同じような「もし」のブロックを作成する

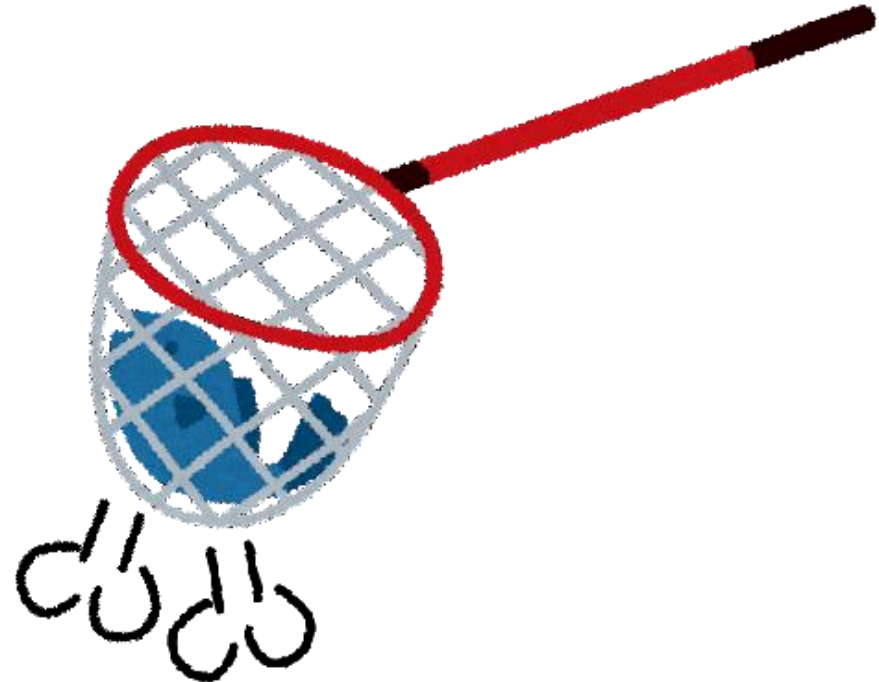


マイナスをつけてあげると
反対方向に動かせる

あみで魚をすくっているように見せる



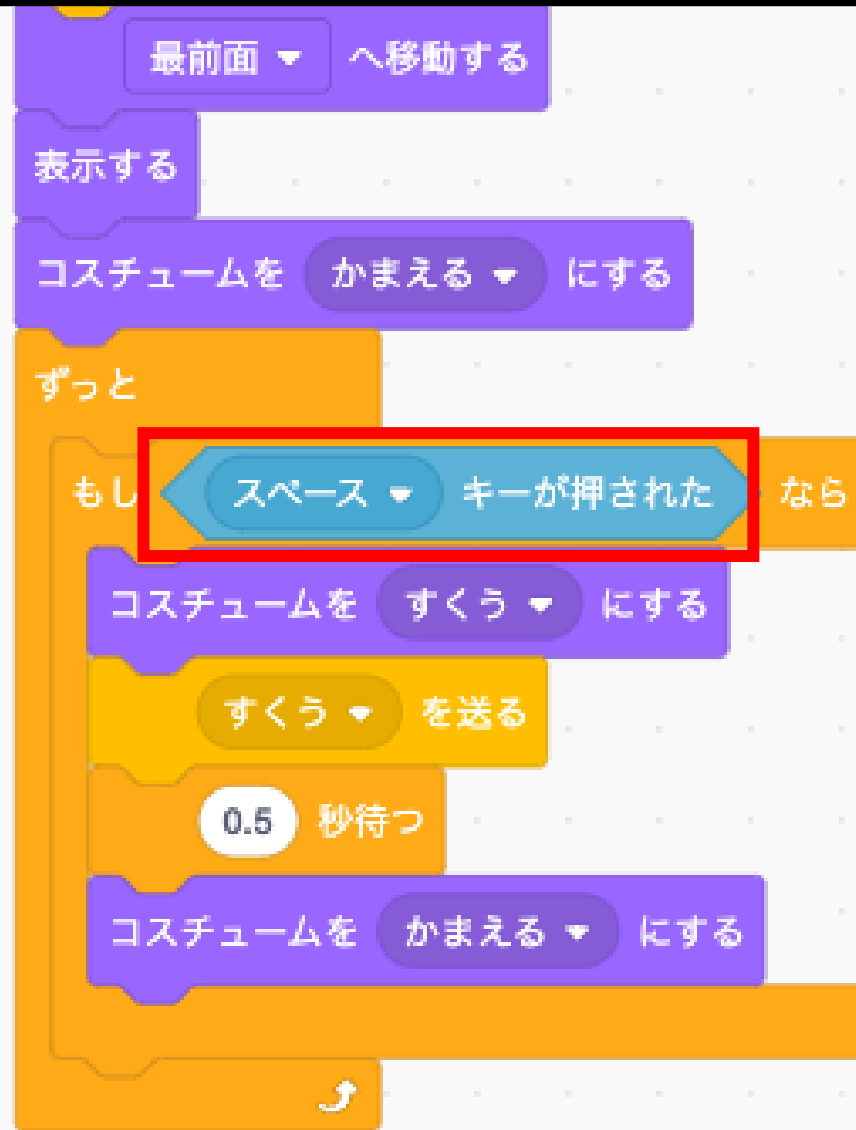
スペースキーを押したら
すくうあみの画像に変えたい



スペースキーが押されると変えたいので、
もし<>ならの部分には.....？



「スペースキーが押されたら」のブロックを入れる







さかなを押して、スプライトを切り替えます



The screenshot displays a software interface for creating an animation. At the top, there is a toolbar with icons for a flag, a red circle, a window, a blue square, and a zoom icon. Below this is a large stage area showing an underwater scene with blue water, green seaweed, pink coral, yellow sponges, and a red fishing net. A small yellow fish is visible in the water. Below the stage is a control panel for the selected sprite. The panel is divided into two sections: 'スプライト' (Sprite) and 'ステージ' (Stage). The 'スプライト' section includes a dropdown menu with 'さかな' (Fish) selected, a '表示する' (Show) section with 'on' and 'off' icons, a '大きさ' (Size) field set to 80, and a '向き' (Direction) field set to 90. The 'ステージ' section includes a '背景' (Background) dropdown menu with '2' selected. At the bottom of the control panel, there are three buttons: 'さかな' (Fish), 'さめ' (Shark), and 'あみ' (Net). The 'さかな' button is highlighted with a red border.

スプライト

さかな

↔ x 37

↑↓ y 19

表示する

大きさ 80

向き 90

ステージ

背景 2

さかな さめ あみ

さかなに置かれているブロックが見えていればOK

The image displays a Scratch-like programming environment with a fish-themed game. The interface is divided into several sections:

- Code Editor (Left):** Shows a list of blocks on the left and a script area on the right. The script area contains the following code:
 - When cloned:** A purple 'show' block, followed by three blue 'set position' blocks: 'set rotation to random degrees from -180 to 180', 'set x position to random from -240 to 240', and 'set y position to random from -180 to 180'.
 - Forever Loop:** A blue 'move 10 steps' block, followed by a blue 'if edge reached, bounce' block.
 - When green flag clicked:** A yellow 'say' block.
- Stage (Right):** Shows a fish named 'さかな' (Sakana) with a score of 0 and a timer of 59.661. The background is an underwater scene with coral and seaweed. The sprite panel shows 'さかな', 'さめ', and 'あみ'.

さかなにある「すくうを受け取ったとき」の、「もし<>なら」の穴あきを埋めたい。



あみの方に「すくうを送る」があるので、
送られた時に「あみに触れたら」もしく>ならの中に行きたい

あみのブロック

さかなのブロック

が押されたとき

- 最前面へ移動する
- 表示する
- コスチュームを かまえる にする
- ずっとループ:
 - もし スペース キーが押された なら
 - コスチュームを すくう にする
 - すくうを送る
 - 0.5 秒待つ
 - コスチュームを かまえる にする

すくうを受け取ったとき

- もし なら
 - 点数 を 10 ずつ変える
 - 隠す

調べるの「マウスのポインターに触れた」をはめて、
マウスのポインター▼を押して一覧から「あみ」を押します。

The screenshot shows the Scratch code editor with the 'Code' tab selected. The script is as follows:

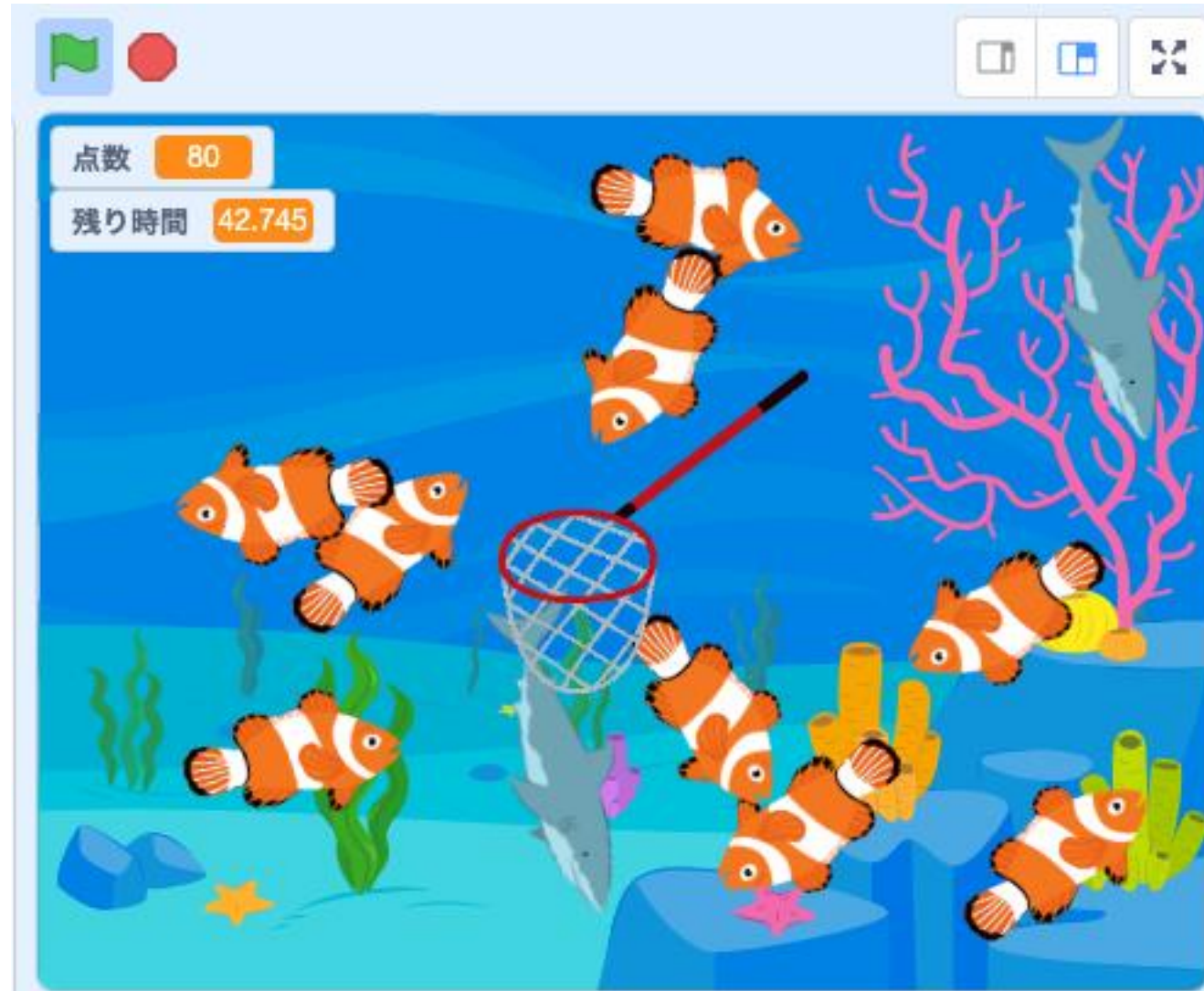
- このクローンを削除する
- 調べる
 - マウスのポインター▼に触れた
 - 色に触れた
 - 色が 色に触れた
 - マウスのポインター▼までの距離
 - クローンされたとき
 - あなたの名前は何ですか? と聞いて待つ
 - 答え -180 から 180 までの乱数 度に向ける
 - スペース▼ キーが押された 240 までの乱数 にする
 - マウスが押された 180 から 180 までの乱数 にする
 - マウスのx座標
 - マウスのy座標 10 歩動かす
 - もし触れたら 跳ね返る
 - ドラッグ できる▼ ようにする
- すくう▼を受け取ったとき
 - もし マウスのポインター▼に触れた なら
 - 点数▼ 10 ずつ変える
 - 隠す

This is a close-up of the 'もし マウスのポインター▼に触れた なら' block. The dropdown menu is open, showing a list of items:

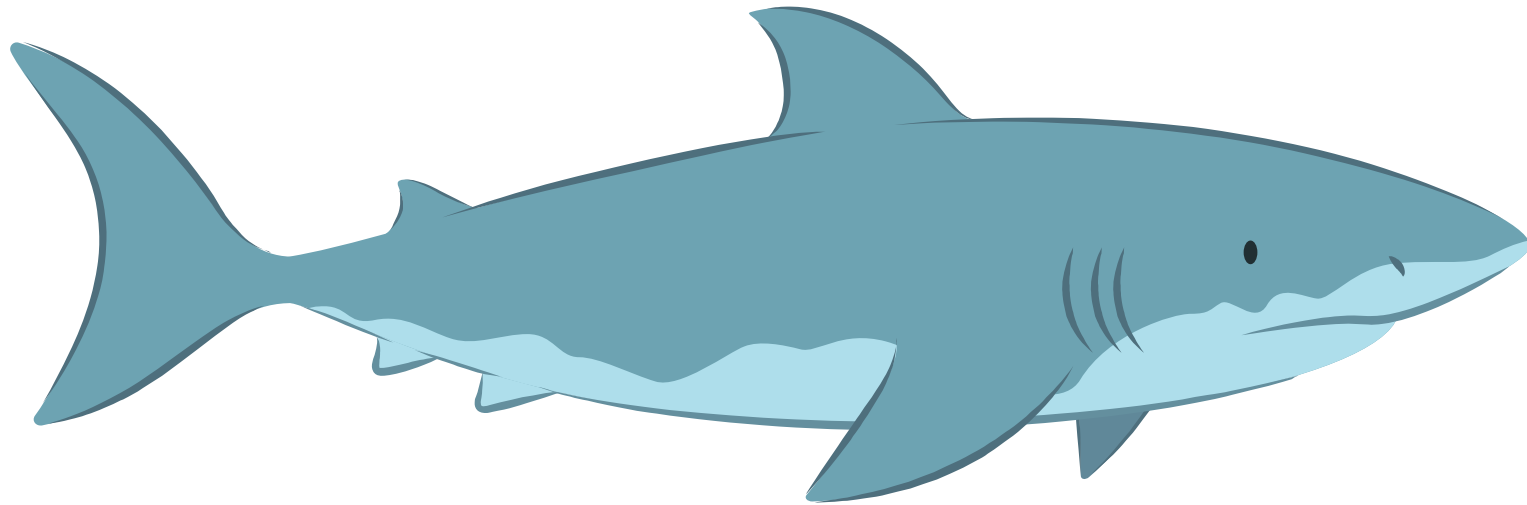
- ✓ マウスのポインター
- 端
- さめ
- あみ

The 'あみ' option is highlighted with a red box.

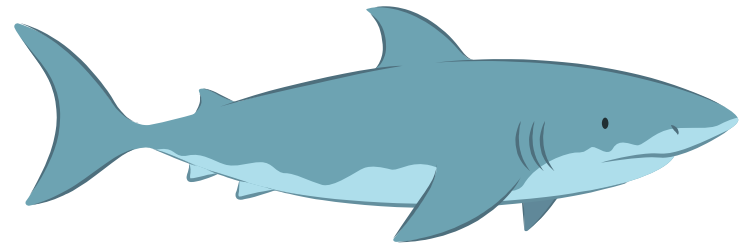
旗を押してゲームスタートして、さかなが消えるか確認しよう



さめも魚と同じようにやってみよう。



さめのプログラム



コード コスチューム 音

調べる

動き

見た目

音

イベント

制御

調べる

演算

変数

ブロック定義

あなたの名前は何ですか? と聞いて待つ

答え

スペース キーが押された

マウスが押された

マウスのx座標

マウスのy座標

ドラッグ できる ようにする

音量

タイマー

タイマーをリセット

マウスのポインター に触れた

色に触れた

色が 色に触れた

マウスのポインター までの距離

クローンされたとき

表示する

-180 から 180 までの乱数 度に向ける

x座標を -240 から 240 までの乱数 にする

y座標を -180 から 180 までの乱数 にする

回転方法を 自由に回転 にする

ずっと

50 歩動かす

もし端に着いたら、跳ね返る

すくろ を受け取ったとき

もし あみ に触れた なら

点数 を -30 ずつ変える

隠す

が押されたとき

隠す

ずっと

自分自身 のクローンを作る

10 秒待つ

Microbitと連動させてみよう

1

Micro bitを振ったときにリンゴがどんだんかじられる
プログラムを組んでみよう！



2

画面の左下にある、拡張機能を追加をクリックしよう



3

← 戻る

拡張機能を選ぶ

必要なもの 協力
Amazon Web Services

必要なもの

Microbitをクリックしよう



micro:bit
プロジェクトを現実の世界と接続する。

必要なもの 協力
Bluetooth Wi-Fi
micro:bit



LEGO MINDSTORMS EV3
対話型ロボットなどを作る。

必要なもの 協力
Bluetooth Wi-Fi
LEGO



LEGO BOOST
ロボット作品に

必要なもの
Bluetooth Wi-Fi



4

この画面が出たらいったん×をクリック

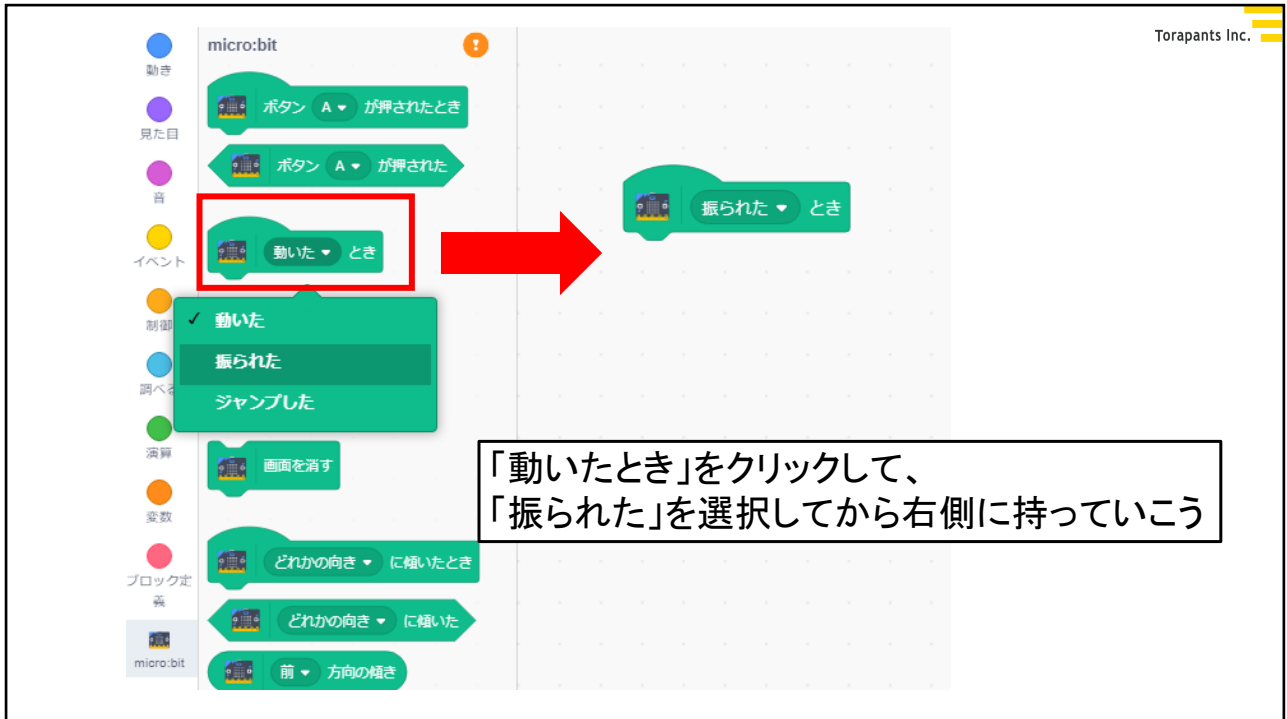


5

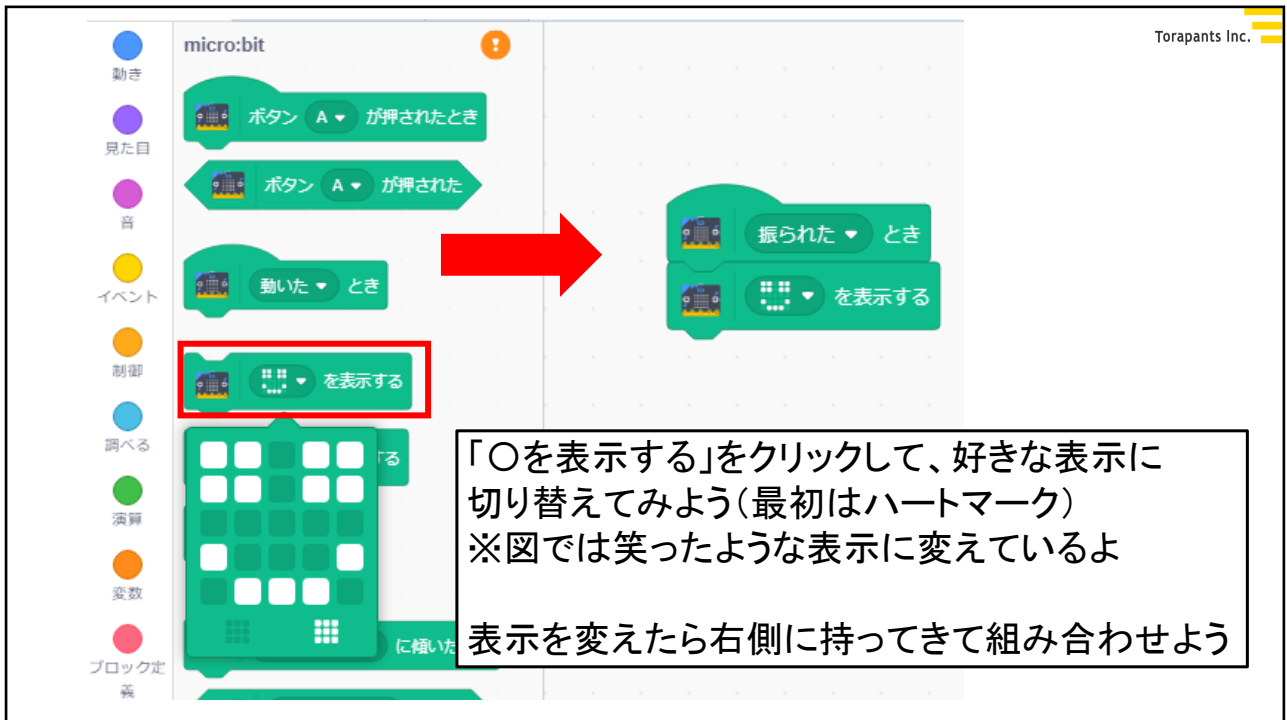


ブロックメニューの一番下に、
Microbit用ブロックが追加される

6



7



8

「音」から「~の音を鳴らす」を持ってきて
組み合わせよう

9

「制御」から「〇秒待つ」を持ってきて
組み合わせよう

秒数を「0.5秒待つ」に変更しよう

10

Torapants Inc.

「見た目」から「次のコスチュームにする」を持ってきて組み合わせよう

Scratch 'Looks' category blocks:

- 動き
- 見た目
- 音
- イベント
- 制御
- 調べる
- 演算
- 変数
- ブロック定義

Scratch script on stage:

- 振られた とき
- を表示する
- Bite の音を鳴らす
- 0.5 秒待つ
- 次のコスチュームにする

11

Torapants Inc.

「制御」から「もし～なら」を持ってきて組み合わせよう

同じ動きをくりかえす処理をつくるよ

Scratch 'Control' category blocks:

- 動き
- 見た目
- 音
- イベント
- 制御
- 調べる
- 演算
- 変数
- ブロック定義

Scratch script on stage:

- を表示する
- Bite の音を鳴らす
- 0.5 秒待つ
- 次のコスチュームにする
- もし なら

12

Torapants Inc.

色 の効果を 0 にする

動き

見た目

音

イベント

制御

調べる

演算

変数

ブロック定

音

画像効果をなくす

表示する

隠す

最前面へ移動する

1 層 手前に出す

コスチュームの 番号

背景の

番号

名前

もし なら

コスチュームの 名前

「見た目」の「コスチュームの番号」を「コスチュームの名前」に変更して右側に持っていこう

13

Torapants Inc.

動き

見た目

音

イベント

制御

調べる

演算

変数

ブロック定

演算

1 から 10 までの乱数

> 50

< 50

= 50

かつ

または

ではない

もし なら

コスチュームの 名前

= apple8

「演算」の「0=0」を右側に持っていこう
内容を「0=apple8」に変更しよう

14

持って来たブロックを赤い四角で囲んだ図のように組み合わせよう

組み合わせたら、「もし～なら」ブロックに合体させよう

The image shows a Scratch script editor. On the left, a palette of blocks is visible. A block labeled 'コスチュームの名前 = apple8' (Costume Name is apple8) is highlighted with a red rectangle. A red arrow points from this block to the right, where it is being placed into a 'もし～なら' (If-then) block. The script on the right includes: 'Bite の音を鳴らす' (Play Bite sound), '0.5 秒待つ' (Wait 0.5 seconds), '次のコスチュームにする' (Switch to next costume), and the 'もし～なら' block with the 'コスチュームの名前 = apple8' block inside it.

15

「制御」から「1秒待つ」を持ってきて図のように組み合わせよう

The image shows a Scratch script editor. On the left, a palette of blocks is visible. A block labeled '1 秒待つ' (Wait 1 second) is highlighted with a red rectangle. A red arrow points from this block to the right, where it is being placed at the end of a script. The script on the right includes: '振られたとき' (When clicked), 'を表示する' (Show sprite), 'Bite の音を鳴らす' (Play Bite sound), '0.5 秒待つ' (Wait 0.5 seconds), '次のコスチュームにする' (Switch to next costume), and a 'もし～なら' (If-then) block with 'コスチュームの名前 = apple8' (Costume Name is apple8) inside it. The '1 秒待つ' block is being added below the 'もし～なら' block.

16

「見た目」の「コスチュームをapple8にする」を「コスチュームをappleにする」に変更しよう
変更したら図のように組み合わせよう

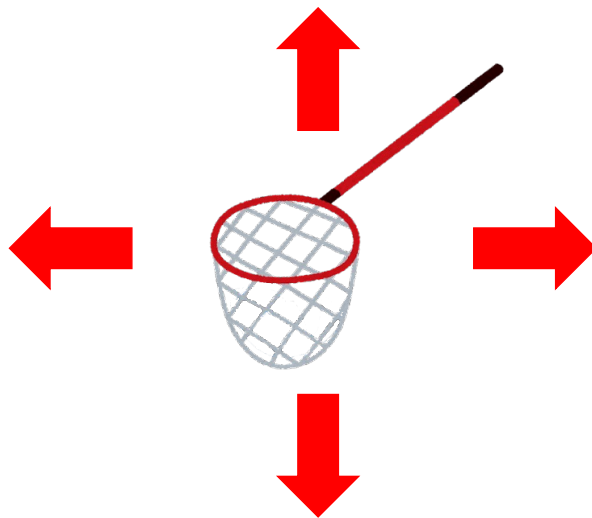
17



完成！

Microbitとパソコンを繋いで、実際に動かしてみよう

18



さっき完成させた魚すくいゲームで、
Microbitを傾けたらあみを移動させるには
どうしたらいいだろう？

19



色々な方向に傾いた時に
座標を変えてあげればOK

20